11－R001

次世代ネットワークと
産業フロンティア情報技術に関する
調査研究報告書
－　ネットワークエージェント技術　－

平成１２年３月

JIPDEC

財団法人　日本情報処理開発協会

KEIRIN 00 この事業は、競輪の補助金を受けて実施したものです。

まえがき

近年、次世代インターネット上の情報資源活用と新しいタイプのコミュニケーションの創出をめざして、ネットワークエージェント技術の研究とその製品開発への取り組みが活発化しています。ネットワークエージェントは、ユーザの代理として問題に応じた様々なアプリケーションソフトウェアや人間と交渉などをする新しいタイプのソフトウェアです。今後の複雑・多様化する情報化社会において、電子商取引、電子図書館、デジタル放送、CAD-CIM、コミュニティビジネスなど広範なアプリケーションの多くの場面で、人間の活動を支援するエージェント技術に対する期待には大きなものがあります。最近、エージェント構築用プラットフォームがネットワークを通して公開され、これらの間のインターオペラビリティを図るための標準化活動が展開されています。

このような状況のもと、当協会では、平成11年度「次世代ネットワークと産業フロンティア情報技術に関する調査研究」として、内外における産業界・学界のネットワークエージェントに関わる研究開発動向を調査して、技術課題および今後の研究開発の進め方について提言をまとめるため、本年度より2年計画で「ネットワークエージェント技術に関する調査研究」を実施することといたしました。

実施にあたっては、ネットワークエージェント技術委員会（委員長　西田 豊明 東京大学大学院工学系研究科電子情報工学専攻 教授）を設置して、調査研究の基本方針、個別テーマの審議・検討を行うとともに、欧米への調査員の派遣、文献調査、ワークショップ開催等を行いました。

本報告書は、平成11年度の調査研究成果を取りまとめたもので、3編から構成され、Ⅰ編は概要編で調査研究成果の要約を、Ⅱ編は本編であり調査研究成果の詳細を、また、Ⅲ編は資料編でエージェント研究開発の国内事例、エージェント技術の内外のプロジェクト一覧等を、まとめています。Ⅱ編は、調査研究の趣旨を述べた第1章、多方面にわたってエージェント技術の動向をまとめた第2章、エージェント技術の現状分析を行った第3章、まとめの第4章から構成されています。

本書が広く各界の方々に活用されることを念願する次第です。

最後に、本調査研究の実施にあたり、ご指導ご協力をいただいた委員各位ならびに関係各位に深甚なる謝意を表する次第であります。

平成12年3月

財団法人　日本情報処理開発協会

ネットワークエージェント技術委員会

（敬称略、50 音順）

委員長：

西田　豊明	東京大学大学院工学系研究科電子情報工学専攻　教授

幹　事：

木下　哲男	東北大学電気通信研究所　助教授
北村　泰彦	大阪市立大学工学部情報工学科　助教授

委　員：

上田　隆也	キヤノン（株）情報技術研究所インターネット技術研究部 インターネット技術第三研究室　主任研究員
栄藤　稔	松下電器産業株式会社　東京通信システム研究所　チームリーダー
大沢　英一	（株）ソニーコンピュータサイエンス研究所　リサーチャー
大須賀　昭彦	（株）東芝　研究開発センター コンピュータ・ネットワークラボラトリー　主任研究員
岡田　誠	（株）富士通研究所　ネットメディア研究センター　研究員
木野　茂徳	三菱電機（株）情報技術総合研究所ネットワークコンピューティング部 オープンネットワークチーム　チームリーダ
金　淵培	日本放送協会　放送技術研究所マルチメディアサービス部 主任研究員
桑原　和宏	日本電信電話（株）NTT コミュニケーション科学基礎研究所社会情報研究部 協調システム研究グループ　リーダ・主幹研究員
小暮　潔	日本電信電話（株）NTT コミュニケーション科学基礎研究所社会情報研究部 分散協調理論研究グループ　リーダ・主幹研究員
須栗　裕樹	（株）コムテック　取締役技術開発部部長
豊内　順一	（株）日立製作所　システム開発研究所　第２部　主任研究員
中嶋　宏	オムロン（株）　技術本部ＩＴ研究所ファジィ・ヒューマンメディア研究室 担当係長

中村	祐一	日本アイ・ビー・エム（株）　東京基礎研究所　主任研究員
長尾	確	日本アイ・ビー・エム（株）　東京基礎研究所　特別研究員
服部	文夫	NTT ソフトウェア（株）　技術開発部 エレクトロニックコマース技術センター所長
藤田	悟	日本電気（株）　C&C メディア研究所基盤ソフトウェア TG 主任研究員
保科	剛	日本ユニシス（株）　システム技術本部知識システム部 担当部長
益岡	竜介	（株）富士通研究所　コンピュータシステム研究所知能システム研究部 主任研究員
間瀬	健二	（株）エイ・ティ・アール知能映像通信研究所第二研究室　室長
三吉	秀夫	シャープ（株）　技術本部マルチメディア推進本部システム開発センター 主任研究員
和田	英彦	横河電機（株）　情報技術開発センター　係長
渡辺	光一	（株）野村総合研究所　情報技術調査室　主任研究員

協力者：

伊藤	修	（株）日本総合研究所　技術戦略クラスター　上席主任研究員

オブザーバ：

福田	賢一	通商産業省　機械情報産業局　電子政策課　課長補佐
横塚	志行	通商産業省　機械情報産業局　電子政策課　技術係長
松本	崇	通商産業省　機械情報産業局　電子政策課

事務局：

茂呂	知明	（財）日本情報処理開発協会　技術企画部技術課　専任調査役
片岡	幸一	（財）日本情報処理開発協会　技術企画部技術課　課長

委員執筆分担

（敬称略）

西田　豊明	Ⅰ編、Ⅱ編1.、3.3、3.4、4.、Ⅲ編A.1、D.2
木下　哲男	Ⅱ編3.1、Ⅲ編A.2、B、D.2
北村　泰彦	Ⅱ編3.2、Ⅲ編A.3、C、D.1
上田　隆也	Ⅲ編A.4
栄藤　稔	Ⅱ編2.1.7、Ⅲ編A.5
大須賀昭彦	Ⅱ編2.1.5、Ⅲ編A.6
岡田　誠	Ⅱ編2.1.8、Ⅲ編A.7
木野　茂徳	Ⅱ編2.1.2、Ⅲ編A.8
金　淵培	Ⅱ編2.1.6、Ⅲ編A.9
桑原　和宏	Ⅱ編2.3.3、2.3.4、Ⅲ編A.10
小暮　潔	Ⅲ編A.10
須栗　裕樹	Ⅱ編2.4.1、Ⅲ編A.11
豊内　順一	Ⅱ編2.4.2、Ⅲ編A.12
中嶋　宏	Ⅱ編2.2.2、Ⅲ編A.13
中村　祐一	Ⅱ編2.3.2、Ⅲ編A.14
長尾　確	Ⅱ編2.2.3
藤田　悟	Ⅱ編2.3.5、Ⅲ編A.15
保科　剛	Ⅱ編2.3.1、2.3.6、Ⅲ編A.16
益岡　竜介	Ⅱ編2.1.1、2.1.3、Ⅲ編A.17
間瀬　健二	Ⅱ編2.2.1、Ⅲ編A.18
三吉　秀夫	Ⅲ編A.19
和田　英彦	Ⅱ編2.1.4、Ⅲ編A.20

注記：本報告書の中にある登録商標および商標は、該当各社の登録商標および商標であります。

目　次

まえがき
委員会および協力者名簿
委員執筆分担

Ⅰ．概要 ……3

Ⅱ．本編
1．調査研究の趣旨 ……7
1.1　背景と目的 ……7
1.2　調査の視点 ……7
1.3　調査研究の内容 ……9
1.4　調査研究の方法 ……10
1.5　スケジュール ……10

2．エージェント技術の動向 －経緯と現状－ ……13
2.1　ビジネス・産業応用 ……13
2.1.1　企業情報システムと知識管理(Knowledge Management) ……13
2.1.2　エージェント技術のEAI (Enterprise Application Integration) への応用 ……19
2.1.3　EC（電子商取引） ……27
2.1.4　加工装置制御システム ……36
2.1.5　ITS（高度道路交通システム）分野への応用 ……44
2.1.6　デジタル放送とエージェントテレビ ……50
2.1.7　情報家電におけるネットワークエージェント技術 ……56
2.1.8　サービス連携 ……61
2.2　インタフェースエージェント ……69
2.2.1　情報メディアとしてのパーソナルガイドエージェント ……69
2.2.2　心を持った機械達 ……79
2.2.3　エージェントの拓く未来 ……88
2.3　プラットフォーム ……95
2.3.1　大規模分散システム環境 System ν [nju:] ……95
2.3.2　現実への利用を指向したe-Marketplaceの開発 ……100
2.3.3　マルチエージェント協調プロトコル記述 AgenTalk ……116
2.3.4　ソーシャルウェア構築用プラットフォーム Shine ……120
2.3.5　パターンに基づく移動エージェントシステムの設計手法 ……122
2.3.6　移動エージェントシステムによるアプリケーションの構築 ……127
2.4　標準化 ……133
2.4.1　FIPA (Foundatiton for Intelligent Physical Agents) ……133
2.4.2　自律分散サービスシステムアーキテクチャの提案
－基本概念・技術と標準化活動－ ……140

3. 技術課題の分析 ―応用から要請される技術課題― ……149
3.1 サービス統合 ……149
3.1.1 エージェントの役割 ……149
3.1.2 現状のアプリケーションの特徴 ……149
3.1.3 エージェント技術の現状 ……150
3.1.4 アプリケーションからみた利点 ……151
3.1.5 エージェント技術の課題 ……152
3.1.6 エージェント技術の技術マップ ……153
3.2 マルチエージェントシステム ……153
3.2.1 協調的マルチエージェントシステム ……154
3.2.2 競合的マルチエージェントシステム ……156
3.3 インタフェースエージェント ……159
3.4 標準化 ……161
3.5 ネットワークエージェント開発方法論 ……163
3.5.1 設計開発支援の現状 ……163
3.5.2 アプリケーション事例にみるネットワークエージェントの設計開発 ……164

4. まとめ ……169

Ⅲ. 資料編
A. 研究開発への取り組み事例
A. 1 東京大学・大学院工学系研究科・電子情報工学専攻・西田研究室 ……175
A. 2 東北大学 電気通信研究所 情報通信システム研究分野
―やわらかい分散システムとエージェント技術― ……179
A. 3 大阪市立大学工学部情報工学科知識情報工学研究室における取り組み
―WWW 情報収集エージェント MetaCommander― ……182
A. 4 キヤノン(株)情報技術研究所におけるエージェント技術への取り組み ……184
A. 5 松下電器産業株式会社 東京通信システム研究所
―Content-Agents― ……186
A. 6 東芝研究開発センターにおける取り組み ……188
A. 7 富士通研究所ネットメディア研究センター
―エージェント技術に関するネットメディア研究センターの取り組み― ……191
A. 8 三菱電機におけるエージェント技術研究開発への取り組み ……194
A. 9 ＮＨＫにおける放送用エージェントの研究開発への取り組み ……196
A. 10 NTT コミュニケーション科学基礎研究所における
エージェント技術への取り組み ……198
A. 11 株式会社コムテックにおける研究開発への取り組み事例 ……200
A. 12 日立製作所におけるエージェント技術への取り組み ……202
A. 13 オムロン株式会社における
インタフェース・エージェントへの取り組みについて ……204
A. 14 日本アイ・ビー・エム株式会社東京基礎研究所における
エージェント技術への取り組み ……207

A.15　ＮＥＣにおける移動エージェント技術への取り組み……208
A.16　日本ユニシスにおけるエージェント技術への取り組み……211
A.17　株式会社 富士通研究所 知能システム研究部
　　－会話型エージェントシステム SAGE と INTERSTAGE AGENTPRO－……213
A.18　ATR 知能映像通信研究所におけるエージェント技術への取り組み……216
A.19　技術研究組合新情報処理開発機構　マルチモーダル機能シャープ研究室
　　－円滑な対話機能を持つインタフェースエージェント－……218
A.20　横河電機におけるエージェント技術への取り組み……220

B. エージェントシステム開発プロジェクト一覧
B.1　北　米……228
B.2　欧　州……244
B.3　日　本……262
B.4　アジア……276
B.5　オセアニアその他……282

C. ワークショップ記録
セッション１：ビジネス・産業応用……289
セッション２：インタフェースエージェント……294
セッション３：プラットフォームと標準化……297

D. 海外調査報告
D.1　米国調査報告……311
D.2　欧州調査報告……319

Ⅰ. 概要

概 要

1990年代になって「エージェント」という概念に基づくソフトウェアが単なる研究上の話題でなく、ビジネスや産業に幅広く応用できるものであるという認識が高まり、本格応用に向けた取り組みが活発化した。これを従来の理論指向のエージェントと区別して、応用指向のエージェント研究開発と呼ぶことにしよう。

2000年を迎え、応用指向のエージェント研究開発の動きは、FIPA (Foundation for Intelligent Physical Agents)、OMG (Object Management Group)などにおけるエージェント標準化活動と、数々のベンダーからのエージェントシステム構築用キットの提供、学会における応用問題への取り組みの活発化と連動して、本格化した。インターネットをはじめ、我々の生活がネットワーク化しつつある現在、ネットワークを活用し、新たな価値を付加するための主要技術の一つとして、エージェントは大きな注目を浴びている。

(財) 日本情報処理開発協会では、産業界・学会におけるネットワークエージェントに関わる研究開発の現状を調査して、現在の動向を取りまとめるとともに、今後の研究開発の進め方について提言をまとめることを目的として1999年度から2年間の予定で「ネットワークエージェント技術委員会(委員長　西田豊明 東京大学大学院工学系研究科電子情報工学専攻教授)」を設置した。この委員会は、学術的視点よりも、ビジネス・産業応用の視点に重点をおいたものであり、主たる構成メンバーは、わが国でエージェント技術開発において先進的な取り組みを行っている企業における研究開発リーダーたちである。初年度は、現在のエージェント技術の分析を行い、エージェント技術への期待、エージェント技術の現在のレベル、エージェント技術の今後の課題を明らかにすることをめざした。

本報告書は、ネットワークエージェント技術委員会の初年度活動成果をまとめたものである。本編は、調査研究の趣旨を述べた第1章、多方面にわたってエージェント技術の動向をまとめた第2章、エージェント技術の現状分析を行った第3章、まとめの第4章から構成されている。

第1章では、本調査研究の背景と目的、視点、内容、方法について述べている。

第2章では、エージェントの技術的側面について、応用的視点にたって動向報告を行っている。2.1節の「ビジネス・産業応用」では、エージェントのビジネス・産業応用として有望であると考えられる8つの領域 (企業情報システムと知識管理、EAI (Enterprise Application Integration：企業間アプリケーション統合)、電子商取引、加工装置制御システム、ITS (Intelligent Transportation Systems：高度道路交通システム)、デジタル放送とエージェントテレビ、情報家電、サービス連携) について、技術動向をまとめた。2.2節の「インタフェースエージェント」では、知識メディアとしてエージェントを捉える立場、心をもつ機械としてエージェントを捉える立場、個人化と擬人化という視点からエージェントを捉える立場からの報告を行った。2.3節の「プラットフォーム」関連の技術動向では、大規模広域分散システム環境 System ν、現実への利用を指向したe-Marketplace、マルチエージェント協調プロトコル記述 AgenTalk、ソーシアルウェア構築用プラットフォーム Shine、移動エージェントシステムによるアプリケーションの構築、パターンに基づく移動エージェントシステムの設計手法について報告した。2.4節の「標

準化」では、エージェントにかかわりの深い標準化活動としてFIPAとADSSについて報告した。

第3章では、サービス統合、マルチエージェントシステム、インタフェースエージェント、標準化、ネットワークエージェント開発方法論の5項目について技術課題の分析を行った。

第4章では、エージェント技術開発の現状を総括し、今後開発すべき技術課題について述べている。

Ⅱ．本編

１．調査研究の趣旨

1. 調査研究の趣旨

1.1 背景と目的

インターネット上の情報資源活用と新しいタイプのコミュニケーションを図るための技術としてネットワークエージェント技術への取り組みが活発化している。ネットワークエージェントは、ユーザの代理として他のソフトや人間と交渉などをする新しいタイプのソフトウェアとして特徴付けられる。各種の情報ネットワークが普及して、人間を取り巻く情報環境が複雑・多様化する今後の情報化社会において人間の活動を支援するために不可欠な技術である。電子商取引、デジタルライブラリ、デジタル放送、CAD-CIM、コミュニティビジネスなど広範なアプリケーションに有効に適用されることが期待される。また、最近二三年の間に、エージェント構築システムもネットワークを通して公開され、それらの間のインターオペラビリティを図るための標準化活動も展開されている。エージェントを巡る国内外の最近の動きは加速・多様化し、その全貌を捉えることは困難になっている。

ネットワークエージェント技術委員会では、産業界・学会におけるネットワークエージェントに関わる研究開発の現状を調査して、現在の動向を取りまとめるとともに、今後の研究開発の進め方について提言をまとめることを目的とする。技術は日々急速に進歩しているので、ある時点での完全な全貌を把握してもすぐに古くなってしまう。そこで、ネットワークエージェントに関わる研究開発の動向が常に把握できるような体制つくりについても討論する。また、エージェント技術は人間の日常生活と密接に関連するために、純粋に技術的側面だけから調査を進めることは適切でないと考えられる。そこで、社会心理学や社会学など人文系の研究者を招いて総合的に議論する。

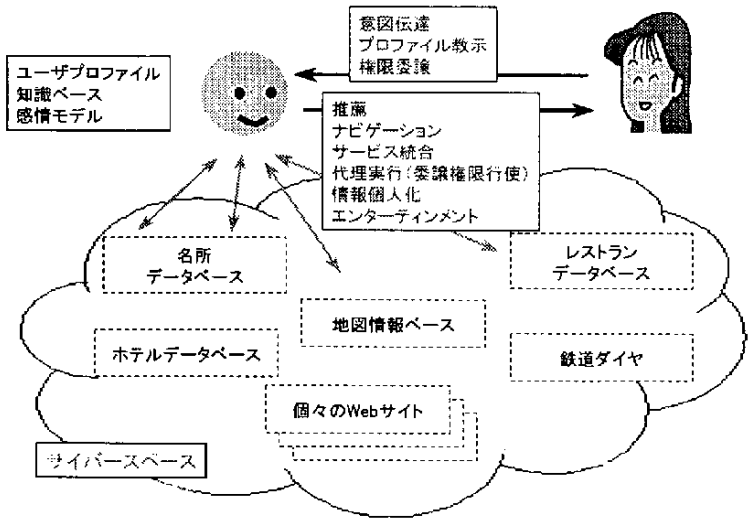
1.2 調査の視点

本報告書では、図表 1-1 のように、エージェントはユーザから一定の権限を委嘱され、ユーザの代理としてサイバースペースのなかで活動し、ユーザに対してさまざまなサービスを提供する知的ソフトウェアとして位置付けられる。概念的には、代理人としてのエージェントはユーザから見たエージェントの姿であり、動作主としてのエージェントはネットワーク上で自律的に行動するソフトウェアと言うシステム的な視点から見たエージェントの姿であると考えることができる。以上を簡潔にまとめると、現在のエージェントは次のように定式化できる。

●エージェントはインターネット上で社会的相互作用（協調・連携・交渉）しながら、ユーザの代理人として活動する知的ソフトウェアであり、ユーザともマルチモーダルインタフェースによって社会的・情動的なインタラクションをする。

エージェントは次のような社会情勢に応えようとした技術であると位置付けることができる。

a) 人工システムの多様化と高度化
b) 情報の洪水による人間のオーバーロード
c) 感性的側面と社会的側面を考慮した人工システムと人間のインタラクションの必要性
d) 新しい空間に即した行動能力をもつ人工システムの必要性



図表 1-1：エージェント技術

エージェント：インターネット上で社会的相互作用（協調・連携・交渉）しながら、ユーザの代理人として活動する知的ソフトウェア。ユーザとも社会的なインタラクションをする。

応用面から見ると、現在のエージェント技術は、次のような側面から特徴付けられる。

1) 新しいヒューマンインタフェース技術：画面上のキャラクタとマルチモーダルインタフェースによって対話する方式のインタフェース（エージェント指向インタフェース）の実現をめざす
2) ネットワーク上での情報流通とサービス統合の技術：ユーザが求める機能とシステムが提供する機能を結びつける自律プログラム（ソフトウェアエージェントとマルチエージェント技術）の実現をめざす
3) 人間同士のインタラクションの知的支援技術：人間関係の形成や交渉など人の媒介（エージェントに媒介されたコミュニケーション）する知的プログラムの実現をめざす
4) エージェントプラットフォーム技術：種々のエージェントシステムを構築するためのプラットフォームを実現する
5) エージェント標準化：エージェントシステムのインターオペラビリティを実現するための標準仕様を策定する

本報告書は、この構造を基本的枠組みとし、研究開発の取り組み状況の多少を勘案して若干の再構成を行ったものである。

他方、理論面からは、エージェントは次のような性質をもつ人工システムとして特徴付けられる。

1) 自律性：自らの目標を達成するために、自らの意思決定原理に基づいて、環境を認識し、行動する。
2) 協調性：環境内にいる他のエージェントと共通の目標を達成するための共同作業ができる。
3) 持続性：自己概念を継続的に保持する。

4) 学習・適応能力：個々あるいは共同で、経験を通して自らの能力を高めていくことができる。
5) 代理性または間接操作：ユーザから一定の権限を委譲され、ユーザの代理としてその権限を行使する。
6) 移動性：ネットワーク上をホストからホストへ移動できる。

ただし、必ずしもこれらの条件の全てが揃う必要はない。

もう一つの立場に、知的エージェントを定式化するには物理概念や機能概念を用いるよりも心的状態に帰着させた方がよいという考え方がある。この考え方に従うと、エージェントは「その動的特性が信念・プラン・意図などの語彙を用いて合理的に説明できるソフトウェアモジュール」と定義されることになる。このような外面からの特徴づけに基づく定式化も、内面的性質による特徴づけと同等に重要である。

本委員会では、理論としてのエージェント技術と。応用に使用されているものとの間には若干の隔たりがあるが、この点についての考え方についても議論する。

1.3 調査研究の内容

ネットワークエージェント技術について、技術的側面と社会的側面から調査を進め、得られた知見に基づいて提言をまとめる。

技術的側面については、次のような項目について調査を行う。

1) レガシィソフト・レガシィデータの知識共有を図るためのインターオペラビリティの実現
2) 電子商取引や電子図書館において情報発信者と情報利用者を結ぶ情報仲介技術
3) 場所に依存しない個人特化型のコミュニケーション環境の提供を狙ったエージェント通信技術
4) デジタル放送におけるネットワークエージェント技術
5) 製造におけるネットワークエージェント技術
6) マルチエージェント技術を応用した高度の協調・交渉技術
7) エージェントプラットフォーム
8) エージェント技術の利用技術
9) ネットワークエージェントの標準化の動向

以上について、技術動向の調査を行うことによって、エージェント技術の核心が何かを明確にし、これまでの研究開発をエージェント技術体系としてまとめる。

社会的側面については、次のような項目について調査を行う。

1) エージェントは人間個人・社会にどのような変化をもたらすか
2) エージェントが人間社会に浸透するための条件
3) エージェント技術を活用したビジネスモデル

以上に基づいて、今後のエージェント技術が社会生活に浸透するシナリオを描くとともに、今後重点的な研究開発が必要とされる研究課題を整理し、調査報告とする。

1.4 調査研究の方法

委員会活動、国内外調査、ワークショップの主催を軸とする研究活動を行う。

委員会については、実装と実証について豊富な経験をもっている産業界のメンバーを中心に構成し、中立的な立場にある学界委員が討論の取り纏めをする。エージェントの研究開発に携わる研究者・技術者を講演者として招聘し、討論をすることによって、調査項目に関する視点を明らかにする。また、委員会内部で分担してエージェント技術調査を実施する。焦点を絞って海外調査も実施する。特に、１回限りの調査にするのではなく、技術動向が継続的に把握できるような体制を作ることを試みる。

また、年に１度程度ワークショップを開催し、現在のエージェント技術の特徴づけや今後の技術展開のシナリオについて公開の場で討論を行う。以上の調査結果を報告書にまとめて公開する。

1.5 スケジュール

年間８回程度委員会を開催する。年間１回ワークショップを開催して、調査結果を公開の場で公表し討論する。

初年度にあたる平成 11 年度は、平成 11 年６月から平成 12 年２月までに計８回の委員会を開催して検討を進めた。海外調査については、平成 11 年 11 月に米国にて開催された ACM EC99 およびワシントン大学を訪問して研究および実用化動向を調査し、また、平成 12 年１月に英国ロンドンで開催された FIPA 会議に参加して標準化動向を調査した。これらについては、Ⅲ編Ｄに報告されている。

さらに、「ネットワークエージェント技術に関するワークショップ」を平成 11 年 12 月９日（木）に芝パークホテルにて開催し、一般参加と委員等を含めて総数 112 名の参加をえた。本ワークショップにおける発表および討議の概要については、Ⅲ編Ａに記録がまとめられている。

Ⅱ．本編

2．エージェント技術の動向 －経緯と現状－

2. エージェント技術の動向 －経緯と現状－

本章では、エージェントの技術的側面について、応用的視点にたって動向報告を行う。2.1節の「ビジネス・産業応用」では、エージェントのビジネス・産業応用として有望であると考えられる８つの領域（企業情報システムと知識管理、EAI（Enterprise Application Integration：企業間アプリケーション統合）、電子商取引、加工装置制御システム、ITS（Intelligent Transportation Systems：高度道路交通システム）、デジタル放送とエージェントテレビ、情報家電、サービス連携）について、技術動向をまとめる。2.2節の「インタフェースエージェント」では、知識メディアとしてエージェントと捉える立場、心をもつ機械としてエージェントを捉える立場、個人化と擬人化という視点からエージェントを捉える立場からの報告を行う。2.3節の「プラットフォーム」関連の技術動向では、大規模広域分散システム環境 System ν、現実への利用を指向した e-Marketplace、マルチエージェント協調プロトコル記述 AgenTalk、ソーシアルウェア構築用プラットフォーム Shine、移動エージェントシステムによるアプリケーションの構築、パターンに基づく移動エージェントシステムの設計手法について報告する。2.4節の「標準化」では、FIPA と ADSS について報告する。

2.1 ビジネス・産業応用

2.1.1 企業情報システムと知識管理(Knowledge Management)

この章では、企業の情報システムと、その中でも特に知識管理（Knowledge Management）へのエージェントの適用について考察する。章の後半では知識管理を含む企業情報システムへのエージェント適用の事例をいくつかあげた。

（1）考察

エージェントは、企業情報システム（企業間の情報システムでも本質的には同じであると考えている）において現在求められている一つの重要な課題に対して、そのソリューションを与えると考えている。ここでいう現在の企業情報システムにおいて解決すべき、重要な課題の一つとは以下のようなものである。

「別々に作られてきた既存のシステムを、 既存のシステムをそのまま使いながら、統合したい。」

統合するための一つのアイデアとしては、情報をすべて一箇所に集めてしまう、あるいは全社のすべての部署に全く同じシステムを入れてしまう集中/均一のソリューションというべきものがある。しかしそれでは、昨今の変化の激しい企業環境には対応しきれない。例えば以下のような要件に対応できない。

a) 会社間の合併/吸収による異なるシステムの接続の必要性
b) 異なる部署には異なる要件があるためにそれぞれに最適なシステムが異なること
c) 個別の部署で異なる管理・運用ポリシー
d) 他社、他部署との差別化

このような要件を満たしながら、なおかつ統合を実現するためには以下のような分散、異種を許したまま（企業内のそれぞれのシステムが定義する）情報源を統合する「仮想情

報源統合」というべきソリューションが必要になる。

e) 分散して自律的に管理されている情報源

f) いろいろなレベルで異種である情報源

異種のシステムを接続するためのソリューションとして EAI (Enterprise Application Integration) [1] が最近注目を浴びている。EAI については次の章 2.1.2 で詳しく述べられるが、EAI とは各種アプリケーション（データベースや ERP パッケージなど）に接続アダプタを用意して、EAI ツールで用意するデータ配信用ルーティング機能、データ変換機能により、業務プロセス、業務アプリケーション、データなどの既存資産をそのままで統合することを実現するものである。EAI によるデータや情報系の業務アプリケーションの統合は企業情報システムにおける一つのソリューションを与える。

章 2.1.2 でも述べられているように EAI 実現の一つの手段としてエージェントがあり、EAI によるソリューションとエージェントによるソリューションは必ずしも独立ではない。しかし企業情報システムにおいて、エージェントによるソリューションは、エージェントによらない一般の EAI によるソリューションにはない強みを与えると考えている。

それは一般の EAI のツールが潜在的に集中管理である点に対してである。一般の EAI のツールでは基本的に中央でルーティングやデータ変換などの要素をコントロールするスタイルであり、中央で連携の仕方なども決めていってしまう。個々の部署の管理・運用ポリシーは中央が対応しない限り、反映されない。このような EAI では e) の管理の分散の要件に関して対応しているとは言いがたいと考えている。

EAI のソリューションで要件 f)中の異種プラットフォームの連携に関してはかなりの範囲で対応できる。しかし各情報源が扱うデータの用語レベルで異種性を持つ情報源の統合に対してはデータ変換機能だけでは足りず、オントロジ（メッセージに用いられる用語と用語の意味の集合。辞書のようなもの）を明確化し、それらのオントロジ管理も考える必要がある（それらの必要性については後で詳しく述べる）。

以上の考察のもと、上記の e) と f) の二つを許しながらの統合を実現するためには以下のような技術が必要であると考えている。

1) 情報源の分散に対処するためには、仲介することが本質である。またその仲介を実現するためには、分散して個別に運用管理されている各情報源が自律的にその情報源の状況を知らせる必要がある。従って「仲介」や「自律性」の技術は必須である。

2) 既存のシステムをそのまま使いながら情報源の異種性に対処するためには、情報源を構造化されたメッセージをやり取りする自律性を持ったモジュールとしてラップ（wrap）することが必要である。さらに構造化されたメッセージの意味を確定するためにはオントロジが必要になる。従って「ラッピング（wrapping）」や「オントロジ」の技術は必須である。

まとめると分散/異種の情報源に対処するために「仲介」、「自律性」、「ラッピング」、「オントロジ」などの技術が必須になる。これらを兼ね備えるのはエージェントである。たとえそのソリューションを別の名前で呼ぶことがあっても、そのソリューションが兼ね備えるべき性質により、その本質はエージェントそのものであると言える。

企業情報システムの中で、エージェントの「ラッピング」の技術はエージェントシステム導入の前提として、特に EAI と異なり、なおかつ重要な部分は「自律性」、「オントロジ」

に基づいた「仲介」であると考えている。

「オントロジ」に関しては、オントロジの明確化、オントロジ変換およびオントロジ管理が必要である。起源の異なる情報源においてはどうしても使われる用語が異なる。それらの異なる用語を使う情報源の連携を実現させるためには互いの用語間の対応付けが必要である。それにはまず対応付けるそれぞれの情報源の用語がはっきり定義されている、すなわちオントロジとして明確になっている必要がある。そうして初めて各情報源の管理者が自分で管理する（従ってよく理解している）情報源のオントロジと他のオントロジを正確に対応付けることができる。このオントロジ間の対応付けによるオントロジ変換によって各情報源はメッセージのやり取りが可能になる。明示的で構造を持つオントロジは、オントロジの対応付けやメッセージの機械的なチェックを可能とする。オントロジ変換もプログラムコードなどに埋め込まれた形ではなく、対応付けを情報として明確化することで、オントロジに変化があった時にも容易に対応付けを入れ替えることができる。そのようにしてできた複数のオントロジやオントロジ間の変換の情報を管理する必要もある。

またオントロジの明確化により、エージェントが自動的に処理できる部分が広がることもメリットとなる。特に仲介などにおいて宣言的な情報とオントロジを組み合わせたサービスの提供が考えられる。

また「自律性」においてはエージェントが自律的に自分の状態を宣伝する（advertise）する行為が重要である。分散管理される情報源においては、情報源の状態はそれぞれの運用管理の主体によって変えられていく。そのような情報源たちの自動的な「仲介」を実現するためには、適切なオントロジに基づいた宣伝が必須となる。

さて企業情報システムの中でも特に知識管理（Knowledge Management）で発生する要件に関しては次のように考えている。

参考文献〔2〕の「知識創造プロセスと知識資産活用の創造的循環」では、「目的・課題」、「知識創造プロセス」、「知識資産のマネジメント」、「価値」の間を以下のようにつないでいる。

1）知識資産の開発・調達・維持
2）知識創造プロセスのリード
3）知識資産の共有・活用
4）知識資産の蓄積
5）知識資産からの収益創出
6）イノベーション・問題解決・知識提供による収益創出

エージェントは全ての要素に関わる可能性はあるが、まず「知識資産のマネジメント」に関わる3)、4）においてその「知識資産のマネジメント」のベースになると考えている。

比較的近い将来の企業内では、まず電子的な文書が大量に一次的な知識として蓄積されていく。それと並行して徐々にそれらの電子的な文書を文脈の中に位置付けるメタ情報（著者、作成日時などから始まって、文書の目的や他の文書との関係など）が二次的な知識として蓄積されていくようになると考えられる。そのような環境の中でエージェントの役割は、個々の部署で蓄積・管理する知識をエージェントとしてラップし、主に知識に関するメタ情報を交換し、それらメタ情報に基づきエージェントが仲介することによる知識資産の蓄積・共有・活用の実現であると考えている。

その他に知識管理で特に重要になる可能性があるのは、「知識創造プロセス」でのインタフェースエージェントによるサポートである。インタフェースエージェントには、ユーザのストレスなくユーザの必要とする知識を提供し、時にはユーザが明示的に求めなくとも積極的に（しかしユーザの思考を妨げることなく）情報を提供することが求められる。そのためにはあらゆるモードを通じてユーザの状況を把握し、ユーザとの非常に高度なレベルのインターアクションをすることを要求されるであろう。また上であげた「知識資産のマネジメント」のベースになるエージェント系とのシームレスな連携も必要になる。

同様のことは「暗黙知（tacit knowledge)」と「形式知（explicit knowledge)」の枠組みでも言える。参考文献〔3〕では知識創造を、個人個人が持っている暗黙知を共有し（共同化)、それら暗黙知の言語（コンセプト）化により形式知とし（表出化)、そうして新たに得られた形式知と既存知識を組み合わせてさらに別の形式知を得（連結化)、その結果が個人に戻されて暗黙知として実践、体得される（内面化）という暗黙知と形式知の間のスパイラルとして捉えている。

暗黙知から暗黙知（共同化)、暗黙知から形式知（表出化）の過程においては、上記の「知識創造プロセス」でも述べたようにユーザサポートの面でインタフェースエージェントが重要な役割を果たすと考えている。

形式知から形式知（連結化)、形式知から暗黙知（内面化）の過程においては、蓄積された形式知や知識に関するメタ情報を交換し、それらの知識に基づく仲介をエージェントが行うことにより実現される形式知の共有・流通が重要な役割を果たすと考えている。

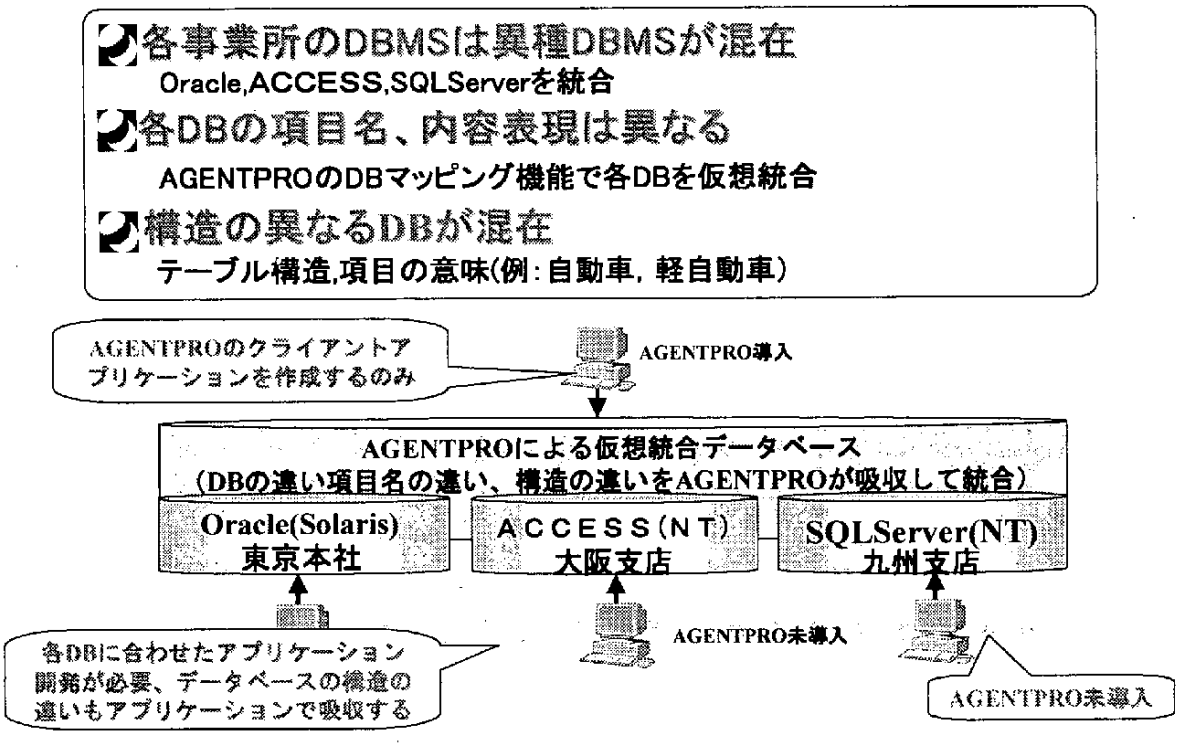
（2）事例

この節では富士通のエージェント製品である AGENTPRO の具体的な事例などを通じてエージェントシステムの企業情報システムへの適用を紹介する。（AGENTPRO の説明については「A. 研究開発への取り組み事例」の「A.17 株式会社 富士通研究所 知能システム研究部」の章を参照） 各事例について主にユーザ要件、エージェントによる仮想情報源統合ソリューションの適用の仕方、およびそのメリットなどを述べている。ここで述べた事例では（1）であげたエージェントの技術で述べた「仲介」、「自律性」、「ラッピング」、「オントロジ」の技術を用いて実現されている。

事例 1） 情報サービス系企業「仮想統合データベース」

ユーザ要件は各拠点に散在したデータを横串で検索したいというものである。各拠点は ACCESS を利用しているシステムが多いが、その他の DBMS も使われている。

この場合は図表 2.1.1-1 にあるようにエージェントシステムでの統合が可能である。ラッピングの技術で異なるプラットフォームをエージェント化し、オントロジ変換の技術で情報源のスキーマの違いを吸収し、さらに自律的な宣伝メッセージを用いて仲介を実現している。エージェントシステムで統合するメリットは図表 2.1.1-1 にある、DB の違い、項目名の違い、構造の違いを吸収して統合ことに加えて、既存システムをそのまま利用しつづけられ、そのままの運用で、新規 RDB の追加も動的に可能であることである。



図表 2.1.1-1　事例 1) 情報サービス系企業「仮想情報統合」

事例 2)　富士通「全社 SE 情報共有システム」

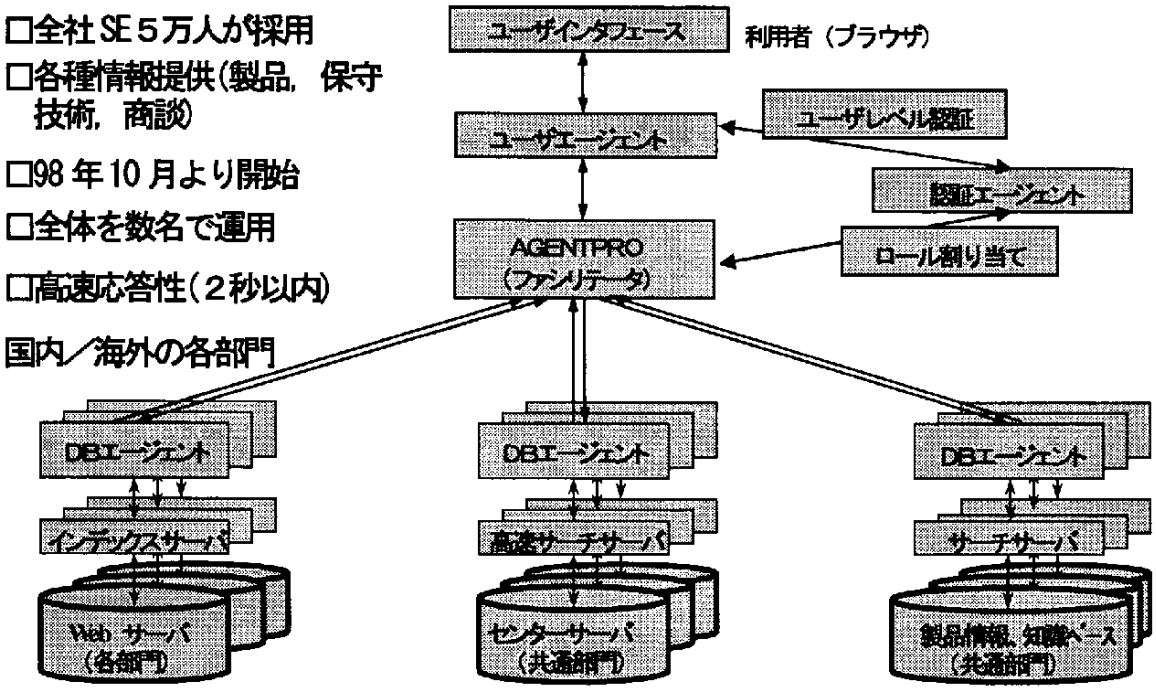
この場合の要件は各部署で管理している文章情報を中央で管理する文書情報とともに横串で検索したいというものである。しかし統合を実現する一方で各部署での運用をあまり変更したくなく、また運用コスト(運用者など)もあまりかけたくないということもある。

それらの要件に対してできた富士通「全社 SE 情報共有システム」では、図表 2.1.1-2 にあるようにエージェントシステムで統合している。ラッピングの技術で異なるプラットフォームをエージェント化し、情報源のメタ情報に関するオントロジを整備し、そのオントロジに基づく自律的な宣伝メッセージを用いて仲介を実現している。適用では、情報検索(URL や要約)にエージェントシステムを用い、実際の文章等は URL により直接アクセスする形態をとる。エージェントシステムで統合するメリットは、他の部署が後からデータベース (DB) エージェントを立ち上げて、動的にこのシステムに加入することができることである。これにより急速に参加する部署を増やしていくことができた。

事例 3)　旅行業者「旅行情報・商品情報提供システム」

ユーザ要件は旅行業者の各店舗などで、お客さまからのリクエストに基づき、商品を検索し、必要ならばその場で商品等の購入を実施したいというものである。例えば、旅行なら、JR チケット、航空券、旅館、ホテル等の予約をブラウザからオンラインで実施するような場合に、各社の商品を見比べてから、ホテルなどの空き状況をもとに予約をする際に適用するものである。店舗などで利用するために結果の即時性が要求される。

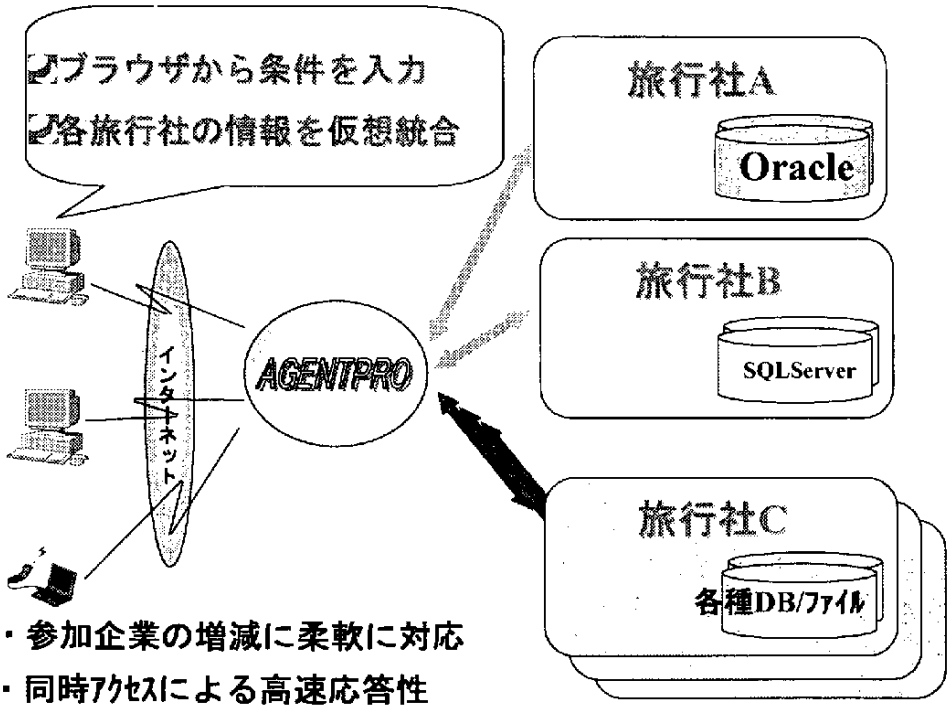
この場合も図表 2.1.1-3 にあるようにエージェントシステムでの統合が可能である。ラッピングにより Web 情報をエージェント化し、自律的な宣伝メッセージに基づく仲介とオ



図表 2.1.1-2　事例 2)　富士通「全社 SE 情報共有システム」

ントロジ変換によって仮想情報統合を実現する。ユーザのお客さまは各社が提供している Web 情報をエージェントシステム経由で検索し、必要な商品情報を見つけ出す。商品が決定したら、その場で必要な情報に対し、予約などの更新をかけることも可能である。

エージェントシステムで統合することのメリットは、即座に情報収集と予約などの更新が可能なことや、情報収集においてオペレータが一つ一つ URL で各情報を検索せずとも、集まる点である。



図表 2.1.1-3　事例 3)　旅行業者「旅行情報・商品情報提供システム」

参考文献

〔1〕　“解説●アプリケーション間連携を自動化する EAI ツール”、日経オープンシステム 1999 年 11 月号、 pp. 124 – 127、 (1999) .

〔2〕　野中 郁次郎、紺野 登、西 高弘、西村 務、山名 一郎:“ナレッジ・マネジメント 実践の技術”、リック、 (1999) .

〔3〕　野中 郁次郎、竹内 弘高:“知識創造企業”、東洋経済新報社、 (1996) .

2.1.2 エージェント技術の EAI(Enterprise Application Integration)への応用

本節では、企業間アプリケーション連携　EAI (Enterperise Application Integration) へのエージェント技術の応用について報告する。

(1) 企業間アプリケーション連携　EAI　導入前の課題と背景

図表 2.1.2-1 は、EAI 導入前の顧客支援（カスタマーサポート）業務における従来のシステム事例を示している。この事例では、Ａ社とＢ社は、電子機器の保守修理というカスタマーサポートサービス（以後ＣＳと略記）を依頼する側（Ａ社）とそのサービスを提供する側（Ｂ社）である。また、Ａ社とＢ社は、それぞれ、Ｘ社、Ｙ社のＣＳ業務アプリケーションパッケージソフト（以後単に業務アプリケーションと略記）を利用し、社内業務の効率化を図っている。Ａ社の情報機器・通信機器の管理部門はＸ社製の業務アプリケーションを用いて社内のこれらの機器を管理し、その保守修理をＢ社に依頼している。Ｂ社は、Ａ社を含む複数の会社に対して、こうした情報機器・通信機器の保守修理のサービスを提供し、これをＹ社製の業務アプリケーションで管理している。

この事例における問題点は、両社で情報機器・通信機器の管理・保守修理のための情報が電子化されコンピュータで管理されているにも関わらず、使用している業務アプリケー

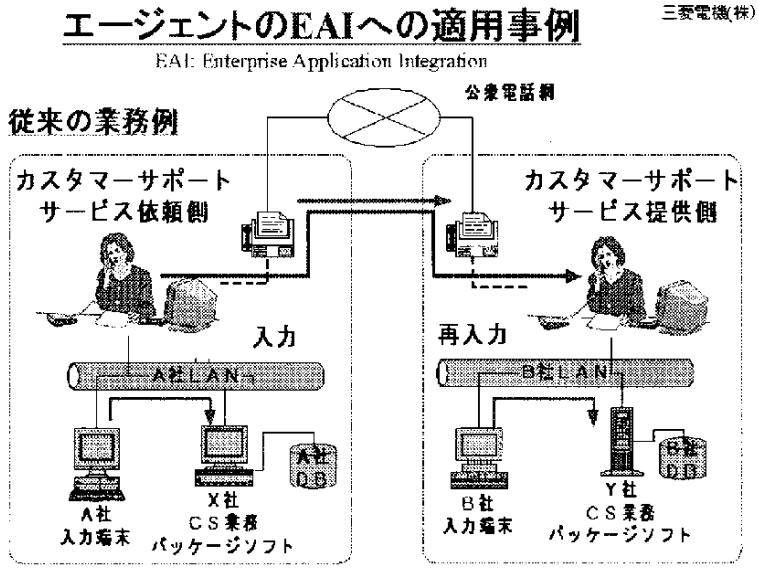


図 2.1.2-1　EAI のツール構成

ションが異なることから、そのデータを相互に利用することができていない点にある。結局、両社間での機器の保守修理依頼・受付け業務は電話やFAXによって行われ、A社側には「修理の依頼、状況確認に時間がかかる」「修理依頼後のサービスの進捗状況が依頼側にわかりにくい」といった問題を生じ、B社側には「事前確認ができていれば、準備できたはずの保守器材の準備ができなかった／省力化できたはずの保守作業を省けなかった」といった機会の損失を生じていた。

次に、両社の業務アプリケーション連携が実現できていない背景について分析する。

ERP（Enterprise Resource Planing）やSCM（Supply Chain Management）と呼ばれる多くの業務アプリケーションでは、業務データへのビューを提供し、そのビューの組み合わせにより、データの業務上の関連付けを行うことで、さらには、その関連したデータを「業務知識」と関係づけて処理を行うことで、利用者を支援するところに大きな特長がある。各アプリケーションのデータは、そのビューに添う形で、それぞれのデータが保管され、システムが構築されている。このため、A社とB社間で単にファイル転送（データ転送）が行われるだけでは、データの相互有効利用は期待できない状況にある。

また、業務アプリケーションを導入した各社は、導入後、それぞれの事情に応じたシステムのカスタマイズを行う。例えば、「以前から使っていたコンピュータとの接続・ソフト資産の活用を図るために、一部のデータ形式を自社向けに変更する」、「会社の組織に合わせて業務アプリケーションが供給する標準的ビュー／業務フローに変更を加える」などのカスタマイズが行われる。このため、同じベンダーの業務アプリケーションを使用していても、ファイルを転送するだけでは、相互のデータ連携はとれない場合が多い。

さらに、それぞれの会社には、他社には見られたくない情報（例えば、取引の割引条件や、関連顧客の情報など）があり、これらが業務アプリケーション中に入力されている。こうした社外に流出してはこまる情報が、会社間で交換されてしまわないようにデータを分離する作業／取引相手に応じて転送内容を選別する作業が必要となる。

(2) EAIソフトの要件

前節で紹介したような背景に加えて、会社を越えた業務アプリケーション連携 EAI を実現するには、次の技術的課題についても答を準備しておく必要がある。

a) 連携に要する通信コストを抑え、同時に、セキュリティを確保すること
b) 交換するデータ形式を標準的に定めること
c) カスタマイズが行われた2社間での連携は個別開発となりがちであるため、そのソフト開発に必要な労力と費用を抑えること

これらは、従来、業務アプリケーション連携が、一部の大型システム同士の連携に限られてきた事情をよく説明している。しかし、これら課題のうち a) 項については、インターネットの普及と暗号・認証を含むインターネット利用技術の進歩により、いくつかの手法が利用可能となってきた。また、b) 項についても、XML技術の標準化と普及により、データの交換形式を標準的な仕様に基づいて定める事が容易になってきた。さらに、米国を中心に、業務アプリケーション導入の重要性が認識され、ERP、SCMといったソフトウェアがパッケージソフトとして企業に導入されるに従い、c) 項の個別開発の部分を少なくする／省力化することも可能となってきた。

そうした状況から、業務アプリケーション連携のためのシステム構築支援ソフトウェア

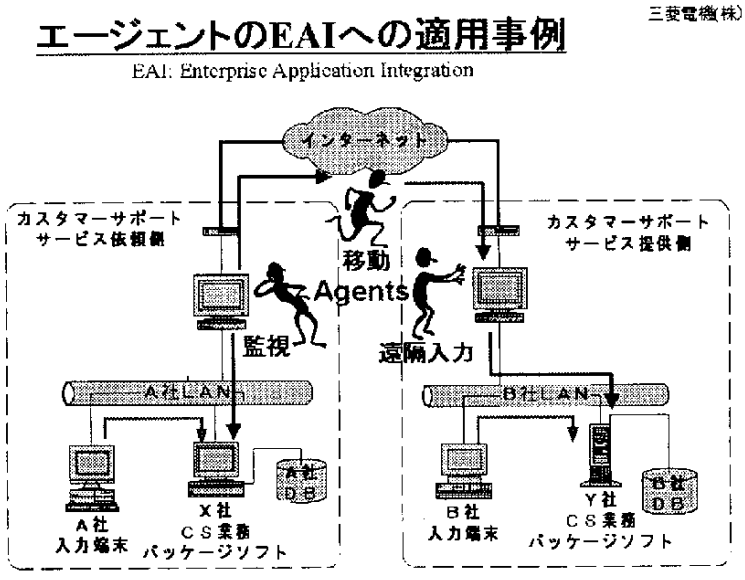


図 2.1.2-2　EAI エージェント技術の適用

が市場に登場してきており、米国では、CrossWorlds Software 社の CrossWorlds Series[*1] をはじめとして、webMethods 社の“webMethods B2B[*2]”、Active Software 社の"ActiveWorks[*3]"などがの製品がある。また IBM 社の“MQ Series[*4] Workflow”も MQ Series[*4] ソフトの EAI 対応版と位置づけることができる。EAI 分野へのエージェント技術適用という点では、三菱電機の“MELBA[*5](Multi-Enterprise Links By Agents)”があり、また、東芝の“Bee-gent[*6] (Bonding and Encapsulation Enhancement Agent)”も EAI 用途を指向したエージェントシステムである。

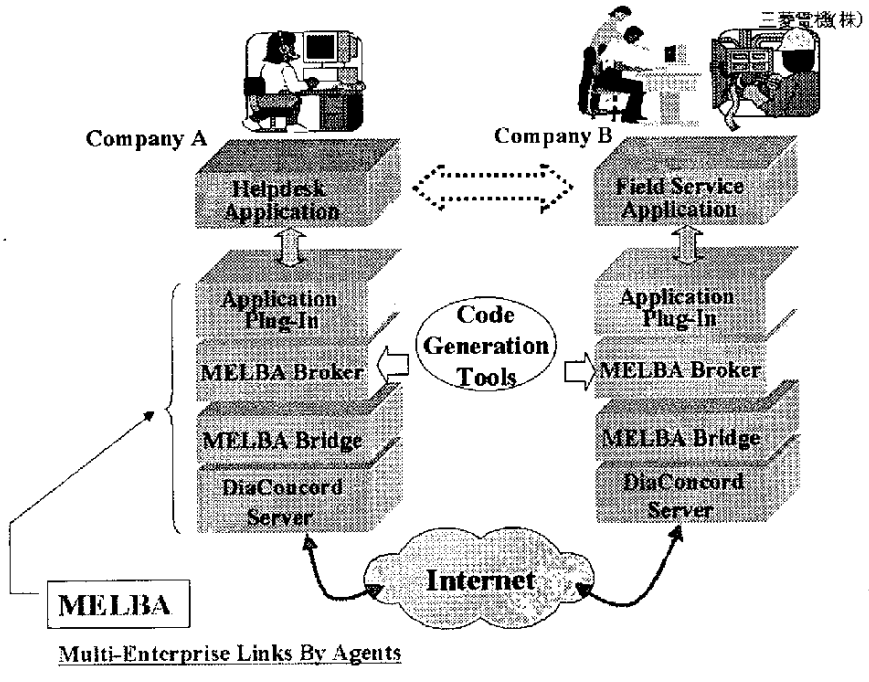


図 2.1.2-3　MELBA の構成

(3) エージェント技術のEAIへの適用事例　"MELBA"

ここでは、先に紹介したMELBA (Multi-Enterprise Links By Agents) を例にエージェント技術のEAIへの適用事例を紹介する。MELBAは、会社間の業務アプリケーション連携の実現を容易にする／短期でのシステム構築を可能にすることを目的として開発されたシステム構築支援ソフトウェアである。上述した業務アプリケーションの相互利用を阻んでいた要因を検証し、従来人手で行われた作業を図表2.1.2-2で示すように、複数のエージェントによって肩代わりさせることで、EAIシステム構築を実現する。エージェントによって代行される作業には、依頼業務の発生検知・状態監視、伝送すべきデータの選択、データ形式の変換・共通化、認証・暗号を利用したデータ転送、相手側業務アプリケーションへの入力作業などがあげられる。

図表2.1.2-3は、業務アプリケーションとして、A社がHelpdeskアプリケーションを、B社がField Serviceアプリケーションを使用した場合のMELBAによる業務アプリケーション連携システムの構成事例を示している。MELBAは、アプリケーション毎にApplication Plug-Inと呼ぶアクセスインタフェースを供給する。DiaConcord Serverは、JavaVM上で動作する移動型エージェントの実行・管理環境である。また、MELBA Bridgeは、このDiaConcordとMELBA Brokerのインタフェース部である。これら移動型エージェント構成要素に関しては、本節の最後に詳しく説明する。

MELBA Broker部は、

a) アプリケーションへのアクセス規則に則って、データアクセス手順を定め、Plug-Inを通してアプリケーションデータの入出力を実行する

b) データの変換規則に則って、データ形式の変換を行う

c) 交信相手毎に異なるデータ処理と例外処理を記憶し、実行する

といったMELBAの要となる「移動しないエージェントソフト」である。このMELBA Broker部は、ユーザ毎・アプリケーション毎に個別の開発が必要となり、従来の開発手法では実用化の点に課題があった。そこで、MELBAでは、MELBA Broker（常駐型エージェント）とその規則を生成するCode Generation Toolsを用意し、MELBA Brokerの短期開発を可能にしている。

Code Generation Toolsの概要を、図表2.1.2-4に示す。

Code Generation Toolsには

1) ビジネスルール記述（例外的処理・変換手順なども記述される）

2) データの変換規則の記述

3) XML化のためのDTD (Data Definition Table) 記述

などの生成機能が含まれている。

これら規則の記述とエージェント処理コードとを一元的に扱えるよう、MELBAでは、システム開発者向けの中間言語Hermを定義し、Herm Compilerを通して、MELBA Brokerコードと規則記述を自動的に生成する仕組みを提供している。また、現在は、Herm中間言語による記述を必要としない、Herm GUI Toolも開発中である。

以後、EAIの各部分でのエージェントの役割を、図表2.1.2-5、6を使って、紹介してゆく。まず、A社側の動作について図表2.1.2-5を用いて説明する。

・A社側では、まず、B社へ通知すべき事象（情報）の発生検知が必要である。「事象の検知」方法としては、a）エージェントが特定のデータの組み合わせを監視し、事象の発生を検知する方式と b）業務アプリケーションから、事象の発生を知らせる方式とがある。使用中の業務アプリケーションに変更を加えたくない／業務アプリケーションそのものをアウトソーシングして利用しており、自社では変更を加えられないというような場合に a）の方式が有効である。

三菱電機(株)

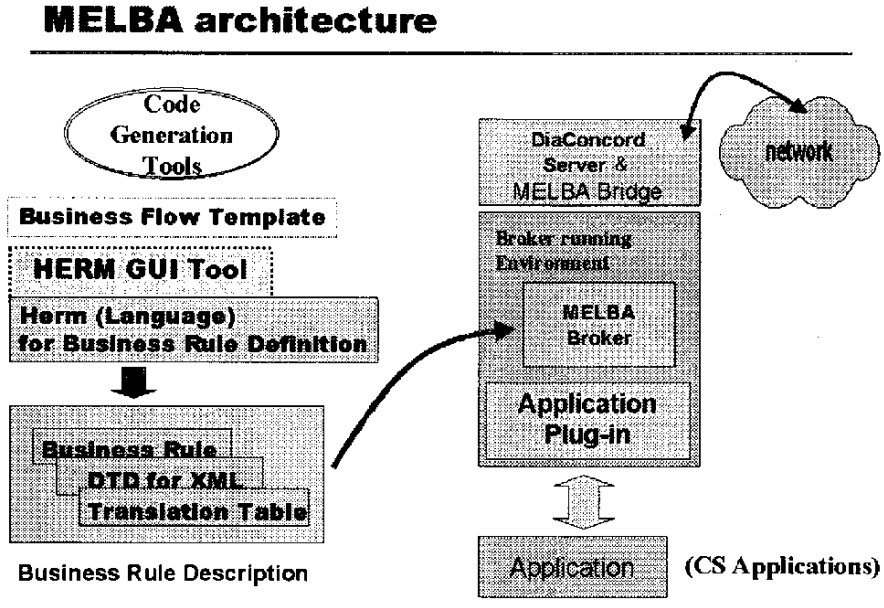


図 2.1.2-4　MELBA のツール構成

三菱電機(株)

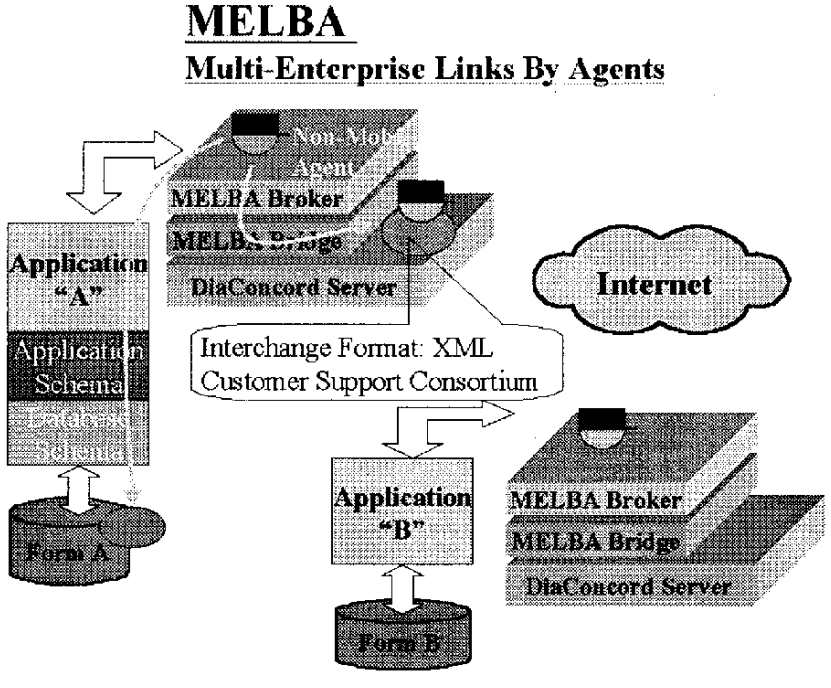


図 2.1.2-5　MELBA の動作（1）

・次に、A社業務アプリケーションからのデータの読み出しを行うが、業務アプリケーションは、そのビューによるデータの関連付けに特長をもっており、直接DBをアクセスしてデータを読み出すことには益が少ない。ここでは、MELBA BrokerがA社業務アプリケーションのユーザと同じ手順で、ビューに従ってデータを読み出す。

・読み出されたデータは、B社向けのデータ形式への変換が必要となる。近年データ交換形式の標準化が進み、XML（eXtensible Markup Language）に基づくデータ交換が推進されている。カスタマーサポート業務分野では、米国 Customer Support Consortiumが、カスタマーサポートアプリケーション間の交換データ形式についての標準化作業（XML 化）を行っており、こうした業種毎に定められる標準的なデータ交換形式を使用することで、Extranetにおける EAI が推進されると考えられる。よって、MELBA Brokerは、記述された変換規則に則って、A社業務アプリケーション内のデータを XML 形式のデータに変換することになる。

次に、図表 2.1.2-6 を用いてデータの転送を受け持つエージェントについて説明する。

上記の手順で XML 化されたデータは、Internetを経由してB社へと伝えられる。このデータ転送機能に要求される条件は、

a）データが相手に安全・確実に伝送すること

b）通信・データ処理に関する例外処理に柔軟に対応できること

などが上げられる。後に紹介する Mobile Agent Platform “DiaConcord” は、移動するエージェントの認証機能・暗号化機能・Queuingによるデータ再送機能等を有し、a）の条件達成に関して、システム構築・保守の負担を大きく軽減することができる。b）の例外処理に関しては、同じ例外事象が発生しても、導入されている業務アプリケーションや各社のシステム運営方針に依存する場合が多く、A社―B社といった１対の業務連携の組み合わせ毎に仕様を定める必要がある。こうした特定の組み合わせで仕様が決められる例外処理ソフトは、２社間を移動する移動型エージェントに実装することで、システムの管理・運用を簡素化できると共に、両社間の仕様・契約の変更の影響が他に及びにくいという効果がある。

次に、B社にエージェントが到着してから、B社業務アプリケーションにデータが書き込まれるまでの処理について、同じく図表 2.1.2-6 を用いて説明する。

到着した移動型エージェントが運んできたデータをB社側常駐エージェントに引渡す前に、実行優先度の判定や依頼側と間で取り決めた特別な処理作業（例えば、データ交換形式標準の中では定義されていない情報の抜き出しなど）を行う。こうした作業は、移動型エージェントが実行する場合（仕様変更・改訂が多いと予想される場合）と、B社側に常駐するエージェントが実行する場合（定型的処理の場合）があり、選択はシステム設計者の判断に委ねられる。

B社業務アプリケーションへの「データ書きこみ」の作業は、業務アプリケーションがデータの書き込みに対する制約条件／規則を設定している場合が多いため、書き込み規則に基づいたデータアクセスが必要であり、また、実行するに十分なデータが与えられているかを確認する作業が必要となる。

以上のステップにより、A社からB社へ必要最小限の情報が渡され、A社からはB社が、特定の業務に関しては、自社と同じ業務アプリケーションを利用しているかのように見え

三菱電機(株)

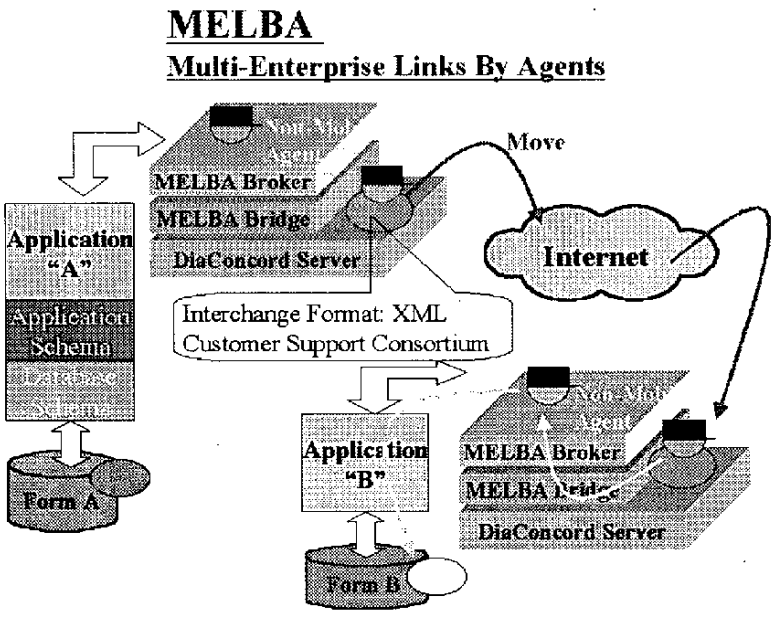


図 2.1.2-6　MELBA の動作 (2)

るようになる（またこの逆手順も同様である)。このように MELBA を使うことで、常駐エージェントと移動型エージェントを組み合わせた業務アプリケーション連携 (EAI) システムを短期間で開発することが可能となる。

Mobile Agent Platform "DiaConcord"

最後に、MELBA で採用されている移動型エージェントのプラットフォーム "DiaConcord" について、簡単にその機能を紹介する。DiaConcord は、移動型エージェントアプリケーションの開発環境・実行環境・運用管理ツールを含んだフレームワークである。DiaConcord の実行環境は、図表 2.1.2-7 に示すように、JavaVM をベースとして、その上位に位置付けられる Mobile Agent Server (DiaConcord Server) が、移動型エージェントの実行・移動を管理している。移動型のエージェントでは、Security (エージェントの認証と内容の暗号化) と Persistency (システム障害発生時にもエージェントが消滅する事態を回避する) を保証することが求められ、多くの移動型エージェント機構がこれらの機能を備えているが、DiaConcord では、さらに移動してきたエージェントの所有者毎にリソースへのアクセス制限を設定するアクセス制御機構をサポートしデータのセキュリティを高めている。

また、DiaConcord には、外部ソフト・外部機器とのインタフェースをとるための Service Bridge 機構が用意されている。Service Bridge は、いわば「移動しないエージェント」ソフトであり、Mobile Agent Server に Service Bridge を登録することにより、遠隔からの Service Bridge ソフトの管理、さらには Directory Lookup による Service Bridge の存在場所の特定などが可能になっている。この機能を、Agent Server の遠隔管理機能や移動するエージェントの追跡機能を持つ Administrator と併せて用いることにより、移動型エージェントを用いたシステム構築と保守作業を容易にしている。

DiaConcord の Security モデルは、３段階の Security 機能を備え、(1) エージェント自身を記憶装置に蓄える際の保護、(2) エージェントの伝送時の保護、(3) サーバ上の

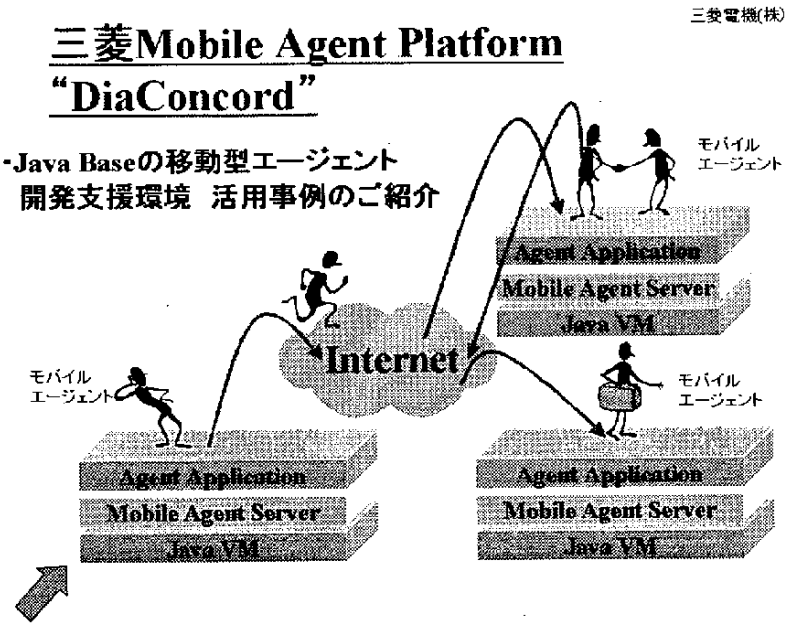


図 2.1.2-7　三菱 Mobile Agent "DiaConcrd"実行図

リソース保護の機能をもつ。エージェント伝送時には、ユーザ ID 照合のためのディジタル認証、移動中のエージェントの暗号化などが実行される。プロトコルには SSL（Secure Socket Layer）とほぼ同様のプロトコルを使用している。

DiaConcord の構成に関する、より詳細な説明は参考資料〔1〕を参照願いたい。

参考資料

〔1〕 Mobile Agent Platform "Concordia" Web site at
http://www.meitca.com/HSL/Projects/Concordia

"Concordia" は、MITSubishi Electric Information Technology Center America の米国登録商標です。DiaConcord は、同じ移動型エージェント技術に対する三菱電機（株）の商標（登録出願中）です。

*1 CrossWorlds は、米国 CrossWorlds Software Inc. の登録商標です。
*2 webMethods B2B は、米国 webMethods Inc. の登録商標です。
*3 ActiveWorks は、米国 Active Software Inc. の登録商標です。
*4 MQ Series は ＩＢＭ Corp. の登録商標です。
*5 MELBA は、三菱電機（株）の登録商標です。
*6 Bee-gent は、（株）東芝の商標（登録出願中）です。

2.1.3 EC(電子商取引)

この章では、EC (Electronic Commerce, 電子商取引) におけるエージェント技術の必要性と EC にエージェントを適用する際に重要となる技術項目を検討する。

(1) EC の現在と将来

EC では、EC での取引の参加者のタイプ (区分け) により、取引を B2B (Business to Business)、B2C (Business to Consumer)、C2C (Consumer to Consumer) といった分け方をするのが一般的である。

EC の経済活動全般に占める割合はいまだ小さい部分もあるが (たとえば B2C の部分に相当する部分はアメリカでの小売業の総売上の１％程度〔9〕)、急速に成長してきており、今後も急速な成長が見込まれる。

EC の急速な成長が見こまれる理由にはまず単純にビジネスモデルをインターネットに持っていっただけでも、取引などのコストが大幅に下がったり、多くのことが自動化されたりするメリットがある (航空券販売の処理やインターネットバンキングのコストの例を参照〔8〕)。

しかし単にコストの問題だけではなく、EC ではインターネット上で全世界を対象に電子的に行うということにより、いろいろな新しい形態のビジネス・サービスが可能になり、実際に新しいビジネスやサービスが次々と生まれている。〔1〕

また EC ではやり取りを電子的に行うことにより、顧客の情報も電子的な形で大量に収集することが可能になり、CRM (Customer Relationship Management) への展開も見ている。これは社内システムの顧客へ向いた方向への変革をもたらし、同時に CRM に基づいた新しいサービスの提供なども始まっている (CRM の場合、顧客とやり取りするモードは電話、ファックスなどインターネットに限らないが、徐々にインターネット (電子メールや Web) にその重点を移している)。

ただ EC ですべてがばら色というわけではない。まだまだ新しい分野であり、EC の入口から出口までいろいろな問題が起きている。

例えば入口ではつい最近のこととして、(商品の人気のせいでもあるが)新たに発売されるゲーム機の販売予約を Web のページで受け付けたところ、アクセスが集中して、予約ができないようになった。

また出口では、実世界との連携 (bit の世界から atom の世界への連携) が重要であり、例えば受け付けた注文はもちろんその注文のとおり処理 (order fulfillment) する必要がある。ところが 1999 年のクリスマスには EC に参入したおもちゃのチェーン店にクリスマスプレゼントの注文が集中して、クリスマスプレゼントの配送をクリスマスに間に合わせられない事態に陥った。

さらにインターネットではどんな商品かの把握が難しいので返品率が高いという問題もある。 現在のネット通販の返品率はカタログ販売の倍程度である。

しかし EC はその成長の過程でこのような問題、あるいはその他の問題点や課題を抱えながらも、それを解決しながら確実に大きくなっていくであろう。

それはまず先にあげたメリットや EC の将来性にかけてより多くの参加者が自ら EC に参入すると考えられるからである。また同時に今後いろいろな国や機関が 〔10〕 にある

ようなポリシーにより、Digital Divide をなくし、あらゆる人がインターネットへのアクセスを持ち、 Small Business でも容易に全世界的な市場にアクセスできる世界[1]が実現していくと予想されるからである。

そのような世界では、EC の市場のサイズはもちろんのこと、EC への参加者の質的な異種性も現在とは比較ならないほど大きくなる。

一般的に B2B の方が B2C より 1 桁程度大きな規模を持っており、近い将来 (少なくとも数年程度) もその比率で発展すると言われている。しかし個人とも企業ともつかない Small Business も大量に参入してくる "Digital eQuality" と呼ばれる世界が実現されたならば、B2B, B2C, C2C などといった取引の参加者のタイプによる明瞭な区別はなくなると思われる。

その中では Small Business, Medium Business を含む個人から大企業までの EC の全ての参加者を連続したスペクトラムとして捉えるような一般的な取引の枠組みを必要とするであろう。その枠組みの中ではたとえば参加者の規模の違いはたまたま一つの属性が違っていることで扱われるであろう。[2]

例えば〔7〕にある "e-commerce inversion" といった B2B と B2C を分けた議論は意味を失い、B2B, B2C, C2C といった区別を持たない "e-commerce convergence" といった世界がくると考えられる。そして〔7〕では、B2B と B2C に分けて、B2C の課題として記述されているユーザの多様かつ曖昧な要件への対応も EC 全体の課題となるであろう。[3]

EC における主に現在の B2C の場合にあたる購買行動に関しては、その段階が以下のように分類されるという。 〔6〕

1) ニーズの把握 (Need Identification)
2) 購入商品の斡旋 (Product Brokering)
3) 購入先の斡旋 (Merchant Brokering)
4) 交渉 (Negotiation)
5) 発注・配送 (Purchase and Delivery)
6) 顧客サービス・評価 (Product Service and Evaluation)

参考文献〔6〕でも述べているように、これらのステップは現在でもいろいろ絡み合っている。しかし将来の EC を考えていく際には、もう一歩進んでこのような分類ももう一度再構成される必要があると考える。

参考文献〔2〕でなされた指摘のように商品やサービスがパッケージ化され経済のアンバンドリング化 (unbundling) が進み、一方でそれをもう一度パッケージ化する editing 機能 (ある種の仲介) が一般的になるであろう。特に EC のような電子的な世界では editing が容易になり、また EC ならではの新たな editing の形式が誕生する可能性がある。

このような世界で 1) -4) は複雑に絡み合うであろう。 誰がどのような条件でどのようなものを売れるかが問題となり、特に交渉を通じた動的な editing (動的な商品構成、dynamic product configuration) が関わるなら、2) -4) は一体となる。1) に関しても例

---

[1] 〔10〕ではそれをタイトルにもあるように "Digital eQuality" と呼んでいる。

[2] これは必ずしも全ての参加者を平等に扱うということではない。同じ枠組みの中で、対象者によるいろいろな属性により差をつけたきめこまやかな対応が必要になるということである。

[3] その場合、ユーザ個人としての要件多様性だけではなく、ユーザのタイプによる要件多様性も加わることになる。

えば 2) -4) の過程で新たなニーズが醸成されたり、当初のニーズが変更されたりする可能性がある。

実際の買い物を考えて見ればわかるように、我々の行動は組織化されておらず、ニーズも曖昧である。必ずしも EC が実際の買い物に全てとって代わる必要はないが、いままではこれらが絡み合った取引が複雑で実現が難しいために EC ではもっぱらこのような段階が成立していたのであれば、その実現も EC の将来の課題であり、質的な複雑性をはらむこういった課題にも対処していく必要があると考える。

以上 EC の現在と将来について概観したが、まとめると以下のように考えている。

a) EC はいろいろな問題を抱えながらも確実に大きくなっていくであろう。
b) EC への参加者の質的な異種性も現在とは比較ならないほど大きくなるであろう。それにともない B2B, B2C, C2C などの区分けも消失するであろう。
c) EC の取引はより複雑かつ動的なものになっていくであろう。

(2) EC におけるエージェント技術の必要性

将来に向けてエージェント技術は（もちろん全てではないが）EC におけるかなり重要な大きな部分を担うと考えている。エージェント技術は EC の参加者に対しては、そのままでは有効に使うことが難しくなるであろう EC を容易に、また参加者に合わせてパーソナライズした形で使うことを可能にするであろう（この章の技術要素「インタフェースエージェント」の説明を参照）。またサービス提供者にとってはサービス提供や新たなサービスを容易に構築するベースになるであろう（この章の技術要素「信用」、「仲介」、「交渉」の説明を参照）。ちなみに「オントロジ」や「XML への対応」は全体を支える技術要素になるであろう。

前の節で述べたように、将来に向かって EC はサイズ、参加者の質的な異種性、取引の動的性や複雑性とも今より格段に大きくなると考えられる。

その世界の中で、EC の参加者は買う側も売る側も将来の EC の巨大なサイズ、異種性、動的性、複雑性[1]に対応しなければいけない。その対応はそれらの巨大さにより人間には難しい、あるいは不可能なこととなる。対応するためにはソフトウェアでのサポートは必須である。

この節でのここからの論理の展開は、そのサポートするソフトウェアの持つべき性質がエージェントの目指すものと一致するということによりエージェント技術が将来の EC において重要な位置を占めるというものである。

そのソフトウェアは人間の要望を確実に把握し（必要ならば繰り返し人間とのインターアクションを持つことにより）、その要望を EC の巨大性の文脈において解釈し、実現する必要がある。そのためには積極的に権限をソフトウェアに与える必要がある。ソフトウェアに権限を与えることと、細かい判断までしなくてはいけないという負担は互いに 相反（trade-off）の関係にある。さらに判断する権限をソフトウェアに委譲するためには、ソフトウェアの方にはそれ自身で判断を行っていく自律性が必要になる。

しかしユーザの意図の適切な解釈や、それに基づく判断は必ずしも一度でうまくいかな

[1] この「将来の EC の巨大なサイズ、異種性、動的性、複雑性」を以下単に「EC の巨大性」と呼ぶ。

いため、ユーザからのフィードバックを受けての学習が必要になる。もちろんソフトウェアに対して絶対にしてはいけないことや、簡単な制約などは事前にユーザが指定することができるようになっているべきではあるが、将来の EC で存在する全ての可能性に対して指定するのは現実的ではなく、おそらく不可能である。ここに学習の必要性がある。

もちろん EC のそれぞれの参加者をサポートするソフトウェア同士が互いに協調・競争するためにやり取りができなければ、自動化が必要な上記のようなサポートの実現はありえない。

従って考えているソフトウェアは、人間の意図と EC の巨大性への対処を結びつけるために、それ自身で解釈や判断を行わなければいけない点で自律性を、その解釈や判断を適切なものにするために学習を、協調や競争を可能にするためにソフトウェア間の連携を必要とする。

このような要素をもったソフトウェアとは一般的にエージェントと呼ばれているものそのものである。(例えばこれら三つは〔5〕であげられているソフトウェアエージェントの要素としてあげられている Autonomous, Learn, Cooperate と一致する。)

以上のように将来の EC の中で必要とされるエージェント技術であるが、その中でも特に重要になると考えられる以下の六つの技術要素を同定し、それぞれに関して必要性や今後開発していなければいけないことなどを議論する。

1) インタフェースエージェント
2) 信用
3) 仲介
4) 交渉
5) オントロジ
6) XML への対応

(3) インタフェースエージェント

インタフェースエージェントの技術は、EC において人間の意図を適切に解釈、実行するための重要な技術要素である。 EC の巨大性の中で人間が EC を使っていくためになくてはならない技術要素である。

全ては EC における人間の満足度を高めるために使われるべきである。人間の意図をいかに適切に[5] エージェント系に伝えることができるか、いかに人間がコントロールを持ちながらも、適切なサポートを受けていると人間が感じうるかが重要な点になると考えられる。これらのことをインタフェースエージェントは、その時点で人間とのやり取りにおいて使用可能な全てのモードを通じて実現する必要がある。

これらのことに関するインタフェースエージェントの能力が、人間のエージェント系へ

---

[5] ここで「正確に」との言葉を使わなかったのは、EC のもたらす巨大なサイズ、異種性、動的性、複雑性の中では、正確性は意味を失い、相対的なものになっていくからである。必ずしも正確でなくても、人間が満足するという意味で「適切さ」であることが重要であり、エージェントの行動の規範になる。もちろんそのような中でも絶対的な条件を人間が指定する(できる)ことがある。(例えば「一月間に絶対に *** 円を越えてものを買ってはいけない」など)そのような条件に対しては正確にその条件を満たさなければいけないが、そのように正確に条件を満たすことは、もちろん適切なことであり、「適切性」の中に包含される。正確性が適切性の中で相対化されるということである。

の「信頼」に通じ、ひいてはエージェントへのより大きな権限の委譲となり、エージェントの自律性を許し、エージェント技術を生かせる土壌となる。

インタフェースエージェントを通じた人間とエージェント系のやり取りの方向は二つであるが、それらの方向を分けて考えるのではなく、やり取りをなるべく両方向含めた全体的な対話として成立させて行く方向を考えることが人間の満足度を向上するであろう。やり取りが人間主導の受身のものだけではなく、インタフェースエージェントから積極的に行われるものも含めて全体として流れを持ったものであることが重要である。

主に人間からエージェント系への流れで人間が明確に意識していろいろなモードを通じて人間が意図のエージェント系への伝達を主導するものがある。しかしその場合においても、エージェントが単に受身な状態でいるのではなく、不明確な部分はエージェントが積極的に人間に問い合わせ、明確化する（clarify）ことが重要である。

また人間が明確に意識して意図をエージェント系へ伝えようとしている時でも、人間がそのために使用している全てのモード（キーボード、音声、画像など）について意識しているとは限らない。その他のモードも使用して、より意図を正確にすることは可能である。例えば音声対話のモードをメインにしていても、顔の表情やジェスチャなど知らず知らずに使っていることは考えられる。さらに言えば一つのモード中においても、意識されている部分とされていない部分があり得る。（それらはモードとして分けるべきかも知れないが）例えば音声対話のモードで、人間が伝えたそのテキストとしての字面だけではなく、例えば早口でそのテキストを言ったとしたら、これは急いでいるなと理解することが可能である

さらには、人間が全く意識していないで発している意図をモニタリングによって積極的に入手することも考えられる。そのような意図を自動的に把握し、例えば EC における広告と結びつけるなら、使っている人も広告主も両者とも利益を受ける形での広告の実現になる。

エージェントが人間の依頼の意図を受けて、何かを実行する際には、状況の変化に応じてその人間にとっての重要性を判断し、適切なメッセージを人間に伝達し、必要なら意図の確認や変更の情報を入手するようにする。意図を伝える段階における確認や、途中の状態を知らせるなど、いかに適切なフィードバックがかかるかは重要である。状況に変化がある場合だけではなく、状況に変化がなくても長い時間がたってしまったりした場合に適切な行動が求められる。

その他、インタフェースエージェントの役割として依頼されたことの結果を知らせることなどがある。この結果の人間へのプレゼンテーションに関しては、人間の明確なあるいは無意識のうちの指示、あるいは理解の確認などにより適切な形にしていくことにより満足度が高いものとすることができるであろう。また依頼されたことに適切な結果が出せなかったときには、代替の結果、あるいは代替の手法などを提案することも考えられる。

さらには、エージェントの意図を人間に伝えることも重要になると考えられる。インタフェースエージェントあるいはインタフェースエージェントを通じてやり取りする他のエージェントから、人間の方で何かを実行することにより、より大きな利益を提供できる、あるいは EC におけるサービスの利便性の向上することができるなどの情報を伝えることが考えられる。

この節のここまでで、主にインタフェースエージェントの自律性に関連した事項を記述した。その他にインタフェースエージェントの実現において学習は必須である。学習は EC におけるエージェント系の他の部分でも必要だが、特にここで学習が重要となる。それは人間の意図を適切に把握し、 EC の巨大性の中で最初から適切に実現することは不可能であるからである。正確性ならまだしも適切性は非常に主観的なものであり、また変化していくものである。多少は最初に一般的なことをインタフェースエージェントに設定することができるが、その後主観的かつ変化していくものを的確に捉えるためには、人間からの意識的あるいは無意識的なフィードバックに基づき学習するシステムがその内部に必要となる。

(4) 信用

参考文献〔2〕でも指摘があったエージェント系におけるエージェント間の信用は EC へのエージェント適用に関し重要な問題である。エージェント間の信用の問題を解決できるかどうかが、エージェントシステムの適用が特定の範囲（企業内の社内適用や特定の企業間の取引など）に留まるか、それとも社会全体への適用を可能にするかを決める重要な技術要素である。エージェント間の信用の問題に適切なソリューションを与えることができるかが、インターネットが社会を変えたように、エージェントシステムが社会を変えるかの分水嶺となるであろう。このエージェント間の信用の問題は現在までにあまり活発に検討されておらず、今後の重要な課題であると考えている。

インターネットが導入された最初の段階の EC では人間と会社が以前より直接的にやり取りすることができた。そのことが中間のコストの削減やすばやいやり取りを実現し、メリットをもたらした。そこでは EC のそれぞれ参加者は、ウェッブブラウザやウェッブサーバを通じて参加し、したがってそれらのすぐ後ろには人間や企業が控えていた。ウェッブページやそのページへのアクセスを信用することは、それらの後ろにある人間や企業を信用することとほとんど同値であった。そこでは信用に関して今までの仕組みを流用して対応することは可能である。

ところが初期の成功で EC が拡大することにより、EC の巨大性の中で人間はより間接的な関わり方しかできなくなっていく。人間は EC に対して、インタフェースエージェントやエージェントによる仲介・交渉におけるエージェントの自律性をもって間接的に対応していくことになるわけだが、そのときに信用についてどうしていくかを考えていく必要がある。

信用には、明確に契約が成立した場合にそれを実現できるかどうか （例えば order fulfillment の問題） だけではなく、契約それ以前の問題 （例えばアクセスできなくなることにより、利益機会損失を与えたりしない） などがあり、さらにはそれらに間接性の問題も絡んでくる。

今まで信用は各種の物理的な制約にも依存していた。特に社会の枠組は「場所に依存していた」（“place based”）〔10〕のであるが、インターネットで状況は一変した。物理的な制約に代わる新たな制約を考え出す必要がある。

現在ある EC のシステムの信用に関係した仕組みとしては、例えば eBay〔3〕では取引における相手の評価を蓄積しておいて、他の人が取引の際に参考にできるようにしている。

また MIT の Kasbah システムでは“Better Business Bureau”と呼ばれる“a trust and reputation mechanism”を採用して、取引の参加者による相手の評価を蓄積し、次回以降の取引を行うかの判断材料にする。

以上のような仕組みを含んで、例えば以下のような枠組を検討して行く必要がある。〔2〕

a) 情報検索・収集

b) 情報の「信用度」の分析

c) リスクをどう取るのか?

まず自分自身の取引の経験、個人、企業など他者による情報、評価を専門とする私的、あるいは公的機関による情報を収集し、それに基づき例えば取引相手がどの程度信用できるかを分析、判断して行く。その際には、それらの情報源の信頼度も含めて学習していく必要があるであろう。

信用して取引しても、故意あるいは不可抗力により取引が実現されないことがあるわけで、最後にはリスクをどう取るかを考えておかなければいけない。最終的な safety net の問題であるが、ここは実世界との連携をどう取るかであり、保険や法律など社会の仕組みとの連携をエージェントなどによる間接性も含めて解決していく必要がある。

もう一つの可能性は、次の節でも挙げている「仲介」である。〔11〕でも指摘されているように、品質の不確実性（quality uncertainty）により、仲介者（intermediary）がそれを保証するような形で信用を提供し、同時にリスクなども負う形が考えられる。すなわち「信用」の仲介である。現時点で Digital Brand が一度確立すると、過度に集中するのは EC の世界でこういった仲介者による信用の仲介がうまく機能していないからと考えている。

いずれにせよエージェント間の信用の問題は、社会システムとの間で、試行錯誤を重ねていく必要があると考えているが、エージェントとともに EC のさらなる飛躍的な発展はこのエージェント間の信用の仕組みを実現することにかかっている。

(5) 仲介

EC において、仲介は必要でありつづける。それをサポートするものとしてエージェント技術における「仲介」の技術（brokerage/match making）がより洗練されることにより、重要な技術項目となる。

現在の EC では直接的なやり取りにより、流通中間業者を中抜きするようなことが目立っているが、むしろ仲介は EC の中核的な存在になって行くと考えている。

一つは「信用」の節でも挙げている信用の「仲介」である。「初めての売り手を信用できるか」という問題があり、信用の「仲介」により EC のスムースな取引が実現され、EC の範囲の拡大にも貢献するであろう。

また〔2〕で指摘があった経済のアンバンドリング化とともに、パッケージの editing 機能としての仲介もやはり重要な役割を果たすであろう。

またマッチメイキング（match making）の機能も、EC の巨大性の中では難しいものとなり、膨大な対象の中から適切な相手を見つけ出す機能としてエージェント技術の中で重要なものとなる。

(6) 交渉

ECにおいて、エージェントたちによる交渉も前節の仲介ともに重要な技術要素になるであろう。

将来のECでは、[2]にある経済のパッケージ化とそのediting機能の充実、あるいはマスカスタマイゼーション（mass customization）やパーソナリゼーション（personalization）の進展により、ユーザのプロファイル、要求、その時の状況など時間的要素に応じて、動的に商品やサービスのパッケージ化が行われ、動的に価格付けが行われるようになるであろう。（[11]も参照）いままでの店のように商品と価格のリストがあって誰がきても同じ価格ということはなく、一人一人個別にパッケージ構成、価格などを含む条件を交渉するようになるであろう。

ユーザの側でも例えば店がいかに信用できるかで、自分のプロファイルをどこまで出すかを決定して、それによって価格が異なるようになる。また店側からだけではなく、ユーザから見ても（仲介者も含む）店が出してくる条件は一軒一軒異なるわけで、自分の条件により適合する店を探し、交渉をして行く必要がある。

したがってユーザも店も複雑かつ膨大な交渉を行う必要があり、そのためには自律的で学習能力を持って互いに会話するソフトウェアモジュール、すなわちエージェントが不可欠となる。

近代より以前には唯一の価格というより、相手や状況に応じての価格付けがなされていたし、また現在でも企業間の取引では一件一件条件を検討するわけである。技術の進展により、個人レベルでも一対一の販売が可能になっていくと考えられる。そのインフラを支えるのがエージェント間の交渉という技術要素であると考えている。

(7) オントロジ

EC において、別々の主体がやり取りするためにはオントロジ（メッセージに用いられる用語と用語の意味の集合。辞書のようなもの）の整備が必要不可欠である。この点に関してはエージェントに限らず、例えばWebで、XMLを使ってやり取りするシステムでも同様である。

参考文献[10]が目指すように、もしECに本当に大小さまざまなビジネスが参加していくならば、膨大な量の商品・サービスあるいはやり取りの仕方が存在しているであろう。それらの商品、サービスあるいはやり取りの仕方を表現するためのオントロジは膨大になり、それらのオントロジの整備とそれらのオントロジを管理するオントロジ管理が必須となる。

オントロジは、まずエージェントが自律的に行動する際に正確に判断できる基準を与える必要として、エージェントたちに自律性を与えるベースとなる。

またオントロジの整備は人間とエージェントとの間の整合性を取るためにも必要である。例えばインタフェースエージェントを構築する際に人間の意図に対してどのようなメッセージを送り出すのが適切かを判断するためにも必要になってくる。

参考文献[6]の以下の文にもあるように

> "The greatest changes may occur, however, once standards are adopted and evolved to unambiguously and universally define goods and services."

もしオントロジが一つの標準に統合されれば、全てのシステムでそれを使えばよいように思われる。しかし実際にはそれはありえないと考えている。それは、ECのサイズと異種性の巨大さともう一つ以下のような理由である。

ECにおいて〔6〕にあるAndersen Consulting's BargainFinderの例のように値段だけの勝負になってしまったら、企業からそっぽを向かれてしまう。企業の存在価値とは、他者（あるいは他社）との差別化にあり、新たな付加価値・異なる価値を与えた新たな商品・サービスを作りつづける必要がある。そのためには新たな概念が必要であり、新たな言葉を必要とし続けるためである。

したがって地理的および時間的に差異化を続ける、異なるオントロジたちをリーズナブルな範囲で結びつづけるオントロジ管理の技術が必要になると考えている。

(8) XMLへの対応

現在ECではXML（extensible Markup Language）〔12〕はかなり重要な地位を占めつつあり、多くのデータ交換用のフォーマットとして使われはじめている。XMLのこのような状況および関連ツールの充実により、今後はエージェントでもECへの適用を中心としてXMLが以下のような部分で積極的に使われていくと考えている。

エージェント間のメッセージは通常、宛先などを示す封筒に相当するエンベロープ部分とメッセージが運ぶ主なデータとしてのコンテント部分に分かれる。

まずエージェントメッセージのコンテント部分にXMLによるデータを用いることが多くなるであろう。これによりXMLのデータを扱うシステムとの整合性が高まる。

また現在エージェントメッセージのエンベロープ部分ではその表現として主にS式（括弧でリスト構造を表すLispなどで使われている表現形式）が使われているが、今後XMLに基づいた表現も使われていくであろう（エージェント標準化団体FIPA〔4〕でもこれについて検討されている）。

XMLの標準などでメッセージとして部分的にエージェントのメッセージの意味論を持っている場合（XML QueryやEDIのXMLによるものなど）においては、ゲートウェイを使う形でエージェント系と連携していくものと考えられる。

XMLにおけるオントロジを表現するための標準といえるRDF（Resource Description Framework）は比較的扱いやすいモデル（Conceptual Graphに基づいたもの）を持っていることにより、ルールを表せないなどの制約はあるが、エージェントシステムでオントロジをあらわす手段などとして多く使われていくようになると考えている。

その他には周辺的な部分であるが、エージェントの各種設定などにもXMLが採用されていくであろう。

一方、連携、学習する自律的なソフトウェアとしてのエージェントに関してすでにあげた他の技術要素は、今後そのモデルや技術がXMLの標準になるなどの形でXMLに影響を与えていくと考えている。

(9) まとめ

エージェント技術はECの将来の発展に必須である。

決してエージェントで全てが解決できるという訳ではない。もちろんエージェント以外

のいろいろな技術が使われて将来の EC が形づくられていくと考えている。しかし将来の EC がもたらす巨大なサイズ、異種性、動的性、複雑性に対処するためにはエージェントの本質的な部分である、自律性、学習、連携などが必要であり、それゆえエージェント技術は将来の EC において重要な位置を占めると考えている。

もちろん今のエージェント技術がそのまま使えるわけではなく、今後の真剣な研究開発にもかかっている。この章では将来の EC においてエージェント技術の中で特に重要な技術要素を六つ同定し、それぞれにおいて EC での必要性、今後の課題などを述べた。これらは必ずしも一つ一つ独立した項目ではなく、密接な関係を持っている。

これらの課題を研究開発によって乗り越えたところに、EC の将来と同時にエージェント技術の未来があると考えている。

参考文献

〔1〕 有元 美津世“全図解 インターネットビジネスの仕組み”、あさ出版、July、(1999).

〔2〕 伊藤 修:“エージェント技術のビジネス化シナリオ -「自律」にひそむ「信用」の罠”、第３回ネットワークエージェント技術委員会、September、(1999).

〔3〕 eBay: http://www.ebay.com/

〔4〕 FIPA (Foundation for Intelligent Physical Agents) : http://www.fIPA.org/

〔5〕 Hyacinth S. Nwana: “Software agents: an overview、” The Knowledge Engineering Review、pp.205-244、Vol.11:3、(1996).

〔6〕 Pattie Maes, H. Guttman, Alexandros G. Mouka: “Agents that Buy and Sell: Transforming Commerce as we Know It、” Communications of the ACM、March (1999).

〔7〕 Munindar P. Singh: “The E-Commerce Inversion、” IEEE Internet Computing、pp. 4-5、Septermber/October、(1999).

〔8〕 U.S. Department of Commerce、“The Emerging Digital Economy、” http://www.ecommerce.gov、(1998)

〔9〕 U.S. Department of Commerce、“The Emerging Digital Economy II、” http://www.ecommerce.gov、(1999).

〔10〕 U.S.Government Working Group On Electronic Commerce: “Towards Digital eQualtiy、2nd Annual Report、” http://www.ecommerce.gov、(1999).

〔11〕 Andrew B. Whinston, Dale O. Stahl、Soon-Yong Choi: “The Economics of Electronic Commerce、” Macmillan Technical Publishing; July、1997.

〔12〕 XML (eXtensible Markup Language) : http://www.w3c.org/XML/

2.1.4 加工装置制御システム

（１）生産制御システムとエージェント

製造業をとりまく生産環境は、多様な消費者のニーズ、マーケット構造の急変、為替の急激な変動など変化が激しい。そのような変化に対応するために、工場を広域、グローバルに展開して、さらにその環境の中で最適な品質、コスト、納期で製品を作ることが求められている。このような生産環境の要求の変化に、現在の生産システムでは対応しきれず、

新たな要求に素早く対応できる柔軟性を持つことが、次世代の生産システムに強く求められている〔1〕。

従来から使われている集中型でシーケンシャルな生産計画、スケジューリングと制御機構は、生産形態や種々の生産要求の変化へ柔軟に対応するには不十分である。従来型のアプローチでは、生産システムの拡張性や構成変更の能力を制限している。従来の階層型／集中型のシステムは、ひとつの障害がシステム全体のシャットダウンへ結びつけるし、システム全体が脆弱で応答時間を増加させている。エージェント技術は、そのような問題に打ち勝ち、分散型の知的な生産システムを設計する自然な方法を提供すると考えられる。近年、エージェント技術は分散型の生産システムを開発するための重要なアプローチと考えられており、いろいろな分野でエージェント技術を適用しようとしている〔2〕。

エージェント技術の生産制御システムへの適用を考えた場合に、想定されるアプローチとしては、

a) レガシーシステムの問題点を解決し、会社全体の活動を統合するために、既存のソフトウェアシステムをカプセル化する。

b) 作業者、セル、機械、ツール、AGV、 製品、部品、操作などの生産リソースをエージェントとして表現し、リソースプランニングやスケジューリングや実行制御を取り扱う。

c) 生産システム内の登録や管理などの特別なサービスをモデル化して提供する。

などの場面が考えられる。本節では、b) における実行制御にエージェントを使用した例として、実験室の機械加工装置の制御システムへ適用した例について紹介する。

（2）概要

次世代の生産システムに求められる柔軟性とは、生産ラインの変更を行なうための装置の追加、変更や削除に対して、システム全体への影響を極力与えずに対応できることと、製品の改良や特注に対しても個別に生産可能であることと考える。このような生産物と生産ラインの柔軟性を実現するために、個々の生産物と、生産ラインに設置された装置にそれぞれエージェントを対応させることにより、協調性および自律性を持たせた分散システムの実現を目指し、カーネギーメロン大学(CMU) と共同で、実験室の機械加工システムを使ってプロトタイプシステムが構築〔3、4、5〕されている。

プロトタイプシステムでは移動エージェントと分散オブジェクトの技術を導入して、柔軟性を持ったシステムの実現を目指している。またプロトタイプシステムでは、開発言語として Java を使用し、移動エージェントとして IBM が開発している Aglets〔6〕を用いている。ここでは、エージェントを用いて生産制御システムを実現する場合の基本的な考え方と、開発したプロトタイプシステムについて紹介する。

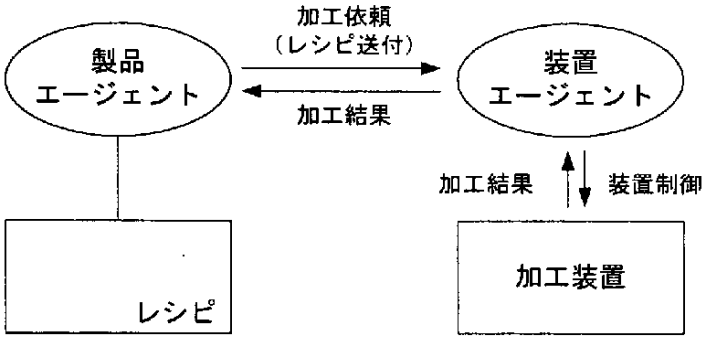
（3）基本コンセプト

開発したプロトタイプシステムでは、柔軟性を持つ生産システムを実現するために、階層型のシステムではなく、エージェントを用いたフラットな構造のシステムを提案している。提案しているシステムの基本的な考え方は次の通りである。

作ろうとする製品一つ一つに対応してエージェントを生成する。このことは、顧客から注文が入るごとにエージェントを生成すると考えてもよい。このエージェントを「製品エ

ージェント」と呼ぶことにする。一方、生産システム内の一つ一つの加工装置に対応させて、「装置エージェント」と呼ぶエージェントを生成する。装置エージェントは、それぞれのデバイスや装置を抽象化したオブジェクトである。

ここで、それぞれの製品エージェントには、作ろうとする製品の製造手順(本節では、この製造手順を記述したものをレシピと呼ぶ) を与え、製品エージェントは、そのレシピの手順に従って、必要な加工装置を選択して、選択した加工装置へ具体的な加工手順を指示し、実行させる。実際には、製品エージェントが加工装置へ直接指示を出すのではなく、製品エージェントは装置エージェントへ加工依頼(加工指示) を出し、装置エージェントが装置やデバイスを制御する (図表 2.1.4-1 参照)。



図表 2.1.4-1 基本コンセプト

製品エージェントは、システムや周りの状況を判断し、その変化に動的に対応しながら、システム内を自律的に動作し、製造手順にしたがって製品を完成させる。また、必要に応じて他のエージェントとネゴシエーションを行うこともある。

このコンセプトを実現するにあたり、製品エージェントを移動エージェントとして、また装置エージェントを非移動型のエージェントとして考える。装置エージェントは、加工装置を制御しているコントローラやコントローラに接続しているコンピュータで動作する。一方、製品エージェントは、移動型のエージェントで、加工を依頼する装置に対応する装置エージェントが動作するコンピュータに加工している製品とともに移動していき、装置エージェントが動作しているコンピュータ上で、装置エージェントに対して具体的な加工依頼、指示を行う。

システム内で、指示は個々の製品エージェントから出されるだけであり、製品を加工することに関しては集中的に管理する部分は必要ではなく、システム全体としてフラットな構造になっている。従って、製品や装置の数が増えた場合に、従来の階層型のシステムでは全体を管理する部分に負荷やメッセージが集中ことがあるが、ここで提案しているシステムでは、そのような負荷やメッセージの集中を避けることができる。

(4) ターゲットシステム

①実験室の概要

Shape Deposition Manufacturing(SDM) 実験室〔7〕は、CMU のロボット工学研究所に

属している。SDM 実験室では、任意の複雑な構造や形状の物を、CAD のモデルのデータから自動的にかつ素早く工作できることを実現している。SDM の研究では、従来の技術では工作できなかった、複数の材料から構成される物や、中にセンサなどを埋め込むような物を工作をする方法の研究に注力している。その応用範囲は、次世代の工作ツールから、個人が身につけるような wearable computer などの広い範囲に渡っている。

実験室では、工作物はパレットと呼ばれる金属製の板の上に作られていく。パレットは、パレット移動用のロボットによって、パレット受け取り機構を持った複数の加工装置にセットされる。現在の SDM テストベッドは、CNC、Deposition、 Shot-peening、 Cleaning の 4 つの装置から構成される。Deposition 装置も複数の工程をまとめるために別のロボット装置を使用している。

②従来のシステム

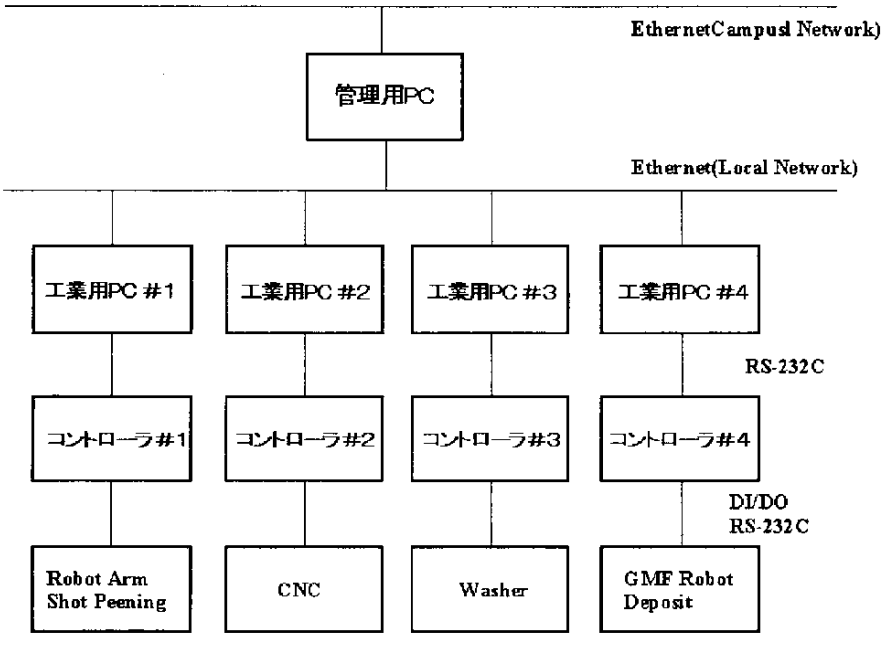
SDM 実験室のシステムは、長年に渡って学生によって改良や拡張が行なわれてきている。そのような ad hoc な拡張の結果、システム構成が複雑となってきており、システムの維持管理や新しい装置を付け加えることが難しくなっている。また、システム全体がひとつのコントローラにより制御される典型的な集中型制御システムであり、二つの工作装置を同時に使用することができない、などいくつかの改善すべき点が発生している。

③システム構成

ここでは、従来のシステムの問題点を考慮しながら、前述の基本コンセプトに基づき開発したシステム構成について述べる。

（ア）ハードウェア構成

本システムでは 6 台のプログラマブルコントローラ（うち 2 台はサブステーションとして使用）を使用している。これらのコントローラは工業用パソコン（Industrial PC、iPC）と各加工装置（Robot Arm、CNC など）に接続されている。工業用 PC は CPU が Pentium 133MHz、 メモリが 32MB の仕様である。さらに管理、操作用 PC（PentiumPro 200MHz、 メ



図表 2.1.4-2 ハードウェア構成

モリ 64MB）を使用して、システムオペレータはシステムの監視や機械の操作を行なう。各 PC のオペレーティングシステムは Windows NT 4.0 である。コントローラと PC 間、コントローラと工作機械の間は RS-232C で接続されている。図表 2.1.4-2 にハードウェア構成を示す。

（イ）ソフトウェア構成

本システムでは以下の 4 つの主要なソフトウェアモジュールで構成されている。

1）製品エージェント
2）装置エージェント
3）管理エージェント
4）ユーザインタフェース

製品エージェントは PC 上を移動する移動型エージェントであるが、装置エージェントは移動せずに特定の PC 上で動作し続ける非移動型のエージェントである。管理エージェントとユーザインタフェースは管理用 PC 上で動作する。それぞれのモジュールの機能と動作について以下に述べる。

1）製品エージェント

製品エージェントは、ターゲットとなる製品やワーク（この場合はパレットと考えてよい）と一緒にコンピュータやコントローラ上を移動する。製品エージェントは、「レシピ」と呼ばれる製品の加工手順を持っており、そのレシピに従ってターゲットの製品を加工していく。レシピの各行には、{機能名、コマンド}が対になって書かれている。製品エージェントは、レシピを 1 行ごとに処理していくことで、製造を完成させる。機能名には、各装置が実行できる作業名（たとえば洗浄機ならば WASH のように）が書かれている。またコマンドには、工作機械への実際の加工指示が書かれている。製品エージェントは、レシピの各行に書かれている機能名を探索のキーとして、次に加工依頼を行う装置を探す。その後、製品エージェントは、加工を引き受けることができる候補の中からひとつの装置を選択し、コマンドに記述されている加工を指示する。実際には装置そのものではなく、装置エージェントが装置選択の要求に対する受け答えや、加工指示を受け取る。

製品エージェントは、装置利用に関する排他制御を行うために、一つまたは複数の装置エージェントを専有する。排他制御は、装置エージェントが発行するアクセス権を得ることで実現する。製品エージェントは、ひとつまたは複数の装置エージェントのアクセス権を同時に持つことができる。装置エージェントに加工依頼や指示を行う場合には、まずターゲットとなる装置エージェントのアクセス権を取得し後に、実際の加工指示を送る。

2）装置エージェント

装置エージェントはそれぞれ工作装置の一つ一つに対応し、工作機械の制御とその状態把握を行なう。装置エージェントはそれぞれのエージェントが担当する工作装置の名前を機能名（例えば CNC、 WASH など）として持つ。いくつかの装置エージェントがひとつの同じ機能名を持っても構わない。機能名はレシピの中でターゲットとして指定される。また、装置エージェントは多くともひとつの製品エージェントに専有され、専有している製品エージェントからの指示によって作業を行なう。

製品エージェントからの指示に従って、装置エージェントはコントローラを介して工作機械を制御し、工作機械やコントローラのの状態を把握する。装置エージェントが動作するPCとコントローラと、またコントローラと工作機械の間は、シリアルライン（RS-232C）で接続されている。

何らかの加工データが必要な場合には、装置エージェントは管理用PCのディスクからデータを転送する。

3）管理エージェント

管理エージェントは通常管理用PC上で動作して、ユーザインタフェースと通信しながら、オペレータからの製品エージェントの生成や状態のモニタリングの指示などを受け付ける。

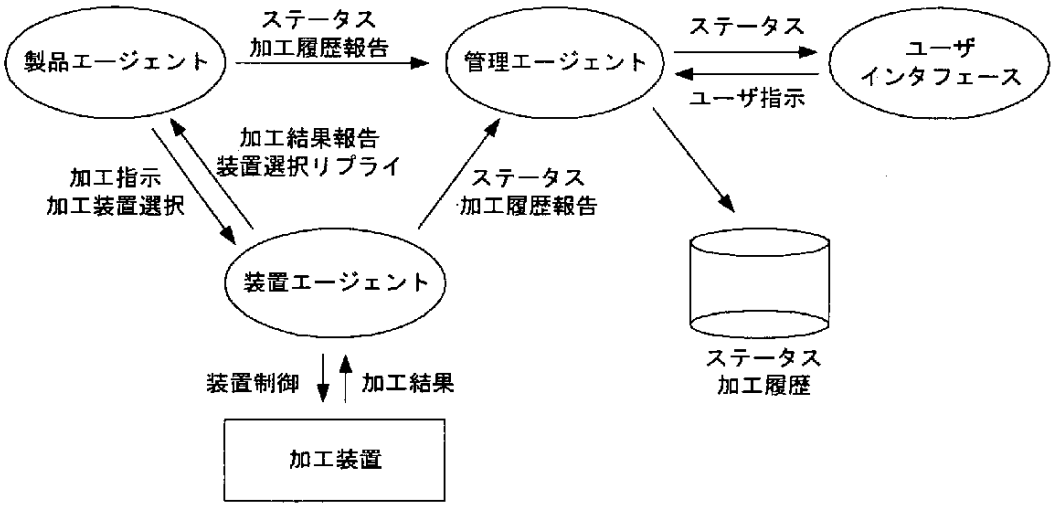
管理エージェントはオペレータに指定されたレシピをつけて、製品エージェントを生成してシステムに投入する。また、製品エージェントの存在場所や内部状態を監視して、管理用PCのディスク上に製品エージェントのログデータを残す。もしも製品エージェントが停止した場合には、管理エージェントはオペレータに通知し、必要ならば製品エージェントをログデータなどを利用して再生成する。

管理エージェントは装置エージェントの状態の監視も行ない、管理用PCのディスク上に装置エージェントのログデータを残す。装置エージェントが停止した場合にも管理エージェントはオペレータに通知し、必要ならばログデータなどから装置エージェントを再生成する。

4）ユーザインタフェース

ユーザインタフェースは、オペレータの指示を管理エージェントを通して製品エージェントや装置エージェントへ伝えたり、製品エージェントや装置エージェントの状態を表示する。ユーザインタフェースは、管理エージェントとIPレベルで接続しているので、ユーザインタフェースのみを遠隔地で利用することもできる。

以上、4つのソフトウェアモジュールの関係を図表2.1.4-3に示す。



図表2.1.4-3 エージェント間の関係

5) レシピ

レシピは、SDM 実験室で従来から使用されている形式をそのまま利用した。一行の形式を以下に示す。

形式

<OP_NUM> <OP_NAME> <FUNCTION> <FILENAME>

<OP_NUM>

行番号

<OP_NAME>

PLACE|PICK|LOAD_FILE|RUN_FILE|DEL_FILE

<FUNCTION>

VMC6030|SPRAY|DRIP|WASH|PEEN|BLAST

例

001PLACEVM6030

002RUN_FILEVM6030data.cnc

FUNCTION はこの行の作業を依頼すべきターゲットが持つべき機能名であり、OP_NAME は依頼する作業を示している。FILENAME は、必要に応じて加工データなどのファイル名を指定する。以上の形式で書かれた行をシーケンシャルに並べて、並べられた順番に加工をしていくと製品が完成する。

④インプリメンテーション

本システムでは、インプリメンテーション言語として Java を使用しており、シリアルポートの制御には native method (JNI) を使っている。Java の実行環境は JDK1.1 ベースである。

また、エージェントの移動とエージェント間のメッセージ通信には、IBM で開発された Aglets(alpha5) [8] を、また、管理エージェントとユーザインタフェース間の通信には HORB [9、10] を使っている。

また、それぞれの PC 上で NTP (Network Time Protocol) を動作させて、複数ある PC 間の時刻の同期を図っている。これは、ログデータのタイムスタンプがそれぞれの PC のローカルな時刻を使って取られるので、システム全体でログデータの時刻の一貫性を持たせるためである。

（5）まとめと今後の課題

本節では、CMU の SDM 実験室と共同で開発した移動エージェントを用いた加工装置制御のプロトタイプシステムについて述べた。開発したプロトタイプシステムでは、生産システムで必要となる柔軟性やアシュアランス [11] を提供するために以下のアプローチを採用した。

●システムとしての柔軟性を持たすために、加工装置の構成が変わっても対応できる

ように、実際に加工を行なう加工装置を実行時に選択する。また、生産する製品の種類を変更しやすいように、それぞれの製品の製造手順を示したレシピを、個々の加工物に対応して持たせる。

- ●システムの可用性を確保するために、製品エージェントと装置エージェントはハートビート信号を管理エージェントに送り、管理エージェントはそのハートビート信号を監視する。
- ●データの信頼性を確保するために、製品エージェントは製造データや品質データを、定期的に管理エージェントに送り、管理エージェントがそのデータを不揮発性の媒体(例えばディスク) に保存する。

今後、エージェントを生産制御システムに適用する場合を考えると、

- ●システムの可用性、エージェントの持つデータの信頼性を向上させる仕組み
- ●集中型機構をもたない可用性向上のための方式
- ●通信相手のエージェント、メッセージやエージェントプラットフォームが信頼できるものであるかの認証
- ●リアルタイム処理への適用

などの課題を解決する必要がある。

参考文献

〔1〕 牧野洋、新井民夫、“自動組立システムの動向”、精密工学会誌、Vol.63、No.11、pp.1503-1509、Nov. 1997.

〔2〕 W. Shen, D.H. Norrie, “Agent-Based Systems for Intelligent Manufacturing: A State-of-the-Art Survey”、http://imsg.enme.ucalgary.ca/publication/abm.htm.

〔3〕 H. Wada、Y. Sakuraba、M. Negishi、T. Yamakawa、K. Kubo、Y. Kashiyama、“A Machinery Control System Using Mobile Agents”、Proc. of the 4th International Symposium on Autonomous Decentralized Systems、March 1999、pp.124-131.

〔4〕 和田英彦、根岸雅子、櫻庭祐一、“移動エージェントを用いた加工装置制御システムにおける信頼性”、信学技報、FTS99-26、June 1999、 pp.47-54.

〔5〕 星哲夫、“エージェントによる次世代生産システムの課題と展望”、人工知能学会研究会、SIG-J-9801-2、 Dec. 1998、 pp.5-10.

〔6〕 D.Lange and M.Oshima、 “Programming and Deploying Java Mobil Agents with Aglets”、 Addison-Wesley、 ISBN:0-201-32582-9、 1998.

〔7〕 Shape Deposition Manufacturing Laboratory、 http://www.cs.cmu.edu/~sdm.

〔8〕 Aglets Workbench、 http://www.trl.ibm.co.jp/aglets/index.html

〔9〕 HORB、 http://ring.etl.go.jp/openlab/horb

〔10〕 S. Hirano、“HORB: Distributed Execution of Java Programs”、Worldwide Computing and ITS Applications'97、Springer Lecture Notes in Computer Science 1274、pp29-42、 1997.

〔11〕 I-Ling Yen、 R. Paul、 K. Mori、“Towards Integrated Methods for High-Assurance Systems”、 IEEE Computer、 Vol.31、 No.4、 April 1998、 pp.32-34.

2.1.5 ITS（高度道路交通システム）分野への応用

（1）ITS 分野におけるネットワークエージェントの必要性

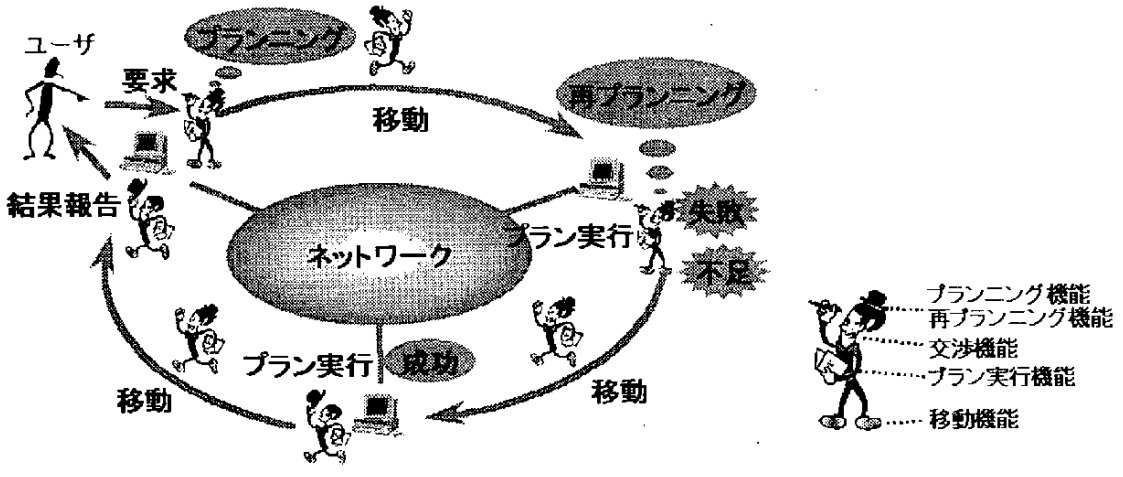
ITS（Intelligent Transport Systems）は、高度なデジタルネットワーク通信技術によって支えられる近未来型の社会システムである。道路交通に限らず家庭やオフィスなどを含め、ユーザがどこにいてもネットワーク経由でさまざまなサービスを受けられるよう、基盤やアプリケーションの開発が進められている。

インターネットの普及により、既に我々は世界の最新情報にアクセス可能である。しかし、いかに目的の情報にたどり着き、それを利用するかはユーザのスキルに任されており、不慣れなユーザはうまく情報を引き出せないといった問題が生じている。これと同様の問題が ITS 分野においても懸念されている。このため、ITS 分野では、ユーザのサービス利用を補助する技術としてエージェントが注目されている。ここで求められるエージェントは、ユーザの代行者である。ユーザの意図を理解し、やるべきことをユーザの代わりに実行してくれるものである。例えば、ネットワークからの情報取得では、情報がどこにあってどのようにアクセスすればよいのかがわかりにくい。情報の存在場所やアクセス方法をエージェントが自分で調べて実行してくれれば、ユーザは本来の目的に集中することができ、ネットワーク利用の負担が軽減される。

本章では、まず、知的モバイルエージェント Plangent の機能概要を説明する。次に、Plangent を ITS 分野への適用した事例として、「ドライバー情報支援 InfoMirror」と「ヒューマンナビゲーションエージェントゆがしまん」を紹介する。最後に、これらの事例の現状と課題について整理する。

（2）知的モバイルエージェント Plangent

Plangent は ITS に限らず、プラントの遠隔監視制御やオフィスにおける会議日程調整など、さまざまな分野に適用可能な汎用のエージェントシステムである〔1～5、14〕。プランニング機能と移動機能を備えており、ユーザの要求に対して最善の行動計画（アクション）を決定し、ネットワーク上を移動しながら自ら生成したアクションを順次実行することで、ユーザのネットワークアクセス操作を代行する。



図表 2.1.5-1 Plangent におけるエージェントの動作概要

現状で車両とネットワークを接続する際、携帯電話のような低速で不安定な回線を用いることが多い。このような環境では、ユーザの要求が完全に実現されるまで回線接続を維持しておくのことは難しい。Plangent では、エージェントが移動する際にユーザの要求やそのための行動計画、実行途中のプログラム状態やデータなどを持ち運ぶので、遠隔コンピュータ上で独立して処理を進めることができる。このため、エージェントの移動後に回線を切断しても、サービスを行なう上で何ら問題ない。必要に応じてエージェントが回線を再接続し、処理を継続することになる。また、エージェントのサイズは非常に小さいので、低速な回線でもユーザのストレスになりにくい。

ネットワーク上では、日々刻々環境が変化する。情報の格納場所やアクセス方法が変更されていた場合など、エージェントが当初計画したアクションを実行しても失敗することが多い。このような場合、Plangent のエージェントは、再プランニング機能を用いて別の計画を生成して実行する。このような試行錯誤を繰り返すことにより、エージェントが環境の変化に柔軟に対応する。図表 2.1.5-1 に Plangent におけるエージェントの動作概要を示す。

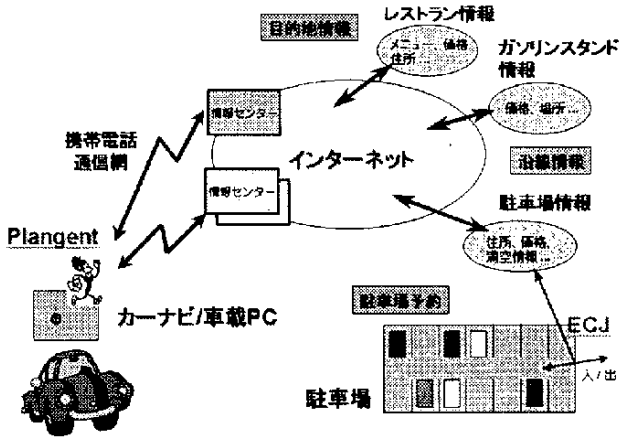
（3）ドライバー情報支援 InfoMirror

InfoMirror はモバイルエージェント技術をカーナビゲーションシステムに応用したシステムである〔6～8〕。カーナビゲーションシステムは、ITS の中でも先行して実用化され、市場を牽引しているものである。最近では交通情報を提供する VICS (Vehicle Information and Communication System) などと連動しており、単に地図表示やルートガイダンスを行なう機器から、ネットワークに接続してさまざまな道路交通情報を取得する車内情報端末に姿を変えつつある。このようなシステムでは、取得できる情報が豊富になる分、取得に要する操作は煩雑化していく。このため、運転中でも容易に操作できるユーザインタフェースの構築や、車の中で本当に必要となるコンテンツの獲得などが課題となる。

InfoMirror において、エージェントは自律的な判断や暗黙の指示の補完などによって、運転中におけるカーナビゲーションシステムの操作の煩わしさを排除し、試行錯誤を繰り返しながらネットワーク上を移動し、ユーザの求めている情報を探し出す。例えば、レストラン情報を検索している際には、運転者は駐車場の確保にまでは注意を払っていないかもしれないが、車で移動しているからには駐車スペースが必要なことは明らかである。エージェントは適当なレストランを見つけたら、近くの空いている駐車場も同時に検索することによって、運転者の操作を簡便化することができる。また、InfoMirror では、高速イベント配送系 ECJ〔15〕を情報伝達のフレームワークとして採用し、これによって外部の情報をリアルタイムに提供することが可能となっている。

ユーザが利用可能な ITS サービスは多種多様であり、今後さらに新しいサービスが増える方向である。ユーザの立場からは、それらのサービスを自分の都合に合わせて組み合わせて使いたい場合が多い。サービス提供の立場では、他のサービスとの組み合わせを意識した機能やコンテンツの充実は煩わしいものであるし、次々と現れる新規サービスに適合させることはほとんど不可能である。この点 InfoMirror では、エージェントがサービス間のインタフェースを調整し、試行錯誤の柔軟さによって複数サービスの連携をはかることができるので、手間をかけずにユーザのニーズに応えることが可能となる。

また、今後のユビキタスコンピューティングの世界では、家庭やオフィスのPC、モバイルコンピューティングで利用するPDA (Personal Digital Assistants)、街角に設置されるキオスク端末、車両における車載PCなど、ユーザの操作するデバイスはさまざまなものになる。サービスの提供者がユーザ端末の多様性を考慮したコンテンツ、すなわちユーザのビューまで考慮したコンテンツを作成するには限界がある。サービス提供者は本来のコンテンツのみに集中し、ビューはユーザ端末側で調整するという形態が好ましいと考えられる。InfoMirrorでは、コンテンツとビューを別のものとして扱い、エージェントがこれらを統合するということも行なっている。図表2.1.5-2にInfoMirrorの構成を示す。



図表2.1.5-2　ドライバー情報支援InfoMirrorの構成

（4）ヒューマンナビゲーションエージェントゆがしまん

ゆがしまんはモバイルエージェント技術をヒューマンナビゲーションに応用したシステムである〔9〕。ヒューマンナビゲーションとは、歩行者に対して現在位置や周辺情報を与えることによって、ユーザの移動を支援するシステムである。

ヒューマンナビゲーションシステムでは、多様なユーザのニーズを満たすため、ユーザの欲する情報をネットワーク上から適切に抽出して提供する必要がある。また、携帯情報端末を介した情報提供の際には複雑な操作を不要とすることはもちろん、ときにはユーザの操作が行なわれていない場合にもシステム側から情報提供を行なうことが有効となる。

伊豆半島のほぼ中央、天城湯ヶ島町にて、Plangentを用いたエージェント“ゆがしまん”を導入したヒューマンナビゲーションシステムの実証実験が実施されている。IPA(情報処理振興事業協会)による地域総合情報化支援システム整備事業 天城湯ヶ島町ヒューマン・ナビゲーションシステムの一環として行なわれているものである。

天城湯ヶ島町はその温泉と景観から温泉療養に適しているとされ、古くは多くの文人が常宿としていた地でもある。実証実験システムでは、PDA およびキオスク端末を用いて、湯道と呼ばれる散策路を中心とする観光や温泉療養のための周辺案内および関連情報が提供される。このシステムの主な機能は次の三つである。

・地図及び現在位置の表示
・周辺情報および関連情報の表示

・キャラクタによるガイド

ナビゲーションを行なうPDAとしては小型のノートパソコンを用いている。また、地域情報センタと称する一室には情報提供を行なうためのサーバ群が稼動しており、PDAは付属のPHSから地域情報センタのPBX（構内交換機）を経由してネットワークに接続、情報の取得を行なう。この際、ユーザの位置はPHSが交信しているアンテナにより特定される。

ゆがしまんはPlangentを利用して構築した上で、さらに次の機能を備えている。

①ユーザに応じた情報提供

あらかじめ登録されたユーザ情報に基づいて状況やユーザの好みに応じた情報を抽出して提供する。ユーザ情報は、年齢や性別などの個人情報や、文学に興味があるか否かなどの嗜好に関する特性により構成されている。一方でコンテンツには、提供すべき季節、時間、ユーザの傾向などの属性を付加しており、ユーザ情報や状況との適合度を算出することによりユーザに合わせた情報提供を行なう。

②自発的情報提供

ユーザの操作入力に応じて情報提供を行なうだけではなく、時間経過、状況の変化やユーザの操作状況に応じて、能動的に情報提供を行なう。ガイドキャラクタは、ツアーガイドとしてユーザの位置に応じて時折出現して道案内や周辺情報の提供を行なうなど積極的な情報提供を行なう。

ゆがしまんはこれらの機能により、ネットワーク上からユーザの個人特性や状況に応じた情報を抽出して提供することを実現している。図表2.1.5-3は、ゆがしまんのエージェントがユーザの要求を求めている画面の例である。

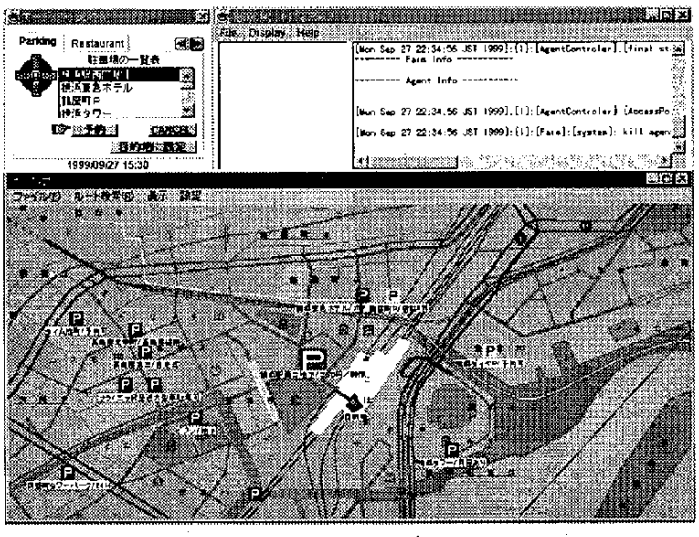
図表2.1.5-3　ゆがしまんの画面例

（5）現状と今後の課題

システム事例として紹介したInfoMirrorとゆがしまんは、それぞれ実証実験を通じて一般ユーザが使用したものである。InfoMirrorでは、自動車メーカの情報提供サービスと提携し、都内の繁華街、新宿・渋谷・銀座の3地域で、駐車場の満空情報をほぼリアルタイムに車両に提供する実証実験が1999年2月から3月にかけて行なわれ、駐車スペースの予約実験も実施された。図表2.1.5-4はエージェントが駐車場の満空情報をネットワークか

ら取得し、ドライバー向けに表示した画面の例である。ゆがしまんは、伊豆天城湯ヶ島町において、World Wide Web向けホームページを整備した上で、現地に設置されたキオスク端末や観光客に貸し出されるPDAに向けて観光情報を提供する実証実験が1999年8月から実施されている。

ところで、ITSのユーザへの情報提供では、ユーザの利便性を確保するためにエージェント技術が不可欠であるが、各社が開発したエージェントの相互接続ができなければ各サービスは隔離され、ユーザは混乱するだけである。このため、異なるエージェントシステム間でのメッセージの相互運用性、あるいは、異なるモバイルエージェントシステム間での移動の相互運用性が重要な課題となる。これについては、エージェント技術に関する標準化団体FIPA (Foundation of Intelligent Physical Agents)〔16〕やOMG (Object Management Group)〔17〕などの活動成果が期待される。また、異種モバイルエージェントシステム間の相互運用性を確保するミドルウェア技術の研究開発なども進められている〔10～13〕。この種の活動を通じて、エージェントの活躍の場が広がることが期待される。



図表2.1.5-4　InfoMirrorによる駐車場情報表示画面例

（6）まとめ

本節では、ITS分野へのネットワークエージェント技術の適用事例として、ドライバー情報支援InfoMirrorとヒューマンナビゲーションエージェントゆがしまんを紹介した。InfoMirrorにおいては、エージェント技術を用いることで、ネットワーク上の膨大な情報からユーザが必要とするものを抽出することが可能となった。また、さまざまなサービス間の整合性をとったり、ユーザ端末の多様性を調整することも可能となっている。ゆがしまんにおいては、エージェント技術を用いることで、ユーザの個人特性や状況に応じた情報提供、ユーザの位置に応じた自発的な情報提供などが実現された。このように、エージェント技術はITSを含めたネットワークコンピューティング分野において、今後の鍵を握る重要技術と考えられる。

参考文献

〔1〕 本位田真一、大須賀昭彦: オブジェクト指向からエージェント指向へ、ソフトバンク (1998).

〔2〕 本位田真一、飯島正、大須賀昭彦: エージェント技術、共立出版 (1999).

〔3〕 Ohsuga, A.、Nagai, Y.、Irie, Y.、Hattori, M.、and Honiden, S.: PLANGENT: An Approach to Making Mobile Agents Intelligent、*IEEE Internet Computing*, Vol. 1、No. 4 (1997)、pp. 50-57.

〔4〕 大須賀昭彦: プランニングモバイルエージェント、bit、Vol. 31、No. 3 (1999)、pp. 88-95.

〔5〕 大須賀昭彦ほか: 知的モバイルエージェントPlangentプログラミング、bit、Vol. 31、No. 8, 9, 10 (1999).

〔6〕 Hattori, M.、Kase, N.、Ohsuga, A.、and Honiden, S.: Agent-Based Drivers' Information Assistance System、Special Issue on New Challenges in Intelligent Systems (Nishida、T. ed.)、*New Generation Computing*、Vol. 17、No. 4 (1999)、pp. 359-367.

〔7〕 Kase, N.、Hattori, M.、Ohsuga, A.、and Honiden, S.: InfoMirror: Agent-based Information Assistance to Drivers、 in Proc. *International Conference on Intelligent Transportation Systems* (ITSC99)、IEEE/IEEJ/JSAI (1999)、pp. 734-739.

〔8〕 加瀬直樹: ITSにおけるエージェント技術、情処研報、Vol. 99、No. ITS-1 (1999)、pp. 21-27.

〔9〕 池谷直紀、加瀬直樹、大須賀昭彦: エージェント技術を適用したヒューマンナビゲーションシステム、情処ITS研究会、99-ITS-3 (1999)、pp. 97-104.

〔10〕 長谷川哲夫, 大須賀昭彦, 粂野文洋, 中島震, 西田豊明, 本位田真一: 移動エージェント相互運用の実現:背景と構想, 第59回情報処理学会全国大会 (1999), pp. 3-279-280.

〔11〕 粂野文洋, 長健太, 長谷川哲夫, 大須賀昭彦, 中島震: 移動エージェント相互運用の実現: 基本アーキテクチャ, 第59回情報処理学会全国大会 (1999), pp. 3-281-282.

〔12〕 長健太, 長谷川哲夫, 大須賀昭彦, 粂野文洋, 中島震: 移動エージェント相互運用の実現:インカネーションエージェントによる相互運用の実現, 第59回情報処理学会全国大会 (1999), pp. 3-283-284.

〔13〕 別府祥之, 中島震, 粂野文洋, 長谷川哲夫, 大須賀昭彦: 移動エージェント相互運用の実現:ディレクトリサーバ, 第59回情報処理学会全国大会 (1999), pp. 3-285-286.

〔14〕 Plangent: http://www2.toshiba.co.jp/plangent/

〔15〕 ECJ: http://www2.toshiba.co.jp/ecj/

〔16〕 FIPA: http://www.fIPA.org/

〔17〕 OMG: http://www.omg.org/

2.1.6 デジタル放送とエージェントテレビ

（1）概 要

2000年からBS(Broadcasting Satellite)デジタル放送が始まる。デジタル放送により、高品質の映像と音声に加えて、文字・図形・静止画やコンピュータデータなどの各種情報やデータを使った、まったく新しい放送サービスがスタートする。デジタル放送によってアナログテレビからデジタルテレビへのシフトが始まり、白黒からカラーへの転換に相当するインパクトが期待されている。一方、デジタル放送、多チャンネル化時代に対して放送用機器全般の高機能化が望まれている。特に、視聴者側のシステムであるテレビ受像機はこれまでの単純に映像と音声を“再生”するだけの装置でなく、より視聴者に密着して、大量の番組から必要な情報や興味を引く番組のみを選択し、それを効果的に視聴者に提示する機能が要求される。このような新たなテレビ受像機を実現する技術として「エージェント」が注目されつつある〔1〕。ユーザの代りに、リソースの確保、環境の変化の感知、与えられたタスクを自律的に遂行するエージェントの特徴は上述した受像機の高機能化にも適していると考えられる。このようにエージェント技術を用いて柔軟でかつ高度な情報処理能力を持つ受像機を「エージェントテレビ」と呼ぶ。

エージェントテレビの基本的な機能は、地上、ケーブル、衛星、通信ネットワークから送られてくる大量の番組と情報から視聴者の好みやそのタイムリーな番組や情報を選択し、見やすい形態で編成された番組列として提示することである。番組の選択には、放送スケジュールや番組の内容に関する属性を記述するEPG（Electronic Program Guide－電子番組案内）を利用するのが効果的であり、放送局または情報プロバイダによってEPGが提供されることが重要である。エージェントテレビは視聴者に関するユーザプロファイル（年齢、趣味、嗜好、習慣などの個人情報）を保持し、このユーザプロファイルとEPG情報から、その時点で提示すべき番組を選択する。各番組の内容を考慮して番組の提示順序が決まれば、CGキャラクタなどの擬人化エージェントを使用して演出の効果を図ることもできる。また、その時々に飛び込んできた情報を効果的に通知する機能もエージェントテレビの重要な機能と言える。これまでの速報や災害情報以外にも、視聴者が興味をもつような娯楽性の高い情報や話題も提供することが望ましい。さらに、ホームサーバなどの大容量の記憶装置を利用することで、利便性の高いノンリニア視聴環境を提供することも可能である。

ここでは、エージェント技術を利用して受像機の高機能化を狙う「エージェントテレビ」に関する研究・開発の現状とその実装方法について述べる。

（2）エージェントテレビの研究・開発の現状

放送分野における受像機の高性能化は、立体テレビやHDTV（High-Definition TV）の研究・開発による画像と音質の改善を主に優先させてきた。その理由として、1）従来放送はチャンネルの数が少ないこと、2）アナログベースであったため計算機処理に適していないこと、3）双方向送受信が可能なネットワークのインフラが欠けていたことなどが挙げられる。一方、計算機処理との適合性を持つデジタル放送と通信の融合を先見して、コンピュータネットワークを利用した新たな放送サービスを視聴者に提供しようとするいくつかのモデルが提案され、実際に運用されているものもある。その中の代表的なものに Web-TV、

DICEMAN、FACTSプロジェクトなどがある〔2〕〔3〕。Web-TVは、米国のマイクロソフト社によって提案された放送モデルで、インターネットとテレビ放送を融合するように設計されている。Web-TVの狙いは、インターネットを利用して放送番組に関連する様々な情報を提供し、既存のテレビサービスを“補完する”ことで、受像機の高機能化を図ろうとすることである。既存のテレビ放送サービスと比較してWeb-TVの最も優れている点は「放送スケジュール、各番組の等級情報（rating)、番組の値段などに関する情報」、すなわちEPGの提供である。視聴者は、毎日更新されるこのEPG情報をもとに自分に適合した視聴プランを立てることができる。しかし、Web-TVは既存のサービスと比べて大変便利な機能を提供しているものの、EPGのサービスを除けば、エージェントテレビの実現に重要な各視聴者に対する柔軟でかつ高いレベルの個人化や優れたヒューマンインタフェースを十分提供していない。Web-TVに関するいくつかの問題点を以下に示す。

1) 初歩的なユーザプロファイル機能を提供するものの、動的に変化する視聴者の好みや状況に自律的に対応する手段は提供しない。例えば、視聴者は自分の嗜好が変わるごとにプロファイルを自分で修正しなければならない。

2) 環境の変化に対する自動感知機能を持たない。例えば、番組のスケジュール変更や事故などの緊急ニュースに対する自動通知機能などがなく、視聴者が監視しなければならない。

3) 視聴者の代わりに番組列を編成し、見やすく提示する手段を提供しない。

4)現在多くの視聴者が見る話題の放送や情報を自動的に検索し、提供する手段がない。

一方、DICEMANプロジェクトは、MPEG-7の規定に従って記述されたあらゆるコンテンツ（番組、映画、音楽、ビデオ、テキストなど）に対して柔軟なコンテンツの検索を実現する手段としてエージェントを利用し、視聴者の好みに適した番組選択の実現を目指している。特に、MPEGやFIPAなどの標準化機関と密接に連携してプロジェクトを現在活発に進めている。

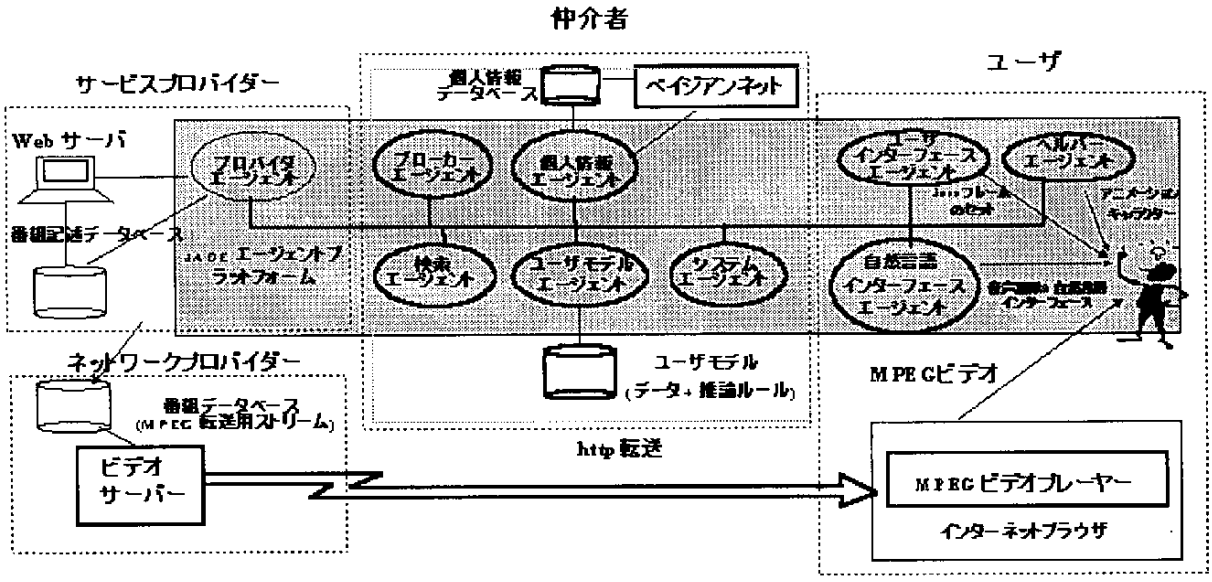
FACTSは、ヨーロッパのACTS (Advanced Communications Technologies and Services) プログラムの一環として、FIPAがエージェント間の相互運用性（Interoperability）を確保するために規定したエージェント標準仕様の検証を行うことを目的として、エージェントをテレビに実装する実験を行っている。ここでは、FACTSプロジェクトを中心にエージェントテレビの構成やその実装方法について述べる。

（3）FACTSにおけるエージェントテレビの研究・開発

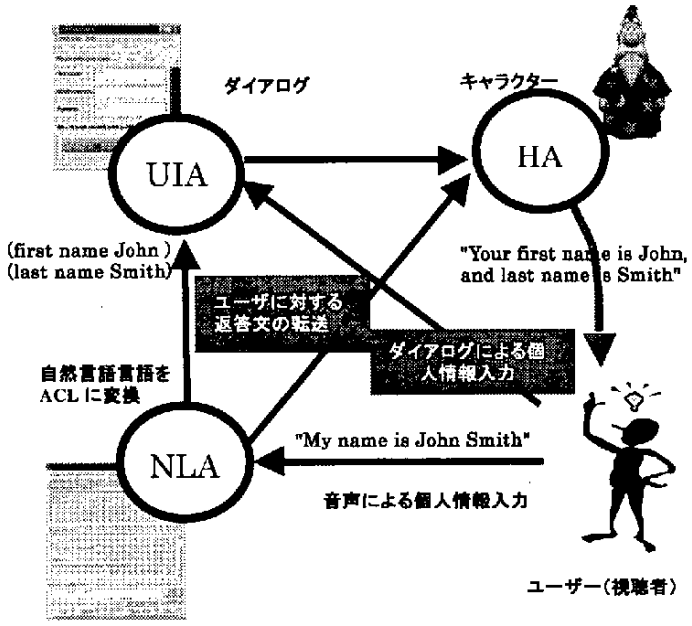
FACTSにおけるエージェントテレビは、図表2.1.6-1で示すように、大きく「ユーザ」、「仲介者」、「サービスプロバイダ」、「ネットワークプロバイダ」、の４つの機能領域（以下ドメインと呼ぶ）とエージェントの管理をサポートする「エージェントプラットフォーム」で構成されている。FACTSでは、エージェントシステムの実装言語として主にJAVAを使っており、エージェントの内部機構にはJESS（Java Expert System Shell）が使われている。以下各ドメインを構成するエージェントについて順次説明する：

（A）ユーザドメイン：ユーザドメインを構成するエージェントは視聴者のために動作するプログラムで、インタフェースエージェント、ヘルパーエージェント、自然言語エージェントの３つのエージェントによって構成される。FACTSでは、ユー

ザドメインを小型計算機や携帯端末上に実装し、ネットワークを介して他のエージェントと通信することで、視聴者はどこでも好きな番組を見ることを可能にしている。このような実装方法によってシステム全体のモビリティーを強化することができる。



図表 2.1.6-1　エージェント間の相互関係



図表 2.1.6-2　エージェントテレビの構成

インタフェースエージェント（UIA）：このエージェントの主な役割は、視聴者と他のエージェント間のやり取りを管理することである。具体的には、1）視聴者の認識と管理、2）視聴者からの要求の把握、3）視聴者の好みに合わせた情報の入出力、4）ヘルプ機能の提供などが挙げられる。インターフェースエージェントは、ヘルパーエージェントと自然言語エージェントを制御して GUI、自然言語、

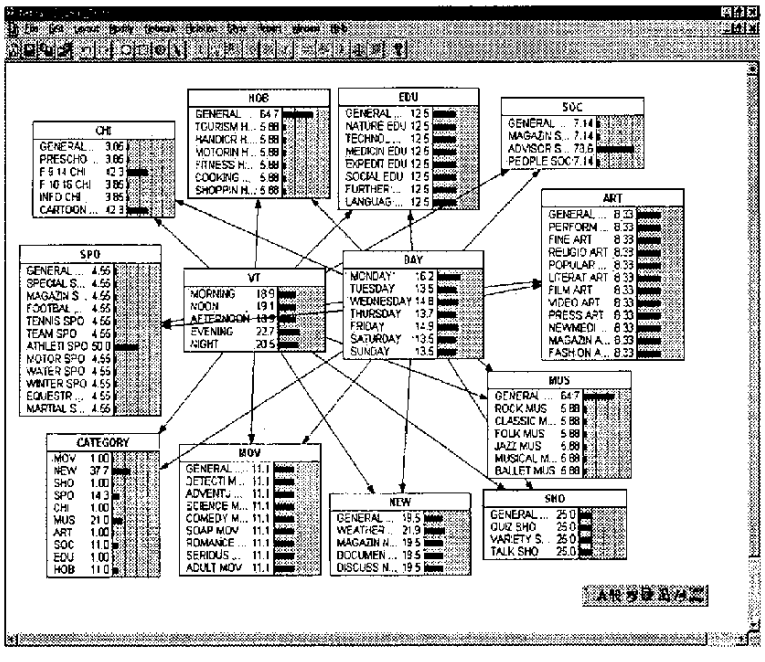
CG キャラクターなどを駆使して視聴者とのやり取りを行うように設計されている。これらのエージェント間の相互関係を図表 2.1.6-2 に示す。

ヘルパーエージェント（HA）：このエージェントは視聴者に対してシステムの使用法やシステムからのメッセージなどを視聴者に効果的に提示するために設計されている。ヘルパーの親和性を強調するため、擬人化キャラクター（MS-Agent を使用）と音声合成装置を用いてヘルプの内容を効果的に伝える。

自然言語インターフェースエージェント（NLA）：自然言語による視聴者の要求をエージェント通信言語（ACL）に変換してインターフェースエージェントに渡す役割を持つ。言語解析は、精密に定義されたコンテキスト（例えば、視聴者の名前の認定、嗜好の入力、検索の要求のダイアログ画面など）によって制約されるので、精度の高い解析結果を得ることができる。また、音声認識装置（IBM の ViaVoice を使用）を用いることで、マウスのようなポインティング装置やキーボードを使わずに音声でシステムとのやり取りを行うことができる。

（B）仲介者ドメイン：コンテンツプロバイダと視聴者の間に立ち、番組情報や個人情報の管理、ブローカリング、コンテンツの手配などの機能を提供する仲介層である。このドメインを構成する主なエージェントとして、プロファイルエージェント、ユーザモデルエージェント、ブローカーエージェント、検索エージェント、システムエージェントが用意されている。

プロファイルエージェントは、インタフェースエージェントと密接に通信しながら視聴者に関する情報（ユーザプロファイル）を管理するのが主な役割である。具体的には、1）新しい視聴者の登録やプロファイルの作成、2）学習による視聴者の嗜好や属性の更新・変更、3）プロファイルに基づく番組情報のフィルタリング、4）視聴者に関する特定の情報の検索および提供などがある。視聴者に関する嗜好の学習は、視聴履歴を基に BBN（Bayesian Belief Network）[4] 上で行われる。視聴時間（VT）と視聴日（DAY）によるある視聴者の視聴嗜好の学習結果を図表 2.1.6-3 に示す。



図表 2.1.6-3　BBN による視聴嗜好の学習の一例

ユーザモデルエージェントは、視聴者に関するより一般的な知識や常識（ユーザモデルと呼ぶ）に関する情報を提供する。この情報は、ユーザプロファイルに記述されていない新しい番組を検索するとき有効な情報源になる。

ブローカエージェントは、放送サービスの“供給者と消費者”間の仲介を行う役割を持つ。具代的には、1）現在利用可能なサービスプロバイダの紹介、2）利用できる新しいサービスの通知、3）ある特定のサービスが提供できるエージェントやプロバイダの検索などがある。このようなブローカリングを容易に行うためには、プロバイダエージェントは自分の提供するサービスに関する情報、例えば、サービスの種類や転送方式、などをあらかじめブローカエージェントに登録する必要がある。

検索エージェントは、視聴者から明確な検索指示を受けなくても、個人情報とユーザモデルに基づいて視聴者の好みに合った番組やまったく新しい番組の検索を自律的に行う。一方、システムエージェントは、仲介者ドメインに属する様々なエージェントに送られてくるACLメッセージをインターセプトして複数のエージェントの働きをモニタリングしたり調節することで、異なるドメイン間のエージェント通信を効率化する。

（C）サービスプロバイダドメイン：プロバイダエージェントはインターネット上の放送関連情報発信サイトや放送局に代わってサービスを提供するのが主な役割である。具体的には、1）番組EPGの管理、2）番組の内容に関する他のエージェントからの問い合わせへの対応、3）番組スケジュールの変更の通知や新しい番組のお知らせなどである。さらに前述したブローカーエージェントと緊密に通信することで、より効率的に視聴者に適したサービスを提供することができる。

（D）ネットワークプロバイダ

番組コンテンツの配送手段の管理・選択を担当し、ビデオサーバと配送システムを制御する。配送媒体として、衛星放送、ケーブルテレビ放送、ATMによるビデオ・オン・デマンド（VOD）などが挙げられる。FACTSでは、NHK番組データベース（約800個）やCSELTとBBCが提供する3つの番組サイトを用意して番組コンテンツの送出をシミュレートしている。

（E）エージェントプラットフォーム：エージェントプラットフォームはエージェントが活動するために必要な基盤と環境を提供する。具体的には、1）エージェント間のメッセージ通信を媒介する通信バスの提供、2）エージェントのライフサイクルの管理や他のプラットフォームへの移動のサポート、3）プラットフォーム上に存在するエージェントの移動とセキュリティーの監視などがある。FACTSではCSELT社が開発したJADE（Java Agent Development Environment）上でエージェントを開発している〔5〕。JADEは、FIPA準拠のマルチエージェントシステムのアプリケーションを開発するため、JAVA言語によって作られたエージェント開発環境である。現在FACTSシステムの実行プラットフォームとしてWindows95/NTとUNIXが使用されている。Windows95上でのエージェントテレビの実行画面を図表2.1.6-4に示す。

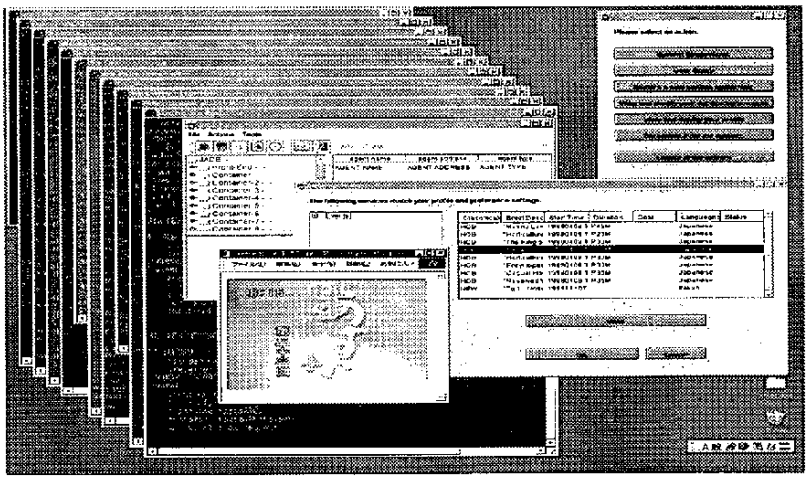


図 2.1.6-4 エージェントテレビの実行画面

（4）まとめ

デジタル放送分野における受信機の高機能化の必要性とエージェント技術を利用した「エージェントテレビ」の現状と FACTS プロジェクトを中心にその実装法について述べた。今後の課題として、1）ユーザプロファイル情報の自動獲得、2）現在話題の番組の自発的な発見、3）エージェントテレビ間の協調と交渉などが当面の課題として挙げられる。これらの課題に対してある程度の対応できる手法やアイデアもいくつか提案されている。ユーザプロファイルの獲得に対しては視聴者がすでに見た番組の頻度をもとに視聴者の好みを推論する手法や視聴者の目のまばだきや視線を観察・分析し、視聴中の番組の興味度を計る手法が考えられる。一方、話題の番組の発見を実現するための１つの方法は、現在のトピックや話題を常時提供するニュース記事や電子新聞からトピック抽出を行い、番組の選択や検索に利用することである。また、エージェント間の協調や交渉を実現する最も現実的な方法として、ドメインを制約し（例えば、放送）効率の良いエージェントインタラクションプロトコルを多く用意することである。インタラクションプロトコルは高度でかつ確実なエージェント間の通信を確保するための鍵になると考えられている。その他にもプラットフォーム間の通信技術、オントロジの共有技術、EPG の国際レベルでの標準化などさらなる進展が期待されている。

参考文献

〔1〕 金淵培：エージェント技術の放送への応用、映像情報メディア学会、Vol.52、No.4、(1998)．

〔2〕 DICEMAN: http://www.teltec.dcu.ie/diceman/index.html

〔3〕 FACTS: http://www.labs.bt.com/profsoc/facts/

〔4〕 Cattoni、R. et al.：Evaluating the FIPA standards on the field: an audio video entertainment application、In Proceedings of IAT'99 (1999)．

〔5〕 Bellifemine、F. et al.：JADE－A FIPA Compliant Agent Framework、In Proceedings of PAAM99 (1999)．

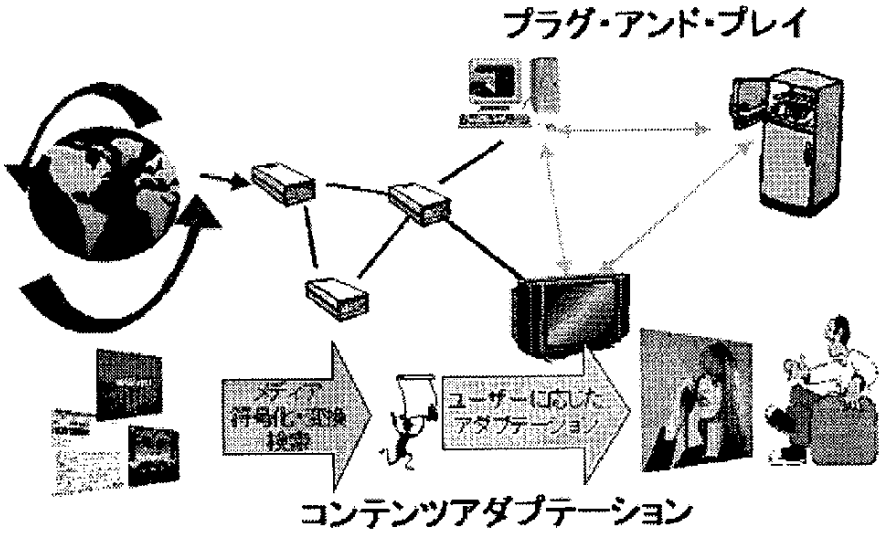
2.1.7 情報家電におけるネットワークエージェント技術

IEEE Computer 誌の記事によればシステム研究の今後 10 年のトレンドは以下のキーワードに要約できるとしてる〔1〕

1) Availability
2) Maintainability
3) Scalability

これらは，サービスがいつでもどこでも受けられること（Availability)、保守・設定が容易であること（Maintainability)、システム機能とサービス品質がスケーラブルであること（Scalability、例：システム全置換しなくても機能追加可能）が今後の研究課題であると示唆している。従来のシステム研究は、能率の向上／省資源化を目指すものであったが、以上のキーワードは“効率化”とは対をなすものである。この観点でのシステム実現技術を語順は前後するが頭文字をとって SAM 技術と呼ぶことにする。本報告では、効率化／省力化を訴求しない SAM 技術のドメインとして家庭を取り巻く情報処理が考え、その要素技術としてのネットワークエージェント技術を報告する。

家庭を取り巻く情報処理の典型として図表 2.1.7-1 に示すプラグ・アンド・プレイとコンテンツアダプテーションを取り上げる。両者に共通することは，その要素技術の一つがサービス、機能、ユーザー記述等のモジュール間通信やオブジェクト（メディア）の抽象化であり、ACL、KQML などエージェント言語とは未だ隔たりがあるものの、今後オープンな分散システム間を流れる情報流通の枠組みを作ろうとする流れが見える。



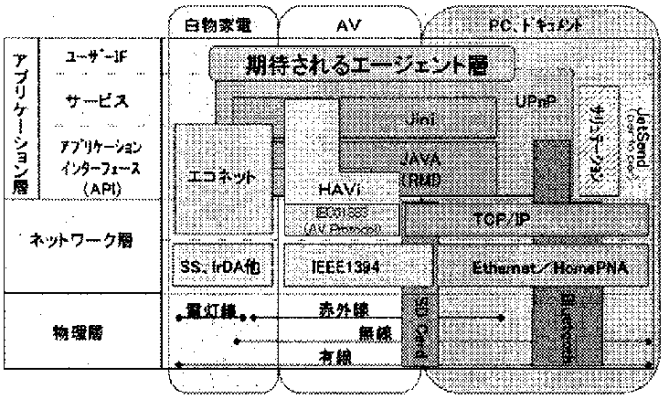
図表 2.1.7-1　ネットワークエージェント技術の家庭への応用

(1)プラグ・アンド・プレイ

家庭内ネットワークの構築は、従来のネットワークにはない SAM 技術が必要となる。この中で、ネットワークにつながったさまざまなデバイスを自動認識し、能力交換を行う仕組みをプラグ・アンド・プレイとよぶ。

図表 2.1.7-2 に、これまで提案または実現された家庭内ネットワークのプロトコルスタックの中で主要なものをまとめた。UPnP は、米マイクロソフト社提案の、機器、電話機、

PC を自動接続して使えるようにするプロトコルであり，同様の規格として、日欧家電メーカー8 社が策定した「HAVi」、米サン・マイクロシステムズが開発した「Jini」がある。UPnP と HAVi との違いは、HAVi が IEEE1394 というインタフェース規格を前提にしているのに対し，UPnP は通信路に依存しない．IP を採用することで，IEEE1394 やイーサネットのほか，電話線を使ったネットワーク規格「HomePNA」や無線の「HomeRF」などにも対応できる。



図表 2.1.7-2　ネットワークエージェント技術の家庭への応用

Jini は UPnP がプロトコルの規定に過ぎないことに対し，機器や PC が Java 実装されていることを前提とすることが大きく異なる．現在　UPnP と Jini が有力視されており，以下概説する。

①ユニバーサル・プラグ・アンド・プレイ(UPnP)〔2〕

UPnP の実行は以下のように、アドレス設定、サービス発見、サービス設定、サービス発行の４ステップで行われる。

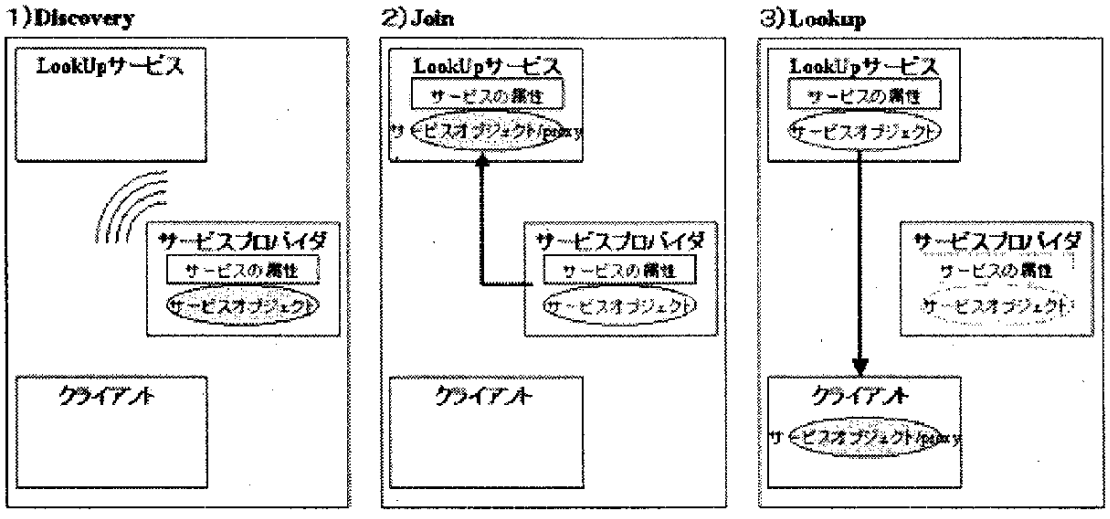
1) IP アドレス設定（IPv4 Auto-Configuration、IETF Internet drafts）
   ①DHCP クライアントとして動作し、DHCP サーバからアドレスを取得
   ②DHCP サーバの応答がない場合には、決められたアドレス範囲から使われていないアドレスを選択して使用（小規模のローカルネットワークでのみ利用可）。
2) サービス発見（SSDP、IETF Internet drafts）
3) SSDP クライアントは HTTP クライアントとして動作し，ローカルネットワーク上に存在する必要な HTTP リソース(SSDP サービス)を発見する(方法は下記の２通り)。
   ①SSDP クライアントが、UDP マルチキャストでリクエストを出し，SSDP サービスはリクエストを聞いて、マッチする場合に応答を UDP ユニキャストで応答。
   ②SSDP サービスが、自身（サービス）の存在を UDP マルチキャストで通知。
4) サービス設定：それぞれのサービス詳細（アイコン、サービスの名称等）を HTTP で取得し，サービスを起動するためのメニュー画面を SSDP クライアント側に作成。
5) サービス実行（SCP）：あるサービスが選択されると、サービスが提供するユーザインタフェース(HTML+Script) を HTTP で取得し、ブラウザを介してサービスを利用する。

サービス発見では、URL に基づいてデバイスの詳細情報（XML〔3〕）を取得し、これに含

まれるサービス定義のURLから，アイコンデータやサービス定義（SCPD）を取得する。注目すべきは、サービス定義（SCPD）はXML-schema〔4〕により表現される点である。

②Jini〔5〕

JiniもUPnPと同様，下位プロトコルは特定しないものの、IPネットワークの存在が強く意識されている。前述したUPnPとは異なり、Java上での実装を前提としており、サービス定義にはUPnPのようなXML-schemaによる構造化・抽象化はない。Jiniに接続されるすべてのデバイスはオブジェクトとして表現される。これにより、分散システムは、分散環境で提供できるサービスやデバイスの集合として理解される[6]。



図表2.1.7-3 Discovery、Join、Lookup プロトコル.

オブジェクトの連携のために２つの機能discoveryとjoinが用意されている。前者は必要なサービスを仲介するコードを見つけて、正しくリンクする機能であり。後者は２つのサービスに出会いの仲立ちをする機能である。これらは、lookupサービスにより行われる（図表2.1.7-3参照)。Jiniの Java実装による数々の恩恵(Java RMIによる分散処理環境)は魅力的であり、またルックアップサービスはサービスのレポジトリ（しかもダウンロード可能）としてメディエーターが実現されている。しかし、一方で可読性、機器接続の柔軟性と独立性ではサービス記述の定義に重きをおくUPnPが優れている。

UPnPもJiniもネットワークエージェントの要素技術であるが、家庭という場のノービスユーザーに利用されるには，さらに上位層で機器接続とメンテナンスをより抽象化して提示するエージェント層（図表 2.1.7-2）を具体化する必要があろう。ここが擬人化されたソフトウェア（＝エージェント）としての出番かもしれない[7]。

(2)ユニバーサル・マルチメディア・アクセス(Universal Muimedia Access、UMA)

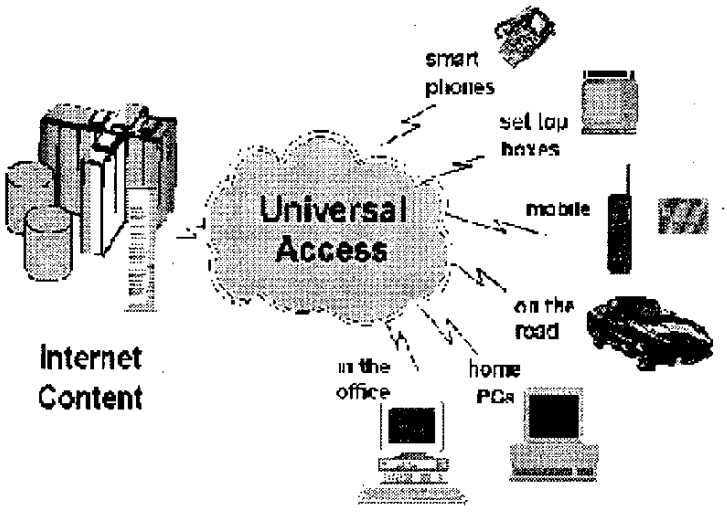
コミュニケーションシステムに人間系の要素を加えると効率化の視点にもう一つの視点

---

[6] JAVAベースの同様の開発にTSpaces - IBM AlphaWorks (http://www.almadan.ibm.com/cs/TSpaces)がある．TSpacesでは、tuple spacesと呼ばれる通信バッファとそれにアクセスするAPI群により anonymous 通信が可能となるなどJini出現以前にそのコンセプトが示されていたことが分かる。

[7] ここで擬人化とは、ソフトウェア工学の立場で、システム内部状態、入出力、挙動を人の動作アナロジーで捉えることを意味している。

が見えてくる〔7〕。それはインタフェースから見たコミュニケーション手段のあり方である。従来、高能率符号化や耐誤り特性の改善であったメディア通信も、コンテンツの内容、端末能力、ユーザー嗜好に応じてコンテンツアダプテーションを可能とすることが議論されている〔8-10〕（図表 2.1.7-4 参照）。これもメディア伝送における SAM 技術と言える[8]。



図表 2.1.7-4 Universal Multimedia Access（〔6〕より引用）

資料編Ａ．５で紹介したように、本報告者らは、ユーザー記述、端末記述、コンテンツ内容に応じて、コンテンツプレゼンテーションを可変とするモバイルエージェントを Content-Agents とよび、エージェントコード（Java コード）と共に S/MIME でカプセル化されたコンテンツが SMTP/HTTP 上で多重ホップしつつ、その内容が変化する実装を行っている〔11〕。アダプテーションのメカニズムを図表 2.1.7-5 に示す。コンテンツ、ユーザー、端末・ネットワークに関するメタデータに基づき、エージェントコードがコンテンツを処理する。これらメタデータには XML／XML-schema に基づく表現が採用／検討されている。

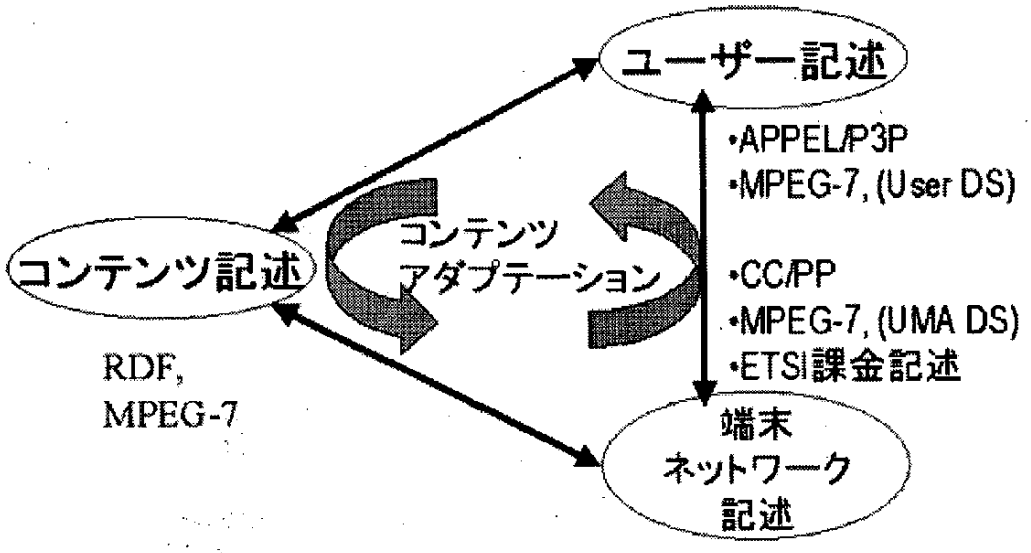
図表 2.1.7-5 の MPEG-7〔12〕は現在 ISO で標準化が検討されているメディアの構造に関するメタデータのフォーマットであり、例えば MPEG1/2/4 等で符号化されたオーディオ・ビデオ情報が、どのようなシーンで構成されているかを、書誌事項を含めて記述する。一方で WWW のコンテンツフォーマットを作成するコンソーシアム W3C ではコンテンツ書誌に RDF を、ユーザーの嗜好に APPEL/P3P を、端末能力に CC/PP を定めている[9]。MPEG-7 においても相当する記述を定めようとしている（図表中 User DS、UMA DS、DS：Description Schema）。

以上、XML（-schema）を軸にメタデータがコンテンツを扱うエージェントの重要な要素技術になりつつある。この他に、ネットワークエージェントの要素技術として重要なものにセキュリティ技術〔13〕がある。これについては別途報告する。

---

[8] 詳説は割愛するが、メディアコンテンツがスケーラブルであること、時間、経路、端末によらずアクセス可能であること、メンテナンスできることの意義は大きい。

[9] フォーマットのみであり，その解釈は応用による．また W3C の資料については，http://www.w3.org/TR/ 以下を参照のこと．



図表 2.1.7-5 コンテンツアダプテーションのためのメタデータ

参考文献

〔1〕 John Hennessy. The Future of Systems Research、Computer、August、1999.

〔2〕 http://www.upnp.org/

〔3〕 http://www.w3.org/XML/

〔4〕 http://www.w3.org/TR/XMLschema-1/

〔5〕 http://www.sun.com/jini/

〔6〕 http://www.research.ibm.com/networked_data_systems/transcoding/index.html

〔7〕 阿部圭一：“情報科学概論でも、情報工学概論でもない情報学概論の試み”，平成９年度　情報処理教育研究集会講演論文集、pp.97-100、1997

〔8〕 J.Smith、R. Mohan、and C.S. Li, “Content-based Transcoding of Images in the Internet”,Proc. of the International Conference on Image Processing、1998.

〔9〕 http://www.research.ibm.com/networked_data_systems / transcoding / Publications/

〔10〕 http://ltswww.epfl.ch/~uma/index.htm

〔11〕 栄藤、三浦　他，“Content-Agents:情報流通のためのモバイルエージェント”，第6回モバイルマルチメディア通信ワークショップ,信学モバイルマルチメディア研資，2000.

〔12〕 江村恒一，妹尾孝憲，“MPEG-7 標準化動向”，映像情報メディア学会誌，vol.54，no.3，2000.

〔13〕 三浦、栄藤：“モバイルエージェントのマルチホップ認証についての考察”，暗号と情報セキュリティシンポジウム（SCIS2000），D-19，（Jan 2000）.

2.1.8 サービス連携

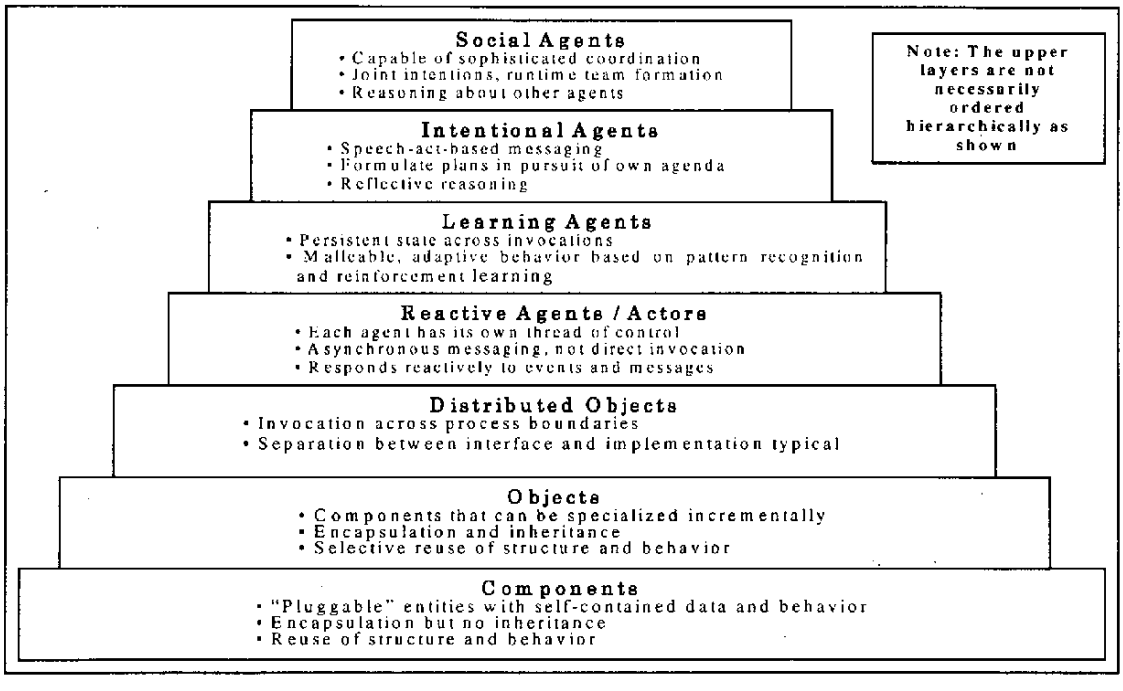
（1）ビジネス・産業応用におけるミドルウェア技術

ビジネス・産業応用においては、アプリケーション・ドメインに特化したいわば垂直的な視点とは別に、システム設計支援の立場から、より水平的な機能的ソリューションを提供するという視点が存在する。いわゆるミドルウェアと呼ばれる、アプリケーションに特化しない共通的な機能モジュール群を提供するレイヤーの技術であり、サプライチェーンマネージメントやグループウェアに代表される企業間の電子的な業務連携や移動型ユーザーに代表される複数の環境下での透過的なシステム実行において求められるものである。

本節では、このミドルウェアとよばれる水平的なレイヤーを対象に、インターネットに代表される動的でオープンな環境の中で、いかにサービスとサービスを柔軟かつ動的に連携させていくかに関して、ビジネス・産業応用におけるサービス連携とエージェント技術との関係を述べることにする。

（2）分散オブジェクト技術とエージェント技術

ビジネス・産業応用におけるサービス連携について考えるとき、分散オブジェクト技術の動向とエージェント技術との関係をまず明確にしておく必要がある。なぜならインターネットの普及・企業間連携に対するユーザー要件は、分散オブジェクト技術に対しても柔軟で動的な連携を要求しつつあるからである。



図表 2.1.8-1　Spectrum of Object and Agent Properties [1]

図表 2.1.8-1 では、オブジェクト指向からエージェント指向に至る技術要素が階層的なスペクトラムとして描かれている〔1〕。この図の中での技術的階層は、図中でも述べられているように厳密なものではなく、オブジェクト指向からエージェント指向へ至る技術要素のイメージを伝えるためのものである。しかし、この図からは、オブジェクト指向技術とエージェント指向技術とを対立的に捉えるのではなく連続的なものとして捉えるという姿

勢と、オブジェクト指向からエージェント指向技術へと至る技術要素とを見ることができる。

このようなエージェント技術をオブジェクト指向技術の延長上に捉えるという姿勢は、現時点での現実的なエージェント研究のアプローチということができる。また分散オブジェクト技術における次の技術を模索している人々にとってそれほど違和感のあるものではない。実際、たとえば、OMG の Agent Working Group〔2〕の Green Paper あるいは White Paper で述べられている視点もこのような立場によるものである。OMG の Agent Working Group は同じく OMG の Electronic Commerce Domain Task Force、Internet Platform Special Interst Group、Analysis and Design Task Force とも関連しながら活動していることからも、このような活動が単に学術的な興味からだけではなく実質的なニーズに裏打ちされていることが伺えるといえよう。

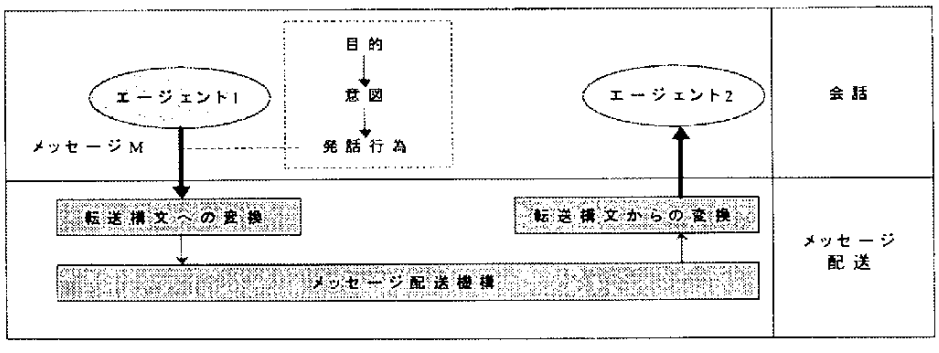
少なくとも、現在の分散オブジェクト指向技術に、図表 2.1.8-1 の、より上位の概念もしくは要素をすこしづつ取り込むことにより、システムはよりエージェント指向的になっていくということができ、またそのような方向での技術開発や試作活動が行われている。

（3）サービス連携におけるエージェント技術の適用レベル

分散オブジェクトシステム技術においては、CORBA〔3〕や DCOM〔4〕に代表されるように、ネットワークに対して透過的なオブジェクトアクセスを提供するための枠組みや分散オブジェクトのインターフェイス記述に関する枠組みが整いつつある。しかし、連携するオブジェクト同士が、互いの内部状態や内部関数に依存しあうような密な関係にあれば、相互作用によって機能を動的に構成していくことは容易でない。したがって、連携によって機能を動的に構成しようと思えば、連携するオブジェクトの疎な関係を規定しなければならない。

ビジネス・産業応用におけるサービス連携におけるエージェント技術の必要性は、いかに連携するオブジェクト間の疎な関係を規定するかということになる。その際、エージェント技術をどのように適用するかにおいてはいくつものレベルが考えられる。

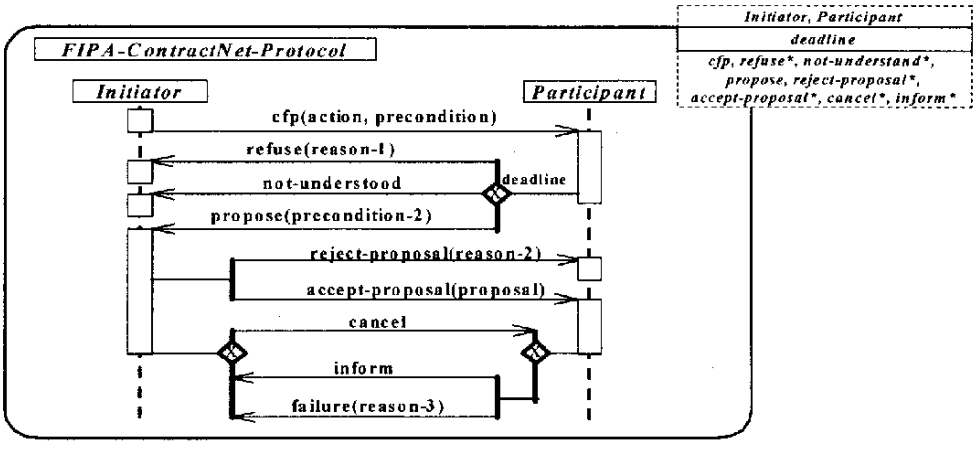
図表 2.1.8-2 は FIPA〔5〕の形式的モデルにおけるエージェント間のメッセージの流れを示している〔6〕。ここでもしエージェント１におけるメッセージ生成およびエージェント２におけるメッセージ処理が完全に目的から意図、発話行為に至る連鎖に準拠したものであれば、これは図表 2.1.8-1 における上位の階層に基づく連携ということができる。その意味で、この連携はもっともピュアで知的な連携といえる。



図表 2.1.8-2　FIPA モデルにおけるエージェント間のメッセージの流れ〔6〕

一方、図表 2.1.8-3 は、FIPA におけるエージェント間のインタラクションプロトコル〔7〕の例を示している。プロトコルで表現することにより、二つのオブジェクトの関係は、両者の会話という意味ではより限定的になる。もちろん、プロトコルによって両者の関係を表現することと、図表 2.1.8-2 に示される形式モデルとは矛盾するものではないが、プロトコル表現することによって、それぞれの内部において目的-意図-発話行為に至る実装がなされていなくても擬似的な会話が可能となる。その意味で、エージェント間のインタラクションプロトコルを規定するということは、図表 2.1.8-1 における Reactive Agent/Actor のレベルの技術による連携を許すものといえる。技術的にも、プロトコル内で用いられるメッセージに対して順次パターンマッチを行いながら会話を進めていくといったより現実的な実装を許すものといえる。またプロトコルに基づく連携において、もし事前にアクション名や変数名が決定されるのであれば、実装的には従来の分散オブジェクトシステムときわめて近いものとなる。実際、プロトコルとそこで用いられる語彙との組み合わせで連携を表現するというアプローチはエージェント指向の技術としてだけではなく、たとえば RosettaNet〔8〕におけるアプローチにも見られる。

もちろん、プロトコルを用いたエージェント間の連携表現は、エージェントらしい自律的な振る舞いを否定するものではない。逆に、動作の範囲に制限を持たせながら、メッセージに含まれる内容によって自律的にその動作を変更するなどを可能にするだろう。その意味では、エージェント連携におけるプロトコル表現は、サービス連携におけるテンプレートを与えるものとする見方も可能である。



図表 2.1.8-3　FIPA インタラクションプロトコルの例（FIPA-ContractNet-Protocol）〔7〕

（4）エージェント間の仲介（Brokerage）行為

柔軟なサービス連携を行うという意味では、１対１通信を基本とするモデルだけではなく、３者以上のモジュールによる対話モデルへの拡張が考えられる。特に仲介的機能を持つモジュールの導入によってシステムの柔軟性、拡張性、変更容易性を拡大させることが可能になる。

仲介的機能を持つモジュールがクライアントとサーバーをつなぐものであれば、このモジュールはクライアントとサーバー双方のメタ情報を持ち、その情報に基づいて仲介動作を行うことになる。このような仲介行為は、エージェントに関わる方法論としては、ファシリテータ〔9〕あるいはメディエータという名称で議論されてきたものであり、KQML を

用いたSubscribe、Broker、Recruit、Recommendとしての仲介行為〔10〕がよく知られている。

メディエータ型の情報連携のひとつの例として、通産省の企業間高度電子商取引推進事業の一環としておこなわれた鉄鋼設備CALSプロジェクト〔10〕での実証実験システムがある。

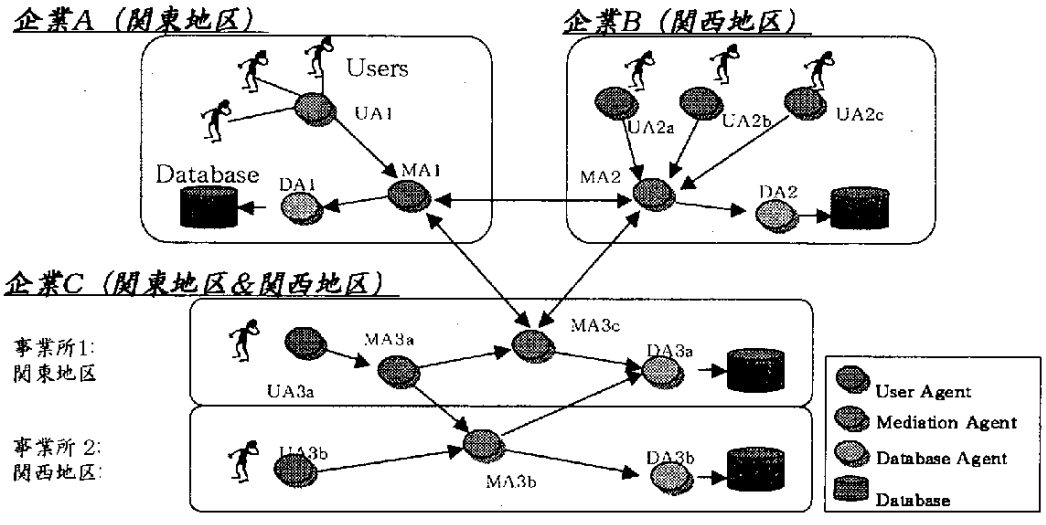
鉄鋼設備CALSプロジェクトの実証実験におけるビジネスモデルは、各製鉄所が保有する予備品の仕様・在庫情報を鉄鋼メーカー・設備メーカー間で共有することにより、業務の効率化・在庫圧縮など管理コスト削減を狙ったものである。実証実験でのエージェント間コミュニケーションにはKQMLが用いられ、データ要求とその回答ではそれぞれBrokering、Recruitingに相当する処理が用いられた。分散型メディエータを利用することで下記のような項目が実現された。

- ●各企業におけるクライアントから複数企業のデータが透過的にアクセスでき、その一方で、各企業毎のクライアントの実装・インターフェイス等の変更は各管理母体毎に閉じさせることを可能とした。
- ●共有するデータの中には技術ノウハウ情報など顧客には開示できても競合他社には開示できない情報がふくまれている。分散型メディエータがデータアクセス経路のコントロールを行うことで、管理母体毎のデータ分散配置とポリシー管理を可能とした。
- ●ユーザー名・パスワードなどは各管理母体内で管理され、他の管理母体へのアクセスにおいてはメディエータによる変換が行われる。このような処理により、各管理母体内でのユーザー名・パスワードが他の管理母体から隠蔽されると同時に、他の管理母体ではデータへのアクセスコントロールを登録されたメディエータでのみ行えばよいことになる。
- ●データ要求の流れを分散メディエータがメタデータの論理表現によるパターンマッチによって処理することで、分散データベースの配置の変更を各管理母体毎に閉じさせることを可能とした。これは同一メーカー内においても事業所毎の差異を吸収したいという要求にも答えるものである。
- ●分散メディエータを用いた複数経路での問い合わせを許すとともに、データ要求の重複を回避する処理をおこない、対障害性やシステム構成の変更に対しての柔軟性を持たせた。
- ●管理母体毎の分散管理が行えることによって、システム管理コストの費用分担を単純化させることが可能となった。

鉄鋼設備CALSの例に見られたように、分散メディエータのような仲介エージェントを用いることで、閉じた世界の利点や資産を生かしながらこれらを柔軟に結びつけ、開いた世界として取り扱うことが容易になる。図表2.1.8-4は、鉄鋼設備CALSの例で用いられた分散メディエータを、企業間連携に一般に適用した場合の模式図である。このような分散メディエータを用いた連携では、仲介するエージェントが、データベース自体のメタデータを管理させることでさらに複雑なアクセス経路コントロールをおこなったり、あるいはデータアクセスのための通信行為を、実際のデータベースアクセスのためのインターフェイスから隠蔽することが可能となるだろう。また、データベース側・クライアント側両方向

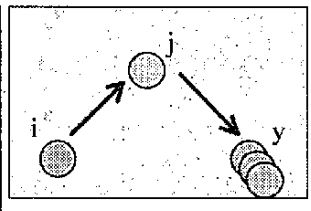
からのアドバタイズによって分散メディエータ内の情報を動的な書き換えることによって、より柔軟なシステム連携が可能になるだろう。また、実際に分散メディエータを企業間共有のプラットフォームに用いるための具体的な方式検討も行われている〔12〕。

エージェント間の仲介行為については、FIPAにおいてもエージェント間の通信行為としても議論が進められている〔13、14〕。図表 2.1.8-5 は、仲介行為を FIPA の形式的モデルで表現したものである。通信行為としての仲介(proxy)の記述は、図表 2.1.8-2 でいえば、完全にエージェント間の対話に基づくものであり、エージェント i がエージェン j に対して、『j が知っているある条件を満たす y に対して、proxy の中に記された通信行為を、条件φが満たされるとき、j が y に対して実施すること』を i が意図（期待）し、それを i が j に通知する行為を表現している。このような直接には指定できない相手 y を対象に対する仲介行為の依頼は、鉄鋼設備 CALS の例に見られた企業間連携など、さまざまな局面で必要となる比較的一般的な行為といえる。



図表 2.1.8-4　企業間連携における分散 Mediator の利用

$$\langle i,\ \mathrm{proxy}(j,\ \iota x\ \delta(x),\ \langle j,\ \mathrm{act}\rangle,\ \phi)\rangle \equiv$$
$$\langle i,\ \mathrm{inform}(j,\ \mathrm{I}_i((\exists y)(\mathrm{B}_j(\iota x\ \delta(x) = y) \wedge \mathrm{Done}(\langle j,\ \mathrm{act}(y)\rangle,\ \mathrm{B}_j\phi))))\rangle$$
$$\mathrm{FP} : \mathrm{B}_i\ \alpha \wedge \neg \mathrm{B}_i\ (\mathrm{Bif}_j\ \alpha \vee \mathrm{Uif}_j\ \alpha)$$
$$\mathrm{RE} : \mathrm{B}_j\ \alpha$$
where
$$\alpha = \mathrm{I}_i((\exists y)\ (\mathrm{B}_j(\iota x\ \delta(x) = y) \wedge \mathrm{Done}(\langle j,\ \mathrm{act}(y)\rangle,\ \mathrm{B}_j\phi)))$$



図表 2.1.8-5　仲介行為(proxy)の形式的表現

実システムとして FIPA の通信行為にみられるようなものを具体的に実装する場合には、それ相当の内部処理を行うモジュールも当然必要となってくる。たとえば、図表 2.1.8-5 における通信行為としての proxy はエージェント i から j を経て y に至る一方向の流れを表現するものであるから、当然、y からの回答を期待する場合には、会話の戻りの制御を行う必要がでてくる。また対話をより自律的に実装しようとすれば、相手のエージェント

がプロトコル外のメッセージを投げた場合の処理もある程度なされなければならない。すなわち、サービス間の連携をエージェント間の通信行為を用いてメタ表現で実装するためには、当然のことではあるが、それ相応の内部処理を行う機能モジュールが必要となってくるのである。

（５）エージェントの連携機構

ビジネス・産業応用におけるサービス連携について考えるとき、連携するエージェントあるいは分散オブジェクトの疎な関係を規定していくためには、連携のための機構についても考慮していく必要がある。

たとえばCSCW(Computer Supported Cooperative Work)分野への応用では、アウェアネス・モデルによる協調が相互作用の機構として議論されている〔15、16〕。たとえば、オーラの接触というモデルでは、仮想空間における人を表現するオブジェクトであるアバタに、その周りの空間的に広がった一定領域にオーラ(Aura)を持たせ、オーラが接触しているもの同士だけが相互作用を行い、コミュニケーション行動もその影響を受けるという方式が提示されている。またこのモデルではパラメータとして、視覚域としてのFocusや周りに対する影響力Nimbusといった概念が導入されている。現段階では、これらの概念はビジネス・応用に直接用いられてはいないが、エージェント群によるサービス連携の範囲の規定など、ドメイン管理やサービス群管理へと応用されていく可能性がある。

アウェアネスの概念をも含む一般的なメッセージの共有という意味では、いわゆる掲示板型サービスの研究も広く行われているといえる。たとえばエール大学で開発された並列分散プログラミングのための通信モデルである Linda〔17〕は、すべてのプロセスによって共有されるTuple-Spaceと呼ばれる空間を考え、そこにTupleと呼ばれるデータを送り出すことで通信を行うモデルであるが、Tuple-SpaceをAwarenessメッセージの共有ボートとして利用することができる。 Lindaのモデルに Notificationの概念を加えたものは、SunのJavaSpace〔18〕、IBMのT-Space〔19〕らによって継承・発展させられ、機器やサービス間のPlug&Playへの利用として検討されている。

さらに、Lindaを源流とした分散した複数の関数や手続きの連携についてはCordination Model〔20〕が存在し、その連携方法や計算モデルについてはデータフロー計算モデル〔21〕が適用できる。さらに、これらを一般化して、メッセージとパターンマッチとを利用したビジネス・サービス応用の研究・開発もすすめられている〔22、23〕。

（６）現状と今後の課題

鉄鋼設備CALSの例では、エージェント間の通信としてKQMLを用いているものの、実装レベルでは、図表2.1.8-1における Reactive Agent / Actorのレベルの分散メディエータに、ユーザー、データベース、ポリシーに関わる知識を持たせ、データアクセス経路のコントロールに利用した事例と考えることができる。その意味では図表2.1.8-2に描かれるFIPA の形式的モデルに基づく実装とはまだ隔たりがあるといえよう。FIPA や OMG Agent Working Group において議論されている内容でも、エージェントマネジメントやプラットフォームに関わる仕様については、メッセージ配送の方式やエージェントというインスタンスの管理に関わる部分などが中心であり、図表2.1.8-2における自律的なエージェント間の会話を通した連携という視点とは異なるレイヤーの議論が中心となっている。

一方、鉄鋼設備 CALS の事例にも見られるように、現実のシステム設計の現場からは、システムの柔軟性や可塑性、分散管理という意味での自律性などがシステム要件として強く求められつつある。その意味では、図表 2.1.8-1 のより上位の概念を取り入れたシステムが求められているといってよい。また、同時にエージェントという用語は使われていなくても RosettaNet や JavaSpace にみられるようにビジネス上の要件から分散システムに関わる技術開発が平行して進められている。これらの応用においても、真に分散的で柔軟なシステムを構築していくためには図表 2.1.8-1 の、より上位の概念を取り入れていく必要があるだろう。

このように、ビジネス・産業応用におけるサービス連携においては、図表 2.1.8-1 の階層の中からシステムとして実現可能な要素技術を抽出し、具体的な事例への現実的な適用を模索している段階にあるということができる。FIPA におけるエージェント通信言語の仕様は図表 2.1.8-2 の会話による連携を記述することを意図しているが、これを実際の具体的なシステムとしてどのように実装していくか、あるいは実装にあたってユーザー要件からくる柔軟性や可塑性をどのように取り込み、ミドルウェアの技術として具体的に実装していくかは今後の課題といえる。

（7）まとめ

本節では、ビジネス・産業応用の視点から、インターネットに代表される動的でオープンな環境の中で、いかにサービスとサービスを柔軟かつ動的に連携させていくかに関して、いわゆるミドルウェアとよばれる水平的なレイヤーを対象に、分散オブジェクト技術とエージェント技術との関係を中心に、サービス連携におけるエージェント技術の適用レベル、エージェント間の仲介（Brokerage）行為、エージェント連携機構について述べた。エージェント間の仲介行為については、その具体的な事例として鉄鋼設備 CALS システムを紹介し、また FIPA の形式モデルによるよ記述を紹介した。鉄鋼設備 CALS についてはまた、分散メディエータというサービス連携のためのエージェント技術を用いることにより、クライアントからのデータへの透過的アクセス、データとポリシーの管理母体毎の分散管理、システム構成変更への柔軟な対応などが可能になったことを述べた。

ビジネス・産業応用の視点からみると、エージェント技術を用いたサービス連携は、オブジェクト技術の自然な発展形とみることができ、その意味からもサービス連携に関わるエージェント技術の研究開発は今後の分散システム構築において重要な鍵を握る技術といえるだろう。

参考文献

〔1〕 Jeffrey M. Bradshaw: 本図は出版されたものではなく、Bradshaw 氏の好意により掲載するものである。Bradshaw 氏については、下記の URL を参照してほしいというのが本人の希望である。http://www.coginst.uwf.edu/~jbradsha/

〔2〕 OMG Agent Working Group: http://www.objs.com/isig/agents.html

〔3〕 CORBA: http://www.omg.org/

〔4〕 DCOM: http://microsoft.com/com/comPapers.asp

〔5〕 FIPA: http://www.fIPA.org/

〔6〕 FIPA: FIPA97 仕様 第２部 エージェント通信言語, http://ias.comtec.co.jp/fIPAtrans/
〔7〕 FIPA Resoponse to UML 2.0 RFI http://ftp.omg.org/pub/docs/ad/99-12-03.pdf
〔8〕 RosettaNet: http://www.rosettanet.org/
〔9〕 M. R. Genesereth: An Agent-Based Framework for Interoperability, Software Agent, MIT Press(1997) , pp. 317-345
〔10〕 T. Finin, Y. Labrou, J. Mayfield: KQML as an Agent Communication Language, Software Agent, MIT Press(1997) , pp. 291-316
〔11〕 竹腰 篤尚：鉄鋼設備 CALS 実用化究, http://www.ncals.cif.or.jp/ncals/pdf/steel.pdf
〔12〕 毛利他: 企業間情報共有のためのマルチエージェントプラットフォーム, CALS/EC Japan 1998 論文集, pp363-370
〔13〕 FIPA Baseline document, http://fIPA.comtec.co.jp/fIPA/Common_docs/f1PA9122.doc
〔14〕 市来他： A proposal on Agent Brokerage, http://fIPA.comtec.co.jp/fIPA/Common_docs/cfp5_008.html, Brokerage 関連の最新の状況については, FIPA: http://www.fIPA.org の ACL 関連仕様を参照のこと。
〔15〕 C. Greenhalgh, S. Benford: MASSIVE: A Collaborative Virtual Environment for Teleconferencing, ACM Transaction on CHI, Vol.2, No3(1995) , pp239-261
〔16〕 T. Rodden: Populating the Application: A Model of Awareness for Cooperative Applications, Proc. CSCW'96, pp. 87-96
〔17〕 N. Cariiero, D. Gelernter: Linda in Context, Communication of the ACM, Vol. 32, No. 4(1989) , pp. 444-458
〔18〕 Sun, JavaSpaces Specification: http://www.sun.com/jini/specs/
〔19〕 IBM, T Space: http://www.almaden.ibm.com/cs/TSpaces/
〔20〕 G. A. Papadopoulous, F. Arbab: Coordination Models and Languages, Advances in Computers, Vol. 46 (1998) , pp. 329-400
〔21〕 M. Amamiya, R. Hasegawa: Dataflow Computing and Eager and Lazy Evalution, New Generation Computing, Vol. 2(1984) , pp. 105-129
〔22〕 T. Iwao et al.: Flexible Multi-Agent Collaboration of Field Reactor Model, LNAI, Vol. 1733 (1999) , Springer, pp. 1-15
〔23〕 岩尾他: Field Reactor Library を用いた分散オブジェクト連携, Proc. DICOMO'99, IPSJ Symposium Series Vol. 99 (1999) , No. 7, pp525-530

2.2 インタフェースエージェント

2.2.1 知識メディアとしてのパーソナルガイドエージェント

本節では人間同士あるいは人間とサイバースペースのコミュニケーションに介在するエージェントをマクロな立場でインタフェース・エージェントとしてとらえる。あえて、「インタフェース・エージェント」を定義すると、「人間やコンピュータシステムのためのあるタスクをもつ自動プログラムで、自律性があり、life-likeness（生き物のような外見や性質のこと）を有して、ユーザである人間とのインタフェースまでかかわるソフトウェアなどの概念」となる。例えば、コンピュータグラフィクスやロボットによる人間型（あるいは動物型）の外見をもち、サイバーミュージアムの展示や電子図書館を案内するような知的システムで自律性を備えているものである。銀行のATM画面に現れるキャラクタを思い浮かべてもよい。現在は誰がきても同じ対応しかしないが、これらが、ユーザ認証後に個別の金融商品売り込みやアドバイスする機能をもつようになれば、インタフェース・エージェントに分類してもよいと考える。

この節では、インタフェース・エージェント技術が要求される背景を整理し、C-MAPというモバイルアシスタントシステムの実装の解説を通して、インタフェース・エージェントの現状と課題を提示する。

（1）背景

1）インタフェースとしてのエージェント

情報処理技術の進歩によってコンピュータが得意とする記憶、計算、検索の能力は人間の能力をはるかにこえてきており、コンピュータの作業内容や結果をそのまま提示しても人間には理解できない状況にある。サイエンティフィック・ビジュアリゼーションや情報可視化といった技術が、膨大な数値データや計算過程を人間に理解しやすく変換して提示するために開発されたように、検索や商取引のようなトランザクションの量が膨大・複雑になったときに、人間にわかりやすく情報を提示したり、人間側の要求や状況を積極的に摂取して橋渡しをするインタフェースとしてのエージェントが考えられる。現在のところ、人間的な外見をもつ擬人化されたキャラクタをもちいることが、多く試みられている。人間世界とコンピュータ世界（サイバースペース）とのメディア変換をするものと考えてもよい。

2）自律的なエージェント

情報技術産業の多くは、ユーザ中心・カスタマー中心のサービス展開をしている。エージェント技術は、画一的なサービスだけでなく、個人の状況に合わせたカスタムサービスを可能にする。ユーザモデリングをし、サービス情報そのもの、サービスメニュー、あるいはサービスの提供の方法を個人化する技術として、ユーザとインタラクションをしながら自律的に動作するコンピュータシステム、すなわちインタフェース・エージェント技術がある。

3）知識メディアとしてのエージェントシステム

人間同士のコミュニケーションあるいは人間と知識ベースのコミュニケーションにコンピュータが介在し、相互理解、話題の整理あるいは話題の展開などの側面を支援する研究分野やアプリケーションがある。そこでは知識そのものがメディアとしてコミュニケーシ

ョンを成立させる。知識の生成、変換、交換、流通、形成、蓄積を可能にするために、ネットワーク上を移動可能で知的なエージェント技術への要請がある。

4) 実世界指向コンピューティング

情報技術は、拡張現実感、ユビキタスコンピューティングやウエラブル、あるいはペットロボットなど、従来の概念のコンピュータやインターネットにとらわれない展開をしている。これらは実空間との関わりが強く、実世界指向コンピューティングと呼ぶことが出来る。実世界ではユーザの状況は多様であり、それを常にセンシングして利用する技術が必要である。インタフェース・エージェント技術はそのひとつと考えられる。

5) エージェントの社会性

エージェントが擬人性をもち、個人化などの知的能力を備えると、個性や仮想人格などの社会性が生まれる。これらをうまく利用することで、コンピュータシステムを人間社会にスムーズに導入できる可能性がある。

（2）C-MAP：文脈を利用した携帯ガイドシステム

インタフェース・エージェント技術の実現例として、実世界指向のモバイルガイドシステムとして試作されている C-MAP(Context-aware Mobile Assistant Project)を紹介する。

C-MAP は、ミュージアムや展示会などのガイドとして活躍できるエージェントシステム提供することを目的としている。ユーザとのインタラクションの中からユーザの興味や背景情報を獲得してユーザの状況を把握することで,「自律的に情報フィルタリングをして展示見学をガイドするパーソナルなアシスタント」と、「展示者と見学者の時空間的な制約をとりはずしてコミュニケーションを活性化させる機能」の提供を目標としている。別の見方では、それは、人間同士あるいは他のエージェントとの非同期的なあるいは同期的なコミュニケーションに介在してコミュニケーションを支援するというタスクになる。また、擬人性を有してノンバーバルな表現で作業内容や状態を可視化したりすることも、ユーザに優しいインタフェースを提供する上で必要な条件と考えられる。

上記のような機能をもつエージェントを実現するため、C-MAP では個人携帯端末と環境に埋め込まれた情報提示キオスクや位置検出装置を使って、下記のタスクを設定している。

1) システムとのインタラクションにより個人の物理的・意味的状況を収集する
2) システムはユーザに個人化されたガイド情報を提供する
3) システムはユーザの見学プランを立てて推薦する
4) 擬人化されたエージェントはマルチモーダルインタフェースを提供する
5) エージェントは社会的インタラクションを調整する
6) エージェントは実世界とサイバー世界を行き来して誘導する

以下の続く節では、C-MAP における試作、実験、評価など、それぞれの段階での試作仕様と評価について説明する。

（3）C-MAP 試作システムの初期仕様（携帯 PC 版）と問題点

1) システム概要

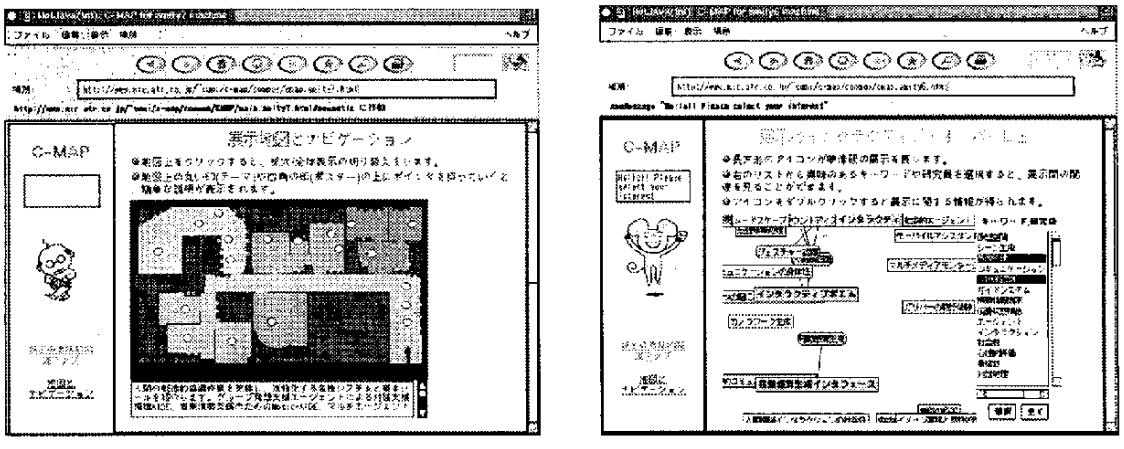
当初システムは、携帯端末としてサブノート型の市販携帯パソコンを利用した。情報ブラウジングなどの操作および情報提供はその携帯パソコンの GUI を通して行った。

図表 2.2.1-1　サブノート PC 型携帯ガイドと ABS 位置センサを使っている様子

システムは基本的に、展示情報やガイド情報を提供するサーバと、無線 LAN を介してネットワークと常時接続する携帯パソコンで構成している。さらに Olivetti 社のアクティブバッヂシステム（ABS）により各ユーザの位置情報を入手するようにしてある。携帯パソコン上のガイドシステムは、WWW ブラウザを介して、視覚的に展示空間の案内を行ない、会場の地理的案内をする地図アプレットと、展示間の意味的な関連を可視化したセマンティックマップを切り替えながら利用する。

携帯ガイドシステム上では、ユーザと会場のコンテキスト（文脈）に応じて次に見学に行くべき展示を推薦する。展示推薦を計算するための制約としては、ユーザが Semantic map 上で選択した複数のキーワードで形成される興味ベクトルと各展示を意味的に定量化しているキーワードベクトルの近さ、ユーザの見学履歴、ユーザの現在位置と各展示との間の距離、展示サイトの込み具合い、展示デモのスケジュール、といったものを利用する。展示推薦計算はコンテキストの変化、つまり、ユーザが新たにキーワードを選択し直したときやユーザが展示サイトを移動したときに行われる。

携帯端末を利用している様子と、端末上の画面の例を図表 2.2.1-1、 図表 2.2.1-2 に示す。



図表 2.2.1-2　展示項目の地理的・意味的案内図

b) エージェントキャラクタの設計

インタフェース・エージェントの擬人性がもたらす機能のうち重要なものを、下記に列挙する。

1) エージェントの作業空間に対する認知性： 振舞い(behavior) の予測や作業(action) の可視化が可能であり、エージェントの作業や振舞いに対する認知性があがる。
2) エージェントの操作性：大規模な情報検索、パターンマッチング、情報提示などにおける膨大なデータを直接見せられても人間であるわれわれは操作できないし処理しきれない。しかしエージェントを介在することによって、これらの情報処理指示、過程および結果の操作が容易になると考えられる。たとえば、エージェントにメタファー的あるいはアイコン的なジェスチャをさせることによって、メッセージの理解や記憶が容易になる。エージェントが存在する空間における身体性も対象となる。
3) 親密性：ツールとして親密性が上がり慣れ親しむことによって、操作性を向上させ、創造性を刺激する可能性がある。

本プロジェクトでは、親密性を高めるため、ブラウザのフレーム内で表示される擬人化キャラクタが 12 個用意された。ユーザには端末貸し出し時に好みのキャラクタを選択してもらい、モーバイルでパーソナルなユーザのガイドエージェントという色合いを強めている。

c) 実験システムの問題点

システムは、研究所公開展示の際に、来訪者に端末を貸し出し、研究展示をガイドするという設定で実証実験を行なった。期待通りに動作した部分もあったが、以下のような問題が浮き彫りにされた。

1) 公開実験中のユーザの主観評価によると、携帯パソコンは重くて使いづらいというコメントが多い。グラフィカルなユーザインタフェースや十分な計算と通信の能力を用いたパーソナルガイドを行うには、現状ではパソコンを携帯するか身につける必要があるのは事実であるが、見学者の本来の目的は実際の展示を経験し人と会うことであるから、現状の携帯パソコンはまだ適切なサイズやインタフェースを提供しているとは言い難い。
2) たくさんの携帯パソコンを準備し、貸し出し、回収したり、ABS のような特別なデバイスを会場に設置するのは主催者にとって大変なコストがかかる。このようなシステムは、結局、特別な装備をした空間内でのほんの短い時間だけの利用に限られ、ユーザと常にいっしょに行動を共にすることができないため、ユーザのコンテキスト情報の獲得の可能性を強く狭めてしまう。
3) ガイドサービスを行うための展示情報や、サービス中に獲得されるコンテキスト情報は同期的/集中的にサーバに蓄積されるようになっていたので、システムアーキテクチャとしての拡張性（スケーラビリティ）に欠けた。

（4）システム設計の変更

ユーザひとりひとりが持ち歩くのは、充分なグラフィカルディスプレイの能力を持ったパソコンではなく掌サイズの PDA (Personal Digital Assistant) であって、それを据え置きの展示ディスプレイや情報キオスク（通常のパソコン）と一時的に接続することでそ

れらを個人化する枠組を採用することにした。展示見学ガイドシステムは、知識メディアの位置づけをもっており、コミュニティを支援するシステム（コミュニティウェア）への展開を考えると、展示者だけでなく見学者も含む誰もが知識の発信者かつ受信者となり得る枠組にする必要がある。

従来から、博物館や観光地案内の展示ガイドシステムの中でやり取りされる知識はあくまでも展示者・情報提供者から発信されるものであり、それを見学者・観光者個人個人が自らの文脈に合わせて利用できるような環境をつくることに主眼が置かれていた。しかし、展示対象が博物館や美術館のように展示者から見学者への一方的な知識の流れを前提としているものだけでなく、例えば、研究所見学や学会のように展示（発表）する側とそれを見る側に対等な立場としての知識の行き来が期待できるような応用を考えたとき、見学者も展示参加を通して情報や知識を発信できるような枠組を再構築することができよう。それをさらに発展させると、個人が観光地の情報をレポートする特派員のようになって、情報を発信する立場にまわることも考えられる。

a) コンテキスト情報の獲得

見学しながら、ユーザ自らグラフィカルユーザインタフェースによる直接操作（direct manipulation）によって情報アクセスすることは面倒であり、もっとガイドエージェントが自発的に情報のフィルタリング、加工、提示をして、エージェントがシステム--ユーザインタラクションを主導するようなシステムデザインをする必要がある。

b) システムの拡張性

ユーザの行動をトラッキングするために特別なデバイスを環境に埋め込むよりは、緩いインタラクションでも動作する robust で分散的なアーキテクチャが好ましい。つまり、どこにいても同じ情報にアクセスできるような均一的なアーキテクチャにするのではなく、各ノードは物理的/意味的に近傍なノード間の局所的制約のみで規定されるようなネットワークアーキテクチャの上でシステムを構築する。

c) プライバシー管理

コミュニティ内での互いの存在、興味、知識のアウェアネスを高めることはコミュニティウェアの重要な機能の一つではあるが、無制限にそれを高めると各人のプライバシーを侵害する可能性がある。例えば、興味を共有する人同士の出会いを支援するにはそれらの人のプロファイルや行動履歴を利用することが望まれるが、それらの個人情報を公開したり利用されることを喜ばない人もいるであろう。その一方、学会のようなコミュニティの場合、自分が何かに興味を持っている事実や、ある展示/発表に対して議論した内容を、コミュニティに対して開示することが参加者の利益になることもある。コミュニティアウェアネスと個人のプライバシーの間のバランスには明快な回答は存在せず、その場その場に応じて設計しなければならない。C-MAP の改良にあたって、システム利用の際に毎回ユーザ認証をし、いつどこで何を見て誰と名刺交換をした、といったような個人的な行動履歴は本人だけが見れるようにし、本人が明示的に自分の興味を開示した場合はその事実をコミュニティに対して提示するようにシステム設計した。

d) システムデザイン

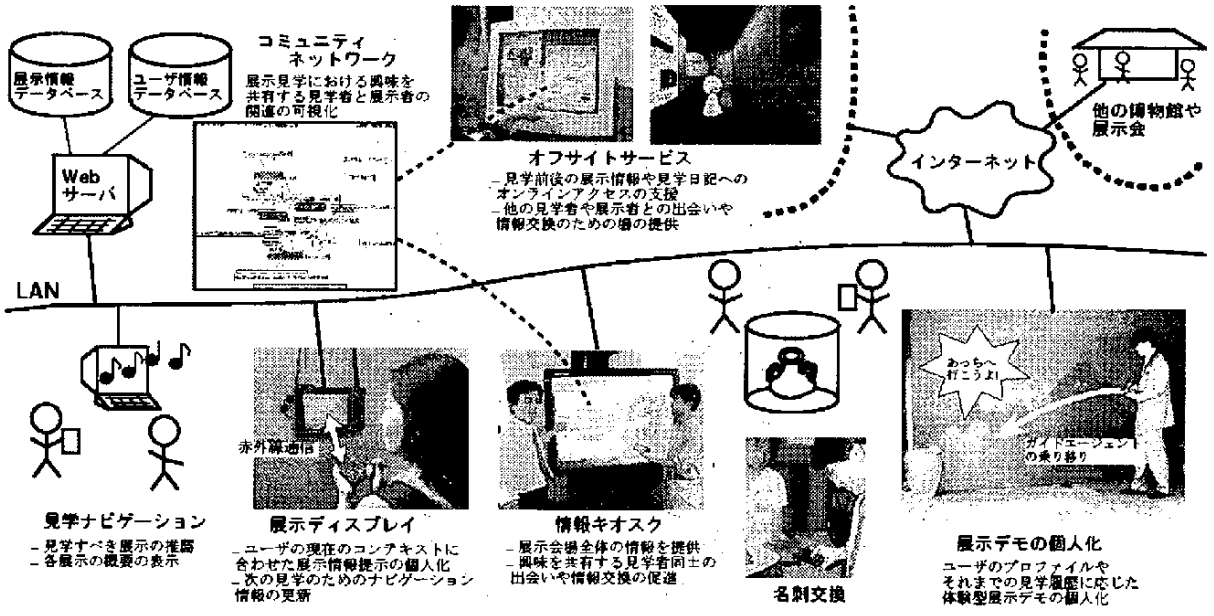
現在、携帯端末として掌サイズの PDA（Personal Digital Assistant）、具体的には 3Com PalmIII と IBM WorkPad を利用して PalmGuide と呼ばれるシステムを用いている（図表

2.2.1-3)。ユーザインタフェースのためのデバイスは小さな白黒液晶のタッチパネルだけなので、ユーザに提示できる携帯情報は簡単なハイパーテキスト程度である。PalmGuide は無線でネットワークに接続しているわけではないし、ABS のような位置検出システムを使っているわけでもない。したがって、マルチメディアを駆使したガイド情報の提示はできないし、ユーザの位置情報に応じた proactive なサービスを提供できるわけでもない。そのかわりに、タッチパネルディスプレイを持った据え置き型パソコンを展示会場に遍在させる方法をとっている。各展示ブースに専門情報を提供する展示ディスプレイを設置し、展示会全体に関わるグローバルなサービスを提供する情報キオスクを数台設置している。PalmGuide と据え置きパソコンは赤外線ポートを持っているので、ユーザは自分の PalmGuide を展示ディスプレイ／情報キオスクと赤外線リンクすることで、状況や場所に応じて個人化されたガイド情報を得ることができる。

この方法の特徴は、手元（PalmGuide）では個人的な情報（プロファイルや見学履歴）の管理と個人のためのガイド情報（次の見学の推薦）の提示といった最低限のサービスを行い、各展示に依存した情報やコミュニティで共有するサービスは会場に偏在／遍在させる、といった割り切りにあり、mobile/wearable computing と ubiquitous computing の適当なバランスを提案するものである。

この手法は、初期のプロトタイプに比べて、運用の実現性、携帯性、応用の拡張性について優れていると考える。つまり、展示者側からすると、特別な端末や設備を準備したり貸し出したりする必要がなく、ユーザが普段から持ち歩いている手帳型の情報端末を利用すれば良いので、（博物館等に限らず）一般的な社会での知識流通メディアとして普及する可能性が高い。また、ユーザや環境の文脈情報や展示情報が分散管理されているので、システムの拡張性やプライバシー管理の点で優れていると考える。

図 2.2.1-3　携帯端末 PalmGuide



図表 2.2.1-4 C-MAP システム構成と利用シナリオ

図表 2.2.1-4 に、現在のシステム構成とその利用シナリオを示す。以下、各サブシステムを概観する。

①展示情報のプレビュー

電子的にアクセス可能な展示情報をオンラインで提供することで、見学者は展示見学に先駆けて展示情報をプレビューすることができる。

②PalmGuide と展示ディスプレイの連携による個人ガイド

PalmGuide 上には個人ガイド（展示概要表示、関連展示の推薦等）が提供される。

③参加型展示の個人化

ガイドシステムが蓄積したユーザの文脈情報を利用することで、見学者参加型の展示を個人化することができる。

④見学中のマルチメディアメモ

携帯ガイドが提示する情報を享受するだけでなく、ユーザが見学中に感じたことや思いついたことをメモし、場合によってはそれを相応しい人に対するメッセージとして発信する。

⑤見学日記

PalmGuide と展示ディスプレイ／情報キオスクとの接続は電子的な足跡を残すので、それを時間軸で構造化すると、自動的な見学日記ができる。

⑥名刺交換

ユーザは PalmGuide 同士を向けて仮想的な名刺交換をすることができる。

⑦情報キオスク

展示会場のバックボーンに蓄積された展示情報やユーザ情報を利用したコミュニティサービスを提供するための据え置き端末である（図表 2.2.1-5）。

図表 2.2.1-5　情報キオスク

⑧コミュニティネットワークの可視化

プレビューサービスや展示会場において獲得・蓄積されたユーザの文脈情報を構造化することでコミュニティネットワークを構成し、興味の共有するユーザの間の出会い支援やコミュニティ形成を助ける。

⑨仮想展示空間への再訪問

時空間的な制約のために展示会場では会えなかった人とのコミュニケーションを支援するために、展示見学後にオンラインで再訪問できる仮想展示空間を提供する。

（５）PalmGuide 版 C-MAP システム

掌サイズの PDA を利用して PalmGuide と呼ばれる新しい携帯ガイドシステムを実装し、PalmGuide と会場に遍在するパソコンを連携して情報提示を個人化する枠組となっている。動作シナリオは次のようになる。

PalmGuide 上では、それを利用しているユーザのためのパーソナルなガイドエージェントが動作しており、展示見学のお供をしてくれる。ガイドエージェントは、手元では、次の見学のお勧めや、その概要を示してくれる。各ユーザが携帯している PalmGuide と各展示ディスプレイは赤外線によって通信を行うことで、ガイドエージェントは展示ディスプレイに乗り移り、そこに偏在する個別展示の詳細情報を取り込み、ユーザ向けに個人化して提示してくれる。ユーザに関する個人情報（ユーザ ID、見学履歴、興味の対象等）は、ガイドエージェントが PalmGuide 内で管理しガイド情報の更新に反映する。

ガイドエージェントは更新されたユーザの文脈情報に応じて、PalmGuide 上のガイド情報（次に見学するべき展示の推薦等）を更新する。現在は、各展示ディスプレイ内にガイド情報のためのシナリオデータベースがあり、ユーザの現時点の見学履歴に応じて加工し直したガイド情報を PalmGuide に返送する、という方法で実装してある。

見学中にある展示を熱心に見たこと、誰かと喋ったこと、展示デモを体験したこと、映像や音声を記録したこと等は、それ自体はそれぞれ独立した事象の集まりでしかない。しかし、ユーザと行動を共にしているインタフェース・エージェントは、その各事象のユーザの文脈（時間や場所、そのときの興味等）を蓄積し、それらの事象に自動的にインデックスを付加する。したがって、ユーザが意識すること無く、ユーザが展示会場で行なった実世界インタラクションには緩い構造が与えられる。そこを特定の視点（あるユーザの文脈）で再構成することで、例えばそのユーザに関連する情報が収集され緩い構造を持つ（コミュニティネットワークのその一例）。その情報は、そのユーザにとっては、自分の興味に関連する展示や他の見学者・展示者との出会いを促進してくれるものであり、知識としての利用価値があるのではないかと考える。

新しいアーキテクチャに基づいたシステムは、1999年6月に開催された人工知能学会全国大会と、1999年11月のATR研究発表会の会場サービスとして提供された。利用者アンケートの結果、PalmGuideの携帯端末としての使用感は、1997年の携帯パソコン端末版のときに比べて、概ね良好であった。また、ガイドエージェントがPalmGuideと遍在ディスプレイの間を行き来するアーキテクチャについても、比較的問題なく受け入れられた。ただし、その方法を楽しく役立つものとして受け入れるユーザと、アクセス作業を煩わしいと感じるユーザの主観評価に大きな差がある。後者のようなユーザにとっては、ABSのような赤外線バッジのみを貸し出し、見学日記が提供されるようなサービスが適していると考える。

（6）関連する研究

ものを介した実世界インタラクションへの回帰については最近多くの研究がなされており、それらの多くがmobile/wearable/ubiquitous computing技術を利用したcontext-awareアプリケーションを提案している

西田や石田ら〔1〕は、コミュニティ内での日常的な知識の共有を促進するエージェント技術やインタフェース技術について体系的に研究を進めており、角ら〔2〕や門林ら〔3〕と動機やアプローチが大変似通っている。Lammingら〔4〕は、日常的に携帯する情報端末を利用してユーザの外部記憶能力を強化する枠組を提案した。その際、ユーザと行動を共にする携帯情報端末が自動的にユーザの文脈情報を獲得して様々な活動に自動的にインデックスを付加し、知識としての利用価値を高めることを指摘した。Lammingらによるシステム Forget-me-notは、獲得した文脈情報（イベントのリスト）を単純に蓄積し続ける個人データベースであったが、C-MAPはさらにその蓄えられた情報をコミュニティ内の知識コミュニケーションに活用することに焦点をあてている。

C-MAPのコミュニティネットワークは、人間と知識の関連性をネットワーク表現することで、情報検索、情報探索、コミュニケーション支援に利用しようとする点で、Kautzら〔5〕の言う社会ネットワーク（Social network）と似ている。また他にも関連研究として、ネットワーク上の情報を構造化することで興味の近い人を探すマッチメーキングエージェント、コミュニティ形成を支援するためのコミュニティ可視化システム、興味の近い人の消費活動に関するデータを利用した協調フィルタリングによる推薦システム（例えば、Ringo）等が提案されている。

（7）まとめと今後の課題

展示見学のガイドシステムのプロジェクトを紹介することで、実世界コンテキストに埋め込まれたコミュニティウェアの考えを示し、context-aware なパーソナルガイドエージェントの構築例を示した。このようなインタフェースエージェントを構築することは、情報サービスの個人化だけでなく、興味に基づいたコミュニティ形成やコミュニティ内での知識交流も促進すると考える。

今後の課題として、

①現行の展示ガイドシステムに限らず日常的な個人アシスタントとしてシステムを拡張し、もっと様々なユーザのコンテキストを獲得しサービスを提供できるような一般的なパーソナルなインタフェースエージェントを構築する必要がある。

②扱われる情報の権限やプライバシーといった社会的な課題に対しての対処を、今後も継続的に考えていかなければならない。

③非同期的/分散的なコミュニケーションを可能にするようなコミュニティウェアを実現するには、人の分身としてのエージェントの自律性、仲介能力が重要である。したがって、ユーザの代理となるエージェント同士が半自動的にインタラクションできるような、様々なゴールを持つスクリプトによって表現されたエージェントの知識の外在化、通信、表現のための手法が必要である。

④エージェントの擬人性については常に効率性の土俵による評価において否定的な立場に置かれている。しかしながら、コミュニケーションを促進するというような、効率で議論できないような対象分野においては、十分な議論がなされていないとみるべきである。

参考文献

〔1〕 T. Ishida, T. Nishida and F. Hattori: Overview of community computing. In Ishida editor, "Community Computing: Collaboration over Global Information networks, chapter 1, pp. 1-11. John Wiley & Sons, 1998.

〔2〕 角ほか: C-MAPContext-aware な展示ガイドシステムの試作", 情処論, Vol. 39, No. 10, pp. 2866-2878, 1998.

〔3〕 門林, 間瀬: 実空間でのコンテキストを利用して仮想空間内をガイドするマルチモーダルなパーソナルエージェント. マルチメディア, 分散, 協調とモーバイルシンポジウム, pp. 653-660, 1998.

〔4〕 M. Lamming and M. Flynn: Forget-me-not, Intimate computing in support of human memory. In International Symposium on Next Generation Human Interface'94, pp. 150-158, FRIEND21. 1994.

〔5〕 H. Kautz, B. Selman, and M. Shah. The hidden web. AI magazine, Vol. 18, No. 2, pp. 27-36, 1997.

2.2.2 心を持った機械達

（１）概要

最近のヒューマンインタフェース研究で提唱される人間中心設計とは、高機能化・複雑化を目指してきた旧来からの物作りに対するアンチテーゼでもある。この設計思想は、単なるインタフェース・デザインに留まらず、物作りの方法論にも影響を与えている。従って、現在の人と機械のインタフェースに関わる技術開発は、非常に広い研究領域で行われることになった〔1〕。本稿ではそれらの研究領域の中でも、インタフェース・エージェントに関連する技術動向として、インタフェース・デザインの観点とアーティフィシャル・マインドの観点から「心」を重要視するアプローチを中心に述べる。くわえて、関連する応用例を紹介する。

（２）インタフェース・デザイン

インタフェース・デザインにおける２つの議論を取り上げ、インタフェース・エージェント技術に対する課題と期待を抽出する。

①直接操作とインタフェース・エージェント

80 年代より GUI (Graphical User Interface) 設計方法について「直接操作 (Direct Manipulation)」という概念が取り上げられている。本方法論に従えば、コンピュータの利用者は、例えばデスクトップなどのメタファーを活用したインタフェースを介し、すべてを明示的かつ直接的に操作する。従って、利用者はすべての入出力イベントを監視することが求められる。しかしながら、訓練されていない利用者が非常に多くの複雑な情報を扱う場合には、このようなメタファーで十分であるかは疑問である。A.Kay は「間接管理 (Indirect Management)」という概念を提案している〔2〕。ここでは、『私達は道具を操作 (Manipulation) し、人を管理 (Management) する。』という彼の言葉からも分るように、インタフェースとしては「直接操作」とは異なるメタファーでなければならない。端的に言えば、インタフェース・エージェントとは間接管理におけるメタファーである。

「直接操作　対　インタフェース・エージェント」のテーマで、B.Shneiderman と P.Maes がディベートをくり返している〔3〕。B.Shneiderman は、『擬人であるとかソーシャル・インタフェースであるとかいったものは、コンピューティングの将来にはない。また、擬人化表現は設計者を過った方向へ導き、ユーザを欺くものであり、コンピュータ利用時の不安を高めるものである』と述べている。一方で、P.Maes は『知的エージェントや自律エージェントなどといった言葉は使いたくない。エージェントはユーザの行動、好みや興味を知っており、それらの情報に基づいて行動を起こす。』また、『（FireFly のように）エージェントは直接に見えないところで働いて価値ある成果が得られた。』などと述べている。そして、これらの点については、B.Shneiderman も賛同している。擬人化や知的などのキーワードを使った Old P.Maes と、このような発言をする P.Maes を区別して P.Maes に賞賛を与えた。P.Maes 本人は明言はしていないが、「擬人的表現」に対して否定的な立場であるという点で両者が一致したとも受け取れる。さらに一方で、彼等はユーザがすべての操作を直接的に実行することは不可能であり、インタフェースとして全面に出てこない裏側で動くようなエージェントについてはその存在を認めている。

これら一連の議論から、インタフェース・エージェントに対する期待は大きいものの、

「インタフェース・エージェントとはどう有るべきか」という点では、明確な結論はまだ得られていないと言える。

②ソーシャル・インテリジェンス

B.Reeves&C.Nass による Media Equation 理論においては、「人間の進化というのは２０世紀の技術の進展に追従していない。すなわち、最新のコンピュータやメディアに対する人間の反応は、人に対する反応と社会的には等しい。」と主張している〔4〕。Reeves&Nass は、Media Equation の研究成果を踏まえ、インタフェース・デザインの方法論の構築を進めている〔5〕。このように機械やシステムの社会的役割の適切な設定や機械の振る舞いを適切に設計することで、機械にソーシャル・インテリジェンスを与えることができる。このようなソーシャル・インテリジェンスを備える応対システムは、人と機械のインタラクションにおいて利用者である人間にシステムの自然な解釈を可能とし、機械の特徴や目的を自然に伝え、電子商取り引きにおいてはユーザの購買行動を促進するなどの利点が報告されている〔6〕〔7〕。

①で述べたように、インタフェース・デザインという観点からは、インタフェース・エージェントはその内側の理解（人によるエージェントの心の理解）の困難性という側面で問題を持つと指摘を受ける。しかしながら、昨今の情報化社会が持つ大問題である情報爆発に対して、直接操作だけで取り組むことも困難であり、インタフェース・エージェントを活用するべきである。例えば、物事の基礎的事項から順序立てて理解したい場合には、本のような線形のメディアを利用するのが適切であろう。あるいはハイパーテキストのような非線形なメディアを利用する方が、素早く目的とする情報を得ることができる場合もある。しかし、自ら本を繰ったり、リンクを辿ったりできないような膨大な情報が背景に潜むとき、あるいは対象に関する知識があまりに少ない場合などには、インタフェース・エージェントが解決できる課題も多い。このような課題解決のためには、間接管理のデザイン方法論は必要とされていることは明白であり、それに対するインタフェース・エージェント技術の研究開発は、今後も力を入れていくべき分野である。

（３）心のモデルの探究について

人と機械の自然で豊かなコミュニケーションの実現は、インタフェース・エージェントに対する重要な課題である。ここでは、自然なコミュニケーションとは、人が特別の訓練や長時間の使用体験を積まずに機械を操作や管理できることであり、豊かなコミュニケーションとはそのコミュニケーションという社会的行為自体に価値を認めるということである。本章ではこれらの課題に対して、インタフェース・エージェントの内部処理についての検討してみる。

人間が機械に対してどのように感じるかという擬人観について検討する。D.Dennett による志向姿勢は、自律性・反応性を示す対象に対してはあたかも知識・欲求・意図をもつ存在であるかのような反応を説明する〔8〕。また、N.Humphrey は、意識は単に自分の心をのぞくものではなく、他者の心の内を類推する社会的・心理的な器官として発展したと述べている〔9〕。すなわち、相手の行為や行動に対する意味付けを行ったり、相手の気持ちを察したり、意図や心的状態を推定したりするのに、人は自らの心を利用している。このことを以降では、「心のシミュレーション」と呼ぶ。「心のシミュレーション」によって、

a) 人から機械への情報伝達において、人の意図や心的状態を推定する。

b) 機械から人への情報伝達において、機械の内部状態を分かりやすく表現する。

という効果を、人と機械のコミュニケーションにおいて追求する心のモデルについて次項で紹介する。

①心のモデルについて

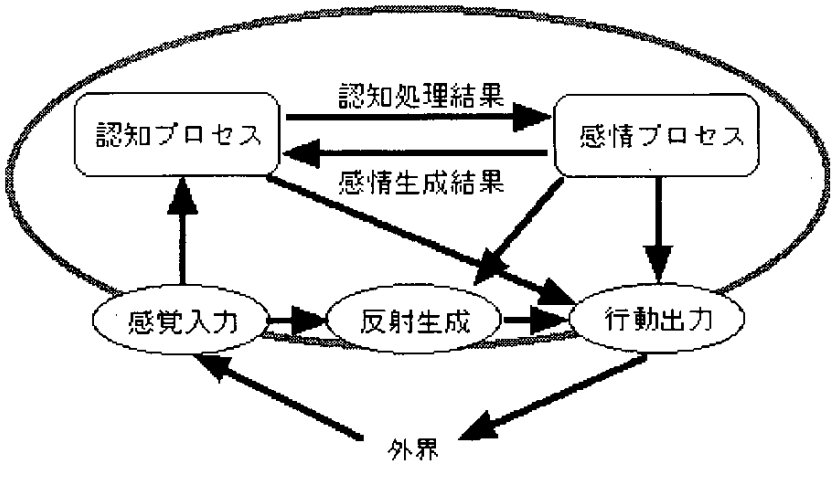
「心のシミュレーション」を実現するためのモデル（Mind and Consciousness model、略して MaC モデル）について紹介する〔10〕〔11〕〔12〕。MaC モデルは、「認知的評価理論（cognitive appraisal theory）」に基づく OCC（Ortony、Clore、Collins）モデル〔13〕を拡張した概念モデルを採用している。OCC モデルの特徴は、認識や判断を行う認知プロセスと、認知結果に基づいて感情を生成する感情プロセスからなり、これら２つのプロセスは相互作用するという点にある。MaC モデルでは、この利点に加えて次の２点の拡張を行っている。図表 2.2.2-1 に MaC モデルの概念図を示す。

a) 感情による価値判断メカニズム

従来の認知的評価理論によれば、外界の認識、目標の設定、および行動の選択などの処理結果が感情プロセスに作用するという特徴があった〔14〕〔15〕。これに加えて、MaC モデルでは感情プロセスを一種の価値判断プロセスとみなすことで、このプロセスの処理結果を認知プロセスにおける目標設定や行動選択に作用させることとした。このため、感情プロセスからの出力を認知プロセスへ与えるというルートを明示的に設けた。このことにより、認知プロセスにおいても感情が与える評価結果を利用することに繋げる。

b) 反射的プロセスと熟考的プロセスのメカニズム

反射と熟考という２階層の処理プロセスのメカニズムを備える。反射的プロセスは、外界から受ける刺激をもとに動作を決定して出力するもので、入力刺激と出力動作の関係は固定的である。例えば、目前の障害物を回避する行動のように迅速な対処を必要とする場合に有効である。一方、熟考的プロセスは、目標を自律的に設定し、その目標を達成するための行動を決定する。また、複数の刺激や目標が存在する場合に、それらの重要性を考慮して、どの刺激や目標に対して注意を向けるべきかという選択的注意も本メカニズムによって得られる機能の１つである。



図表 2.2.2-1　MaC モデルの概念図

②MaC モデルの構成と特徴

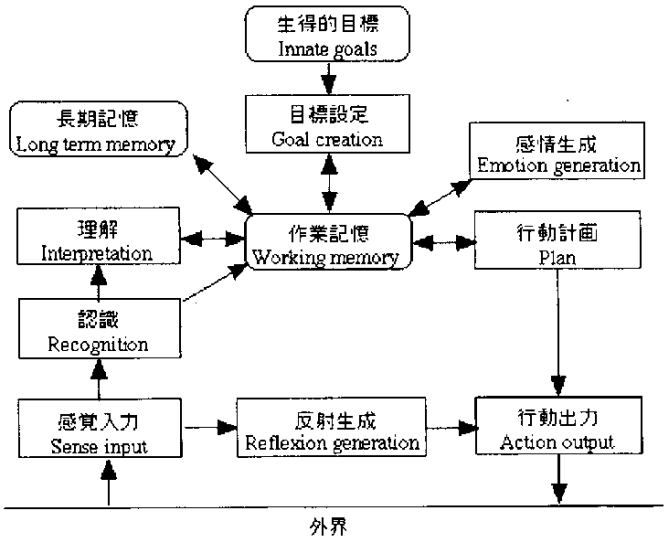
MaC モデルの構成および主な機能について紹介する。図表 2.2.2-2 に MaC モデルの構成を示す。図中、生得的目標、長期記憶、作業記憶は記憶モジュールであり、他の矩形で示されるものは処理モジュールを表している。アーキテクチャとしては、作業記憶を中心に情報共有を行う黒板モデルを採用し、さらに認識・理解→目標設定・感情生成→行動計画・行動出力といった処理順序の制約をかけている。

a) 感情生成

MaC モデルの感情生成メカニズムは、感情の生起は論理的に説明できるというアージ理論〔9〕に基づいている。感情は、感情生成モジュールを中心として生成される。感情生成モジュールでは、アージ理論および認知評価理論から抽出した規則を基に、ファジィ推論規則を用いて感情要因を推論している。また、それらの感情要因から「喜び・悲しみ・怒り・恐怖・嫌悪・驚き」の６つの基本感情に重み付きネットワークによって結合することによって、各基本感情の生起を決定する。認知的評価理論では、生成される要因によって感情を分類している。例えば、次のような要因および規則がある。

1) 目標達成の成否に関する要因：目標を達成すると喜びの感情が増加し、目標が紛糾すると怒りや悲しみの感情が増加する。
2) 刺激の快不快に関する要因：快適な刺激を得ると喜びの感情が増加し、不快な刺激を得ると嫌悪の感情が増加する。
3) 危険に関する要因：他者からの攻撃などにより苦痛を受けたり危険に遭遇すると恐怖の感情が増加する。
4) 意外性に関する要因：予想外の刺激を受けると驚きの感情が増加する。

それぞれの感情の程度は感情強度と呼ぶ値で表現される。感情強度は、感情同士の影響や減衰などを考慮した関数によって計算される。



図表 2.2.2-2 MaC モデルのアーキテクチャ

b) 生得的目標と経験的目標

MaC モデルで扱う目標には、生得的目標と経験的目標の２種類がある。このうち生得的目標は、エージェントの設計者が予め設定しておく目標である。経験的目標は、エージェントが行動を通じて経験的に学習することによって自らが設定する目標であり、図表３・２では長期記憶にて管理される。

生得的目標には、異なる複数の目標を設定することが可能である。各目標はその目標を達成する欲求の強さを「欲求度」というパラメタで表現している。欲求度を活用することで、空腹を満たすという目標のように時間経過にしたがって増加し、欲求が満たされると減少する場合や、苦痛を避ける欲求のように外部から与えられる刺激の強さに応じて変化を与える場合などに応じて、さまざまな種別の目標を設定することが可能である。

経験的目標は、経験をもとに推論規則を用いて目標設定部で設定される目標である。この目標は、内面の変化あるいは外部環境の変化を引き起こしたであろう、自らの行為や外部環境の変化との因果関係を学習することによって設定される目標である。例えば、空腹を満たすという生得的目標の達成に対して、ある物体を食したという行為の貢献度合いを計算することによって経験的目標を学習する。

c) 個性表出

MaC モデルでは、モジュール内のパラメタ調整によって、数多くの基準を変化させることができる。行動基準の変化によって、表出される個性に変化を与えることができると考えている。例えば、下記の４つのモジュールのパラメタを変化させることによってさまざまな個性を表出することができる。

1) 生得的目標：欲求度が増加する割合の大きさにより、目標の設定頻度を変化させることができる。例えば、空腹を満たすという欲求度が増加する割合を大きくすると、空腹を満たす目標が設定される頻度が増え、結果として食欲旺盛に見えるようになる。

2) 経験的目標：固執度が減衰する割合を調整することにより、目標に対する執着心の強さを設定する。

3) 感情生成部：感情の表出しきい値を下げると、その感情の表出頻度は高くなり、しきい値を上げると表出頻度は低くなる。例えば、怒りの感情強度に対するしきい値を下げると怒りやすくなる。目標の達成や紛糾による感情強度の変化量によっても個性を設定できる。目標が達成したときの喜びの増加量や、目標が紛糾したときの怒りや悲しみの増加量を調整することにより個性が変化する。

4) 行動計画部：それぞれの行動を選択する条件を変化させることで、出力される行動の傾向が変化する。ある目標が設定されているときの行動を選択する確率を変化させる。

心のシミュレーションを実現する MaC モデルについて概説した。今までに複数の異なった領域のアプリケーションに適応することで、その有効性を確認している。一方で、現在のレベルでは当初想定した自然で豊かなコミュニケーションが実現できているわけではない。体験の共有のための記憶のメカニズムや対話における文脈理解など手付かずの機能も多々ある。モデルの拡張という技術的課題に加えて、キラーアプリケーション探索を通じ

た課題抽出についても取り組むべきである。

（4）インタフェース・エージェントの事例紹介

本章では、インタフェース・エージェントの応用事例を紹介する。なお、先述のMaCモデルの応用事例については、別項の「研究開発への取り組み事例」として、サイバーキャラクタ“クリーチャ・ワールド”、ペットロボット“たま”、擬人化エージェント“Stellar”を紹介している。

①Microsoft “Office97 Assistants”〔16〕

Microsoft 社の Office97 におけるアシスタントは、ヘルプ機能をアクティブに提供する。エージェントのキャラクタ・タイプは数ある中から選択し、これらの単純な個性（おせっかいなど）も選べる。個人の好みや文化的な相違を考慮し、さまざまなタイプのキャラクタを提供している。従来は、ユーザは困ったときにヘルプを参照するという操作を行ったが、ここではユーザが困っている状況をエージェントが認識し、能動的にヘルプするのである。ユーザはこれらのエージェントの動作を管理するのである。

②SRI International “InfoWiz”〔17〕

InfoWiz は、キオスク端末において案内役をなすキャラクタが元のアイデアとなっている。従来のポインティングデバイスによるインタフェースに変えて、音声を利用したインタラクションを基にして Web ページの閲覧を行うものである。WWW のように、内容が非構造でダイナミックに変化するような情報については、直接操作のようなインタフェースは向かないとの考えで研究開発されている。

③電総研“マルチモーダル対話エージェント Mr. Bengo”〔18〕

人間の上半身をCGにて模倣したエージェントと人間との間で、視覚および音声を利用した対話を行う。エージェントは、感情の顔表情および身振り手振り表現、視覚による個人照合、音声認識／合成機能などを備える。これらの機能を活用し、秘書エージェントや簡単な対話をこなす論争支援マルチモーダル実験システムが研究されている。論争支援マルチモーダルシステムは、ICOT で開発された法的推論システム〔19〕にインタフェースをかぶせたもので、裁判の対象となっている事件に関連する法令・判例に基づき、エージェント同士が原告と被告の立場に立って論争を行うシステムである

④ソニー “PostPet”〔20〕

郵便配達人のメタファーを利用したインタフェースによる電子メーラを提供する。これは、操作性の向上のみならず、そこにペットというキャラクタを配達人とすることでエンタティメント的な要素を新たに付加した。ここでは、ペットは単にメールの送受信時に働くだけではなく、ユーザとの対話から、あるいはペット同士の対話から、踊りを踊ったりケンカしたりするパフォーマンスを見せる。ユーザは単にペットを育てることや、ペットとコミュニケーションを取るのみではなく、ポストペットを一種のメディアとして他のユーザとのコミュニケーションをとるという場を提供している。

⑤富士通 “TEO-もう一つの地球-”〔21〕

エンタティメントは非日常的な場を提供することが目的だとすると、これに対して TEO では日常性のソフトウェアであることを目指している。ここでは、単に人に媚びたり遊んだりするようなキャラクタを作るのではなく、自分自身の感情や生活の場を持って自律的

に行動する「生き物」を作ることを目指している。コンピュータアニメーションで描かれるイルカに似たフィンフィンというキャラクタが主人公であり、ユーザとコミュニケーションをとるばかりでなく、自分自身の住む生活の場を持っている。また、フィンフィン以外にもキャラクタが登場し、フィンフィンとコミュニケーションを取る。機械が提供する機能を中心として設計されたのではなく、フィンフィン達キャラクタが存在していること自体に意義を見い出すような存在であるといえる。

⑥MIT Media Lab. "ALIVE"〔22〕

ALIVE (Artificial Life Interactive Video Environment) ではビデオ環境の中に人間が取り込まれ、その環境の中でキャラクタ達とインタラクションする点が他のインタフェースの手段とは異なっている。画像処理による姿勢や身ぶりの認識から、行動生成モデルに基づきキャラクタがそれに反応する動作をおこなう。

⑦Stanford Univ. & Extempo System Inc. "Improvisational Agents"/ "Interactive Characters with Smarts and Hearts"〔23〕... "Tigrito"〔24〕"Erin"〔23〕、"Jennifer"〔23〕、etc.

B.Hayes-Roth が中心となって研究開発している、Stanford 大学 Knowledge System Lab. および、Extempo Systems Inc. のインタフェース・エージェントがある。このエージェントは、個性、感情、気分、そしてそれらを利用して即興応答を行う心のモデルを利用して構築されている。そして、この心のモデルに加えてキャラクタの持つパーソナリティがしっかりとデザインされ、それに基づいたシナリオが作成されている。パーソナリティは心のモデルのみで決まるのではなく、それらキャラクタの体と振る舞い方にも大きく依存しており、シナリオはこれらの体や振る舞いにおける設定にまでも含んでいる。これらのキャラクタは、エンタテイメントや電子ペット的な存在のみならず、情報探索ナビゲータ、あるいはエレクトリック・コマースにて利用されるような販売員エージェントなどへの応用も検討されている。

⑧MIT Media Lab. "Storyteller Agents"〔22〕

ぬいぐるみをインタフェースとして利用し、インタラクティブな物語作成の研究を行っている。物語はそれを構成する断片があり、それらが分岐ノードによって結ばれている。これらの分岐点においてユーザとインタラクションすることで物語を構成していく。ぬいぐるみは物語の語り手として、また、インタラクションの相手としての役割を果たす。音声合成による語りと、目が光るなどの動作を行う。余談であるが、筆者がこのロボットを抱いた感想は、サーボのフィードバックに起因すると思われる「震え」と、モータの「熱」が感じられ、まるで生きているかのような錯覚に陥ったことを付け加えておく。

⑨Microsoft "ActiMates Barney"〔16〕

子供向けのエデュテイメント (Education + Entertainment) 製品である。Barney と呼ばれるぬいぐるみロボットは、喋り、歌い、身ぶり手振りで子供とインタラクションする。Barney 単体で喋れる言葉は 2000 ワード (製品は英語版) だが、パソコンと接続することにより 14000 ワード以上を喋ることができる。目の部分に光センサ、手先に触覚センサが内蔵されている。ユーザが、Barney の目を覆ったり、手を握ると反応する。反応するのは Barney だけでなく、Barney が接続されたパソコン画面に映し出されるキャラクタも連動して反応する。また、Barney はユーザとインタラクションすることに加えて、パソコン画面

上のキャラクタとの掛け合いも行う。

⑩ソニー“AIBO”〔25〕〔26〕

AIBOはロボット・エンタテイメントという新しいアプリケーション領域の開拓を狙って開発された。この領域の目的は人を楽しませるロボットを提供することである。従来のおもちゃのロボットとは明らかに異なる次のような高機能が特徴である。

a) メカ：18自由度のメカ機構により歩行動作を含む多様な動作を行える。

b) センサ：18万画素のCCDカメラの搭載し、特定色のボールを追い掛ける動作を行なうことができる。また、赤外線方式の距離センサを用いて、自分と対象との距離を知ることにより、対象に近づいて立ち止まるという動作を行なう。

c) 感情と学習：6種類の基本感情と、4種類の欲求を生成する。また、ユーザの接し方により行動パターンが変化する。この他、「幼年期」「少年期」「青年期」「成年期」という成長過程があり、それぞれの過程により行動パターンが異なる。

⑪テレフォニー・エージェント

General Magic〔27〕のPortico（以前はSerengetiと呼ばれていたもの）とWildFire〔28〕のものをあげる。双方ともCTI（Computer Telephony Integration）のアプリケーションとして脚光を浴びている統合メッセージングシステムである。これらのシステムは、電子メール、電話、ボイスメール、FAXなどのさまざまな媒体からなるメッセージを統合的に扱おうというもので、PorticoもWildFireもこれらのメッセージを電話、携帯電話に集中したものである。Porticoの方は、音声認識、音声合成ともに従来のIVR(Interactive Voice Response)システムと比較すると自由度がかなり高いものを利用している。利用者は単語を話すのではなく、自然な発話文で、また、エージェントの方も録音および合成音によって発声し、ユーザとダイアログを形成する。一方、WildFireのほうは、単語レベルの音声認識機能に、人間の声を録音した合成音声を利用している。電話では音声メディアが中心であり、情報伝達の量を増やせないなどの制約を受ける分、インタフェース・エージェント側の実行しなければならない仕事の量も増えると考えられる。この点からはテレフォニー分野での応用は、インタフェース・エージェントが生きるアプリケーションであると言える。

（5）まとめ

インタフェース・デザインおよび心のモデルという2つの異なった観点からインタフェース・エージェントに関する技術動向について概観した。また、インタフェース・エージェントの事例と思われるものをいくつか紹介した。これらの事例を振りかえってみると、「インタフェース・エージェントとは何か」また「インタフェース・エージェント技術とは何か」ということ自体が明らかになっていないと思える。さまざまな考えの基に、さまざまな応用が渾沌としているのが現状である。今後は、技術指向とアプリケーション指向の活動のマッチメーキングによって、技術発展とキラーアプリケーションの発掘が期待される。

参考文献

〔1〕 田村博編, “ヒューマンインタフェース”, オーム社, 1998.

〔2〕 Alan Kay, “User Interface: A personal view”, In B. Laurel, ed., The Art of Human-Computer Interface Design, Addison-Wesley, Reading, Mass., 1990. (邦訳　人間のためのコンピュータ, ブレンダ・ローレル編, アジソン・ウェスレイ, 1994)

〔3〕 Ben Shneiderman, Pattie Maes, “Direct Manipulation VS. Interface Agents”, (excerpts from debates at IUI97 and CHI97), Interactions November+December, Vol. 4, No. 6, pp. 42-61, 1997.

〔4〕 Byron Reeves, Clifford Nass, “The Media Equation”, Cambridge University Press, 1996.

〔5〕 NetSage Corporation : http://www.netsage.com

〔6〕 Katherine Isbister, “サイバー空間での社会的インタラクションのための設計”, 情報処理, Vol. 40, No. 6, pp. 569-574 (1999).

〔7〕 Youngme Moon, “When the computer is the “salesperson” :Consumer responses to computer “personalities” in interactive marketing situations”, Harvard Business School Working Paper 99-041 (1998).

〔8〕 ダニエル・C・デネット, 若島正・河田学訳, “志向姿勢の哲学”, 白揚社, 1996.

〔9〕 ニコラス・ハンフリー, 垂水雄二訳, “内なる目”, 紀伊国屋書店, 1993.

〔10〕 Ushida, H., Hirayama, Y., and Nakajima, H.: Emotion Model for Life-like Agent and ITS Evaluation, Proceedings of the 15th National Conference on Artificial Intelligence (AAAI-98), pp. 62-69 (1998).

〔11〕 牛田, 平山, 中嶋, 田島, 齋藤:心のモデルに基づくインタラクティブエージェント, 第4回知能情報メディアシンポジウム講演論文集, pp. 165-172 (1998).

〔12〕 牛田, 平山, 中嶋:自律的行動決定モデルに基づくインタラクティブキャラクタ, インタラクション99論文集, 情報処理学会シンポジウムシリーズ, Vol. 99, No. 4, pp. 51-52 (1999).

〔13〕 23) A. Ortony, G. L. Clore, A. Collins, “The Cognitive Structure of Emotions”, Cambridge University Press, 1988.

〔14〕 C. Elliott: The Affective Reasoner: A Process Model of Emotions in Multi-agent System, Ph. D. Dissertation, Technical Report No. 32, Northwestern University, 1992.

〔15〕 W. S. Reilly and J. Bates: Building Emotional Agents, Technical Report CMU-CS-92-142, School of Computer Science, Carnegie Mellon University, 1992.

〔16〕 Microsoft Corporation : http://www.microsoft.com

〔17〕 SRI International : http://www.ai.sri.com/~oaa/

〔18〕 電総研知能情報部画像班 : http://www.etl.go.jp/etl/gazo/

〔19〕 ICOT 法的推論システム : http://www.icot.or.jp/AITEC/COLUMN/legal/legal.html

〔20〕 ソニー「PostPet」 : http://www.so-net.ne.jp/postpet/

〔21〕 富士通「TEOの世界」 : http://www.teo-world.com

〔22〕 MIT Media Lab. : http://www.media.mit.edu

〔23〕 Extempo Systems, Inc. : http://www.extempo.com
〔24〕 Heidy Maldonado, Antoine Picard, Patrick Doyle, Barbara Hayes-Roth, "Tigrito: A Multi-Mode Interactive Imporovisational Agent" Proceedings of the 1998 International Conference on Intelligent User Interfaces, pp. 29-32., 1998.
〔25〕 Fujita, M and Kageyama, K: An Open Architecture for Robot Entertainment, Proceedings of the First International Conference on Autonomous Agents, pp. 435-442 (1997).
〔26〕 SONY : http://www.world.sony.com/robot/
〔27〕 General Magic : http://www.genmagic.com
〔28〕 WildFire : http://www.wildfire.com

※ 文中の製品名称は各社の登録商標です。

2.2.3 エージェントの拓く未来

(1)エージェントの起源と現状

AlanKayによると、いわゆる（ソフトウェア）エージェントのアイディアは、1950年代の中頃に、当時MITに所属していたJohnMcCarthyが考え出し、数年後、同じくMITにいたOliverG.Selfridgeがこの名前をつけたそうである〔1〕。彼らが考えていたのは、あるゴールを与えられると適切な操作を実行していき、途中で障害が発生すると人間に助言を求め、それを自然言語で受け付けるようなシステムであった。まさに、コンピュータネットワーク上の「ソフトウェア・ロボット」と呼べるような存在である。

その後、エージェントは、人工知能における主要な研究テーマの一つになり〔7〕、また、ユーザーインタフェースや情報検索においても重要なアプローチの一つになっている。もっとも、ユーザーインタフェースの研究者の一部には、依然としてエージェントという間接的なインタフェースメタファーに対する懐疑心が強いようで、従来からの直接操作インタフェースの研究者とのディベートがよく行われているようである。

人間にとって、文句一つ言わずに休みなく働きつづけ、自分の興味などをよく知ってくれて、さまざまなアプリケーションの利用法を心得ていて、適切なタイミングで適切な情報を提示してくれる賢い「助手」の存在は極めて有益であることは間違いがない。ゆえに、エージェントシステムに対する人々の期待は小さなものではないだろう。

現在、とてつもない量の情報がオンラインでアクセスできる。しかし、個人がすべての情報をブラウズできるわけではないから、個人のアシスタントとして働くエージェントは、膨大な情報の中から、その個人にとって有益なものを選び出さなければならない。

ただ、現在の技術ではエージェントの知能を人間並みにすることは不可能だから、人間がエージェントに知能を期待すればするほど落胆の度合いは大きいと予想される。これが、今のところ自然言語による対話型エージェントが失敗している主な理由である。しかし、何らかの手段でユーザーを適切な文脈に誘導できれば、機械との対話も人間にとって十分に許容できるものになりうる。

実際には、知能があり文脈に適応しているのは人間自身であっても、人間が容易に選択

できる程度の文脈を用意して、人間との対話が適切なフィードバックループを構成するように導いていければ、自然な対話は成立するのであり、エージェントは賢い助手としての幻想を持続させることができるのである。

これがエージェントを設計するための基本原理であり、今後、巧妙な仕組みに基づいた、うまいエージェントがいくつか出てくるだろう。

では、そのようなうまいエージェントとは、いったい何をすれば実現できるのかについて、筆者の考えを述べていこうと思う。

(2)エージェントの諸相:擬人化と個人化

人間と対話するエージェントを人間にとって親しみやすくする手段の一つとして、エージェントの擬人化（personication）がある。それは、エージェントシステムに、人間同士の対話において重要な役割を果たすモダリティ（対話の様式）を加えていくことに相当する。そのようなエージェントを人々に印象付けた功労者はナレッジナビゲーターである。

ナレッジナビゲーターとは、AppleComputer が 1988 年に発表したコンセプトビデオに登場する未来のコンピュータである。二つ折りになった本のようなマシン（いわゆるペンコンピュータをたたんだようなもの）を開くと、待ってましたとばかりに画面上にフィルと呼ばれる人間型エージェントが現れ、その日のスケジュールや受け取った電子メールの内容などを教えてくれる。また、ユーザーが言葉で質問すると詳しく答えてくれる。ある場合は、ユーザーと別の人間が話している内容を聞いて、「それは、○○ですよ」と情報を付け加えてくれる。

ナレッジナビゲーターは仮想のマシンに過ぎないが、そのビデオは、人間の言葉を理解し自発的に話しかけてくれる擬人的キャラクターというヒューマンインタフェースのイメージを強烈に印象づけた。以降、それに触発されたと思われる研究がずいぶん行なわれた。たとえば、エージェント指向インタフェースやマルチモーダルインタフェースの一部がそうである。

擬人化は、ユーザーにとって親しみやすくすると同時に、人間並みの知能を期待させてしまうという点で諸刃の剣であると言える。しかし、エージェントの機能の本質が対話であることを考えると、人間同士の対面対話に伴うさまざまなモダリティをできるだけ多く備えるべきであるという見方もできる。その結果、エージェントがある種の擬人性を持つのは必然であろう。

擬人性には、外観に関するものと、挙動に関するもの、そして内面に関するものがある。外観はコンピュータグラフィックスによって実現される場合が多く、その技術の進歩には目ざましいものがある。挙動には、ユーザーからの働きかけに対する適切な反応だけでなく、自発的な行為、つまりエージェントからの積極的な働きかけも含まれる。これには、即応から熟考までのさまざまなレベルの行為が含まれる。ここで即応とは、反射のような、刺激に対して複雑な過程を経ずに生じる行為であり、熟考とは、十分に計画が練られた行為である。挙動の擬人性は知能に直結する問題であるため、設計に注意を要する部分である。ただし、人工無脳として有名な対話システム ELIZA〔8〕や、最近のゲームソフトのシーマンのように、非常に単純なやり方で臨場感のある対話を作り出すことが可能である場合もあるので、工夫次第で、ある程度、エージェントを賢く見せることは可能だろう。ま

た、内面に関する擬人性はエージェントにある種の心のモデルを持たせることであり、人間の感情のモデル化などが応用できる場合もある。AI の長老である Marvin Minsky が Emotion Machine と呼んでいるシステムは、感情のモデルを備えることにより、人間がエージェントにいだく幻想をより長く持続させると考えられている〔2〕。

ただし、今のところ、擬人化のための技術が不十分なことや、考慮すべき心理学的・社会学的要因が非常に多いために、擬人化によって効率的な対話が実現できるという状態にはまだ至っていない。

さて、当然ながら擬人性のみがエージェントが持つべき性質というわけではない。エージェントの本質と言えるもう一つの性質は、エージェントがユーザーである個人に適応するということである。

一般に、システムの情報処理を特定の個人に適応させることを個人化(personalization)という。個人化エージェントは、繰り返される人間の行動やあらかじめ入力されたユーザーの個人的データから、その人間の習慣や嗜好などを獲得し、その人間に合った適切な振舞いをするエージェントである。特定の人間に特化された情報は、暗黙的にその意図を認識する手助けになり、提示すべき情報の選択や、対話を開始するきっかけを決めるのに役立つ。

個人化の典型的な例は特定話者にチューンされた音声認識である。また、言葉と実世界の事象を対応付けることも、個人を特定することによって、より容易になる。このように対話の文脈をユーザーに適したものにして、ユーザーを誘導し、ユーザーの次の行動を予測して文脈を切り替えていくことが、個人化エージェントにとって重要な機能である。

個人化には、ユーザーモデルのような個人の興味や嗜好などの特性を表す記述を獲得することや、特定の時間や場所とユーザーの行動を関連付けるような、個人に強く依存した文脈情報を獲得すること、さらに、発汗や心拍などの精神生理学的情報に基づいて個人の心理的状態を感知し、その状態に適した対応法を獲得することなどが含まれる。これらについて、これまで個別のアプローチがなされてきた。最も古くから行なわれているのはユーザーモデルに関する研究である。これにはプロトタイプモデルと呼ばれるあらかじめ用意されたユーザーのパターンに、パラメータを調整しながら当てはめていくやり方や、プラン認識と呼ばれる手法により、対話の進行に伴ってユーザーの意図や背景知識を推論していくというやり方が考えられている。また、ユーザーを取り巻く環境にさまざまなセンサーを埋め込んで、実世界の文脈まで含めてエージェントを個人化しようという動きも活発になってきた。たとえば、今いる場所、話している相手、近接している対象などのさまざまな情報が入手できれば、エージェントはユーザーに関するより多くの情報を考慮して、その習慣や嗜好を獲得し、対話に役立てることができるだろう。さらに、MIT Media Lab で進められている aective computing と呼ばれる研究では、精神生理学の成果に基づき、各種センサーを人間に接触させて、生理情報を検出し、心理状態を知ることによって、より高度に個人化されたシステムが構築できると考えている〔6〕。確かに、対話において相手の感情を知ることは、意図を認識するための重要な手がかりになるだろう。

これらのような、個人化に対する複数のアプローチを統合すれば、さまざまなレベルでユーザーに適応するエージェントを設計できるだろう。

擬人化と個人化は、エージェントに対する異なる観点であるが、共にエージェントと人

間の対話に大きな効果をもたらすものであり、うまいエージェントを作るためには十分に考慮する必要がある。しかし、やはり、それだけではまだ不十分で、エージェントが行うタスクについて、さらに多くの配慮が必要である。

では次に、検索などの情報処理を主なタスクとするエージェントは、どうしたらもっと賢く、ユーザーにとってより受け入れやすいものになるのかについて述べてみたいと思う。

(3)インテリジェントエージェントからインテリジェントコンテンツへ

筆者は、WWW（World Wide Web、以下 Web）上の検索ロボットについて調べるにつれ、「こんなに単純なものをエージェントと呼べるのだろうか」と考えるが、ほんの少しのインテリジェンスでも、操作の対象となるデータこの場合は Web コンテンツが大量になれば、人間の予測をはるかに上回る効果が得られるのだということを痛感している。

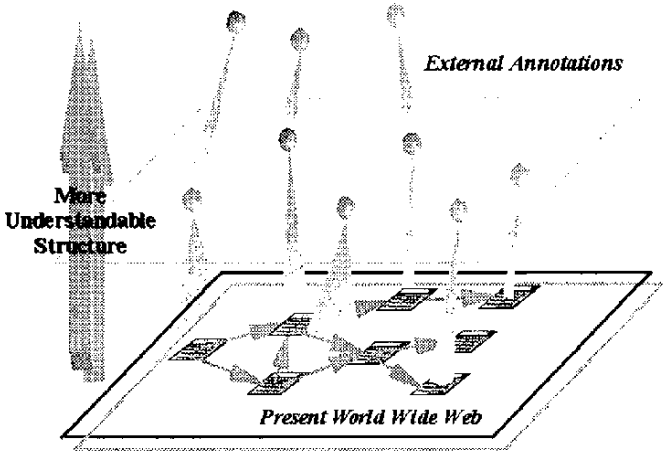
人工知能の研究は、深く狭い知能から、広く浅い知能へとシフトしてきたと見ることができる。これは、より実用的なものを作ろうという動機に基づいており、より高度であっても適用範囲が狭ければ多くの人にとって役に立たないが、単純であっても適用範囲が十分に広くロバストであれば多くの人に受け入れられるものになるだろういうことである。さらに、適用範囲やロバストさはそのままに知能を高度化できれば、いずれ深くて広い知能が実現できると考えるのである。

しかし、より高度なことをやろうとして複雑なルールを構築すれば必ず例外が発生する、それによってロバストさは崩れ、再びユーザーの不信感を煽ることになる。それでは、ある程度まで普及した技術をさらに高度化するのは不可能になってしまう。

筆者は、エージェントをある程度以上に複雑にする代わりに、コンテンツの方を人間がエージェントにとって都合の良い形に変えていき、人間とエージェントとコンテンツが構成するシステム全体をより高度にするアプローチを取るべきだと考え、そのためのプロジェクトを推進している。それは、現在の Web をインテリジェントコンテンツにする、というものである〔3〕。

従来の Web は一枚の平面上に存在するグラフとして捉えることができる。筆者らは、Web を平面から立体に拡張する手法を提案している。それは、Web 上に新たな上位構造を構築していくことに相当する。

図表 2.2.3-1 は筆者のイメージする Web の上位構造を表している。



図表 2.2.3-1　WWW 上に構築される上位構造

WWW上に構築される上位構造Webの上位構造とは、コンテンツに対するメタコンテンツ、さらにそのメタコンテンツに対するメタコンテンツという具合に、Web をより立体的に捉える構造である。ここでは、そのようなメタコンテンツを外部アノテーション各々のコンテンツの外部に存在するアノテーション。以下では、単にアノテーションと表記するとして一般化する。よく知られているアノテーションの例は、外部リンクである。これは、XML（eXtensible Markup Language）〔9〕の仕様の中で議論されていることであるが、まだ実装されていないこともあり、現在のWebのアーキテクチャの上でどのように実現されるのかわかっていない。アノテーションのその他のよくある例は、コンテンツに対するコメントや評価などの、著者以外の人による、そのコンテンツの読者に対するメッセージである。これはもちろん無制限に許可することはできないかも知れないが、コンテンツを理解する上での重要な手がかりになる。たとえば、視覚障害のある人にとって、内容記述のないイメージは理解不能であるが、第三者が付けたコメントを読み上げることによって、イメージの理解が可能になる場合がある。

筆者らは任意のドキュメントの任意のエレメントにアノテーションのXMLデータを関連付ける仕組みを開発した。具体的には、XPointerを用いてエレメントを指定し、アノテーションデータのファイル名を関連付けるテーブルを用意する。この場合のアノテーションは大きく分けて種類に分類される。一つは、テキストに言語的情報を付与するもので、言語的アノテーションと呼ばれる。二つめは、ドキュメント内の任意のエレメントに注釈を付けるもので、コメントアノテーションと呼ぶ。三つめは、ビデオなどのマルチメディアデータにその内容に関する構造を付与するもので、マルチメディアアノテーションと呼ばれる。

アノテーションを作成し、公開することが容易になれば、Web の表現力は大きく拡張され、その利用価値は飛躍的に向上するものと思われる。

さて、このようにコンテンツそのものがより理解しやすくなれば、それを扱うエージェントが可能なタスクもより高度になるだろう。その一つが、知識発見である。これは、アノテーションを含む大量のWebコンテンツから、機械的に何らかの発見をさせようとするものである。従来の検索エンジンのように、キーワードから複数のWebページを検索するのではなく、ユーザーのある要求を満たすような情報を複数のコンテンツを合成して作り出すのである。たとえば、IBM製品に関する一年分のWeb情報から、IBMのその一年の製品戦略に関するサマリーを生成する、ということが実現できる。

エージェントはアノテーション付きのコンテンツを対象にすることによって、単純な手続きを繰り返すだけで、より高度のサービスを提供できる。これは、見かけ上、エージェントが賢くなったように見えるが、実際はコンテンツそのものが人間の努力によって賢くなっているのである。

近い将来に、人々はWebから情報を得るために、検索エンジンを用いるのではなく、エージェントを用いて、大量のハイパーリンクのリストの代わりに、短時間で容易に理解できるように個人化された情報のサマリーを読むことができるようになるだろう。それは来るべき情報の洪水から自分自身を守る最良の方法になると思われる。アノテーションを大規模化・共有化していくためには、今後多くの労力を人間に強いるかも知れないが、ドキュメントを人類共有の知識とするために人間が力を合わせて努力をしていくことがなけれ

ば、人々は今後も無限に拡張していく情報の圧迫から自分自身を解放することができないだろう。

(4)おわりに:エージェントの拓く未来

最後に、以上のような、うまいエージェントを作る試みによって、どのような未来が拓かれるか展望してみたいと思う。

エージェントによってもたらされる一つの未来のあり方として、まず考えられるのは、筆者が以前から提唱している「エージェント拡張現実感」である〔4〕。

エージェント拡張現実感とは、エージェント技術と拡張現実感の技術を統合する新しい研究領域である。拡張現実感の基本コンセプトは実世界の対象に情報世界の対象を重ね合わせ、実世界の対象の操作が情報世界にも反映され、ユーザーが意識しなくても情報を操作することができる、ということである。ここにエージェント技術を導入することによって、実世界に関わる情報の間接的な操作も可能になる。これは、物理的な対象や位置に関する操作に連動して、エージェントを起動し、個人化された情報処理を行わせることができるということである。これによって、実世界状況を手がかりとして、ユーザーの意図を暗黙的に理解して、情報世界を動き回って適切な情報を収集・加工するシステム、あるいは道案内や買い物支援、さらに、目の前にある機械の操作など、状況に依存したタスクを行ってユーザーを支援するシステムが実現できる。

また、エージェント拡張現実感は人同士の結び付きを強化することにも貢献するだろう。たとえば、パーティなどの参加者の中で興味の一致する人をエージェントが情報世界において探し出し、ユーザーに知らせるのである。これによって、初対面の相手とも比較的楽に会話できるようになると思う。また、遠隔地にある相手と話をするときに、事前にエージェントに相手の都合をプライバシーを侵害しない程度に調べさせ、問題がなければコミュニケーションのチャネルを開く、ということも可能になるだろう。エージェントはユーザーのプライバシーを守りつつ、他のエージェントからの問い合わせに答えて、必要に応じてユーザーの現在の状態を伝達するようになると思われる。

このようなエージェントは、実世界認識という新たなモダリティを用いることによって、ユーザーの意図をより容易に認識することができるため、人間とのインタラクションを大きく改善できる可能性がある。

つまり、人間とエージェントが環境や状況を共有していることを暗黙的に理解できる場合には、自然言語のようなあいまいな情報伝達手段が有効になると思われる。実世界状況に依存したエージェントは、人間のいる状況、その人間の習慣・興味などを知った上で、対話を理解する。人間はエージェントがそういったことを知ってくれていることを前提として、安心して言葉を使うだろう。

エージェント拡張現実感が、個人化エージェントの延長線上にあるとすれば、擬人化エージェントの延長には何があるだろうか?

これも以前に筆者が提案したものだが、エージェントの社会と人間社会の橋渡しをする存在としての「社会的エージェント」と呼ばれるものがある〔5〕。このエージェントは、擬人性を備え、不特定多数の人間と共同作業ができるように設計されている。

エージェントがある種の社会的存在であるためには、自発性と協調性を持って人間社会

に参加する必要がある。そのため、社会性を備えたエージェントは人間どうしの会話の中に何らかの誤解や事実誤認が含まれていたときに、それを検出して適切なタイミングで人間に知らせることを試みる。誤解の検出は、発話者の発話プランニングをシミュレートし、その意図のくい違いを調べることによって行われる。また、このエージェントは、視覚を使って人間の体の動きを見たり、話の間合いを窺いながら、発話のタイミングを決める。また、エージェントは表情や首の動きによってノンバーバルなコミュニケーション信号を発信し、エージェントの内部状態人間の発話を理解しているかどうか、などを即座に人間に知らせることができる。

このようなエージェントは、オフィスや自宅に限らず、さまざまな場所に存在する。ある意味で、環境そのものが社会的エージェントによってインテリジェント化されていると考えることもできる。社会的エージェントと個人化エージェントがコミュニケートすることによって、ユーザーの個人情報と、環境に依存した情報、さらによりグローバルな情報が統合され、人間はさらに都合良く情報を処理できるようになるだろう。

人間がエージェントを受け入れ、うまく使いこなすことによって、ノイズを含む情報に振り回されることなく、適切に自分の行動を選択できるような社会になるだろう。そうなったときに人間には何ができるだろうか？

筆者は、人間がより創造的な方向に進むことを期待している。よく言われることであるが、「優れた道具は人間をダメにするのでは？」という問いには、筆者は自信を持って「ノー」と答えたい。かな漢字変換のおかげで、漢字を書けなくなったとか、カーナビのおかげで道を覚えなくなったとかいうことは、本来人間がなすべきことを見失っている人の言うことである。漢字を書くことが重要なのではなく、より良い文章を書いて自分の考えを適切に表現することが重要なのであり、また、道を覚えることが重要なのではなく、迅速に場所を移動して、目的地においてなすべきことをすることが重要なのである。翻ってエージェントの場合はどうかというと、エージェントが上手に情報を集めて提示してくれるようになれば、情報の探し方そのもののスキルは低下するだろうが、その分、情報の活用のスキルが向上して、結果として人間はより創造的で行動的になるだろう。エージェントは創造や決定を「支援」してはくれるが、最終的に何をするかを決めるのは人間であり、そうであり続けなければならない。人工脳などと称して、制御不能な複雑系マシンを作って、それに決定そのものを委ね、人間がやるべき仕事を放棄してしまったら、そのときこそ、機械が人間をダメにしてしまうのである。

参考文献

〔1〕 AlanKay. Computer Software. *Scientic American* Vol. 251, No. 3, pp. 52〔59, 1984.

〔2〕 Marvin Minsky. *Emotion Machine*. MIT Press, 1999.

〔3〕 Katashi Nagaoetal. Semantic Transcoding:Making the World Wide Web more understandable and reusable by external annotations. *TRL Research Report*. IBM Tokyo Research Laboratory, 2000.

〔4〕 Katashi Nagao. Agent Augmented Reality:Agents integrate the real world with Cyberspace. InToru Ishida, editor, *Community Computing:Collaboration over Global Information Networks*. JohnWiley & Sons, 1998.

〔5〕 Katashi Nagao and Akikazu Takeuchi. Social interaction: Multimodal conversation with socialagents. *In Proceedings of the Twelfth National Conferenceon Articial Intelligence (AAAI-94)* , pp. 22〔28. The MIT Press, 1994.

〔6〕 RosalindW. Picard. A ectivecomputing. Technical Report321, MIT Media Laboratory, 1995.

〔7〕 Stuart RussellandPeterNorvig. *Articial Intelligence*:A Modern Approach. PrenticeHall, 1995.

〔8〕 Joseph Weizenbaum. ELIZA〔Acomputerprogram for the study of natural language communicationbetweenmanandmachine. *Communications of the ACM*, 1966.

〔9〕 World Wide Web Consortium. Extensible Markup Language. http://www.w3.org/TR/PR-xml-971208.

2.3 プラットフォーム

2.3.1 大規模広域分散システム環境 System ν[nju:]

現在、多くの企業はメインフレームでの集中処理型基幹システムを稼働させているが、その一方では、昨今のPCおよびネットワーク技術の進展、低価格化にともない、環境変化に対応するための分散システム指向、ダウンサイジングなども着実に浸透しつつある。よって、ビジネス形態の変化にダイナミックに対応できる柔軟性、発展性をもち、さらに系として十分な信頼性が保証されるような環境への円滑なトランスフォーメーションを可能とするシステム構築技術が待望されている。

このような目的を実現するための技術として、エージェント技術が有効であった。本項では、その技術を適用した製品である「SYSTEM ν[nju:]」について述べる。

（1）概要

SYSTEM νは、『次世代高速通信網をバックボーンとする大規模広域分散システムを安全かつ容易に構築・運用するためのプラットフォーム』という目的で開発されたTransactional CORBA 開発実行環境である。ここでいう"大規模広域"とは、数千から数万ノードの全国あるいは世界規模のネットワーク環境を想定している。

新しいネットワーク・アーキテクチャによって、分散オブジェクト環境に企業の情報基盤としての機能を付加し、大規模ネットワーク上のさまざまなオブジェクトを網の目のように結ぶサブシステム（エージェント）により、ネットワークにおけるオブジェクト（プログラムとデータ）の自律性(Autonomous)と位置透過性(Transparency)を実現する。SYSTEM νのエージェント技術により実現するネットワークの自律性が、クリティカルなビジネスを支える信頼性、運用性、そしてハイパフォーマンスを提供する。

また、SYSTEM νは、開発／実行／運用の生産性向上を実現するソフトウェア・プラットフォームでもある。CORBA準拠のORB(Object Request Broker) 機能に加え、SYSTEM νが

独自に提供する自律性と位置透過性が大規模ネットワーク・アプリケーション開発におけるプログラマの作業効率を飛躍的に向上させるとともに、異機種混在の分散環境におけるオブジェクト間の高度な相互運用性を実現する。他の構成要素に影響を与えず任意のオブジェクトだけの修正／変更も可能である。

（2）エージェントモデル

SYSTEM νは、クライアント／サーバ・モデルを包含する自律的／対称的な多対多通信を基盤とする分散システムを実現する。

SYSTEM νを使用して分散システムを構築するには、まず対象となるシステムをオブジェクト・モデル化しなければならない。モデル化が行えたなら、それを実際のシステムで使用するためにプログラム・コードに書き換える必要がある。SYSTEM νでは、オブジェクトのインタフェースを定義するための言語として、CORBA で定める仕様に準拠したIDL(Interface Definition Language) を提供する。IDL を使用して、オブジェクトの属性とオペレーションの定義を記述することができる。

IDL は、SYSTEM νが提供する専用の IDL コンパイラでコンパイルする。IDL コンパイラは、IDL で定義したオブジェクトを生成し、SYSTEM νへの登録を行うとともにスタブと呼ばれるファイルを生成する。スタブは、IDL の記述を C++や Visual Basic などのプログラム言語に展開したもので、以降で説明するエージェントを作成するときに使用する。

SYSTEM νはエージェントと呼ばれるサブシステムから構成される。エージェントとは、データとそれを処理する手続きとをカプセル化したオブジェクト指向におけるオブジェクトをさらに機能拡張し、データや手続きを状況に応じて適切に処理する自律的なプロセス（実行可能プログラム）のことである。SYSTEM νのエージェントは TA、UA、AA から成り立っている。

①TA (Transfer Agent　トランスファ・エージェント)

SYSTEM νにおけるオブジェクトの管理とデータ／処理要求の制御を行うエージェントである。SYSTEM νが提供するエージェントであり、SYSTEM νで構築された分散オブジェクトの制御を行う。ユーザは、TA へ直接アクセスすることはなく、ユーザが必要な処理はすべて UA と AA を使用して行う。TA には、「オブジェクトの管理」と「通信」の 2 つの機能がある。

a) オブジェクトの管理

TA は SYSTEM νで使用するオブジェクトを管理する。各 TA は、自分が管理するオブジェクト情報をテーブルとして保持し、他のエージェントからの要求がくると、自分のテーブルに登録されているオブジェクトに対して、参照、更新などの処理を行う。

さらに、TA はオブジェクトを制御する機能をもつ。例えば、1 つの TA の管理するオブジェクトに処理が集中すると、システムの効率が悪くなることがある。このような場合、TA は別の TA にオブジェクトを分散できるため、システムの効率低下を防ぐことができる。

TA は最適なシステムリソースの配分を実現するための代表的な機能として、「インタフェースリポジトリの管理」、「オブジェクトの位置（実際の場所）の取得」、「オ

ペレーションの起動」、「属性の参照/変更」、「オブジェクトの動的移動」、「オブジェクトの動的コピー」を持っている。

b) 通信

TA は UA と AA の通信の仲介役であり、UA と AA が直接通信することはなく、必ず TA を経由して情報のやりとりを行う。

また、TA 同士も通信を行い、互いの管理しているオブジェクトの情報をやりとりする。

②UA (User Agent　ユーザ・エージェント)

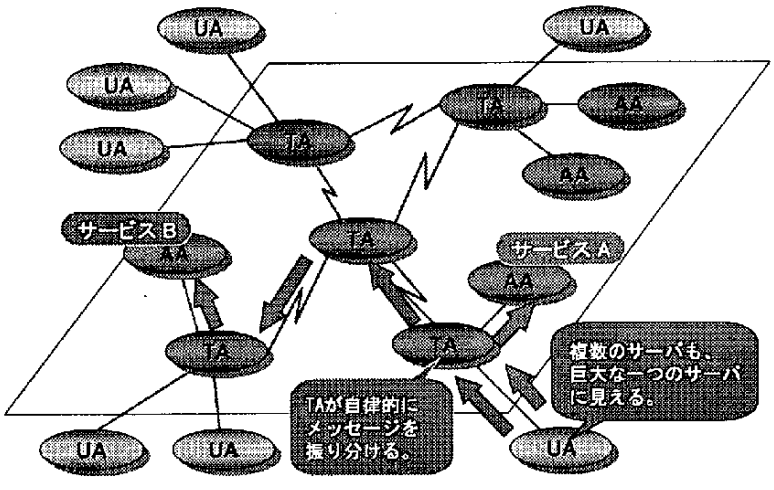
ユーザ・インタフェースを実現するエージェントである。フロントエンドの処理 (GUI) 制御を行う。すなわち、ユーザがオブジェクトに対するメッセージを送ったり、メッセージに対する返答を受け取ったりするための「窓口」としての役割を担っている。

UA は、ユーザが作成するエージェントであり、ユーザが使用する画面を定義し、画面からオブジェクトへ送信するメッセージを SYSTEM ν から提供される API(Application Interface)を使用して記述する。

UA は自分のメッセージがどのような経路で送られ、どこで処理されるかなどを一切考慮する必要はない。つまり、UA は TA や AA で構成される「TA 空間」と通信を行っていることになり、UA 自身がオブジェクトを所有しているようにプログラムを作成することができる。

③AA (Application Agent　アプリケーション・エージェント)

処理 (メソッド) を担当するエージェントである。すなわち、オブジェクトのオペレーションのインプリメンテーションを実行するエージェントである。オペレーションの実行要求は、必ず TA から送られてくるようになっている。



図表 2.3.1-1 System ν エージェントモデル

AA もユーザが作成するエージェントであり、ユーザが SYSTEM ν から提供される API を使用してオブジェクトのオペレーションのインプリメンテーションを記述する。

例えば、画面からあるデータを検索しようとした場合、「検索する」というメッセージをオブジェクトへ送るのは UA の仕事であるが、実際の検索処理を行うのは AA の役割である。つまり、AA はオブジェクトのインプリメンテーションを実行する働きをもっている。

MVC(Model-View-Controller)モデルとしてとらえると、TA は「Controller」、UA は「View」、

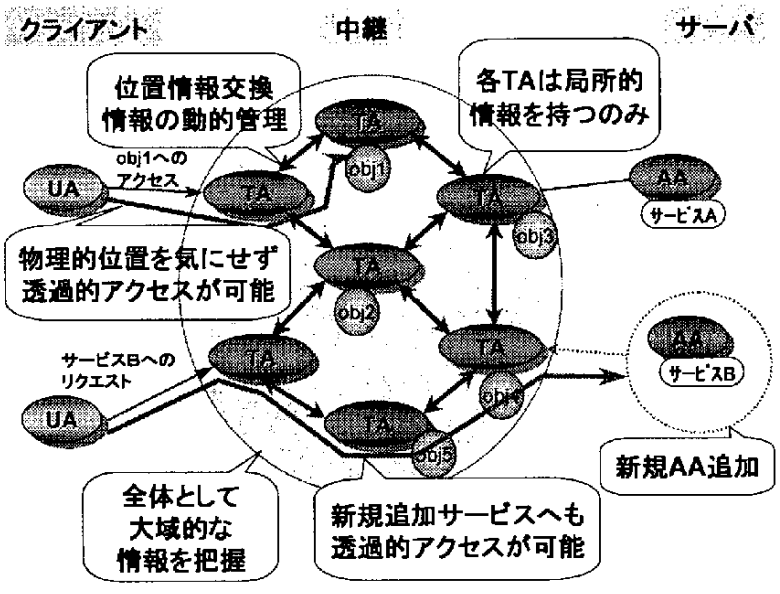
そしてAAは「Model」部分に相当する。なお、UA、AA、TA間のメッセージは、IIOPでやり取りが行われている。

（３）エージェント技術による効果

大規模なネットワーク上で、複雑な要素から構成される分散オブジェクト環境を人手に頼った方法で管理することは、もはや不可能な作業になろうとしている。

SYSTEM νでは、先進のエージェント技術によって、システム管理者に代わって自らを管理することを可能とした。通常のアプリケーションがユーザ（人間）の指示によって受動的に動作するのに比べて、エージェントは、つねに自律的に働くプロセス（ソフトウェア）である。複数のエージェントが互いに協調して働くことにで、分散オブジェクト環境の管理といった膨大な要因が絡み合い、そしてつねに変動する環境における作業をリアルタイムにこなすことが可能となる。

TAは、他のエージェント（UAやAAなど）から発せられるデータや処理要求などの通信制御と分散オブジェクトの管理を行っている。このTAのTA間での通信によってネットワークの動的な管理を行っている。



図表 2.3.1-2 エージェントの機能イメージ

①最適経路選択

各 TA は、オブジェクトやネットワークに関する情報を TA 同士で交換し、分散管理している。TA のこの自律的な協調動作により、オブジェクトへのアクセスに最適な経路を自動選択することが可能となった。特定のサーバへの経由を必要としないため、C／SS で問題となっていたトラフィックや処理の集中を防ぐことができる。

②障害時対応

各 TA は、オブジェクトやネットワークに関する情報を TA 同士で交換し、分散管理している。TA のこの自律的な協調動作により、オブジェクトへのアクセスに最適な経路を自動選択することが可能となった。特定のサーバへの経由を必要としないため、C／SS で問題となっていたトラフィックや処理の集中を防ぐことができる。

③オブジェクトの複製・再配置

アクセス頻度や処理負荷などの状況に応じ、ネックとなっているオブジェクトの複製や再配置を事前の設定に基づき、自動的に実行する。従来大きな問題となっていた大規模ネットワークの管理業務が大幅に削減される。オブジェクトの物理的な位置管理は TA にまかされており、アプリケーションも、開発者も煩雑な位置指定を考慮する必要がなくなる。SYSTEM ν の位置透過性によって、オブジェクトの論理的なつながりから独立して、より効率的なネットワーク構成に自由に変更ができる。また、ネットワーク間の連携や階層化といった構成上の大きな変更も、ソフトウェアの修正を伴わないで済むのである。

また、System ν 内のオブジェクトは、ネットワーク・トラフィックや CPU 負荷、さらにディスクやメモリの容量などを考慮し、SYSTEM ν ネットワークの中で最適な TA へ自動的に移動する（オブジェクト・マイグレーション）こともできる。セキュリティの保持やオブジェクトの性質に応じた細かなチューンアップのために、オブジェクトごとにマイグレーション範囲を指定することもできる。

④プラグ＆プレイ

サーバなどの物理的な位置を意識せずに、分散オブジェクトを駆使したプログラミングが可能となり、広域ネットワークを基盤とした大規模なアプリケーションの開発や変更がスピーディに行える。ハードウェアの追加変更に際しては、LAN に接続したクライアントに UA をインストールし、認証キーを登録するだけで、SYSTEM ν への参加が可能となる。

⑤ディレクトリ機能

SYSTEM ν の分散オブジェクトは、生成時にシステムで一意なオブジェクト識別子（OID）が付けられる。UA や AA がオブジェクトへのアクセスを行うときは、この OID を指定する。サーバ名などの物理的な位置指定は不要である。SYSTEM ν はオブジェクトの位置を示すディレクトリテーブルを持っている。各オブジェクトの OID とそのオブジェクトを保持している TA の物理的位置が表形式で管理され、システムの初期状態では各 TA は自分の保持しているオブジェクトの情報のみを持っている。このディレクトリを自動的に更新するしくみをインクリメンタルと呼ぶ。各 TA は、最新のディレクトリ情報を他の TA から取得し、自らのディレクトリを更新する。このインクリメンタルによって、動的なディレクトリ管理が可能となる。

（4）まとめ

本項では、エージェント技術をメッセージ基盤（CORBA）を構築する上での基礎技術と位置づけ、開発した製品を紹介した。本製品のもう一つの特徴は、ユーザからエージェント技術を隠蔽している点にある。そうすることによって、エージェントに不慣れなユーザがアプリケーション構築時にエージェントを意識することなく、しかしながら、エージェント技術の恩恵に預かりミッションクリティカルなシステムを構築できる点にある。

このように、エージェント技術は、大規模分散、ミッションクリティカルな分野においても適用される重要技術の一つとなっていくものと思われる。

参考文献

〔1〕 大規模広域分散システム開発環境 System ν A White Paper

〔2〕 OMG:http://www.omg.org/corba/corbaiiop.html

2.3.2 現実への利用を指向したe-Marketplaceの開発

(1)概要

ここでは、現実の問題やビジネスアプリケーションへの利用を指向して作られたエージェントシステムについて述べる。最初に、Aglets上にエージェントマーケットプレースを作ることを目的として作られたe-Marketplaceについて、その利用である商用旅行サイト「たびCan」とともに説明する。そして、e-Marketplaceと同様のものを性能の観点から再考し、作り直したCaribbeanと呼ばれるエージェントシステムについて述べる。

(2)e-Marketplaceのコンセプト

①対象とする範囲

e-Marketplaceはインターネット上でのショッピングを対象にして、この中での顧客や店舗の振る舞いをモデル化することを目的としている。しかしながら、ショッピングは様々な過程を含んでおり、それらの過程を整理し、我々がどの部分にフォーカスしているのかを明確にする必要がある。CBB（Consumer Buying Behavior）のモデル〔1〕では、顧客行動の観点から、６つのステージを同定している。

1）必要なものの同定
2）商品の選択
3）店舗の選択
4）交渉
5）購入と配達
6）サービスと評価

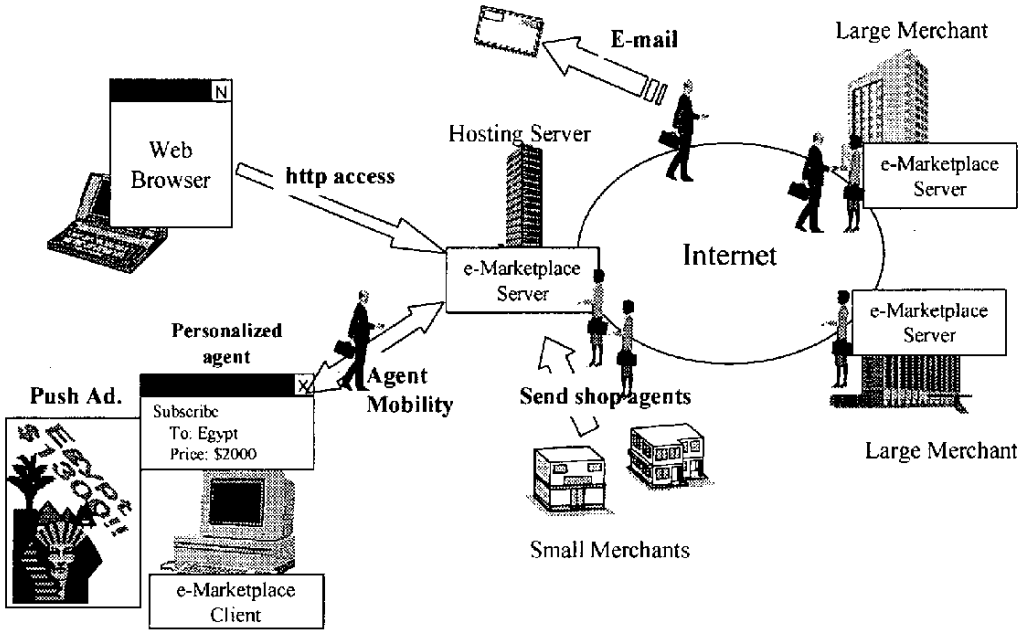
この中で、5）と6）に関しては、セキュリティや信頼性等が重要であり、業界の標準的なプロトコルに従って処理される。例えば、SET（Secure Electronic Transaction）〔2〕やOTP（Open Trading Protocol）〔3〕等である。これに対して、1）から4）に関しては様々な形態による実現が可能であり、エージェント技術も主にこれらのステージに対して適用される。本研究では特に2）と3）に着目して、エージェントシステムを開発しており、将

来的には 1）と 4）についても取り組んで行きたいと考えている。

②仮想コミュニティ

ここでは、エージェント・マーケットプレースに関して我々が描いている全体像に関して述べる。現在、インターネット上には多数のショッピングサイトがあるが〔4、5〕、これらは単に独立して存在しているに過ぎない。そのため、利用者が商品を探したい場合、それぞれのサイトに個別にアクセスし、条件入力等を逐次実行しなければならない。これに対して、我々は関連するショッピングサイトをまとめて、仮想的なコミュニティを作りたいと考えている。

我々が考えている仮想的なコミュニティの典型的な構成例を図表 2.3.2-1 に示す。小規模の店舗は、ホスティングサービスを提供しているサーバーに店舗エージェントを送り込んでおく。大規模な店舗は、自分でサーバーを設置し、店舗エージェントもレガシーシステムとリンクされている。一方、顧客はブラウザーによりアクセスし、自分用の顧客エージェントを生成する。顧客エージェントは、顧客からの要求を持って、様々なサイトを巡回して情報を集め、電子メール等によって結果を通知する。さらに、顧客がエージェントの実行環境を持っている場合には、顧客エージェントは端末に帰って、画面を表示したりすることもできる。



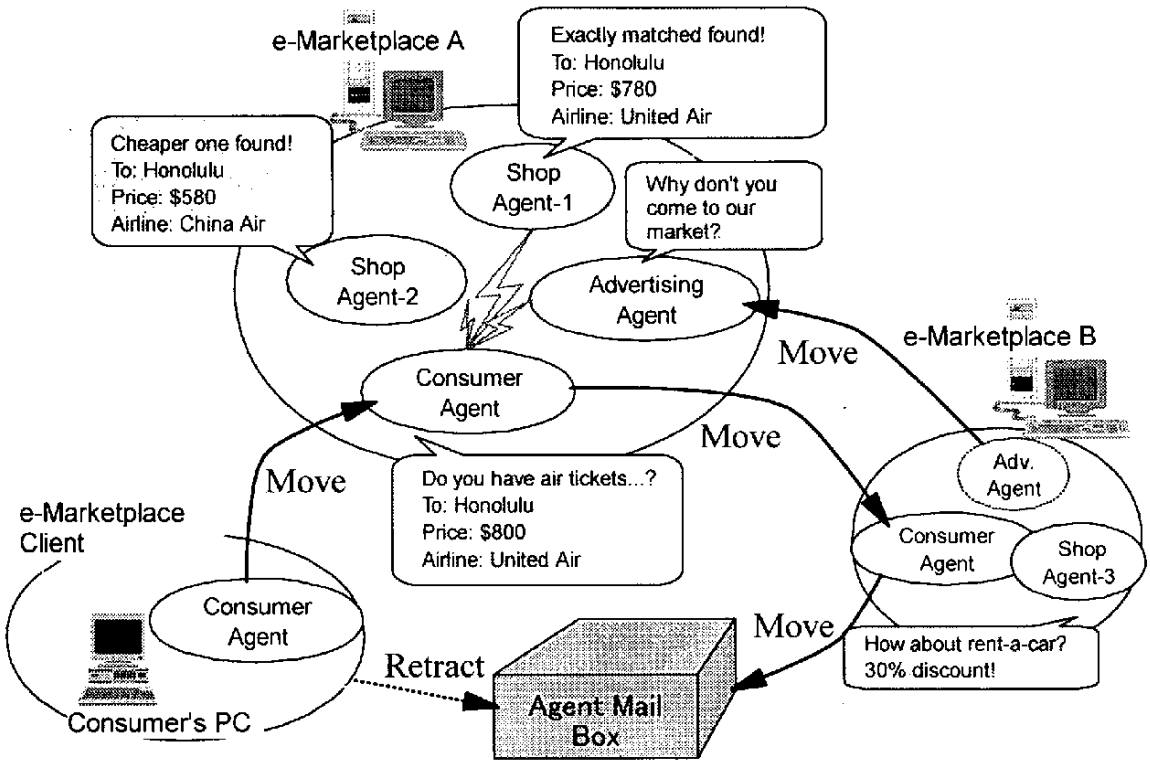
図表 2.3.2-1　移動エージェントを利用した仮想的なコミュニティ

③典型的なシナリオ

前節で述べた仮想コミュニティについて具体的に説明するために、旅行情報の検索を例としてアプリケーションのシナリオを示す。図表 2.3.2-2 は旅行代理店のコミュニティにおいて、顧客のエージェントが旅行情報を検索する様子を示したものである。顧客はエージェントの実行環境（e-Marketplace クライアントと呼ぶ）により自分専用の顧客エージェントを生成し、航空券に関する条件を入力し送り出す。顧客エージェントはマーケット A に到着すると、「ユナイテッド航空でホノルルまで$800 ドル以下でいく航空券が欲しい」

と叫ぶ。各店舗エージェントは、この質問に対して自分の都合の良い商品情報を提供する。例えば、店舗１は条件にマッチしたものを返しているが、店舗２は航空会社を替えて安い航空券を返している。一方、宣伝エージェントはマーケットＢを宣伝するために送り込まれてきたものであり、顧客に対して客引きを行う。その結果、顧客エージェントはマーケットＢに移動していき、そこでは店舗３がレンタカー情報を提供している。

我々は、エージェントは自分勝手な（self-serving）ものと仮定し、これらのエージェントがやり取りをする場所、すなわちマーケットプレースを提供したいと考えている。そして、このようなマーケットプレースを複数つなぎあわせて仮想的なコミュニティを作るための手段を提供するのが e-Marketplace の目的である。



図表 2.3.2-2　e-Marketplace における典型的なシナリオ

(3)e-Marketplace の構成要素

（２）節において、e-Marketplace の構想ならびにエージェントの振る舞いについて、大まかに述べた。本節では、e-Marketplace の構成要素を網羅的に列挙すると共に、それぞれに関して詳細に述べる。

①実行モジュール

e-Marketplace の構成を考える際の最も大まかな構成要素は実行モジュールであり、これには４種類のものがある。

e-Marketplace サーバー：エージェントの実行環境であり、たくさんのエージェントがここに集まることから、サーバーと呼ぶことにする。サーバーは多数のエージェントの実行を制御する機能を持つ必要があり、管理者のためのツール群を提供することも重要となる。

e-Marketplace クライアント：顧客や小規模の店舗が持つエージェントの実行環境であり、少数のエージェントのみが実行される。エージェントにとってはサーバーとの差異はほとんどないが、実行環境としてはサーバーほどのスケーラビリティや安定性は必要とされない。

ウェブ・ブラウザ：顧客や小規模の店舗が、e-Marketplace クライアントの代わりに利用する。機能的には e-Marketplace クライアントには劣るが、ユーザが新たなソフトウエアとして何もインストールする必要がないという利点がある。

エージェント・メール・ボックス：与えられたタスクを実行したエージェントが、端末に帰れないときに、一時的に保存されるための場所である。メールボックスと同様に、e-Marketplace クライアントから取りにいくことによって（retract）、エージェントは端末へと戻ることができる。このモジュールはユーザが接続を一端切ってしまう場合や、ファイアウォールの中から接続している場合に利用される。

②エージェント

e-Marketplace で実行されるエージェントは以下の３種類に分類されている。

顧客エージェント：顧客の代理となって、様々なマーケットを廻って商品情報を集める。顧客端末において、顧客から欲しい商品に関する条件や好みを受け取り、e-Marketplace サーバーに移動する。移動先では、店舗エージェントとやり取りして、商品情報を収集し、場合によっては他のサーバーに移動して情報収集を続ける。

店舗エージェント：店舗の代理となって、やってきた顧客エージェントに商品情報を提供する。通常はサーバー上で生成され、顧客からの要求に応じて自分の商品データベースを検索して結果を返す。この際、店舗の販売ポリシーに基づいて、正確に一致したもの以外にお薦め商品を返すこともある。また、小規模の店舗では、店舗エージェントを店舗の端末で生成し、これをホスティングサーバーに送った後で接続を切ることもある。

宣伝エージェント：ある特定の e-Marketplace サーバーを宣伝することを目的としている。あるサーバーで生成され、そこから別のサーバーに移動して自分のサーバーを宣伝する。

③共有資源

各 e-Marketplace サーバーには、参加しているエージェントに様々なサービスを提供できるように**共有資源**と呼ばれるものが用意されている。共有資源の種類はサーバーごとに固有であるが、以下のようなものが典型的である。

掲示板：特定の商品（例えば、特価品）情報を一定期間載せておくための場所である。顧客エージェントは自分が欲しい商品に関する条件を登録しておき、店舗エージェントが条件に一致するものを掲載した時に、その商品情報を知らせてもらうといった機能が提供される。

ファシリテータ：特定の要求を処理できるエージェントにはどのようなものがあるのか、またどのサーバーにいるのかといった情報を紹介するためのサービスを提供する。

④対話プロトコル

e-Marketplace では、未知のエージェント同士が対話することを想定している。従って、

エージェント間の対話に関する規則、すなわち対話プロトコルを定義しておくことが不可欠となる。実際の対話の中味は、商品に関する情報のやり取りであることから、商品データの表現規則（メタデータ）と、提供されるサービスという２種類に分けて考えることができる。

商品のメタデータ：あるタイプの商品がどのような属性によって表現されるかを定義したものである。例えば、航空券は「出発地」「行き先」「出発日」「値段」等の属性により表現される。

サービス：商品のタイプを特定しても、その商品タイプに関して様々なサービスを定義することが可能である。例えば、商品の検索、関連商品の検索、関連情報の検索、等である。これらのサービスは明示的に表現されることが必要であり、これによりエージェント同士が特定のサービスに関してやり取りすることが可能となる。

エージェント間の対話に加えて、エージェントは共有資源とも相互作用するので、共有資源とやり取りするための規則も定義しなければならない。共有資源に関しては、以下のような３種類のメッセージが用意されている。

更新メッセージ：エージェントが共有資源を更新する際に使われるメッセージである。このメッセージは非同期であり、更新要求を出して実際の更新まで時間差が生じる場合もある。

参照メッセージ：エージェントが共有資源の内容を参照するためのメッセージである。このメッセージは同期型であり、答えは戻り値によって返される。

イベント：共有資源上に変化があった場合に、その変更を伝えるために送信されるメッセージである。エージェントはイベントを受け取るために予めどのようなイベントを受け取りたいかを登録することが前提であり、そのためのサブスクライブ機能も必要となる。

(4)e-Marketplace フレームワークの実現

①アーキテクチャ

図表 2.3.2-3 に e-Marketplace フレームワークのアーキテクチャを示す。このフレームワークは、Aglets〔7〕上に構築されており、特に多数のエージェントをホスティングした場合にもシステムの負荷を軽くすることを目的として設計されている。このようなシステムのオーバーロードの問題は、我々が「たび Can」（（５）節で述べる）を開発した際に、実際に直面したものである。

図表 2.3.2-3 に示すように、本フレームワークでは、エージェントと共有資源の抽象クラスを提供し、これらを具体化することによりアプリケーションを開発するようになっている。さらに、対話プロトコルやスケジューリング方法はアプリケーションごとに異なることから、これらの定義をファイルに記述し、これらを読み込みようになっている。

本システムの特徴は、対話プロトコルに基づいたスケジューリング機構を提供している点にある。仕組み自身については文献〔7、8〕に詳述してあるので、ここではその概要のみを述べる。エージェントは別のサーバーまたはクライアントから移動してきて、他のエージェントと対話する。この際、エージェント間の対話プロトコルがあらかじめ決められており、メッセージモニターは、対話が正しく行われているのかを常に監視している。対話プロトコルには一連のメッセージの流れ、すなわちセッション（後述する）に関する規則も定義されており、このような情報に基づいて、メッセージモニターはあるセッション

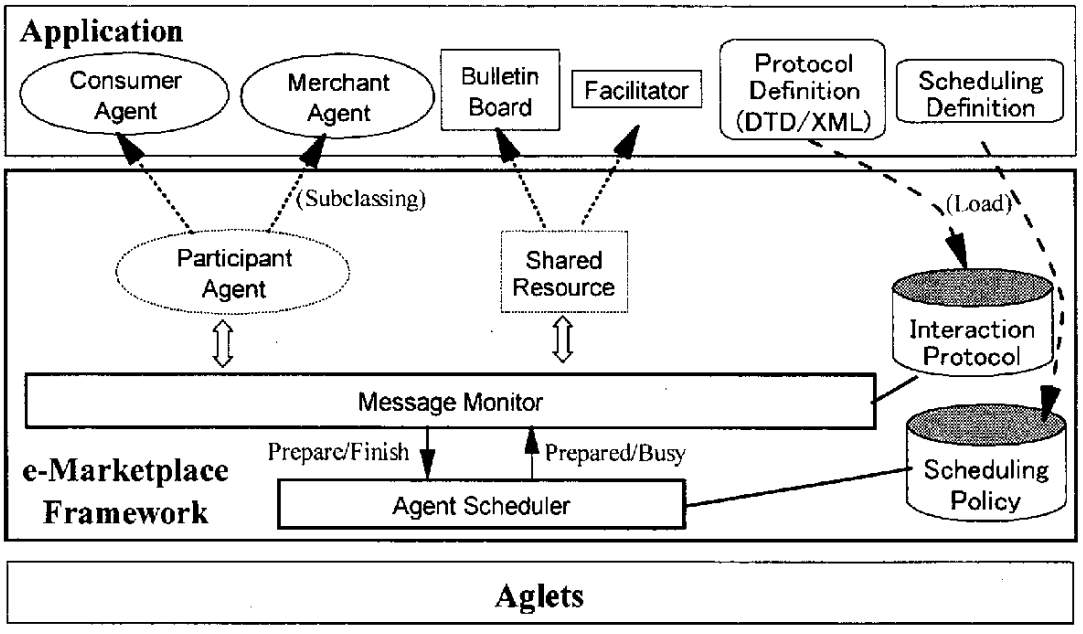


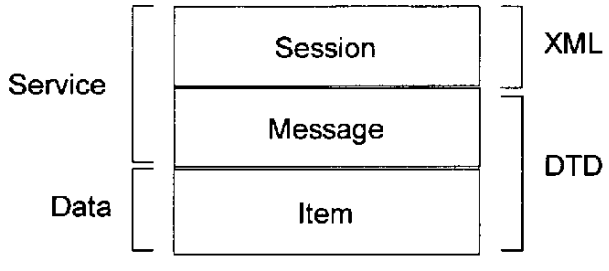
図 2.3.2-3　e-Marketplace フレームワークのアーキテチャ

にどのエージェントが参加しているのかも知ることができる。このようなエージェントの利用に関する情報はエージェントスケジューラーに送られ、この情報に基づいて、エージェントをディスクからメモリーにロード（Activate）したり、逆にディスクへと格納（Deactivate）したりする。このような仕組みにより、メモリーの消費を大幅に抑えることができ、多数のエージェントのホスティングが可能となっている。

②対話プロトコルの表現

図表 2.3.2-3 にも示してあるように、対話プロトコルは XML ならびに DTD によって記述してある〔9〕。本節では、たび Can からの例を使って、対話プロトコルの表現法について具体的に説明する。なお、紙面の制約上、ここではエージェント間の対話規則のみを示し、共有資源に関する規則は別途報告する。

（3）－④でも述べたように、エージェント間の対話規則に関しては商品データとサービスに分けて考える必要がある。商品データに関しては単にメタデータを定義するだけであるが、サービスに関しては様々な可能性が考えられる。筆者らは、図表 2.3.2-4 に示すように対話プロトコルのレイヤーを「アイテム」「メッセージ」「セッション」に分け、上位２つのレイヤーによってサービスが定義されるものとしている。ここでアイテムとは、やり取りされる商品のことであり、このレベルでは商品に関するメタデータが定義される。



図表 2.3.2-4　対話プロトコルのレイヤー

メッセージは商品に関しての要求を発行したり、商品情報を返すためのものであり、それぞれに関してメッセージタイプが定義される。セッションは一連のメッセージの流れを規定するものであり、例えば、要求メッセージの後に回答メッセージが送信される、といったことを定義する。

表現方法に関しては、アイテムのメタデータとメッセージのタイプに関しては DTD (Document Type Definition) により表現し、セッションは XML により表現している。これは、アイテムとメッセージはエージェント間の対話で実際にやり取りされる実体であり、これを XML により表現できればエージェント間のやり取りを全て XML とすることが可能となるからである。一方、セッションに関しては、エージェント間のやり取りが正しく行われているかを検証するために内部的に使われるものであり、実際にセッションに関する情報がエージェント間でやり取りされるわけではないので、記述力の高い XML で表現することにしている。以下では、それぞれの表現法に関して例を示しながら説明する。

1) **アイテム**:

商品アイテムのメタデータは DTD で表現されるが、次のような２つの点で通常の DTD よりも表現が拡張されている。１つは属性の型を指定できる点であり、これにより属性には、String、Integer、Long、Date などの型を指定することができる。さらに、コレクションや列挙型といったものも指定できるようになっている。もう一つの拡張は、属性の継承を表現できる点である。アイテムのメタデータはカテゴリーと呼ばれており、上位のカテゴリーを特定することにより継承を表現できるようになっている。なお、最上位のカテゴリ

```
<!ELEMENT Pack
  (ProductName,TourID,Title,Priority,
   Area,LeavingCities, Interests,
   Destinations,Duration,LeavingDate,
   Price,MemberPrice,AgencyPrice,Child,
   Status,Airline,Hotel,MinParticipant,
   Promoter,Staff)>
<!ATTLIST Pack
   super (TravelGoods) #REQUIRED>
<!ELEMENT ProductName (#PCDATA)>
   <!ATTLIST ProductName
       type (String) #REQUIRED>
<!ELEMENT TourID (#PCDATA)>
   <!ATTLIST TourID
       type (String) #REQUIRED>
.....
<!ELEMENT Destinations (AbraodCity*)>
   <!ATTLIST Destinations
       type (Collection) #REQUIRED>
       <!ELEMENT AbraodCity (#PCDATA)>
          <!ATTLIST AbraodCity
              type (String) #REQUIRED>
.....
<!ELEMENT LeavingDate (#PCDATA)>
   <!ATTLIST LeavingDate
       type (Date) #REQUIRED>
<!ELEMENT Price (#PCDATA)>
   <!ATTLIST Price
       type (Integer) #REQUIRED>
.....
<!ELEMENT Status EMPTY>
   <!ATTLIST Status
       type (String) #REQUIRED>
   <!ATTLIST Status
       enum (E|C|F) #REQUIRED>
```

図 2.3.2-5　カテゴリーの表現

ー（Java における Object）は Thing と呼び、全てのカテゴリーは Thing のサブカテゴリーとなっている。例として、たび Can における旅行のパッケージツアーに関するカテゴリーを DTD で表現したものを図表 2.3.2-5 に示す。

図表 2.3.2-5 において、最初の要素型宣言における Pack はカテゴリーの名前であり、ここではパッケージツアーというカテゴリーを定義することを意図している。それに続く引数は Pack の属性名を列挙してあり、ここでは 20 の属性が列挙してある。次の属性リスト宣言では、上位（super）のカテゴリーが TravelGoods であることが示されており、その属性が継承されることを意味している。

ProductName 以下は実際に属性を定義している部分である。ProductName と TourID はそれぞれ String 型であることが定義されており、Destinations は目的地が複数の場合も想定して String のコレクション型であると定義されている。また、Status は予約状況を表すものであり、十分に空あり（E）、空きが少ない（C）、満席（F）のどれかであることを意味する列挙型による指定がなされている。

2）メッセージ：

メッセージタイプもカテゴリーと同様に DTD により表現される。図表 2.3.2-6 にメッセージタイプを表現したものを示す。最初のメッセージ（RequestTravelGoods）は顧客から店舗へと商品情報を要求するものであり、引数には要求内容を問い合わせ（Query）文として表現したもの（Requirements）、ならびにその問いあわせに対して最大何個の答えが欲しいかを示すもの（MaxSize）が定義されている。ここで、問い合わせ文とは特定のカテゴリーに関する条件を SQL の Select 文のような形で表現したものであり、XML によって表現することができる。

第二のメッセージ（ProvideExactGoods）は、最初のメッセージによって与えられた問い合わせ文に対して、正確に一致する商品リストを返すものである。引数は Thing のコレクション型と定義されているが、Thing が全てのカテゴリーの最上位クラスであることから、この引数にはどのカテゴリーの商品でも入れられることを意味している。

最後のメッセージ（AdvertizeMarket）は、（３）－②で述べた宣伝エージェントが、自分のマーケットを宣伝する際に使うものである。引数には自分のマーケットの URL アドレスを指定することが定義されている。

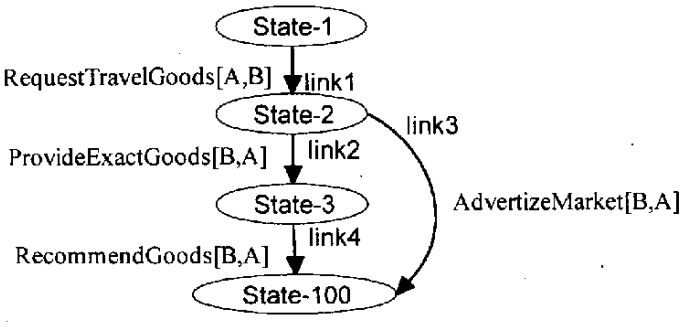
3）セッション：

エージェント同士がやり取りをする場合、単にどのメッセージタイプを使うかだけではなく、それらのメッセージ間の流れも決めておかなければならない。例えば、Request-TravelGoods で質問されたら、ProvideExact-Goods で答えを返す、等である。このようなメッセージの流れをセッションと呼ぶが、我々はセッションを状態遷移と捉え、XML により表現している。

図表 2.3.2-6 はたび Can において定義されている状態遷移をネットワーク表現したものである。State-1 が初期状態であり、State-2 と State-3 が中間状態、State-100 が最終状態となっている。link1 から link4 までが遷移を表しており、各遷移はメッセージタイプと送り手と受け手のエージェントを表す変数によって特徴づけられている。たび Can における典型的な対話は link1、link2、link4 を辿って推移する。すなわち、顧客エージェント C から店舗エージェント S へと RequestTravel-Goods が送信されると、最初に、

正確に条件に一致する商品を返すために S から C へと ProvideExactGoods が送信され、次に、お薦め商品を返すために S から C へと RecommendGoods が送信される。一方、link1 から link3 を辿る遷移は、顧客エージェントと宣伝エージェント間の対話を規定したものである。すなわち、宣伝エージェントは顧客から RequestTravelGoods のメッセージを受け取ると、AdvertizeMarket により、顧客へと自分のサーバーの URL を返す。

以上のような状態遷移を XML で表現したものを図表 2.3.2-7 に示す。最初の宣言は、これから request という名前のセッションを定義することを意味している。続く４つの宣言は遷移を表しており、図表 2.3.2-6 で示した link1 から link4 までを定義している。遷移を特徴付ける属性について説明すると、name がリンクの名前、utterance は対象とするメッセージタイプ名、prev と next は遷移前と後の状態、sender と receiver は送り手と受け手のエージェントを表す変数、type は遷移が初期状態からのものか (initial)、最終状態に行くものなのか (terminal)、それ以外か (intermediate: type のデフォルト値) を表している。



① メッセージタイプの表現

```
<!ELEMENT RequestTravelGoods
   (Requirements,MaxSize)>
<!ELEMENT Requirements (#PCDATA)>
   <!ATTLIST Requirements
      type (Query) #REQUIRED>
<!ELEMENT MaxSize (#PCDATA)>
   <!ATTLIST MaxSize
      type (Integer) #REQUIRED>

<!ELEMENT ProvideExactGoods
   (ExactGoods)>
<!ELEMENT ExactGoods (ExactGood*)>
   <!ATTLIST ExactGoods
      type (Collection) #REQUIRED>
   <!ELEMENT ExactGood (#PCDATA)>
      <!ATTLIST ExactGood
         type (Thing) #REQUIRED>

<!ELEMENT AdvertizeMarket (NewMarket)>
<!ELEMENT NewMarket (#PCDATA)>
   <!ATTLIST NewMarket
      type (URL) #REQUIRED>
```

② セッションの状態遷移

図表 2.3.2-6

状態遷移に続いて、メッセージの引数間の制約を記述する部分がある　(constraint)。この中で、transition はこの制約をいつチェックするかを指定しており、ここでは、link2 においてチェックすることが定義されている。GreaterThanOrEqual 以下は具体的な制約であり、link1 における RequestTravelGoods の MaxSize の値が、link2 における ProvideExactGoods の ExactGoods の要素数よりも大きくなければならないことを示している。なお、制約記述においては link1.MaxSize のようにメッセージ名は省略されているが、これは遷移を特定すればメッセージは一意に決まるからである。また、GreaterThanOrEqual は比較のためのオペレータであり、left と right はその引数、size はコレクションの要素数を返す関数である。

```
<session name="request">
<transition name="link1"
   utterance="RequestTravelGoods"
   prev="1" next="2"
   sender="A" receiver="B"
   type="initial" />
<transition name="link2"
   utterance="ProvideExactGoods"
   prev="2" next="3"
   sender="B" receiver="A" />
<transition name="link3"
   utterance="AdvertizeMarket"
   prev="2" next="100"
   sender="B" receiver="A"
   type="terminal" />
<transition name="link4"
   utterance="RecommendGoods"
   prev="3" next="100"
   sender="B" receiver="A"
   type="terminal" />
<constraint transition="link2">
   <GreaterThanOrEqual>
      <left> link1.MaxSize </left>
      <right>
         <size> link2.ExactGoods </size>
      </right>
   </GreaterThanOrEqual>
</constraint>
</session>
```

図表 2.3.2-7　セッションの表現

(5) たび Can

我々は、e-Marketplace の考え方に基づいて、97 年 12 月より「たび Can」という旅行情報サービスを開始している〔10〕。現在は、サーバー 1 台を IBM が運営し旅行代理店が出店するというホスティングサービスの形態となっている。さらに、98 年 8 月にはサーバーを 2 台にし、実際にエージェントがマシン間を移動するようなシステムを開発しサービスを開始した。但し、これらの 2 台のマシンは IBM の計算機センターにおかれており、エージェントがインターネット上を移動するわけではない。むしろ、エージェントが大量に移動したときのシステムの負荷や安定性に関してのシステムテストの意味合いが強い。

(6)Caribbean の実現

前節までに述べた e-Marketplace は Aglets ベースで構築されているが、元々のコードが移動、セキュリティ、複雑なメッセージングなどの様様な機構を次々と取り入れていったために、性能の向上には限界があった。そこで、e-Marketplace と同様の機能を特に性能の観点から設計しなおし作られたのが Caribbean である〔11〕。ここで考慮したシステム的な問題は、メモリ管理とスレッド管理、障害対策であり、以下ではこれらを順次説明していく。なお、実際の管理方法は、エージェントの振る舞い方により異なる。例えば、ユーザ・エージェントは、その数が数万になり得るので、メモリ管理を必要とするが、サービス・エージェントは、数が数十程度であるので、常にメモリ中に置くことができる。したがって、Caribbean では、エージェントをグループ化し、それぞれに管理の仕方を定義できる機能も提供している。

また、実システムを運用する場合、バグ対応やバージョンアップによるアプリケーションの変更は重要な課題である。Caribbean では、エージェントのインスタンス変数としてデータを保持する。したがって、エージェントのクラスを変更すると、それ以前のエージェントは無効になり、それが保持していた情報は失われてしまう。Caribbean では、このような場合、古いバージョンのエージェントが保持していたデータを新しいバージョンのエージェントに渡すための機構を提供している。以下では、これに 2 つの機能に関しても言及する。図表 2.3.2-8 に Caribbean のアーキテクチャを示す。

1) メモリ管理：

Caribbean では、1 つのエージェント・サーバー上で数千、数万というエージェントが活動する。エージェントは、その実態はデータをもつオブジェクトである。1 つのオブジェクトの大きさは、アプリケーションに依存するが、典型的なアプリケーションでは数 K バイト～数十 K バイト程度である。しかしながら、このようなエージェントが大量に存在すると大量のメモリを必要とすることになる。例えば、10K バイトのエージェントが一万個存在すると、100M バイトのメモリを必要とする。インターネットでの利用を考えると、数万人という利用者数が考えられる。その場合、数百 M バイトのメモリになる。したがって、エージェント・サーバーは実メモリ使用量を制限しながら大量のエージェントを管理するための機構が必要である。

このために、Caribbean は、エージェントをスワップイン・アウトする機構を提供している。これは実行環境により自動的に行われるため、開発者が明示的に行う必要はない。実行環境は、メモリ上に置くことができるエージェントの数を設定された値以下に制限する。スワップイン・アウトのスケジュールは、なるべく性能を向上するように、スワップ

イン・アウトの回数を減らす方法で行われる。

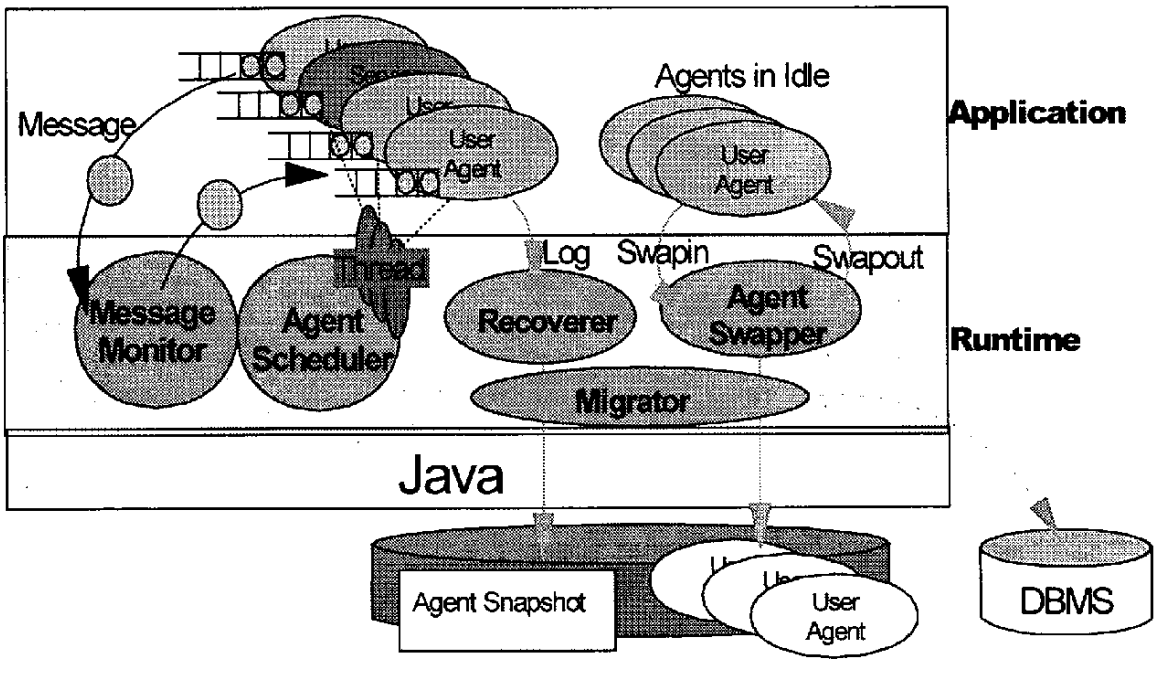
2) スレッド管理：

ユーザ・エージェントは先に述べた通り数千から数万存在する。このような数のエージェントが勝手に他のエージェントとコミュニケーションを行うと、同時に多数のスレッドを使用することになり、システムは過負荷になる。また、情報を提供するエージェントが多数存在し、情報を検索するエージェントがそれらのすべてのエージェントから情報を取得するというアプリケーション（例えば、多数の店舗が集まる電子モールで個々の店舗をエージェントとして実現した電子モール・システム）の場合、利用者のエージェントは、一回の検索処理で多数のエージェントとコミュニケーションを行うことになる。ここで、もし、すべての情報提供エージェントが同一のデータベースマネージャ（DBMS）に接続しているとすると、この DBMS に同時に多数のアクセスが集中し、性能を低下させてしまう可能性がある。したがって、エージェント・サーバーには、同時に使用されるスレッドの数を制限して、エージェントの活動を適切にスケジュールする機構が必要である。

Caribbean の実行環境は、スレッド・プールを持ち、スレッド数を設定された値以下に制限する。エージェントへのスレッド割当てのスケジュールは、エージェントがメモリ中にあるか、二次記憶にスワップアウトされているかを考慮して、スワップイン・アウトの回数を減らし、なるべく性能を向上するような方法で行われる。

3) 障害対策：

エージェントは、論理的にはその生存期間内において永続的である。これは、システムに障害が発生しても、障害回復後にエージェントが復活しなければならないことを意味する。内部状態を変更しないエージェントは再度生成すればよいが、ユーザ・エージェントのように情報を保持するエージェントは、その情報を保持した状態で復活しなければならない。



図表 2.3.2-8　Caribbean のアーキテクチャ

したがって、エージェント・サーバーには、このような障害復帰機能が必要である。

Caribbean の実行環境は、エージェントがその内部状態を変更した場合に、自動的にエージェントのメモリ・イメージを二次記憶に書き出すロギング機能を提供している。エージェントの状態は、メッセージを受信したときに呼び出されるコールバック・メソッド（handleMessage）内で変更される。このメソッドの戻り値は Boolean であり、その値が TRUE の時に、エージェントの内部状態が変更されたと解釈する。実行環境は、その再起動時に、ログに保管されているメモリ・イメージからエージェントを回復する。

このロギング機構は、エージェント・サーバーの起動パラメータでオン・オフできる。

ロギングの性能を向上させるため、一度のディスク・アクセスで複数のエージェントのメモリ・イメージが書き出されるように「グループ・ロギング」機構を採用している。

エージェントのグループ化：Caribbean では、色々な種類のエージェントが一つのエージェント・サーバーで活動する。これらのエージェントの振る舞いは、その役割い応じて幾つかに分類される。例えば、インターネット・バンキングの例では、ユーザ・エージェントは、その数が非常に多く、それぞれのエージェントは利用者のための情報を保持する。また、個々のユーザ・エージェントは、そのエージェントの利用者がアクセスするか、サーバー内で生じたイベントに反応する時だけ活動する。これに対し、サービス・エージェントは、その数はせいぜい数十程度であり、エージェントが自分の状態を変化させることはない。しかしながら、このエージェントはすべてのユーザ・エージェントに共有されるため、頻繁に呼び出される。このような振る舞いを考慮すると、使用メモリを制限するために、ユーザ・エージェントをスワップイン・アウトする必要があるが、サービス・エージェントは常にメモリ上に置く方がよい。また、ユーザ・エージェントは、それが持つ情報が更新されるため、障害対策のためのログをとる必要があるが、サービス・エージェントは、システム回復時に新しいエージェントとして生成しても問題はないのでログをとる必要はない。

Caribbean では、エージェントをグループ化して管理し、メモリ中に保持できるエージェントの最大数、スレッド数、ロギングのオン・オフを、それぞれごとに設定できる機能を提供している。これらは、エージェント・サーバーのパラメータとして設定できるので、アプリケーション開発後にシステムの性能等を考慮して決めることができる。

4）プログラムのバージョンアップ：

システムを運用していく場合、バグ対応や機能追加など、既存のプログラムを変更したり、別のプログラムに入れ替えることはよく行われることであり、重要なことである。一方、Caribbean では、永続的なエージェントのデータはエージェントのインスタンス変数として保持される。そのため、そのようなエージェントのプログラムを変更すると、そのエージェントを新しいプログラムで作り直すことになり、古いエージェントが保持していたデータが失われてしまう。

このようなことを避けるために、Caribbean の実行環境は、古いエージェントから新しいエージェントへデータを移項するための柔軟な仕組みを提供している。これは、コンバーター・プログラムと呼ぶプログラムを実行環境に登録することで行われる。

永続的なエージェントは、ロギングやスワップアウトにより二次記憶へ書き出される。コンバーター・プログラムが登録されていない場合は、エージェントのオブジェクトが

Java の Serialize 機構を使用して二次記憶へ書き出される。このプログラムが登録されている場合、二次記憶へ書き出される直前に、エージェントのオブジェクトはこのプログラムのオブジェクト書き出し用のコールバック・メソッドに渡される。ここで、コンバータープログラムは、エージェントが持つデータを別のオブジェクトにコピーし、そのオブジェクトをコールバック・メソッドの戻り値として返す。Caribbean の実行環境は、ここで返されたオブジェクトを二次記憶に格納する。二次記憶から再びメモリへ読み込まれるときも、コンバーター・プログラムが登録されている場合は、そのプログラムのオブジェクト読み込み用のコールバック・メソッドが呼び出され、二次記憶に格納されていたオブジェクトからエージェントのオブジェクトに値をコピーして、エージェントとして再生成される。

この仕掛けにより、コンバーター・プログラム内で、エージェントのプログラムを入れ替えることが可能となる。コンバーター・プログラムが変換するオブジェクトは、Java で定義する Serializable なものであれば、どのようなものでもよい。また、オブジェクトとして格納する代わりに、データをデータベースへ格納することも可能である。この場合、コンバーター・プログラムの書き出し用のコールバック・メソッドは、データベースへ値を格納して、戻り値に NULL を返す。エージェントが読み込まれる場合は、コンバーター・プログラムの読み込み用のコールバック・メソッドにエージェント ID が渡されるので、それをキーにしてデータベースからデータを取り出し、エージェントを再生成すればよい。

エージェントの内部構造はアプリケーションに依存しているので、コンバーター・プログラムは、プログラムを変更するエージェントのクラスごとに作らなければならない。このプログラムは、エージェントのクラスごとに登録できるので、永続的でないエージェントやプログラムを変更しないエージェントに対しては、コンバーター・プログラムを登録する必要はない。さらに、通常はコンバーター・プログラムを使用せず、プログラムを変更するときだけこのプログラムを登録し、対象となるエージェントすべてのプログラムを変更し、その後でそのプログラムをはずすことも可能である。

Java の Serialization 機構でも、オブジェクトの Serialize を、Java のデフォルトの機構を使用せず、開発者がプログラムすることも可能である。しかしながら、この場合、まったく別のクラスに変更することはできない。例えば、OldAgent.class のエージェントを NewAgent.class で作り替えるということはできない。ところが、システムを運用する場合、このような変更は十分ありえる。そのため、Caribbean では、このようなマイグレーション機構を提供している。もちろん、Java が提供する方法を用いることも可能であり、この機構と Caribbean のマイグレーション機構を併用することも可能である。

(7) Caribbean の性能評価

ここでは、Caribbean のエージェント・サーバーがどの程度のパフォーマンスを提供するのか、どのようなパフォーマンス特性を持つのかに関して述べる。

ここでは、以下のようなテスト・アプリケーションにおけるスループットを示す。

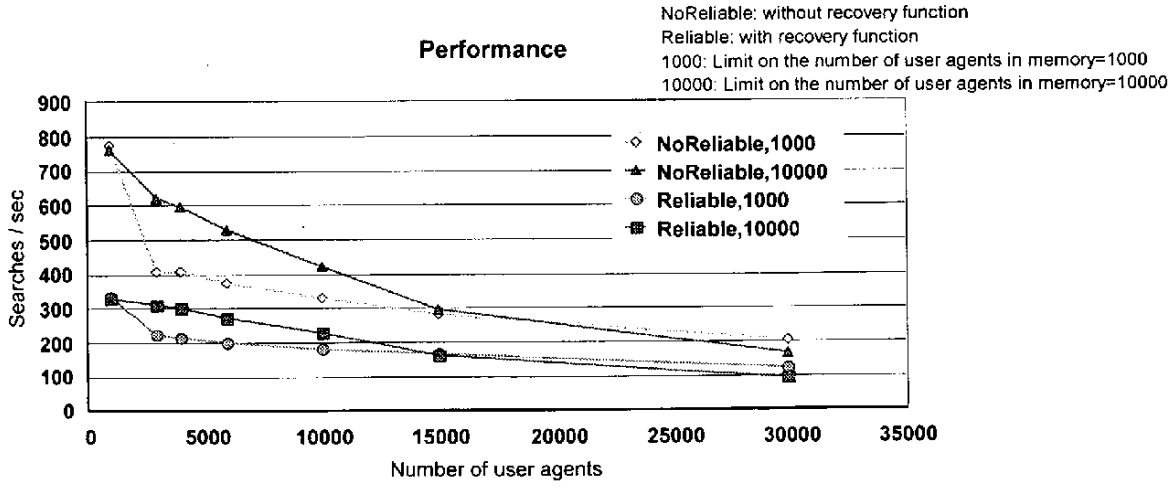
- エージェント・サーバーには、情報を検索するエージェントと情報を提供するエージェントが活動している

●情報検索エージェントは、サーバー内にいるすべての情報提供エージェントに対し、情報検索要求メッセージをマルチキャストする
●情報提供エージェントは、このメッセージを受けると情報を表すオブジェクトを10個作り、これらのデータを情報提供メッセージに載せて要求を発行したエージェントに渡す
●情報検索エージェントは、すべての情報提供エージェントからメッセージを受信したら一連の検索処理を終了とする
●情報検索エージェントは、これら一連の情報検索処理を連続して5回繰り返し、それ以降は何も処理を行わない
●1つの情報は10Bytesのバイト列で表され、情報を表すオブジェクトはこのバイト列を持つ。
●情報提供エージェントはサーバー上で5つ活動している。これらのエージェントが持つ情報はすべてメモリ上に置かれている。情報提供エージェントは常にメモリ上に置かれる。情報提供エージェントの最大スレッド数は5であり、すべてのエージェントにスレッドが割り当てられる。情報提供エージェントは永続的でなく、ロギング機能は使用されない。

この実測では、情報取得エージェントの数を、100から30000まで変化させる。また、メモリ中に置かれる情報検索エージェントの最大数は、1000と10000のそれぞれの場合で実測を行う。情報提供エージェントの最大スレッド数は5である。情報検索エージェントのロギング機構はオンの場合とオフの場合の両方を実測する。

実測用の計算機は、PentiumII 333MHz、メモリ256MBを搭載した計算機である。オペレーティング・システムはWindowsNT4.0 Workstation、Javaは1BM JDK1.1.7で、JITコンパイラを使用する。Javaの設定は、初期ヒープサイズを100MB、最大ヒープサイズを150MBである。

実測結果を図表2.3.2-9に示す。



図表2.3.2-9　エージェント・サーバーの性能

この結果からまず言えることは、エージェント・サーバー自体の性能は、実用上十分使用可能な範囲にあるということである。性能特性としては、情報検索エージェントの数が増加するにつれ、検索処理のスループットが低下していくことがあげられる。この原因は、主に２つある。一つは情報検索エージェントのスワップのコストである。エージェントの数が増えれば、その回数は多くなり全体の計測時間を大きくする。その結果、スループットは小さくなる。もう一つの理由は、Java のガーベッジ・コレクションのコストである。このコストを計測したところ、情報検索エージェントが 30000 の場合で全体の計測時間の20%を占めていた。

この実測は、あくまで現在の実装における実測値であり、今後性能を向上させるように実装を改良していく予定である。また、この実測では、エージェント数を 30000 までしか行っていないが、別のテストではエージェント数を 10 万にして行っており、エージェントの数においても実用的なレベルにあることを付け加えておく。

(8)まとめ

本節では、現実の問題やビジネスアプリケーションへの利用を指向して作られたエージェントシステムについて、開発の変遷に沿って述べた。最初に、Aglets 上にエージェントマーケットプレースを作ることを目的として作られた e-Marketplace について、その利用である商用旅行サイト「たび Can」とともに説明した。そして、e-Marketplace と同様のものを性能の観点から再考し、作り直した Caribbean と呼ばれるエージェントシステムについて述べた。結果的には、Aglets ベースのものに比べ、性能比で 10 倍ほどの効果となっており、現在想定しているアプリケーションではある程度使える見込みがついている。

参考文献

〔1〕 R. Guttman, A. Moukas, and P. Maes: "Agent-mediated Electronic Commerce: A Survey". To appear, Knowledge Engineering Review Journal.

〔2〕 Secure Electronic Transactions. http://www.mastercard.com/set/

〔3〕 Open Trading Protocol. http://www.otp.org/

〔4〕 Amazon. http://www.amazon.com/

〔5〕 CDNow. http://www.cdnow.com/

〔6〕 Aglets Workbench. http://www.trl.co.jp/aglets/

〔7〕 Y. Nakamura and G. Yamamoto: Aglets-based e-Marketplace - Concept, Archi-tecture and Applications, IBM Research Report, RT-0253 (1998).

〔8〕 G. Yamamoto and Y. Nakamura: Architecture and Performance Evaluation of a Massive Multi-agent System, pp. 319 - 325, Autonomous Agents '99 (1999)

〔9〕 XML. http://www.XML.com/XML/pub/

〔10〕 たび Can. http://www.tabican.ne.jp (99 年末でサービス終了)

〔11〕 G. Yamamoto and H. Tai: Architecture of an Agent Server Capable of Hosting Tens of Thousands of Agents, IBM Research Report No. RT0330 (1999)

2.3.3 マルチエージェント協調プロトコル記述 AgenTalk

マルチエージェントシステムの考え方を使って、アプリケーションを構築しようとする時、エージェント間でいかに協調させるかが大きな課題となる。一般にエージェントはお互いにメッセージを交換して協調を実現する。ここでは、協調を実現するためにエージェント間でやりとりされる一連のメッセージの交換におけるハイレベルなプロトコルを協調プロトコルとよんでいる。典型的な協調プロトコルの一例が契約ネットプロトコルであり、多くのアプリケーションに応用されている。

多種多様なアプリケーションの構築では、それぞれの応用領域に適した協調プロトコルが必要になろう。特にユーザの代わりとなって働くエージェントの実現では、ユーザ自らプロトコルを記述することも考えられる。このためには、プロトコルを一から記述していくのではなく、既存のプロトコルをベースにして段階的にプロトコルを定義することにより、応用に適したプロトコルを容易に実現できることが望まれる。

本節では、このような背景のもと、マルチエージェントシステムにおける協調プロトコルを記述する枠組みとして、ＮＴＴコミュニケーション科学研究所（当時）と京都大学石田研究室とで開発された AgenTalk〔1〕を紹介する。AgenTalk では協調プロトコルの記述にオブジェクト指向言語に見られるような継承機能を導入し、段階的なプロトコル定義を可能にしているのが大きな特徴である。これにより、あらかじめ設計したプロトコル群をライブラリとして保存しておき、新たなプロトコルを記述する時は既存のプロトコルとの差分だけを定義すればいいというメリットが生じる。

以下、AgenTalk の設計方針、その機能、さらに AgenTalk による契約ネットプロトコルの記述例を述べる。

（１）AgenTalk の設計方針

AgenTalk は協調プロトコルの形式的仕様を定義するというよりも、むしろ実際に動くプログラムの記述を目標にしている。さらに、エージェント自体がさまざまな枠組（例えば黒板モデル、プロダクションシステム）で記述できるように、エージェントの実現方法までは深く関わらず、プロトコル記述に注目する立場をとっている。プロトコル記述に関しては以下の方針で設計されている。

①プロトコルにおける状態の明示的な記述

AgenTalk では通信プロトコル記述でよく用いられている拡張有限状態機械をベースに協調プロトコルを記述する。拡張有限状態機械は有限状態機械に変数を許すように拡張したものである。ここでは、この拡張有限状態機械の表現をスクリプトと呼んでいる。

②エージェントごとのプロトコルの容易な特化

エージェントは、基本的には共通のプロトコルに従うものの部分的にエージェント独自の振舞いをすることが必要になる。例えば、契約ネットプロトコルでは、具体的なタスク公告の内容はエージェントごとに異なってこよう。AgenTalk ではエージェント関数とよぶ、スクリプトを特化するためのインタフェースを設けている。エージェント関数はスクリプト内の状態遷移規則から呼び出される一種のコールバック関数で、エージェントが独自に定義できるものである。

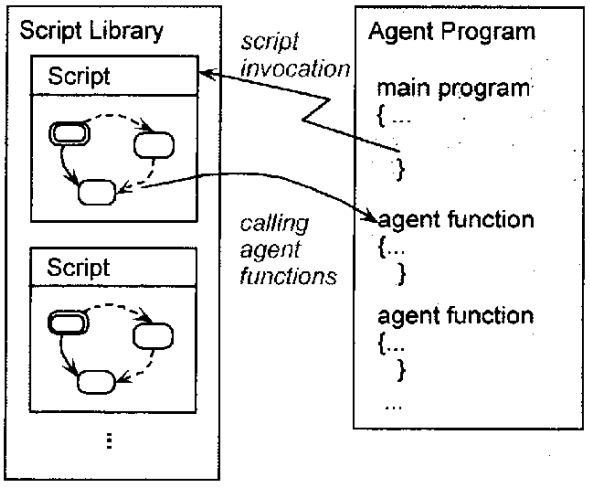
③エージェント内で複数の協調プロトコルの並行実行

例えば、契約ネットプロトコルにおいて一つのエージェントが複数のタスク公告に対し並行して入札する場合を考えよう。この時、一つのエージェントが複数の協調プロトコルの実行に関っていることになる。すなわち、エージェント内で複数の協調プロトコルが並行して実行できるようにする必要がある。AgenTalk では、複数のスクリプトを一つのエージェント中に動作させることにより、複数の協調プロトコルを並行的に実行できるようにしている。

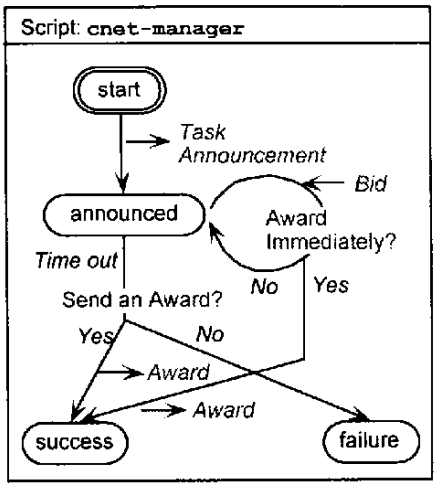
以上に加えて、前述のように AgenTalk では段階的なプロトコル定義を実現するためにオブジェクト指向言語にみられる継承の考え方を導入している。具体的にはスクリプト間で継承を実現し、既存のスクリプトを拡張する形で新たなスクリプトを定義できるようにしている。AgenTalk では状態を明示的に定義できるため、スクリプトの継承において、状態を単位とした継承が容易に実現できる。この点が一般のオブジェクト指向言語におけるメソッドなどの継承とは異なる点になっている。

（２）AgenTalk の機能

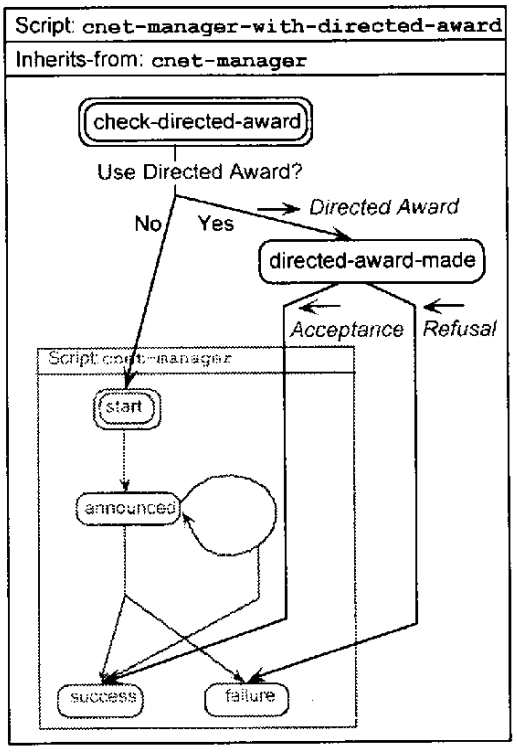
エージェントのプログラムから見ると、スクリプトは一種の手続き群としてみることができる。エージェントがある協調プロトコルにしたがって行動するときは、エージェントのプログラムから対応するスクリプトを起動する（図表 2.3.3-1）。一方、スクリプトからみると、エージェントはスクリプトの状態遷移規則から呼ばれる関数（エージェント関数）を提供することになる。エージェントごとにエージェント関数を用意することにより、大枠ではスクリプトにしたがって動くものの、個々の振舞いはエージェント独自なものとすることができる。もし、対応するエージェント関数が定義されていない場合はスクリプトの定義の一部である（同名の）「スクリプト関数」が呼ばれる。すなわち、スクリプト関数がデフォールトの振舞いを定義していることになる。したがって、既存のスクリプトを継承した新たなスクリプトを定義し、その中で、スクリプト関数だけを定義し直すことによって、スクリプトを段階的に定義することもできる。



図表 2.3.3-1 スクリプトの起動



図表 2.3.3-2 契約ネットプロトコルにおけるマネージャの状態遷移記述例



図表 2.3.3-3 契約ネットプロトコルの拡張（指定落札）

（3）AgenTalk の記述例

ここでは AgenTalk の記述例としてエージェント間でタスクの割当てを行う契約ネットプロトコルの記述をとりあげる。契約ネットプロトコルにおいては基本的には次のようなメッセージのやりとりによりタスクの割当てが行われる。まず、実行すべきタスクを持っているエージェント（マネージャ）がタスク公告メッセージを放送する。これに対して、タスクを請け負いたいエージェントが入札メッセージを出す。マネージャが入札の中から実際にタスクを割り当てるエージェント（コントラクタ）を決定し、落札メッセージを出す。

このような契約ネットプロトコルにしたがうマネージャの状態遷移は図表 2.3.3-2 のように書ける。それをもとにマネージャのスクリプトを定義する。AgenTalk では Lisp ベースのスクリプト記述のシンタックスを定義しているが、ここでは説明のため状態遷移をもとにした直観的な図表を用いている。図表中のタスク公告を放送するところ、落札を出すかの判断するところなどはエージェント関数として定義され、エージェントごとに具体的な内容を定義できる。割り当てられるべきタスクが発生した時にエージェントはこのスクリプトを起動すればよい。

次に契約ネットプロトコルの拡張として、タスクを割り当てるべきエージェントがあらかじめわかっている場合を考えよう。このような場合に対応するものとして指定落札メッセージが定義されている。この場合、指定落札を受け取ったエージェントは受理か拒否かの返答を送ることなる。この拡張を導入した場合の状態遷移は図表 2.3.3-3 のようになる。この図表に示すように基本的な契約ネットプロトコルをもとに、指定落札を含むように拡

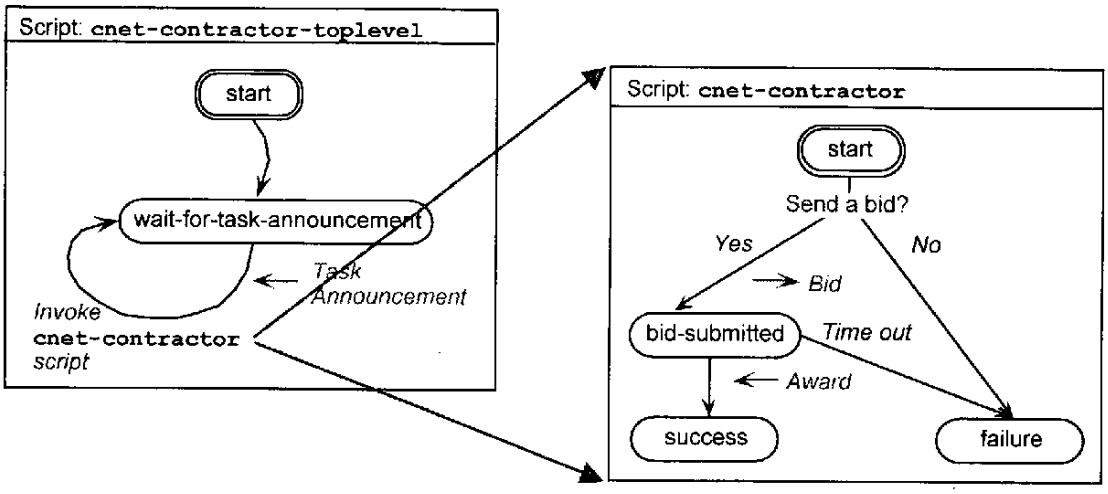


図 2.3.3-4 コントラクタの動作例
（タスク公告を監視するトップレベルのスクリプトから、
コントラクタ本体のスクリプトを起動）

張したプロトコルが定義できる。

一方、タスク公告に対して入札を出すコントラクタの振舞いはどうなるだろうか。例えばエージェントにおいて一種のトップレベルのスクリプトでタスク公告を監視していて、適当なタスク公告が来た場合に、コントラクタ本体のスクリプトを起動するような実現方法が考えられる。単純なコントラクタ本体のスクリプトでは、対象になっているタスクが自分のところで実行できる場合は、入札を返し、落札のメッセージを待つことになる（図表 2.3.3-4）。

さらに、この拡張としてタスク公告におけるタスクが自分のところだけでは実行できない場合は、タスクを複数のサブタスクに分解し、個々のサブタスクを他のエージェントに割り当てることが考えられる。具体的にはコントラクタ本体のスクリプトでは、タスク公告のタスクをサブタスクへ分解する必要があるかを調べ、もし、必要であれば各サブタスクに関して契約ネットプロトコルのマネージャのスクリプトを起動して、サブタスクの割当てを行う。すべてのサブタスクの割当てができた段階（success の状態）で、もとのタスク公告に対して入札することになる。

（4）まとめ

マルチエージェントの協調プロトコルを記述、実行するプラットフォームとして AgenTalk を紹介した。なお、AgenTalk の初期バージョンが CommonLisp を用いて実装されている。そこでは Lisp 処理系で提供されるマルチスレッド機構を用いて一つのスレッドを一つのエージェントに割り当てており、一つの Lisp 処理系の中で複数のエージェントが存在できるようになっている。また、エージェントが別のマシンにあるときは TCP/IP を用いて通信するようになっている。

参考文献

〔1〕 桑原和宏, 石田亨, 大里延康: AgenTalk: マルチエージェントシステムにおける協調プロトコル記述, 電子情報通信学会論文誌 B-I, Vol. 79, No. 5, pp. 346-354 (1996).

2.3.4 ソーシャルウェア構築用プラットフォーム Shine

マルチエージェントの考え方を使ってアプリケーションを構築しようとする時、アプリケーションのドメインに特化した機能を持ったプラットフォームがあると便利である。本節では、アプリケーションのドメインに特化したプラットフォームの例としてＮＴＴコミュニケーション科学基礎研究所の吉田らによって研究が進められているソーシャルウェア構築用プラットフォーム Shine〔3〕を紹介する。ここで、ソーシャルウェアとはネットワーク上のコミュニティ活動を支援するシステムを指している〔1〕。ソーシャルウェアの一例としてネットワーク上のコミュニティ形成の支援を目的とした Community Organizer〔2〕などがある。

Shine では、エージェントによって人々のコミュニケーションを支援するマルチエージェントシステムとしてソーシャルウェアを実現することを目指している。実社会に見られるような集団のダイナミズムをサイバー社会でも実現し、さらにサイバー社会ならではの利便性を提供するためには、人と人との関係に関する制御機構やデータを分散して保持することが必要である。Shine では、マルチエージェントアーキテクチャを用いてこの機能を実現する。すなわち、各ユーザに対して社会的行為に介在するエージェントを用意し、それらエージェント間でデータを交換しながら、仲介、紹介、挨拶といった集団の動的変更に必要な行為を協調して実現する。

ソーシャルウェアのアプリケーションは様々なものが考えられるが、それらに共通して求められる機能として、

a) 交友関係や集団構成の動的な変更、

b) コミュニティにおける各人の特徴・役割・立場の分析、

c) 柔軟で直観的な通信手段の提供、

d) ユーザのコミュニティ感覚とシステムの論理的情報とをリンクするインタフェース

などがあげられる。Shine では様々なアプリケーションの基盤となるこのような機能を提供することでソーシャルウェア構築用プラットフォームを実現しようとしている。

例えば、コミュニティにおける各人の特徴や役割を分析するには、そのためのユーザモデルが必要である。各アプリケーションは、特定のユーザモデルにしたがって、他人の属性や状態に関するデータをデータベースに格納し、またデータに対する操作を行うことになる。ユーザモデルは、どのような観点から人間を捉えるかの違いにより、様々なものが考えられる。例えば、人間の興味を、興味の対象を表わすキーワードとその興味への関心の度合を表わした数値で表現し、多次元の特徴ベクトルとして表現する手法などが考えられる〔2〕。

さらにソーシャルウェアのアプリケーションでは、システムが持つユーザモデルに基づいたデータ集合をユーザが直観的に捉えられるように提示するユーザインタフェースが必要になる。Shine では種々のユーザモデルについて、コミュニティ感覚を助けるような機能をライブラリ化する。例えば、前述のようにその人の興味を表わした特徴ベクトルを用いて人間を表現する場合、バネモデルなどの可視化手法が提供されれば便利である。バネモデルを用いて興味の近さ遠さに応じて人々を２次元平面上に配置することができる。この手法を用いて、Community Organizer では興味の近い人々の集団を可視化し、ネットワーク上でのコミュニティ形成を支援しようとしている〔2〕。

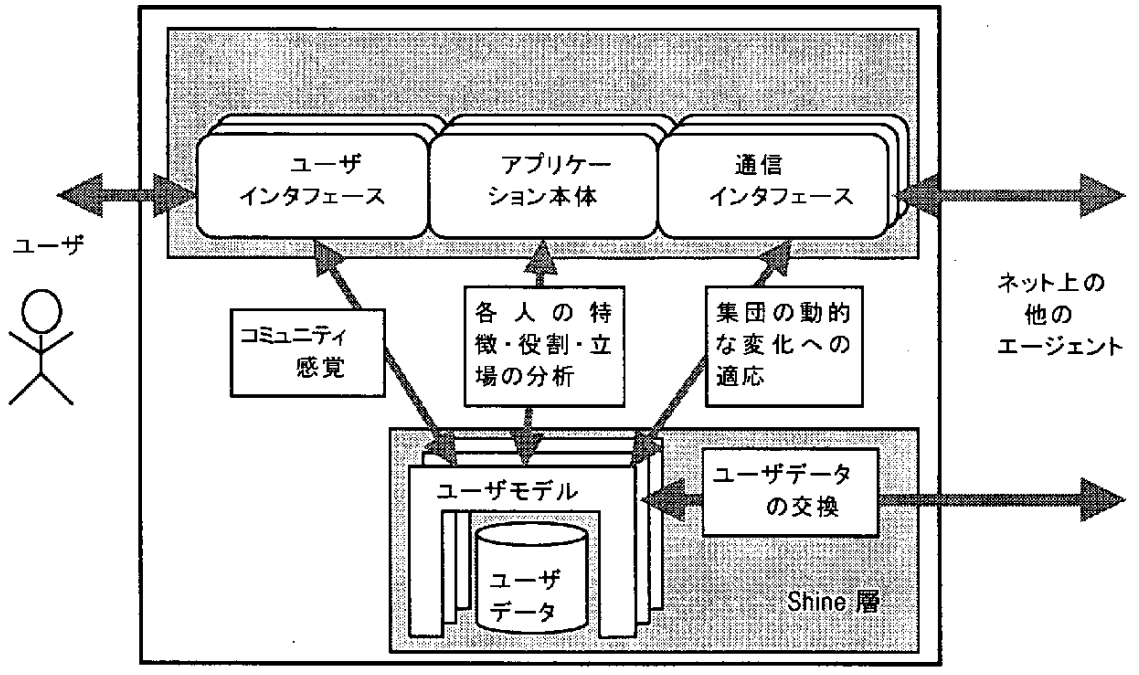


図 2.3.4-1 Shine におけるエージェントの構成

Shine におけるエージェントの構造を抽象的に表すと、図表 2.3.4-1 のようになる。各エージェントはアプリケーションに依存する層（アプリケーション層）と共通の機能を提供する Shine 層に分かれる。Shine 層は、アプリケーションのユーザインタフェースについて、ユーザのコミュニティ感覚とシステムの論理的情報とのリンクを助ける。また、アプリケーションの本体については各人の特徴・役割・立場の分析をすることでコミュニティの特長を活かせるようにしている。また、他とのコミュニケーションのインタフェースについてはそれが柔軟で直観的なものになるようにする。さらに、交友関係や集団構成の動的な変更を協調して行うために Shine 層同士が通信することになる。

なお、Shine は Java のクラスライブラリとして実装がすすめられている。

参考文献

〔1〕 Hattori, F., Ohguro, T., Yokoo, M., Matsubara, S., and Yoshida, S.: Socialware: Multiagent systems for supporting network communities, Comm. ACM, 42 (3), pp. 55 – 61 (1999).

〔2〕 亀井剛次, 吉田仙, 大黒毅, 服部文夫: Community Organizer:ネットワークコミュニティの形成支援, ヒューマンインタフェースシンポジウム'99 論文集, pp. 333 - 336 (1999).

〔3〕 吉田仙, 大黒毅, 亀井剛次, 船越要, 桑原和宏: サイバーコミュニティのアプリケーションのためのプラットフォーム Shine の提案, 第 13 回人工知能学会全国大会論文集, pp. 461- 462 (1999).

2.3.5 パターンに基づく移動エージェントシステムの設計手法

(1) 概要

移動エージェント技術は、実行中のプロセスを別のコンピュータに移動させ、実行を継続させる技術であり、通信効率の向上、回線切断への耐性、環境適応性、設計容易性等の様々な効果が期待されている。しかしながら、実システムに適用する場合、その効果は対象となるシステム環境や利用条件に大きく依存するため、設計時に対象システム環境を熟知し、それに適合した方式を選択することが必要である。

この問題に対し、筆者らは「ネットワーク形状」と「移動エージェントの移動・通信に関する振舞い」を用いたパターン手法を提案した〔1〕。このパターンを用いることで、移動エージェントシステムの開発設計時に、システム運用時における移動エージェントの効果をあらかじめ予測できるようになり、移動エージェントシステムの設計に的確な指針を与えうると期待している。

本節では、パターンの記述法とパターン間の性能比較実験を中心に移動エージェントのデザインパターンの基本的な考え方について説明する。

(2) 従来のパターン化手法との比較

エージェントの設計にオブジェクト指向のデザインパターンの概念を採り入れる研究はE.Kendallらの研究〔2〕をはじめとして数多く提案されており、特にY. Aridorの研究〔3〕では、移動エージェントを対象にしたパターンについてまとめている。これらは基本的にエージェントの設計を「設計の柔軟さ」「設計再利用の容易さ」の視点に基づいてパターン化する手法であり、エージェントシステムの設計に上記視点における指針を与えている。また、田原らの研究〔4〕は、エージェントの上流設計過程において移動の基本パターンを取り入れた設計を行うことを提唱しており、筆者らの研究に最も関係が深い。

しかし前節の通り、移動エージェント技術の運用時における効果（通信量削減、回線切断耐性等）は対象システムの環境に依存して大きく変化する点について触れている研究は少ない。この「システム運用時の効果」は、そもそも移動エージェントを使用する際の主な動機のひとつであり、本提案の趣旨は、この運用時の効果を活かすために必要な設計指針を移動エージェントパターンとして設計者に提供することにある。

ここで提案する新しいパターン化手法では、ネットワークの形状、帯域幅、通信遅延、各ホストの処理速度、リソース量、その他多数の環境状況を考慮し、それらの環境に適したパターンを適切に選択する指針を与える。中でも、移動エージェントシステムで特に重要な「ネットワーク形状」を強く意識したパターンの分類を目指している。

(3) 移動エージェントパターンの記述方法

移動エージェントパターンは、オブジェクト指向におけるデザインパターンの記述方法に習い、名称、構造、適用可能性など12の記述項目を流用するが、「構造」には新たに定義する「トポロジー図」「通信シーケンス図」を使用し、「適用可能性」よりシステム運用時の効果を重視する。また、「評価」という項目を追加し、通信性能等に関する評価式やその考察などを記述する。

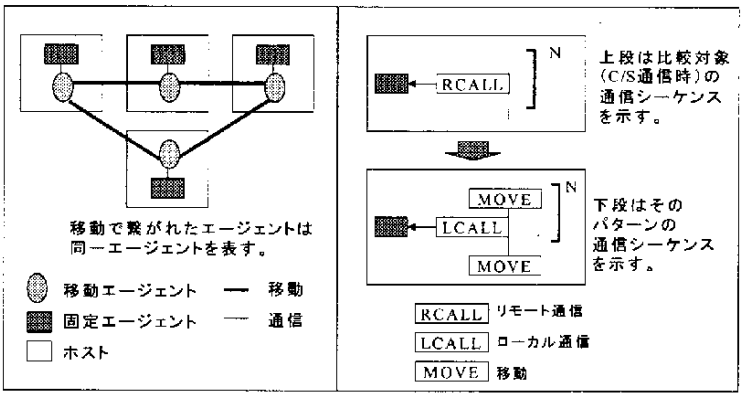


図 2.3.5-1　移動エージェントパターン記述形式

図表 2.3.5-1 は、複数箇所のリソースに巡回アクセスする「Round」パターンを例としてトポロジー図、通信シーケンス図の記述法を示した図である。

トポロジー図（図表 2.3.5-1 左）には、パターンを適用するために必要な環境（ネットワーク形状）が、ホスト、ネットワーク接続によって示されており、その上に、移動・通信を行なうエージェント、移動しない場所固定のリソース（またはそれをラッピングするエージェント)、そしてその間の通信や移動の行程が示されている。図表中の太線（移動線）で結ばれた 4 つの丸は全て同じ移動エージェントを表しており、1 つのエージェントがこの軌跡を辿って各ホストに移動することを意味している。

通信シーケンス図（図表 2.3.5-1 右）には、ローカル通信、リモート通信、移動という 3 つのアクションを用いて、移動中のエージェントの処理内容を記述している。従来のクライアントサーバ通信から移動エージェントを用いたシステムに変更することは、通信時におけるローカル、リモートの関係が大きく変化することになり、特に実行性能に強く影響を及ぼす。通信シーケンス図では、従来のクライアントサーバ通信使用時の通信シーケンス図も記述することで、そのパラダイムの差異を明瞭にする。

このパターン記述によって、移動エージェントによる分散システムのパラダイム変化を簡単に表現することできる。設計者は各々のパターンが持つ長所、短所を事前に把握し、また「評価」の項から、本手法の重視する「運用時の利点」（通信性能等）の定量的評価を得ることができる。

(4) 移動エージェントの様々なパターン

現在、筆者らによって記述されているパターンは 13 種類存在し、各パターンは「RemoteAccess グループ」「Notifier グループ」「Meeting グループ」の 3 つのグループに分類される。「RemoteAccess グループ」は、ある 1 箇所から複数のホストにアクセスすることを目的とするパターン、「Notifier グループ」は、別のホストに情報を伝達することを目的とするパターン、「Meeting グループ」は、複数のホスト間で情報を共有することを目的とするパターンである。以下に、パターン記述の例として「RemoteAccess グループ」の 7 種類のパターンについて、トポロジー図、及び簡単な解説を示す（図表 2.3.5-2)。なお、これらはオブジェクト指向デザインパターンと同様、パターンの網羅性を考えて作成

されてはいいないことには留意する必要がある。

RemoteCall パターン、クライアントの端末から複数のサーバに対して、従来通りの C/S 通信を行なうパターンであり、他のパターンに対する比較対象基準となる。各ホストへのアクセス回数が少ない場合や、通信遅延が小さい場合に効果を発揮する。

Round パターンはエージェントが巡回移動する形でアクセスを行なう。各サーバでローカル通信を行なうため、各サーバへのアクセス回数が多い場合に、大幅な通信量削減効果が得られる。また、回線切断にも強い耐性を持つ。一方、移動中のサーバでエージェントが喪失すると、そこまでの全ての結果データが失われる危険性も持つ。

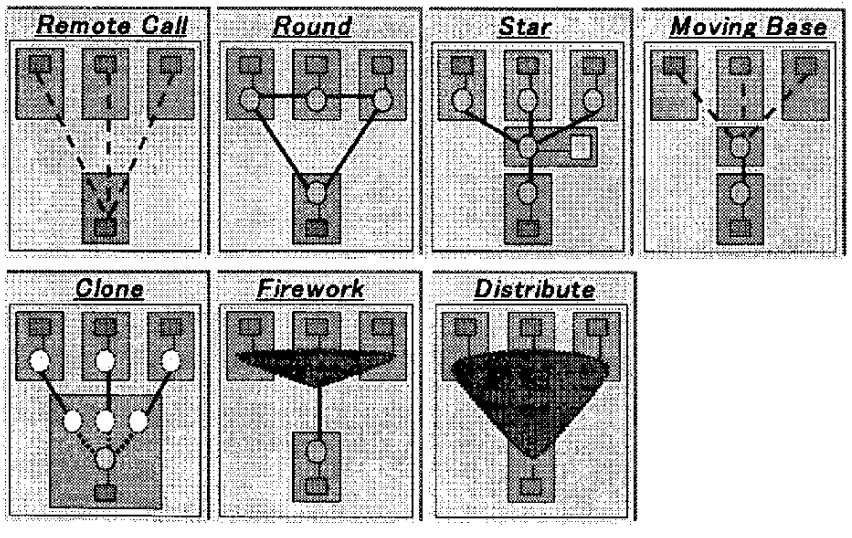
Star パターンは移動の中継地点を設け、各サーバにアクセスするごとに中継地点に戻ってデータを保存する。エージェントの移動回数は増加するが、Round パターンのようなデータ消失の危険性を回避できる。また、サーバ数が増加しても、エージェントの持ち運ぶデータ量は累積的に増加しない。

MovingBase パターンはエージェントが中継地点まで移動し、そこから各サーバに C/S 通信を行なう。中継地点までの通信が遅く、中継地点から各サーバへの通信が速いような非対称なネットワークで効果を発揮する。

Clone パターンはサーバと同数のクローンエージェントを生成し、各々のサーバに移動させることで並列アクセスを実現する。巡回アクセス型のパターンと比較して圧倒的に速いレスポンスタイムが期待できるが、ある箇所のアクセス結果が他の箇所のアクセス方法に影響を与える場合ような、データに依存性があるアクセス場面には適用できない。

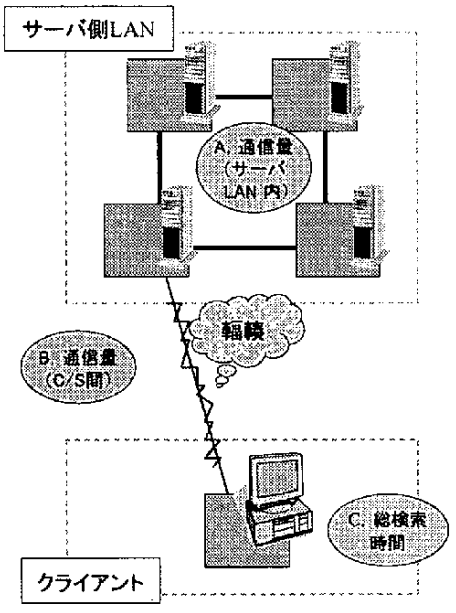
Firework パターンは、サーバ側に移動したエージェントが、複数のサーバ間に分散して並列アクセスを行なう。筆者らの開発した Mobidget のような分散機能を持つエージェントシステムにて利用できるパターンである。Clone パターンと類似の効果を持つが、新規エージェントの生成コストがなくなり、また、分散エージェント間で相互に中間結果を参照したいケースに有効である。ただし、障害等により分散中のエージェントの一部を失うような事態に弱い。

Distribution パターンはクライアント、サーバの全てのホストに分散したエージェント



図表 2.3.5-2　各種パターンのトポロジー

を生成し、並列アクセスを行なう。サーバにアクセス中、クライアントとも適度な通信が必要な場合に効果を発揮するが、クライアントとサーバ間の通信回線の信頼性、速度によってはFireworkパターンのほうが望ましい。



図表 2.3.5-3 実験環境

(5) パターン間の性能比較実験

システム設計の際には、実際の適用環境に合わせて移動エージェントのパターンを決定する。パターンカタログの文章説明に記された「効果」「適用ケース」などの項目を参照することによって、従来は設計者各個人によって行なわれていた暗黙の利得計算に、客観的な評価基準を与えることができる。

パターンカタログには、実行環境のパラメータに基づく性能評価式が与えられているが、ここでは、RemoteCall、Round、MovingBase、Fireworkの4種類のパターンを例に、実際に図表2.3.5-3のような環境設定で通信効率測定の実験を行なった結果を示し、パターン間の性能比較を試みる。あるクライアント（図表下部）から複数のサーバ(図表上部）にあるデータベースに対して、各々複数回のアクセスを行ない、通信量、検索時間を測定する。実験では、以下のような環境条件を設定し、Mobidget上にエージェントを実装して実験した。

- ●C/S間の通信は低速（115.2Kb/s）で、サーバ内のLANの通信は高速（10Mb/s）である。
- ●C/S間通信部分は多くのユーザの検索によって輻輳するため、サーバ側のLAN内の通信よりC/S間の通信削減を優先する。

このとき、各パターン適用時において、サーバ数を1から50まで増やした際の、A:サーバLAN内通信量、B:クライアントーサーバ間通信量、C:総検索時間の3つを測定、比較した結果が図表2.3.5-4～7である。

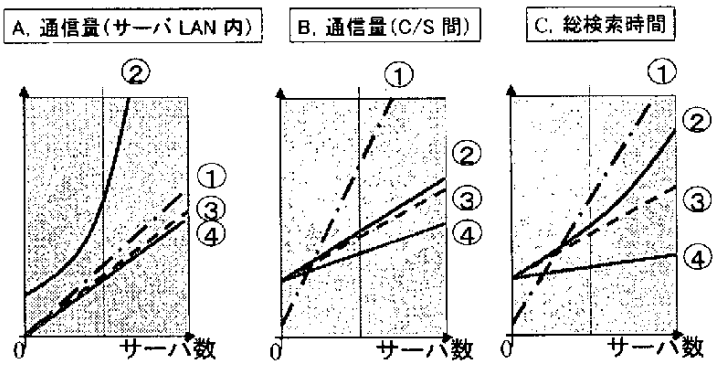
RemoteCallパターン①は、3つのグラフがいずれも、サーバ数に比例しており、これを1つの基準線とする。Roundパターン②の場合、B、Cに関してはサーバ数の増加に対する

上昇がRemoteCallに比べて低く抑えられているが、Aは獲得したデータ量を累積的に抱え込むため、サーバ数に対して2乗に比例するグラフを描いてしまう。一方、MovingBaseパターン③の場合、B、Cについては Roundパターンと同程度に抑えられ、なおかつAについてもRemoteCall①以下の通信量を達成している。Fireworkパターン(4)では、MovingBaseよりさらに、Cに関して分散並列アクセスの効果による高いレスポンスを達成している。

このように、ある環境を設定すれば、どのパターンがその環境下でより適切であるかについて、明確な指針をもって判断することが可能である。具体的な応用例を挙げると、インターネット上のショッピングモールに対して価格検索を行なうようなケースでは、この実験と同様の結果を期待することができる。単純にRemoteCallパターン(C/S通信)からMovingBaseパターンに変更するだけでも、ダイヤルアップ接続時間を減少させることができ、Fireworkパターンを適用することによって更に並列効果を生み出すことが期待できる。

(6) まとめ

以上のように、移動エージェントシステムにおけるデザインパターンに関し、従来考慮されていた設計容易化、再利用性向上等の利点に加え、運用時における通信効率、耐切断性の向上等の利点を活かすために必要な、新しい移動エージェントのデザインパターンと



① Remote Call パターン ③ Moving Base パターン
② Round パターン ④ Firework パターン

図表 2.3.5-4 パターン毎の通信性能比較

表 2.3.5-5 A. 通信量(サーバLAN内 : Kbyte)

サーバ数	1	2	5	10	20	50
RemoteCall	0.05	80.6	241.9	560.7	1195	2943.9
Round	0.1	202.6	460.4	1695.6	5197.3	26548.3
MovingBase	0.05	77.4	235.9	551.4	1191	2945.4
Firework	0.05	69.7	200.7	434.4	905.2	2188.5

図表 2.3.5-6 B. 通信量(C/S間 : Kbyte)

サーバ数	1	2	5	10	20	50
RemoteCall	63.9	130.4	328.9	656.8	1314.9	3278.8
Round	159.9	177.8	229.8	316.2	490.9	1014.3
MovingBase	159.3	172.3	224.8	312.8	491.8	1014.8
Firework	153.4	156.6	182.9	199.8	290.5	592.7

図表 2.3.5-7 C. 総検索時間(秒)

サーバ数	1	2	5	10	20	50
RemoteCall	1.21	2.36	6.02	12.86	26.13	64.53
Round	7.61	7.51	9.47	11.72	20.51	58.62
MovingBase	7.88	6.86	8.9	11.49	19.73	33.36
Firework	6.63	6.62	8.71	9.23	11.92	21.71

その記述手法の有効性を示した。ネットワークのトポロジーと、移動エージェントの移動・通信に関する振舞いを中心としたパターン化によって、分散システム設計時における移動エージェントの有効的活用に、より明確な方向性を与えることが可能であることが、実験によっても示されている。

今後もパターンの洗練と追加に加え、これらを設計レベルから実装レベルに移行するためのライブラリ/フレームワーク化技術についての研究を進める予定でいる。

参考文献

〔1〕 小松千尋,藤田悟,山之内徹,パターンに基づく移動エージェントシステムの設計手法,　電子情報通信学会信学技報,AI99-49,1999.

〔2〕 Kendall,E. A.,Pathak,C. V.,Krishna,P. V. M.,and Suresh,C. B.,The Layered Agent Pattern Language,in Proceedings of PLoP'97,1997.

〔3〕 Aridor,Y. and Lange,D.B.,Agent Design Patterns: Elements of Agent Application Design,in Proceedings of the Second International Conference on Autonomous Agents, pp.108-115,1998.

〔4〕 Tahara,Y.,Ohsuga,A.,and Honiden,S.,Agent System Development Method Based on Agent Patterns,in Proceedings of the 21st International Conference on Software Engineering,acm PRESS,pp.356-367,1999.

2.3.6 移動エージェントシステムによるアプリケーションの構築

日本情報処理開発協会(JIPDEC) において、「次世代電子図書館システム 研究開発事業」(http://www.dlib.jipdec.or.jp/)が推進されている。そこでは、不特定多数の使用者が、任意のタイミングで多様なアプリケーションを起動させることを想定され、この要求を的確にかつ効率的に稼動させるための仕組みが必要となる。

このような目的を実現するための技術として、移動エージェント技術が有効であると考え、本項ではその研究開発について述べる。

（1）概要

本件研究開発においては、まず移動エージェントとは何か以下のように定義した。

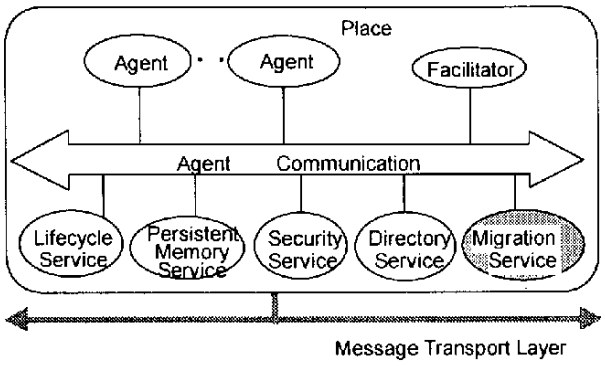
●プログラムが処理ロジックとユーザデータを保持し、ネットワークを通して他の計算機上に移動し、処理を継続可能なこと

つまり、移動対象はプログラムコードとデータとし、プログラムのスタック情報は移動させない、という「Weak Migration」の形態の実現を目標とした。これは、スタック情報を移動対象外とすることにより、比較的簡便な移動エージェント基盤が実現でき、機能性も図書館等の公共性の高いサーバシステムに供するに十分なものが実現できるからである。ただし、このことにより、本移動エージェントは、エージェントの状態（ライフサイクルの状態やどのサーバ上で稼動しているか）に応じてユーザがプログラムロジックを記述する必要がある。使用するプログラミング言語は、図書館サーバのハードウェアの多様性を考慮して Java を想定した。

（２）参照モデル

移動エージェント機能は、本研究開発事業におけるサブアーキテクチャの１つであるエージェント基盤のリファレンスモデルに従う。移動エージェント機能は、図表 2.3.6-1 に示すように、エージェント基盤上のサービスの一つである MigrationService として提供される。

Message Transport Layer として、OMG の CORBA(Common Object Request Broker) のアーキテクチャを前提としているが、WAN などファイアウォールを経由する通信が必要となる場合には、代替のプロトコルが選択可能でなければならない。本件研究開発では、SMTP をこの代替プロトコルとして取り上げている。



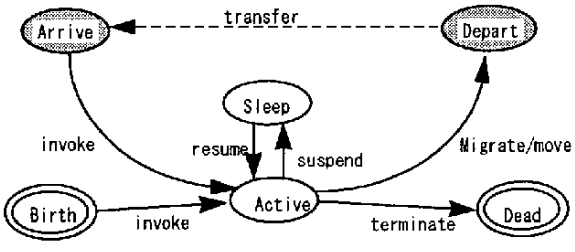
図表 2.3.6-1　エージェント基盤参照モデル

（３）エージェント基盤との関係

移動エージェント機能は、エージェント基盤上の以下のサービスと連携をとり、エージェントの移動を実現する。

①ライフサイクルサービス

ライフサイクルサービスは、スタティックなエージェントの状態に加えて、図表 2.3.6-2 に示されるように、Arrive、Depart という状態を管理する。



図表 2.3.6-2　状態遷移

マイグレーションサービスは、ライフサイクルサービスと連携し、以下のことを保証する。

a）移動前後でエージェントに割り当てられるエージェント ID は不変である。

b）エージェントは、移動前に Depart に状態遷移した後終了する。

c）エージェントは、移動後に Arrive の状態にて生成される。

②永続メモリサービス

エージェントが使用していた永続メモリ上の情報は、エージェントプログラムと

ともに移動され、移動後に再び永続メモリ上に展開され使用可能となる。

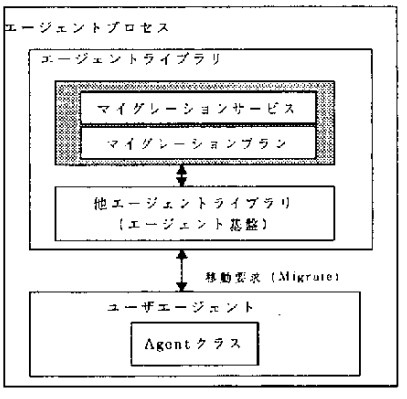
③ディレクトリサービス

エージェントは、サービスを指定して移動することができる。その時、マイグレーションサービスはディレクトリサービスと連携し、サービスからプレースへのマッピングを行い、エージェントを、サービスを提供しているプレースへ移動させる。

（4）移動エージェントの機能構成

移動エージェント機能は、移動エージェントライブラリと移動エージェント基盤オブジェクトから構成される。

移動エージェントは、エージェントプログラマが作成したエージェントプログラム部と、エージェントプログラマに対して各サービスのインタフェースを提供するエージェントライブラリ部からなる。マイグレーションサービスライブラリもエージェントライブラリの一部として提供され、マイグレーションサービスおよびマイグレーションプランからなる（図表 2.3.6-3）。



図表 2.3.6-3 エージェントプロセスの構成

マイグレーションサービスは、エージェントの移動要求を受けつけ、移動要求を解釈し、移動エージェント基盤オブジェクトへ移動要求を伝達する。マイグレーションプランは、エージェントの移動計画を保持する。マイグレーションプランを使用することで、エージェントは移動の定型化及び管理を行なうことができる。

エージェント基盤オブジェクトは、通常 CORBA オブジェクトとして実装されるオブジェクト群であり、その中の一つとして移動エージェント基盤オブジェクト（マイグレーションサービスオブジェクト）が提供される。エージェントの移動は、複数のマイグレーションサービスオブジェクト間のメッセージ通信として実現される。

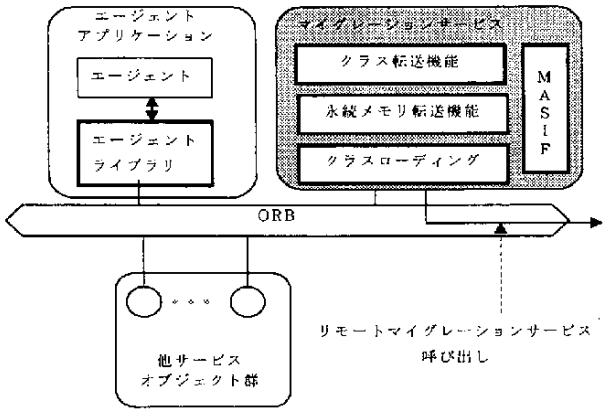
マイグレーションサービスオブジェクト（図表 2.3.6-4）は、エージェント移動のために

①エージェントのクラス転送

②永続メモリの転送

③エージェントクラスのローディング

を行う。

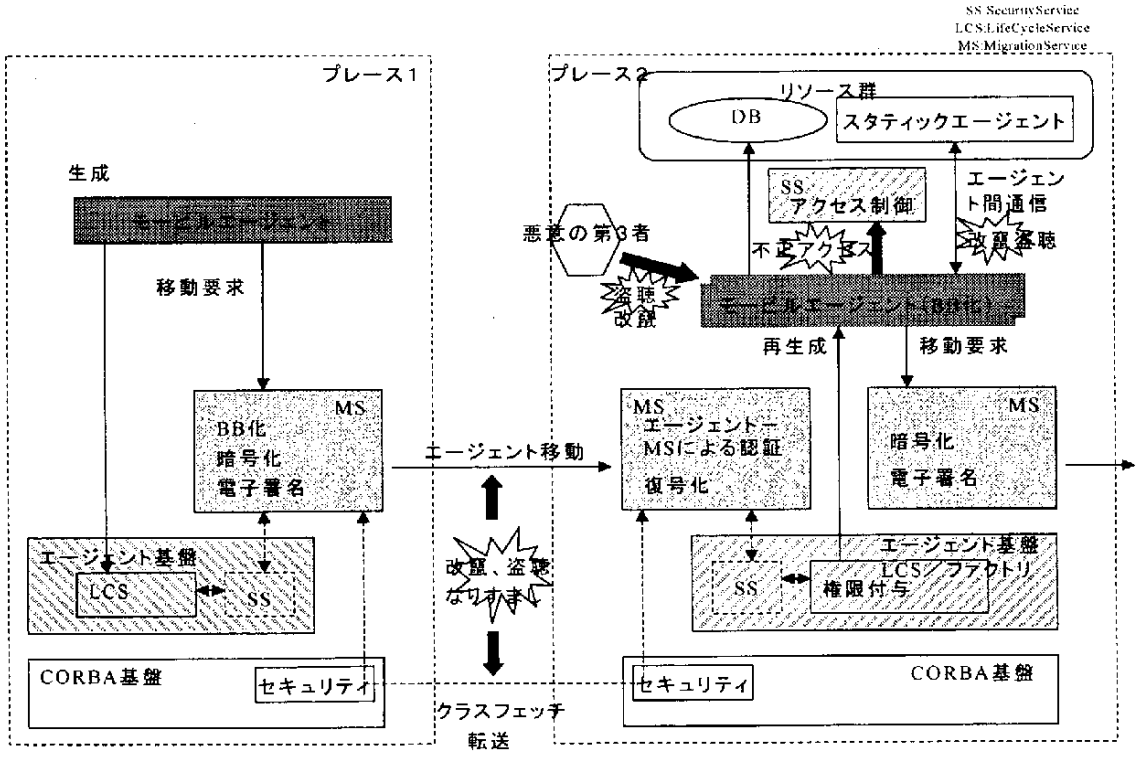


図表 2.3.6-4 マイグレーションサービスオブジェクト

（5）移動エージェントにおけるセキュリティ

エージェント基盤は、基本的に CORBA に閉じた世界であり、CORBA セキュリティサービスに依存するという立場をとる。しかし「移動」を考えた場合、前述のように WAN を介した通信となり、トランスポート層セキュリティである CORBA セキュリティのみに依存することはできない。また、エージェント独自のセキュリティに対する考え方も必要となる。そこで、本件研究開発では移動エージェントにおけるセキュリティについても考察した。

エージェントとりわけ移動エージェントに係るセキュリティ上の脅威からいかにエージェントおよびそれにまつわるリソースを保護できるかは大変重要なテーマである。移動エージェントの生成から移動、消滅にいたるライフサイクル全般にわたるセキュリティ概念は図表 2.3.6-5 のようになる。



図表 2.3.6-5 移動エージェントにおけるセキュリティ概念図

エージェントの生成から移動までの、セキュリティ関連の前提と全体のフレームを以下のように考えた。

1）オリジナルホストにおけるエージェント生成から移動まで

エージェントの生成およびローカルなリソースアクセスは、既存のセキュリティ技術にて保護されるものとする。エージェント移動時に、盗聴・改竄を防止するための電子署名・暗号化等を施す。また、移動先ホストからの攻撃から保護するためにブラックボックス化（後述）を行なう。

2）移動中

エージェントがネットワークを移動している間は、通常のメッセージにおけるセキュリティ技術により、盗聴・改竄・なりすまし等の脅威から保護される。

3）移動先ホストにおけるエージェント再生成から移動まで

移動してきたエージェントと、それを受け入れるエージェントサービスとの間で相互認証・復号化された後、再生成されたエージェントはローカルリソース（静的エージェントやデータベース等）をアクセスする。そのためのアクセス権限制御が行われる。また、エージェント自体がホストからの悪意ある攻撃を受けた場合には、オリジナルホストにて施されているブラックボックス化で対処する。

上記の通り、移動エージェントにおけるセキュリティを考慮する場合、システム外の第三者からのエージェントに対する攻撃以外に

1）悪意あるエージェントからのホストの保護

2）悪意あるホストからのエージェントの保護

の二つの観点が必要となる。

前者は、コンピュータウイルスの侵入防止と似た考え方であり、既に種々のセキュリティ対策が講じられている。それに対して後者は、新しい概念と言える。脅威には、システム上の第三者がエージェントプログラムのコードやデータを盗聴・改竄を行い、システムコールに対して不正な結果を返送する、エージェントコード自体の実行を拒否する、等が考えられる。これらの脅威に対して現在考えられているのが、エージェントのブラックボックス化手法である。（図表 2.3.6-6）

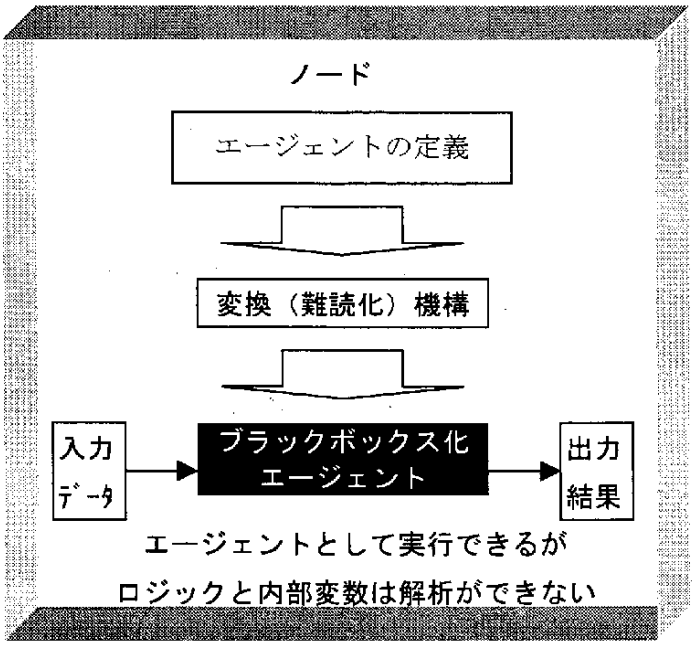
ブラックボックス化とは、エージェントを実行する計算機上で、エージェント自身が、他のモジュールからの攻撃に対処するための技術である。主にエージェント生成ホストにて、エージェントのロジックとそこで利用される変数を、難読化機能を使用することにより解析できないように変換する。変換後のエージェントは実行可能であり、機能そのものはブラックボックス化前のエージェントと同等である。ブラックボックス化する手法は、数種類発表されているが、データに対して施す手法と処理ロジックに対して施す手法の２種類がある。ブラックボックス化をより強固にするためには、両方の手法を併用することが望ましい。

（６）成果物

本研究開発の成果物として、以下のものがあげられる。

①移動エージェント機能実装規約書

本研究開発の内容を実装規約という形で記述したものである。内容は、移動エージ



図表 2.3.6-6　ブラックボックス化のしくみ

ェント概論とインタフェース規定からなる。インタフェースは、OMG で規定している IDL インタフェースに即して述べられている。

②移動エージェント機能プロトタイプシステム

上記インタフェースを実装し、簡単な図書検索アプリケーションを作成した。アプリケーションは各図書館を巡回し、ユーザに適切な図書情報を提供するものである。

（7）まとめ

エージェント基盤をベースとした移動エージェントの機能定義と実装規約の制定を行ったが、移動エージェントの特性から、稼動環境の拡大が必要とされるであろう。例えば、モバイル端末からのエージェントの移動が実現されると、エージェントの活用範囲が広がる。現在のエージェント基盤を軽量化し、モバイル端末のような小型の端末でもエージェントが利用できるような仕組みの検討が必要である。

標準化に関して言うと、本移動エージェントの機能は、OMG の MASIF と似た概念であり、MASIF への対応も容易に図られると思われる。また、今後は FIPA の動向も注視しなければならない。近年 FIPA においてはエージェントの移動に関しては積極的な活動が見られないが、キラーアプリケーションの開発等なんらかの契機により標準化が進展することを期待する。

参考文献

〔1〕 OMG:MASIF (MobileAgentSystem Interoperability Facilities)
http://www.mg.rg/docs/orbos/98-03-09.pdf

〔2〕 FIPA:FIPA98 Agent Mobility
http://www.IPA.rg/spec/ fIPA8a27.oc

〔3〕 Fritz Hohl: Time Limited Blackbox Security: Protecting Mobile Agents from Malicious Hosts, in: G.igna (Ed.: Mobilc Agents and Security, Springer

2．4　標準化

2．4．1　FIPA(Foundation for Intelligent Physical Agents)

（1）概要

FIPAは、スイスのジュネーブで登記された非営利団体である。組織の詳細についてはウェブを御参照頂きたい。

http://www.fIPA.org/　（オリジナル）

http://fIPA.comtec.co.jp/fIPA/（日本のミラー）

http://fIPA.umbc.edu/mirror/（米国のミラー）

設立は1996年、現在14ヶ国から56メンバーが参加している。（以下「現在」「現時点」等は、2000年2月末を指すものとする。）年4回、欧州、北米、アジアの持ち回りでこれまでに16回の会合を行っている。年会費は従来3,500スイスフランであったが、2000年1月に、5,000スイスフランに値上げされた。会費の使途は、これまでは会合のホストに対する補助が主であったが、今後は広報宣伝や具体的なアプリケーションの開発と運用の促進のために使われることも多くなると予想される。学会のために、会費を免除される「アカデミー」の制度がある。仕様の実装など、FIPAの活動へ寄与を行う学生に対する補助もある。内部ではFIPAボードが組織を監督する。1999年10月の第15回会議で、それまでプレジデントを勤めていた伊CSELTのLeonardo Chiariglione氏の任期が満了したことから、新プレジデントとして、独シーメンスのDonald Steiner氏が選ばれた。それ以外のボードメンバーも大幅な入れ替わりがあった。日本からは、NTTの小暮氏が新たにボードメンバーに選ばれた。仕様は技術委員会（Technical Committee、TC）が各部分を分担している。1999年度の技術委員会の構成は以下の通り。

①TC A－抽象アーキテクチャー。エージェントとは何かという（抽象的な）定義から、FIPA 97/98がその一つのインスタンスであるようなアーキテクチャーを定める。これによって、より高度な見地から互換性や相互運用性が確保されるようになることが期待されている。

②TC B－エージェント管理。名前管理、具体的な転送構文を含む通信管理、構成管理を担当する。

③TC C－エージェント通信。コミュニカティヴアクトの意味論（構文はTC B）、相互作用プロトコル、コンテント言語について責任を持つ。

④TC D－相互運用性。異なるベンダーによる複数の実装のあいだの相互運用性、互換性、仕様の検証と保守、アプリケーション開発の促進、実証実験、開発者ガイドを受け持つ（2000年1月のロンドン会議でスペシャルインタレストグループに変更）。

⑤TC E－遊牧的アプリケーションサポート。98年の移動エージェント管理をふまえ、移動デバイス上のエージェント管理をアプリケーションの立場から考える。

これらのTCによって議論されてきた、FIPA 99と呼ばれてきた仕様はリリースされていない。仕様の品質を高めるために、組織とポリシーに大きな変化があり、仕様書の作成プロセスも影響を受けたからである。まず、1999年10月の第15回会議では、従来の年次のサイクルをやめ、各TCが非同期に仕様書を発行することとした。また、2000年1月の第16回会議では、アーキテクチャーボードが新設された。これは、仕様のリリースをはじめとする、FIPAの技術面に最終の責任と権限を持つ機関である。FIPAボードとしての任期が

うまく満了したため、須栗がアーキテクチャーボードのチェアに選ばれた。具体的な活動内容については後述する。本文では主にFIPA 98までの公開されている文書を基に活動の紹介を行う（図表2.4.1-1）。

図表2.4.1-1　FIPAの仕様書一覧

Spec	FIPA 97	FIPA 98
1	エージェント管理（J）	エージェント管理
2	エージェント通信言語（J）	
3	ソフトウェア統合　（J）	
4	個人旅行支援	
5	個人秘書	
6	視聴覚娯楽と放送	
7	ネットワーク管理	
8		人間との相互作用（J）
9		（欠番）
10		安全性管理
11		移動エージェント
12		オントロジーサービス（J）
13		FIPA 97 開発者ガイド

下線：アプリケーション仕様

（J）：日本語訳あり http://ias.comtec.co.jp/fIPAtrans/

（２）仕様の概略

FIPAの仕様で大切なことは、これはあくまでも、相互運用性を保証するための、エージェントの外部のインターフェースを規定するものだということである。プログラミング言語を含む内部の実装については仕様の範囲外である。このことにより互換性と機能性のバランスを取る。ただし安全性や移動性など、内部の構成に踏み込まざるを得ない箇所では、隔靴痛痒の感は否めない。

FIPA 97仕様は1997年10月に策定され、文書の最終編集を経て1997年12月に公開された。FIPA 98仕様は1998年10月に策定され、1998年12月に公開された。いずれの場合も提案を公募し、技術委員会による議論を行い、何度かのドラフトの公開を経て最終版が規定された。FIPA 97仕様にはバージョン１とバージョン２がある。バージョン２はバージョン１のバグフィクス版であり、本質的な変更はない。仕様書の番号は、以前はパート何番と呼んでいたが、スペック何番という呼称に変更された。図表2.4.1-1に仕様書の一覧を示す。

大別して、技術仕様とアプリケーション仕様がある。技術仕様は規定であり、アプリケーション仕様は参考であると言われる。規定というのは、仕様をすべて実装しなければならないという意味ではない。仕様書に記述されている機能を実現するならその通りに実装しなければならないが、機能の取捨選択は実装者の自由である。また規定の仕様と矛盾をおこさない限り、拡張には寛容である。異なる実装によるそれぞれの拡張部分が矛盾を起こす場合、それを解決するのは各実装者の責任である。

技術仕様の範囲となる標準化の項目を定める前に、FIPAではアプリケーションから考えた。どのようなアプリケーションが、エージェント技術、とりわけ標準化されたエージェ

ント技術の恩恵を被るかを議論した。その結果選ばれたのが、スペック４から７までのアプリケーションである。次に、そのようなアプリケーションを構築し、相互運用性を確保するためにはどのような技術が必要であるかを抽出したものが、スペック１から３までの技術仕様である。さらに、技術仕様をアプリケーションにフィードバックした。すなわち実際に技術仕様を使うアプリケーションのシナリオを作り、実証実験を行うこととした。

（３）実装と接続実験および実証実験

各社で技術仕様とアプリケーション仕様の実装が行われている。実装をオープンソースとして公開しているメンバーも多い〔1〕〔2〕〔3〕〔4〕。それらの実証実験や接続実験も、ようやく成果が出始めている。まず、1999年1月には、相互接続性試験が行われた。具体的には、Broadcom、コムテック、CSELT、Siemensの各社が実装を持ち寄り、FIPA 97の技術仕様の範囲での相互接続性を確認した〔5〕。4月には、ヨーロッパのFACTSプロジェクトの中間成果が出た〔6〕。2000年3月には最終報告が出る予定である。FACTSでカバーされていないアプリケーションシナリオについては、FIPA自体が費用を負担して、開発を公募した〔7〕。ソースコードが公開される予定である。

（４）FIPA 97 および FIPA 98 仕様

以下にそれぞれの仕様の概略を説明する。

①エージェント管理

エージェント管理の仕様はFIPA 97と98の両方にある。97版は基本的な仕様を規定し、98版は安全性管理や移動エージェントのサポートのために使われる拡張機能を定義している。主な仕様として、エージェントのホワイトページとイエローページのディレクトリーサービスを定義している。ホワイトページとはエージェントの名前から機能へのインデックスであり、イエローページとは機能から名前へのインデックスである。ホワイトページを管理するのがAgent Management System（AMS）であり、イエローページを管理するのがDirectory Facilitator（DF）である。AMSもDFも、それら自体がFIPA準拠のエージェントである（余談であるが、98年度の段階では、FIPAでは、エージェントとは何かという定義を積極的に行ってはいない。FIPA準拠のエージェントと言う場合、それは、FIPAの仕様に従う何らかの実体という程度の意味である）。

AMSによって管理されるエージェントの集合がエージェント基盤を構成する。エージェント基盤とは、エージェントの物理的な構成である。例えば一台の計算機であるが、これに限られるわけではない。基盤の内部におけるエージェント間の通信は、標準仕様の対象外であるので、IPC、RMI、ソケット直など自由に行なえる。基盤と基盤の間を結ぶのがエージェント通信路 Agent Communication Channel（ACC）というエージェントである。ACC同士の通信にはIIOPを用いることによって汎用性を高めている。ただしORBが必須であるわけではない。何らかの手段によってIORを交換できればよいだけであるが、そこが一意に定められていないため、相互運用性の障害になることが判明している。

DFによって管理されるエージェントの集合がエージェント領域と呼ばれる。領域はアプリケーションを実行するエージェントの論理的な構成であり、複数の基盤にまたがることができる。物理的に一つの基盤に帰属するエージェントが論理的に複数の領域に参加する

こともできる。

このような管理エージェント（AMS、DF、ACC）に対する登録、検索、転送といった操作、管理されるエージェントのライフサイクルが定められている。

②エージェント通信言語

エージェント通信言語（Agent Communication Language ACL）の仕様は、20個の通信行為、7個の相互作用プロトコル、例題用のコンテント言語SLとそのプロファイル0、1、2を定めている。通信行為はKQMLではパフォーマティヴと呼ばれている。FIPA ACLも同様の発話行為理論に基づき、話者の意図の伝達、聴者に対する行為の要求、命題やオブジェクトの質問などの通信行為を定義している。KQMLと似て非なるものをどうしてつくったのかとよく聞かれる。最大の理由は、Yannis Labrouによる意味論的アプローチの以前には、KQMLには標準の形式意味論が存在しなかったことである。このため複数の実装の間の会話の互換性を維持することが困難であった。そこで厳密な形式意味論を持つ言語を開発した。またエージェント管理の関係はスペック1に任せた。ただし構文はなるべくKQMLに似せるようにしたのである。Tim FininとYannis LabrouがFIPAに参加したのはACL策定の最終段階になってからであった。従って初版で彼らの意見を全面的に取り入れることはできなかったが、その後も活発に議論を盛り上げているので、今後は両者のギャップをどう埋めるかが重要なテーマとなろう。

コンテント言語SLも悩ましい立場である。もともとSLは、FIPA ACLの基礎となったフランステレコムのARCOLという言語の形式意味論を記述する言語Semantic Languageであった。それに別の構文を与え、コンテント言語として使えるようにしたのである。通信行為によって、別の通信行為をコンテントとして対話者に要求するような場合、通信行為とコンテント言語にこのような連続性があることは非常に便利である。ただしエージェント通信言語の仕様書という立場からは、SLが唯一のコンテント言語であるわけではない。SLはあくまでも例である。ただし、SLの表現力を弱めたプロファイル0、1、2が、エージェント管理や非エージェントソフトウェア統合の仕様書で規定として採用されている。SL自体は強力な様相一階言語であるが、それだけ実装が重くなる。表現力を弱めることにより、すべてのエージェントで必要とされるエージェント管理のコンテント言語を軽くする目的でサブセットを規定したのがプロファイルである。

③エージェントとソフトウェアの統合

すべてのソフトウェアサービスがエージェント化されるわけではないので、エージェントにとって、エージェントでないソフトウェア（以下単にソフトウェアと呼ぶ）とのインターフェースを確保することは肝要である。本仕様では、そのためのラッパーとエージェント資源ブローカーを規定している。ラッパーとは、ソフトウェアを包み、一方で当該ソフトウェアのAPIによってソフトウェアと会話し、他方でACLによって他のエージェントと会話するエージェントである。ラッパーの初期化、ソフトウェアの起動と停止、情報交換などの機能が規定されている。エージェント資源ブローカーは、ラッパーのディレクトリーサービスを提供する。登録、変更、削除、検索などの機能が規定されている。ラッパーは自らをブローカーに登録しておかなければならない。ソフトウェアを使いたいエージェントは、まずブローカーにそのソフトウェアの特性を問い合わせる。ブローカーは、自分に登録されているラッパーのなかから、条件に合致するものを検索し、質問に答える。

解答を得たエージェントは、そのラッパーとACLで会話することにより、ソフトウェアの機能を利用する。

④人間とエージェントの相互動作

本仕様は、人間であるユーザーに対するエージェントの対話サービスと、ユーザーのプロファイル管理を行うサービスの二つを規定する。これらを使用するためにDFに登録するサービスタイプ、オントロジーと行為を定めている。ユーザー対話管理サービスは、ユーザーインターフェースをラップするエージェントである。一方でユーザーインターフェースを通じて人間と対話し、他方でACLによってアプリケーションエージェントと対話を行う。ユーザー個人化サービスは、人間のプロファイルを管理するエージェントである。固定情報と、学習により変化する情報の二種類を取扱うことができる。

⑤エージェントの安全性管理

本仕様はきわめて未熟である。マルチエージェントシステムにどのような安全性管理が必要であるか、心構えを述べている程度の水準である。基盤での安全性確保のためのインターフェースとして、新たにエージェント基盤安全性管理者というエージェントを設けた。またAMSとDFについても言及している。エージェント間通信の安全性として、トランスポートレイヤーでの保護と、メッセージ構文をラップするエンベロープを規定している。また監査の必要性が述べられている。今後の発展が期待される仕様である。

⑥エージェント管理における移動エージェントのサポート

本仕様は移動エージェントをFIPAのエージェント管理の枠組みのなかでサポートするための二つの抽象プロトコルと関連するオントロジー、ライフサイクルを規定している。一つは単純移動プロトコルである。単純移動プロトコルでは、移動したいエージェントは、現在いるエージェント基盤に対して、移動先の基盤の識別子を含む移動の要求を出すだけである。エージェント基盤は移動先の基盤にその要求を転送し、移動先の基盤がそのエージェントの実行を再開する。つまり移動の実作業は基盤によって行われる。もう一つは完全移動プロトコルである。この場合、移動するエージェントが自分で、コードやデータの移送、失敗などの例外処理、ホームエージェント基盤に対するアドレス変更等を行わなければならない。単純移動プロトコルに比べて面倒であるが細かい制御を行うことができる。

この二つに大別された抽象プロトコルのそれぞれに対して、具体的に、移送、複製、呼出しの三つが規定されている。移送は、移動エージェントがある基盤から別の基盤に移ったら即座に、元の基盤からは消去されるプロトコルである。複製は、別の基盤に移動後も、元の基盤でそのエージェントが実行を継続する。呼出しは、あるエージェントが別のエージェントを、別の基盤で起動する。

気になるのはOMGのMASIFとの関係だが、これらは互換で相補的であり、統一された枠組みを形成することができる。すなわち、MAFAgentSystem対AMS、MAFFinder対DF、ORB対ACCというように両者の機能をマップすることができる。

⑦オントロジーサービス

本仕様は、OKBCの知識モデルをベースとして、ACLによってオントロジーを操作するための機能を規定している。DARPAの認可のもと、OKBC仕様書の該当箇所を本仕様書に取り込んでいる。OKBC自体は、最初にも述べた通り、クライアントサーバーモデルのプログラミング言語APIなので、エージェントの高レベルの標準インターフェースを定めようとい

う FIPA の目的とは相性が悪い。そこで、OKBC やそれ以外のオントロジーサーバーを、オントロジーエージェントと呼ぶラッパーで包むことにより、一方で ACL でクライアントとなるエージェントと会話し、他方で OKBC の API や、XML/RDF 等でサーバーに接続するモデルを採った。

基本的に OKBC に準拠し、オントロジーサーバーに対する行為として、オントロジーの格納、取得、変更等を規定した。また必要な述語を定めた。さらに、オントロジー間の翻訳もしくは表現言語間の翻訳をサポートするために、ontol-relationship という述語により、二つのオントロジーの同一、拡張、翻訳可能などを表現する。また翻訳を実行する行為を定めた。付録として、オントロジーとは何かの解説と、オントロジーを開発する際のガイドラインを収録している。口の悪い人はこの付録のほうが本文よりも大切だと言う。

（４）他の組織との関係

OMG と FIPA はメンバーの多くが重なっている。また、個々の企業としてだけでなく、組織対組織としても、連絡を密にしている。OMG においては、ECDTF における Agent Working Group が、FIPA と協調して作業を行っている。FIPA のバイスプレジデントである、FujITSu Laboratories of America の Frank McCabe 氏が、Agent WG のリエゾンである。第 14 回、第 16 回の会議では、Agent WG のコチェアーであり、UML のチェアーでもある James Odell 氏を招き、活発な討論を行なった。エージェントの相互作用プロトコルを記述するために UML を拡張した AUML （Agent UML） を、OMG のプロセスにおいて UML の次版に含めるべく努力している。さらに、1999 年に OMG はエージェント技術に対する RFI を出したが（ec/99-02-03）、FIPA としてこれに対する応答を提出している。他方、W3C や IETF との連携は進んでいない。今後の課題である。

（５）今後の展開

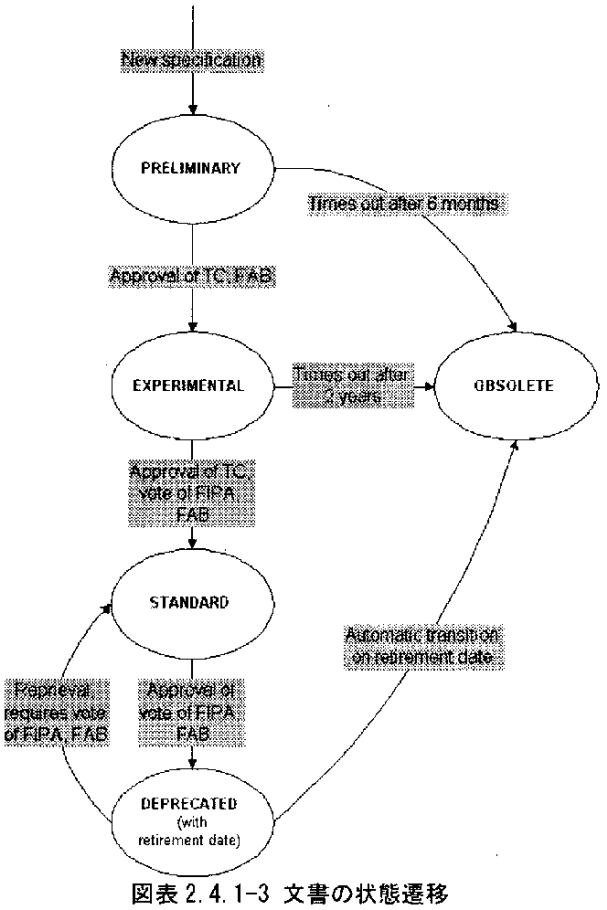
まず現在審議中の仕様を早期にリリースすることが求められる。FIPA では従来、FIPA 97、FIPA 98 というように仕様に年号をつけて、一年単位で作業を回してきたが、今後は年単位のサイクルにはこだわらず、各 TC が非同期に仕様をリリースする。具体的には、文書の種類として、Component と Profile の二つを定義した。Component は最小限の個別の技術仕様であり、Profile は Component を束ねることによって仕様の使用者に意味のあるまとまった単位を提供する。また、Preliminary、Experimental、Standard、Deprecated、Obsolete という文書の状態を定義した。これらの意味は図表 2.4.1-2 の通りである。

これを図示すると図表 2.4.1-3 のようになる。

また、大切なことは、既存の FIPA 97/98 の仕様書を上記の枠組みに収め、保守を行うことである。アーキテクチャーボードで作業を開始している。1999 年度の FIPA は目に見えるアウトプットが少なかったために、外からは活動が停滞していたように見えたかもしれないが、助走期間を終え、2000 年にはいっそうの活躍が期待される。

図表 2.4.1-2　文書の状態

Preliminary	TC の外部に出して、リビューが可能な状態。異なる TC の仕様との間で不整合がありうる。TC のみが議決すれば本状態になる。6 ヶ月以内に Experimental に昇格しなければ、自動的に Obsolete となる。
Experimental	TC 間および既存の仕様の整合性がとれており、実験的な実装および相互接続試験が可能となる状態。アーキテクチャーボードの承認が必要。2 年以内に Standard に昇格しなければ、自動的に Obsolete となる。
Standard	実装がなされ、相互接続実験によって実証された仕様。商業的な開発および運用が推奨される。メンバーの投票で過半数を獲得した後で、アーキテクチャーボードの承認を得なければならない。投票で見逃された欠陥がある場合、アーキテクチャーボードは拒否権を発動できる。
Deprecated	アーキテクチャーボードの決定により、Standard が Obsolete にされるまでの猶予期間。期間は個別に設定される。アーキテクチャーボードは、必要があれば再度 Standard 状態に戻すことができる。
Obsolete	無効となった文書。



図表 2.4.1-3 文書の状態遷移

参考資料

〔1〕 Comtec Agent Platform: http://fIPA.comtec.co.jp/fIPA/glointe.htm

〔2〕 Nortel Networks FIPA-OS:
http://www.nortelnetworks.com/products/announcements/fIPA/index.html

〔3〕 British Telecom Zeus: http://www.labs.bt.com/projects/agents/zeus/

〔4〕 CSELT JADE: http://sharon.cselt.it/projects/jade
〔5〕 http://fIPA.comtec.co.jp/fIPA/interopl.pdf
〔6〕 http://www.labs.bt.com/profsoc/facts/
〔7〕 http://fIPA.comtec.co.jp/fIPA/chronicle/nice/fIPA9407.PDF

2.4.2 自律分散サービスシステムアーキテクチャの提案－基本概念・技術と標準化活動－

（１）はじめに

イントラネット/エクストラネットの普及に伴い、大規模な情報サービスシステムを構築するニーズが増大している。このニーズに対応するため、システムやサービスの柔軟な拡張や変更、ユーザ毎にカスタマイズされたサービスの提供、複数の異なるサービスシステム間の連携、またサービスの信頼性や安全性などの保証を可能とする自律分散サービスシステム（ADSS: Autonomous Decentralized Service System）概念とシステムアーキテクチャを紹介する。また、ADSS概念とアーキテクチャを基に研究開発を進めている、ADSSプラットフォームについて概説する。ADSSプラットフォームは、類似興味を持つエージェントのグループを管理・協調させる「コミュニティ管理」、ユーザの要求と状況に応じたサービスの選択・提供・保証を行う「動的ブローカリング管理」、およびシステムやサービスの信頼性を保証する「フェデレーション管理」などの機能を持つミドルウェアであり、ADSSアーキテクチャに基づくシステムを構築することを容易にする。

同時に、オブジェクト指向分散システム基盤の標準化団体であるOMG(Object Management Group)内のADSS DSIG（Domain Special Interest Group）とその活動内容を紹介する。ADSS DSIGは、日立製作所の提案により1997年９月に設立され、ADSS技術を議論するとともに、ADSS概念の普及と技術の国際標準化を進める母体となっている。

（２）システムニーズと自律分散システム

グローバルネットワークの発達に伴って、電子的に情報やサービスを交換するコミュニティは、日々拡大の一途をたどっている。より付加価値の高いサービスの提供、あるいは企業間競争力の維持のためには、既存の情報システムの柔軟な再構成が緊急の課題である。すなわち新たな情報システムを構築するのみならず、既存の莫大なシステム財産を有効に活用し、個別のシステム群を相互接続することが、重要なソリューションとなる。

この際に重要なのは、このようなシステムの連邦（以下、「システムのシステム」と呼ぶ）は場当たり的に個別のシステムを相互接続していくのではなく、ある統一したコンセプトのもとに構築されねばならないということである。すなわち、システム統合の理念としてのコンセプトを確立するとともに、社会インフラとして「システムのシステム」を支える共通的な基盤が必要となる。

「システムのシステム」の姿を考える上で、我々が出発点としようとしている自律分散概念について述べる。これまでの工学概念の多くは、あらゆる思索を事前に巡らして目標とする姿を設計図に描き、設計図どおりにものを作り上げようとするものであった。しかし情報システムにおいては、このような伝統的な工学システムと全く異なっている。特に、システムが大規模になればなるほど、システムのライフサイクルに対する視点が重要とな

ってくる。

情報システムは生まれ、育ち、お互いに連携し、そして、その一部は逐次入れ替わってゆく。これは、あたかも生物の如きであり、生物として対象をみたシステム構成概念が不可欠である。生命論では、できあがった姿は過程の結果であるとして、姿そのものよりも過程を重視する。これに倣えば、新陳代謝とか動的な安定性を許容するといったことを積極的に取り込んでシステム構成論を展開することが大規模なシステムでは特に必要である。自律分散概念は、このような生命論的な見地に立ったシステム構成論を展開しようとするものである。その骨子は、我々が大規模な情報システムを建設するとすれば、生物世界を手本として、「要素の自律化と要素の相互作用を通じての機能実現」をシステムの構成原理にすべきであると言える。

この原理をシステム要素同士の通信方式（内容コード通信）で具体化し[10]、鉄鋼や鉄道などの大規模制御システムに適用してきている。これらの巨大な制御システムにおいては、一挙に全システムを建設することは不可能であり、成長型のシステム構成が望まれる典型例といえるが、自律分散概念によるシステム構成を用いることで、この要求に応えている。

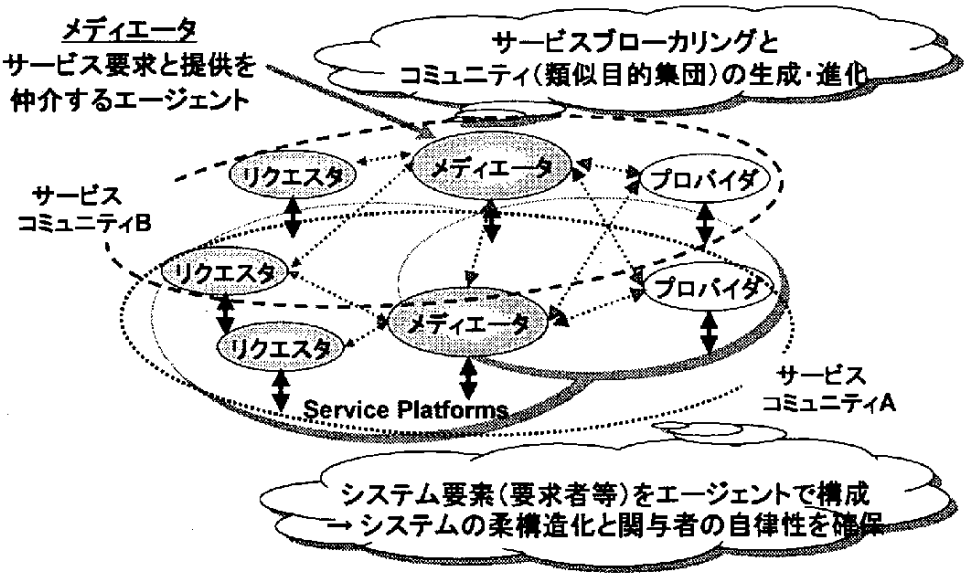
（3）自律分散サービスシステムの構成と機能

ここでは、自律分散概念を情報サービスシステムに適用したADSSのモデルを提起する。

①ADSSの一般モデル

ADSSでは、サービスの提供者（プロバイダ）とサービスの消費者（リクエスタ）の間に立つ存在であるメディエータの概念を導入した。メディエータは、情報サービス分野におけるブローカー的存在であり、日々莫大な数のサービスが生まれ変化していく環境において、ユーザに最新のサービスの情報を提供することでサービスの交換を活発に役割を果たすものである。さらに、1対1のみならず多対多の複雑な取り引きの場合や、デジタルコンテンツを流通させる場合などには、メディエータは交渉やコンテンツ交換の場を提供するとともに、プロバイダとリクエスタの間の調整を行う仲介役にもなり得る。以下では、メディエータを軸としたシステムモデルと、それを実現するための技術の基本要素について述べる。ADSSでは、図表2.4.2-1に示すようなメディエータが管理する場（以下、サービス・コミュニティと呼ぶ）において、プロバイダのエージェントとリクエスタのエージェントとがサービスの情報とサービスコンテンツを交換するモデルを基本とする。

[10] K. Mori, et al.：Autonomous Decentralized Software Structure and its Application, Proc. of FJCC'86(1986)



図表 2.4.2-1　自律分散サービスシステム基本モデル

メディエータは、このサービス・コミュニティ内のメンバの管理、情報配信範囲の管理、コミュニティ内で用いられる語彙（オントロジ）の管理、また他のメディエータとの連携などの業務を担う。プロバイダとリクエスタが、どのメディエータに仲介委託の契約を結ぶかは自由であり、一つのプロバイダ（あるいはリクエスタ）が複数のメディエータと契約しても構わない。また、リクエスタ、プロバイダというエージェントの区別は、一つのサービス交換の中での役割に過ぎず、実際の商取引と同様に、プロバイダも次の瞬間にはリクエスタになり得る。リクエスタもまた同様である。

②メディエータの機能

サービス・コミュニティは、メディエータに仲介を委託したリクエスタやプロバイダの集合である。サービス・コミュニティは類似の興味や目的を持ったエージェントのグループであり、コミュニティによるメンバの絞り込みによって、サービス仲介の効率化、および仲介の結果に対するコミュニティのメンバの満足度向上が期待される。コミュニティが繁栄し、有効に機能するかどうかは、メディエータがいかにそのメンバから知識を引き出し、コミュニティの運用にフィードバックし、仲介するサービスに付加価値を与えられるかにかかっている。この観点から、図表 2.4.2-2 に示すように、メディエータが提供する機能として以下のものが想定される。

a) コミュニティ管理機能：コミュニティを生成、管理し、コミュニティの成員間での通信を実現する機能である。より詳細には、以下の機能に分割されるであろう。

・グループ管理機能：コミュニティを生成し、コミュニティに属するリクエスタとプロバイダに関する情報の管理を行う機能である。同時に、個々のエージェントの権限に応じて、アクセスできる情報や利用できるサービスの範囲も管理する。

・サービスプロファイル管理：コミュニティ内でサービスの情報を交換するためには、サービスを共通の属性（サービスプロファイル）を用いて記述する必要があ

る。コミュニティ内のオントロジを管理し、それに基づいてサービスの属性を定義する機能である。

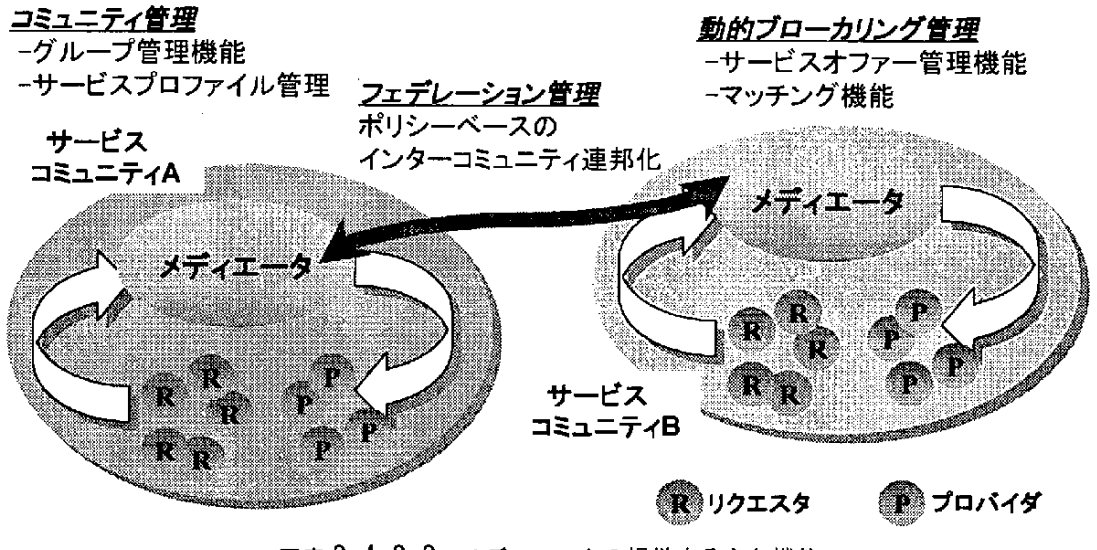
b) 動的ブローカリング機能：上記のサービスプロファイルに基づいて登録されたサービスと、要求の内容とを結び付ける機能である。本機能により、任意のタイミングで、プロバイダからのサービス登録とリクエスタからの要求の内容とを結び付け、その結果をリクエスタあるいはプロバイダに提示する。そして、どのリクエスタが、どのサービスを選択するかが決定すれば、プロバイダ側で実際にサービスが実行される。より詳細には、以下の機能に分割されるであろう。

・サービスオファー管理機能：プロバイダが提供できるサービス（サービスオファー）を登録し、管理する機能。

・マッチング機能：リクエスタからのサービス要求の条件と登録されたサービスオファーを比較し、合致するサービスオファーを選定する機能。マッチングの方式は、ポリシーというルールで規定される。

c) フェデレーション機能：コミュニティ内で情報が閉じてしまってはその発展性は期待できない。フェデレーション機能は、あるコミュニティの中で、リクエスタからの要求を満たすサービスが無い場合に、他のコミュニティと連携し、サービスを取得する。一般的に、異種のコミュニティでは、異なる言語体系（オントロジ）に基づいて定義された、異なるプロファイルと異なるポリシーが用いられているので、必ずしも連携ができるとは限らない。しかし、コミュニティ間で共通なオントロジが用いられている場合、あるいはオントロジの変換が可能なような定義(メタ定義)がなされている場合には、言語体系の変換を行うことにより、他コミュニティのサービスも結び付けの対象とすることができる。すなわち、他のメディエータと連携してサービスの取り寄せを実現する機能である。

d) サービス実行機能：メディエータによって仲介されたサービスオファーはリクエスタからプロバイダに対して実行要求を出すことも可能だが、メディエータがリクエスタに代わって実行要求しても良い。この方式のメリットは、リクエスタとプロバイダの間の関係を疎にすることができ、例えばリクエスタがプライバシ保護上の理



図表 2.4.2-2　メディエータの提供する主な機能

由から、「誰がそのサービスを要求したか」をプロバイダに知られたくないという問題を解決することができる。また、プロバイダへのサービス実行要求が特定の契約を事前に結んでいる者だけに許可される場合、メディエータが実行要求を代行することで、リクエスタとプロバイダとの煩雑な契約手続きを省略できるという利点もある。

e) ディペンダブル機能：コミュニティが繁栄するためには、コミュニティの安定運用による信頼関係が形成されねばならない。この一助として、通信や各種メディエータの機能に障害が発生した際にも、メンバ間のサービス交換への影響を最小化するような救済機能を提供する。具体的には、リソースの多重化によるシステムの高信頼化、各種情報のバックアップによる障害時の情報の消失防止などである。

（４）ADSS プラットフォームのソフトウェア構成

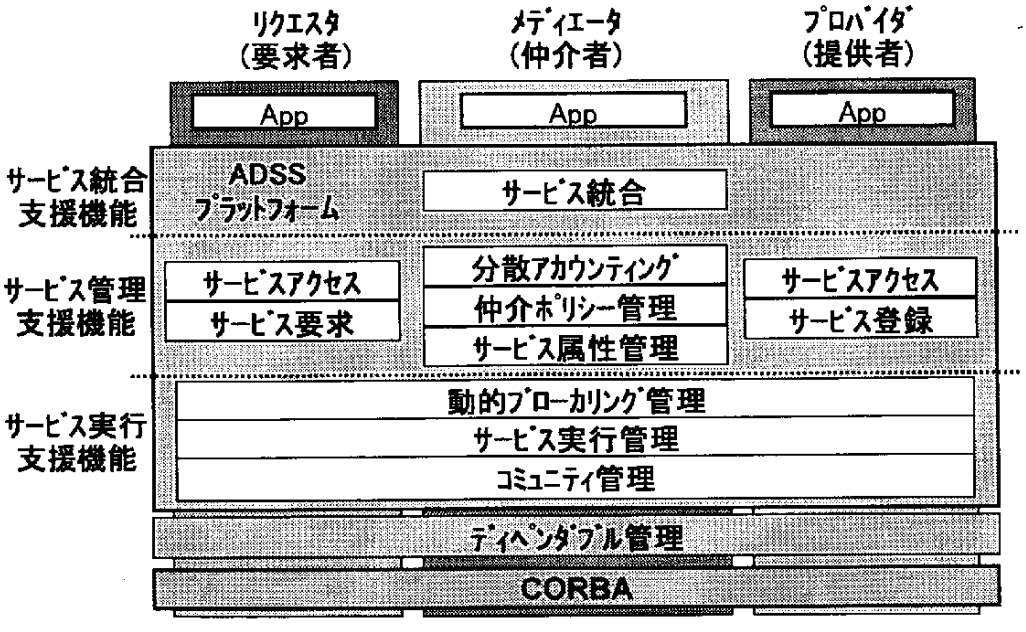
図表 2.4.2-3 に、ADSS 概念に基づいた情報サービスシステムを実際に構築するために、共通の基盤として用いられるミドルウェアである ADSS プラットフォームの機能構成を示す。

現在の ADSS プラットフォームの機能は分散オブジェクトシステム構築用ミドルウェア CORBA 上の関数として実装されている。リクエスタ、メディエータ、プロバイダの各アプリケーションは、ADSS プラットフォームの必要な機能コンポーネントを利用して、作成される。

（５）OMG での標準化活動

①現状

OMG(Obeject Management Group) [11]は、分散システム環境でオブジェクト同士がメッセージを交換するための共通仕様を作成している米国の標準化団体である。CORBA(Common Object Request Broker Architecture) は、実際に存在している位置に関係なく、オブジ



図表 2.4.2-3　自律分散サービスシステムプラットフォーム機能構成図

ェクトの相互運用性を保証するために、OMG が規定したオブジェクト間の通信プロトコル、インターフェース定義言語などの仕様、機能およびソフトウェアを意味する。

OMG における ADSS 概念の普及と技術の標準化の活動は、1997 年 6 月に開催された Montreal 会議で、ADSS Working Group が設立されたことに始まる。同年 9 月には、ADSS DSIG として活動拠点を立ち上げ、また、11 月には ADSS の基本概念と技術とを説明した ADSS Whitepaper[12]が承認された。

1998 年 9 月には、Telecom DTF(Domain Task Force) を通じて、図表 2.4.2-4 に示される Access に相当する、プロバイダとリクエスタがメディエータと契約するためのインタフェースと、サービスの登録およびサービスの要求をするためのインタフェースの RFP (Request For Proposal) [13]が発行され、現在最終版の仕様を作成中である。

②他の標準化団体との関係

OMG においては、他の標準化団体のメンバも積極的に技術の提案や標準化を進めている。2.4.1 で報告のあった FIPA のメンバも、OMG に参加し、特に Agent Working Group で、エージェント技術や FIPA の成果の紹介を行っている。ただ、OMG の標準化対象は CORBA のある機能コンポーネントのインタフェースであるのに対し、FIPA では、より上位の概念やプロトコルなどを扱っているため、この二つの団体が対象としている技術はレイヤが異なっているという印象である。しかし、OMG のメンバもエージェント技術分野での FIPA の成果は評価しており、有効活用できる部分はできるだけ尊重していこうという前向きな取り組みが感じられる。

また、通信のディジタル化が新たな通信サービスを生み出すとの期待の下、テレコム企業を中心にコンソーシアム TINA-C[14] (Telecommunication Information Network Architecture-Consortium) が結成され技術開発が進められている。TINA-C では、異なるネットワーク・キャリアをまたがったサービスの交換を容易に実現するため、ネットワークを構成する装置の管理、ネットワーク自体の管理、およびネットワークにより提供されるサービスの管理を統合した、ネットワーク・オペレーションシステムのソフトウェア仕様の標準化を進めている。TINA-C では、ADSS と共通する非常に優れたビジネス・モデルや情報システムのアーキテクチャが提案されており、サービスシステムのアーキテクチャ立案の出発点とすることができる。

TINA-C では、策定した仕様の実装を行う際に CORBA を用いることを想定している。そのため TINA-C のメンバも、OMG 内において多方面で参加しており、特に Telecom Task Force では積極的な提案活動を行っている。

③今後の予定

ADSS DSIG で標準化を目指すインタフェースの全体像を図表 2.4.2-4 に示す。今後は、同報的にサービスの登録や要求を行う際に重要な Multicast Service や、サービスの利用量情報を管理するための Accounting インタフェースなどの標準化が予定されている。テレ

---

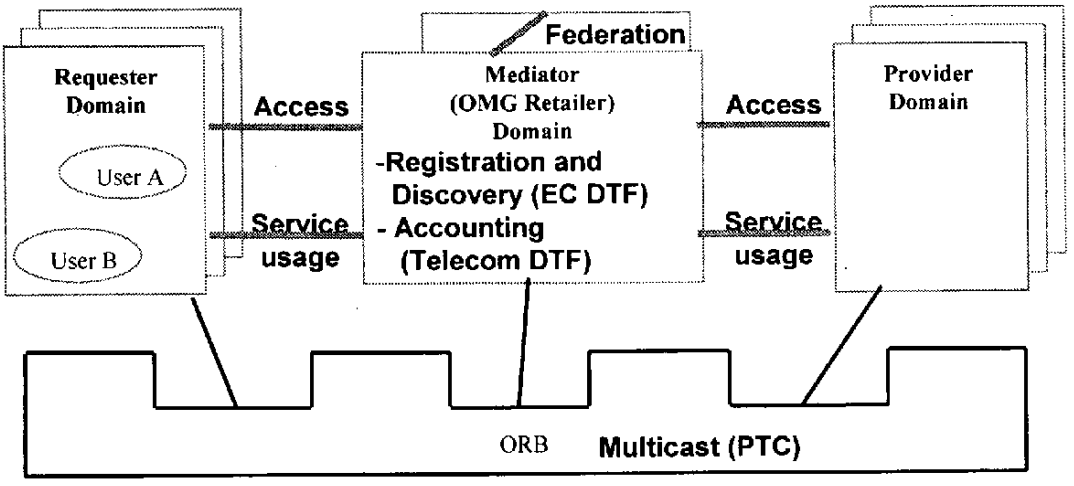
[11] Object Management Group, http://www.omg.org/

[12] ADSS Domain Special Interest Group Whitepaper Ver.1.0 : ads/97-12-01(1997), http://www.omg.org

[13] Telecommunication Service Access and Subscription: Request For Proposal: telecom/98-09-07(1998), http://www.omg.org

[14] TINA-C (Telecommunication Information Network Architecture-Consortium) URL: http://www.tinac.com

コム分野に関わる提案の際には、特にTINA-Cのメンバと積極的に協力を要請していく。



図表 2.4.2-4 ADSS DSIGでの標準化対象インタフェース

（6）むすび

次世代の情報システムの共通基盤として開発を進めている自律分散サービスシステムの構成を概説した。この考えを具体化するには、世界の関係者達との合意が不可欠である。幸いにして、分散オブジェクト技術の国際標準化団体であるOMGにこの意義が認められ、この中でADSS DSIGを設けて具体化のための審議を進めるに至っている。日本から発信したシステム技術として実が結ぶよう、一層の努力をしてゆきたい。

Ⅱ．本編

３．技術課題の分析

－応用から要請される技術課題－

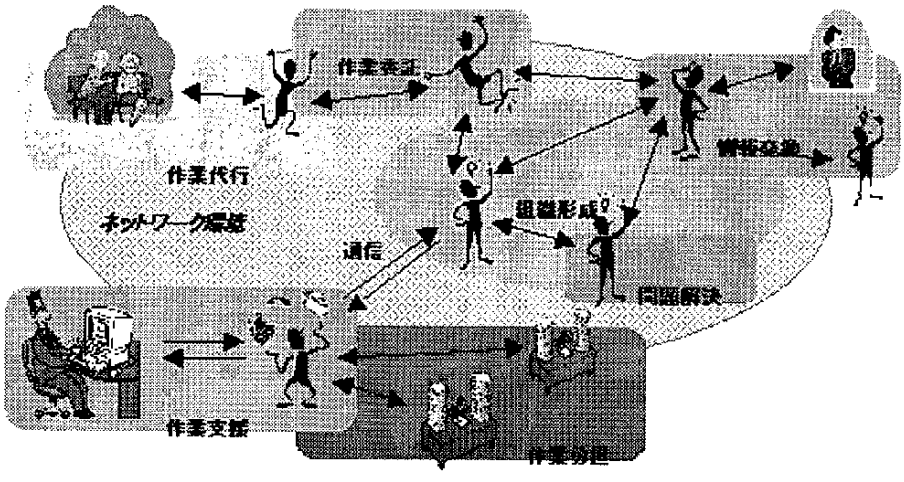
3. 技術課題の分析
－応用から要請される技術課題－

本章では、応用面からみた技術課題について検討する。サービス統合、マルチエージェントシステム、インタフェースエージェント、標準化の４つの側面に焦点をあてる。

3.1 サービス統合
3.1.1 エージェントの役割

アプリケーション技術とエージェント技術との関わりについて、EC などを事例として現在の技術状況と今後の課題について概観する。

アプリケーションの立場からみたエージェントに期待する役割／働きは図表 3.1.1-1 のようにまとめられる。



図表 3.1.1-1　エージェントの役割

すなわち、エージェントは、エージェント相互間での通信と情報交換、処理の分担や委託、エージェント組織形成と協調問題解決などの機能を利用して、単体、もしくは、複数で動作してサービスを提供する。例えば、EC などのアプリケーション領域では、これらの機能をベースとして、ネットワーク環境上での商品情報の検索や配布、プロシューマエージェントによるオークションや価格交渉、利用者の商品選択支援や購入処理の代行など、応用領域の特性を反映したサービスが実現される。アプリケーション構築の立場からみると、エージェントを導入したアプリケーション、或いは、そこでのサービス／機能の実現レベルが問題となる。

3.1.2 現状のアプリケーションの特徴

そこで、まず、エージェントの概念や技術に基づいて開発されつつあるアプリケーションの性質／特徴を次の４つの視点で整理してみた。

１）対象問題：

アプリケーションによって扱われる問題は、一般に、問題の構造や処理手順が整理、明確化されているか否かによって、定型問題と非定型問題に分類される。所定の手順に基づく会計処理などは定型問題であり、発注者の要請を反映した建築物の意匠設計などは非定

型問題に分類される。一般に、非定型問題では、広範な問題解決知識や高度な問題解決機能が必要となり、問題解決プロセスのシステム化なども困難となる。こうした観点から、情報検索、故障診断、スケジューリング／プランニング、設計／製造などに関わる問題を対象として研究開発や試作が進められているアプリケーションの性質をみると、現時点では、その大半が定型問題を扱っている段階にあるといえる。

2）規模：

エージェント指向のアプリケーションは、単体のエージェントとして稼動するものから、複数のエージェントが連携して動作するものまで、様々なタイプが存在する。そこで、こうしたアプリケーションの構成要素となるエージェントの個数や、各エージェントが持つ機能や知識の量（エージェントの粒度）などの側面から現時点のレベルを評価すると、現時点では、ほとんどのアプリケーションは小規模(エージェント数でみれば１～数個程度)、もしくは、高々中規模（エージェント数：数個～数十個程度）の段階にある。

3）形態：

前項と関連して、エージェントを新たな機能要素とするアプリケーションの構造や形態をみると、情報検索やチュータリング(コンピュータ利用支援)などのアプリケーション、或いは、EC応用のオークションにおけるプロシューマエージェント間での情報交換などにみられるように、現時点では、単体、もしくは少数のエージェントによる連携処理を扱っているといえる。複数のエージェントが協調的に組織を構成して問題解決を実行するマルチエージェント型アプリケーションはこれからという段階である。

4）適合性：

アプリケーションと対象問題との適合性という側面からみると、現時点では、既存の問題やアプリケーションシステムをベースとした試験的／代替的な問題、或いは、アプリケーションで要求される機能とエージェントのもつ機能との親和性が比較的高い問題が選択されている場合が多い。アプリケーションが扱う問題の性質やシステム設計条件などからみて、エージェント技術が必要不可欠といえるアプリケーションが模索されている段階といえる。

3.1.3 エージェント技術の現状

次に、アプリケーションの構築や運用を支えるエージェント技術の現状を次の観点から概観する。

a) エージェント通信言語（ACL）：

エージェントが相互作用する際に用いられる通信方式やそこで利用される言語系などに関する技術である。これまで、KQMLを嚆矢として、同言語をベースとした言語処理系やアプリケーション構築に関する研究が進められてきている。しかし、現時点では広く合意のとれた言語体系は確定しておらず、これがエージェント間での相互運用性やエージェントレベルでの再利用性の面で障害となることが指摘されたりしている。これに対して、現在、ACLを含むエージェント技術全般に関する標準化作業が複数の組織（FIPA、OMG、etc.）で進行している。

b) コーディング／記述言語：

エージェントの知識や動作を記述（プログラム）する形式的な言語系や、それらの

記述に基づいてエージェントを動作させる処理系に関する技術である。現時点では、前述したACLと同様に、固有のエージェントモデルやアプリケーションに準拠した様々な言語処理系が開発・利用されている。最近では、モバイルエージェントシステムを筆頭として、Java言語などを基盤とした処理系なども多数提案されてきている。現時点では、その多くが研究・実験の段階にあり、処理系としての評価や一般ユーザ向きの言語処理系の提供などは今後の課題である。

c）既存ソフト／資産との連携：

アプリケーション構築においてエージェントを導入するにあたり、既存のアプリケーションとの連携や従来型ソフトウェア（レガシーソフトウェアを含む）の再利用などを考慮する必要が生じてくる。現在、従来型ソフトウェアなどをエージェントとして利用するラッピングと呼ぶ概念に基づいて、種々のソフトウェア／情報資源の相互連携や再利用を図ることが徐々に可能となりつつある。但し、これはソフトウェア工学分野では長年の課題となっているソフトウェア部品化の問題に直結しており、部品再利用のための知識記述をはじめとして解決すべき課題も多く残されている。

d）方法論／開発支援環境：

アプリケーション設計の観点からみると、現時点では、エージェント指向のシステム設計／実装に関するガイドラインや方法論がほとんど提供されておらず、アプリケーション設計で採用されたエージェントモデルや言語系に依存した設計が行われている。しかし、ここ１～２年の間に、エージェント／マルチエージェント概念に基づくアプリケーション構築の支援を目指す方法論や設計手法に関する研究が徐々に活発化してきており、今後の発展が期待されている。

e）費用対効果／生産性：

アプリケーションの開発や運用の側面からエージェント技術を評価してみる必要がある。現時点までの開発事例においては、評価の視点がそれぞれ異なるとはいえ、本節冒頭で述べたエージェントの諸機能がアプリケーション構築に有効であったことが報告されている。一方、現状では、系統的な評価手法が確立されていない、評価事例が乏しい（ほとんどない）、また、設計方法論や対象問題との関係等も考慮する必要があることから、評価に関する議論は時期尚早という感もある。

3.1.4 アプリケーションからみた利点

上述した現状のもとで、エージェント技術に基づくアプリケーションにとって有利な点として以下のものが挙げられる。

1）要求指向情報処理への接近：

エージェントの導入により、サービスやシステムの利用者の個別的な要求に対処する新たな手段が得られる可能性がある。例えば、商品情報の選択や商品購入に関する利用者固有の特性を学習して商取引を支援するエージェントなどがその例となる。

2）拡張性／柔軟性の向上：

エージェントを主要な構成要素とするモジュール性の高いシステムを構成することにより、例えば、サービス／機能の追加・変更などが容易となり、多様な利用要求や設計条件とその動的変動に対処するアプリケーションが構築できる。

3) 分かり易さ／馴染み易さの向上：

エージェントが種々の状況に即した合理的な動作や利用者の要求／好みを考慮した動作を自律的に選択する、或いは、エージェントの外見や振る舞いの特性を人間のそれに近付けたりすることにより、利用者からみたシステムやサービスの理解性や親密感を高めることができる。これは、利用者とサービス／システムとの仲介役となるインタフェースエージェントなどの応用にその例をみることができる、

4) 新規サービス実現への足掛かり：

エージェントに埋め込まれた知識に基づく問題解決や相互連携（協調）機能を活用することにより、例えば、オンデマンドのプル型サービスの高度化や、利用者特性に準拠した新たなプッシュ型サービスの提供など、より高度な利用者指向サービスが実現できる可能性がある。

3.1.5 エージェント技術の課題

一方、サービス／アプリケーションからみた現下の課題としては以下のものが挙げられる。

a) 相互運用性と標準化：

様々な目的／用途で構築されたエージェントを必要に応じて探索・起動して活用するためには、エージェントの相互運用性を高める必要があり、現下の標準化活動の進展とその成果が期待される。

b) マルチエージェント化：

複数のエージェントが状況に応じて効果的に連携・協調して問題解決を実行するマルチエージェント型アプリケーションの実現にむけた研究開発が必要である。

c) 知識処理レベルと問題解決能力：

現下のエージェントの問題解決能力を高め、高度なアプリケーション／サービスの設計開発に耐え得る知的エージェントの実現技術を充実する必要がある。

d) エージェントらしさ：

老若男女すべてのユーザに対する適切なサービスや知的な支援を提供するエージェントを実現するためには、前項で述べたエージェントの問題解決能力だけでなく、ユーザからみた親近感や存在感を産み出す手段としての擬人性や社会性を高める工夫が必要となる。

e) セキュリティ／安全性：

利用者が抱える仕事を安心して任せられる信頼性の高いエージェントを実現するために、エージェントが扱う情報だけでなく、仕事を分担するエージェント自身のセキュリティや安全性を確保する技術を開発する必要がある。

f) 開発環境：

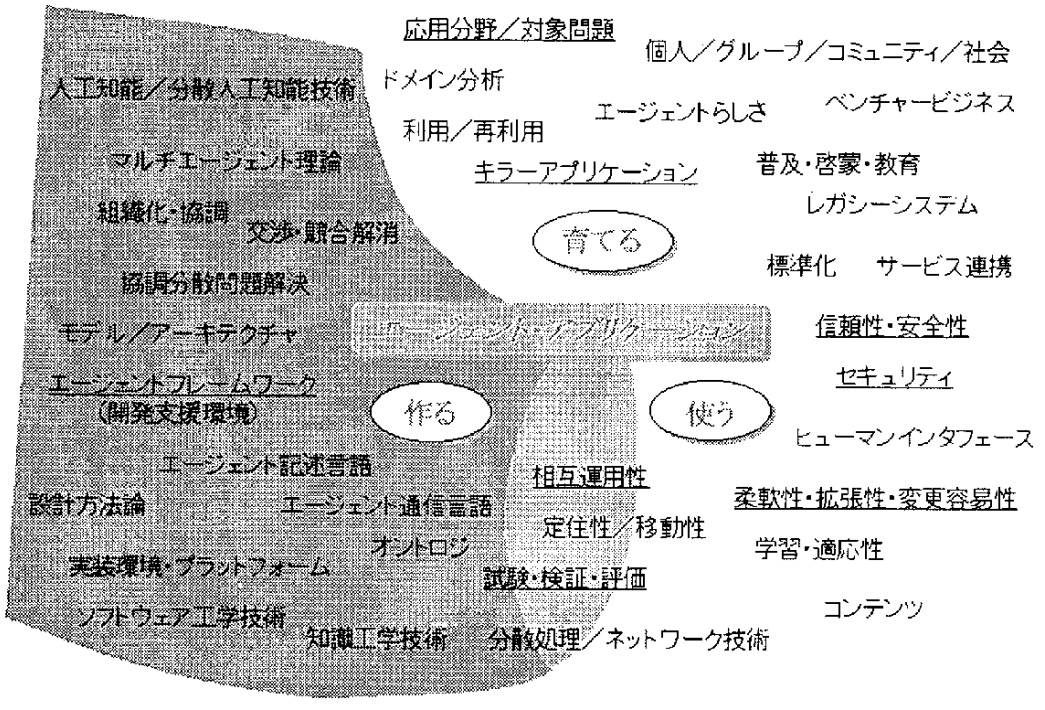
様々なサービス／アプリケーションの設計開発を効果的に支援・促進する手段となる設計方法論、エージェント記述（プログラミング）言語、設計支援ツール、試験・検証・評価手法などを開発・整備・拡充する必要がある。

3.1.6 エージェント技術の技術マップ

上述した以外にも、エージェント技術には解決すべき多くの課題が残されている。これらの課題を、「作る」、「使う」、「育てる」という３つの視点からみた技術キーワードによって整理したものを図表 3.1.6-1 に示す。

すなわち、「作る」観点からみると、エージェント指向アプリケーションの構築の基盤となるモデル／アーキテクチャ、設計支援環境としてのエージェントフレームワーク、エージェント高度化のためのマルチエージェント理論などが主たる課題となる。

また、「使う」観点では、相互運用性、セキュリティ・安全性・信頼性、柔軟性・拡張性といったシステム特性とならんで、アプリケーションの試験・検証・評価が重要となる。



図表 3.1.6-1　技術マップ

更に、「育てる」観点では、適切な応用分野／対象問題の選択、特に、エージェント技術適用の効果を確認できるキラーアプリケーションの開発が重要なことは当然として、エージェント以外のアプリケーション／システム等を対象とした利用／再利用技術、エージェントらしさの向上なども大きな課題となる。

3.2 マルチエージェントシステム

マルチエージェントシステムは相互作用可能な複数のエージェントが協調的あるいは競合的に動作するシステムである。これまでも述べられてきたように、今後様々なエージェントが開発され、相互作用が可能になるにつれて、それらを一つの組織としてどのように動作させるかということは重要な課題になる。

マルチエージェントシステム（MAS）は協調的 MAS と競合的 MAS に大きく分類することが

できる。協調的MASでは組織の中に共通の達成すべき目標が存在し、いかにしてエージェントを協調的に動作させ、その目標を達成するかが課題になる。一方、競合的MASではエージェントそれぞれが個別の目標を持ち、それらが同時に達成できない場合に、いかにしてエージェントを調整するかが課題になる。本稿ではこのような協調的MASと競合的MASのそれぞれにおいて、典型的な応用事例を示しながら、今後の技術課題を議論していく。

3.2.1 協調的マルチエージェントシステム

協調的マルチエージェントシステムの典型的な応用事例はサプライチェーン管理システム〔1〕であろう。サプライチェーンとは何らかの商品を製造する企業において、顧客からの受注に始まり、原材料の確保、商品の製造、在庫管理、顧客への配送に至る企業活動の連鎖である。現在の熾烈な企業競争を勝ち抜くためには、受注から配送までの時間を短縮する努力が必要であり、効率的なサプライチェーン管理が必要とされている。このようなサプライチェーンには、企業内の営業、資材調達、製造、流通などの各部門が緊密に連携し、協調的に動作する必要がある。一方で顧客の要求や市場の動向、製造や流通ラインのトラブルなど様々な不確定要素が企業を取り巻いており、これらの外乱に対応しながら、製造スケジュールを柔軟に調整してゆく必要がある。

このようなサプライチェーン管理を自動化するには協調型MASは有効な手段であると考えられるが、その実現のためには以下のような機能が必要となるであろう。

①自律分散型アーキテクチャ

サプライチェーン管理機能はそれぞれの機能に応じて、複数の自律的エージェントにより実現されるべきである。このような自律分散型のアーキテクチャを採用することにより、いずれかのエージェントにトラブルが生じても、他のエージェントは動作しつづけることができ、システム全体に影響を及ぼすことが少なくなる。

このようなエージェントを構築する上で考慮すべきことは既存のレガシーソフトウェアの存在である。レガシーソフトウェアを廃棄して、すべてのエージェントを一から作り直すことは膨大なコストを要する。そこでレガシーソフトウェアをエージェント化する技術が重要になる。このような問題に対して、Genesereth〔2〕は変換器、ラッピング、書き換えによる手法を議論している。

次にエージェントを結びつけるエージェント通信言語やプラットフォームの研究も重要になる。エージェント通信言語としては言語行為理論（Speech Act Theory）をベースにしたKQML〔3〕が代表的である。プラットフォームの事例としてはABE（Agent Building Environment）、JAT（Java Agent Template）、JESS（Java Expert System Shell）、Voyager、OAA（Open Agent Architecture）などがある〔5〕。これらのプラットフォームの多くはその実装の容易性からJavaで記述されているものが多い。またFIPA〔4〕によりエージェント通信言語やエージェントプラットフォームの国際標準化活動も行われている。

エージェントプラットフォームの整備によりエージェント間でのメッセージ交換が可能になったとしても、エージェント間での相互作用を実現するためにはオントロジの整備も重要な課題である。すなわち、システム全体で用いる用語やその意味を統一しなければエージェント間でのコミュニケーションは成り立たない。オントロジの整備は地道で手間のかかる作業であるが、Cyc、DARPAオントロジ共有プロジェクト、Ontology Base、WordNet

などの開発事例がある〔5〕。また企業活動に特化したオントロジ開発の試みとしてはTOVEプロジェクト〔1〕がある。

②協調的問題解決能力

達成すべきタスクが単一エージェントの能力を越えたり、複数のエージェント間で調整が必要な場合には、他のエージェントにタスクを割り当てたり、それぞれエージェントの制約が充足されるように調整する機能が必要になる。負荷の変動やエージェントの故障など、動的に変化する環境下でのタスク割り当てに関するマルチエージェント研究のもっとも代表的な事例は契約ネットプロトコル〔7〕である。契約ネットプロトコルは人間社会の中で行われている契約の概念をタスク割り当ての枠組みの中に組み込んでいる。プロトコルはタスクの仕様を記述したタスク通知（task announcement）メッセージがマネージャエージェントからブロードキャストされ、実行可能なエージェントはその能力を記述した入札（bid）メッセージをマネージャに送る。マネージャは受け取った入札メッセージの中からタスク実行にもっとも相応しいエージェントに対して判定（award）メッセージを送ることによりタスク割り当てが行われる。このプロトコルの特徴はエージェントがタスク通知メッセージを選択し、マネージャが入札メッセージを選択できるという相互選択の機能である。この機能により一方的なタスク割り当てではなく状況の変化に応じた柔軟なタスク割り当てが可能になる。契約ネットプロトコルの製造ラインへの応用事例としてはParunak〔8〕によるものがある。

さて、状況によっては割り当てられたタスクが独立ではなく、その達成のためにはエージェント間で調整を行うことが必要になることがある。このような問題に対処する手法としては分散制約充足法がある。この定式化ではエージェントはそれぞれ変数を持っており、エージェント間には制約が存在する。横尾ら〔9〕は解が存在する場合に、無限ループに陥ることなく充足解を導き出す非同期バックトラックアルゴリズムを提案している。このようなエージェント間の制約を解消する手法としてはその他に、Multiagent Truth Maintenance〔10〕、Multistage Negotiation〔11〕、Hierarchical Behavior Space〔12〕などの研究がある。

エージェントが協調的にふるまうのはエージェント同士に限ったことではない。時には人間のオペレータに支援を求めたり、人間と協調的に作業することも必要になるであろう。システムをエージェントにより完全に自動化するよりも、人間の介在を許す方が現実的であるともいえる。今後はこのような研究の重要性はより高くなるであろう。

③自己組織化機能

サプライチェーン管理において新たな部門や製造ラインの拡充、廃止などによりシステムの規模が変化することは頻繁に起こりうることである。またインターネットを通じた外部の企業との連携も今後さらに盛んになることが予想される。したがってサプライチェーン管理システムもその構成要素や規模を柔軟に変化させる機能が必要になる。このためにはエージェントをプラグインとして動作させるために、インタフェースの違いを自動的に吸収したり、初期設定を自動的に行うような機能が必要になる。また、さらに高度な要求としては、環境の変化に応じてエージェントの役割や組織を動的に変化させるような機能も必要になるであろう。このような研究事例としてはマルチエージェントプロダクションシステムにおける自己組織化の事例〔13〕が挙げられる。

④局所的問題解決能力

マルチエージェントシステムでは先に述べた明示的な協調手法だけでなく、エージェント自らの局所的判断に基づく暗黙的な協調手法も重要である。すなわち明示的な協調手法は一般に多量のメッセージ通信が必要になるのに対して、暗黙的な協調手法は通信が必要でない。もちろん協調の効果と通信量はトレードオフの関係にあると思われるが、これらをうまくバランスを取ることが重要になる。Partial Global Planningの研究〔14〕では、エージェントはそれぞれ高度なプランニング機能を持っており、抽象度の高い partial plan を交換することにより分散協調問題解決における通信オーバヘッドの問題に対処している。

3.2.2 競合的マルチエージェントシステム

競合的マルチエージェントの典型的な応用例は電子商取引システムであろう。このところのインターネットの社会への急速な普及に伴い、インターネット上でも企業間、あるいは企業対顧客の商品の売買が可能になった。その中でも eBay(http://www.ebay.com/) や OnSale (http://www.onsale.com/) などのインターネットオークションサイトは顧客が商品の値段を決定する新しい商取引システムといえる〔15〕。このような電子商取引では売り手はできるだけ高く商品を販売したいと願うであろうし、買い手はできるだけ安く商品を購入したいと願う。このような取引をエージェントに代行させるとするならば、それは競合する個別目標をもつ競合的マルチエージェントシステムとなる。システム設計者がこのような競合的システムを設計する際には、取引戦略を含んだ個別のエージェントを設計することはできず、利用者に委ねられるのが普通であろう。(取引戦略は利用者の秘密事項である。) したがって設計者ができることはエージェント間の取引プロトコルの設計である。インターネット上の商取引では不特定多数の利用者が自由に参加できることから、さまざまな不正が起こりうる。したがって取引プロトコルの設計ではエージェント間の調整とともに不正への対処が必要になる。

それでは以上のような競合的マルチエージェントシステムを実現するために必要な機能を議論しよう。

①公正なプロトコル設計

エージェント間で取引が行われている際に、その取引が公正なものであることを保証することは重要である。例えば、あるエージェントが嘘の申告をすることにより利益を得るならばそれは望ましいプロトコルとはいえない。例えばある商品のオークションにおいて参加者A、B、Cがおり、入札予定価格がそれぞれ18万円、20万円、23万円であったとしよう。(ここでは入札は封筒にいれて一回限りでおこなわれるとする。) この場合、最高価格を入札した参加者Cが落札するのが普通である。ところがもし参加者Cが他者の入札予定価格を知ったとすると、20.1万円で入札することが考えられる。したがってこのプロトコルでは自らの利益を増加させるために、入札予定価格を操作することが可能なプロトコルであるといえる。これに対してVickreyオークションと呼ばれる手法がある。このプロトコルでは最高価格を入札した参加者が落札するが、その価格は第二のものとなる。すなわちこの例では参加者Cが(参加者Bが入札した)20万円で落札することになる。このプロトコルでは落札者と落札価格を切り離すことにより入札者が入札価格を操作すること

の意味を無くしている。例えば参加者Cが入札価格を20.1万円に下げたとしても落札価格は変わらない。むしろ他に参加者がいた場合には落札できない可能性が増加する。一方、他に参加者がいた場合に、落札の可能性を増加させるために入札価格を引き上げることは逆に23万円以上で落札する可能性があり、その場合は赤字になってしまう。このようにVickreyオークションは各参加者が当初の入札予定価格を正直に申告することが最良の選択となるプロトコルであるといえる。以上は取引プロトコルを設計する一例に過ぎないが、エージェントが虚偽の申告やふるまいを行わないように競合的エージェントの調整を図る様々なプロトコル設計に関する議論は文献〔16〕にまとめられている。

②市場指向コンピューティング

オークションの議論は主に1対多の小規模な商品取引のために用いられているが、さらに規模の大きな取引を扱う手段も必要になる。利己的な人間から構成される人間社会が中央集権的な機構がなくても、ある程度秩序あるふるまいを見せるのはそこに市場機構が機能しているからであるということができる。したがって、人工エージェントからなる競合的マルチエージェントシステムを自律分散的に制御する機構として、経済学の知見を導入することは有用であろう。このような試みとしてはタスク割り当ての問題に市場機構を組み込んだ市場指向コンピューティングの研究〔17〕がある。

③利用者プライバシの保護

電子商取引においては買い手が必要なものをシステムを介して購入するだけでなく、売り手が積極的に商品を広告することも可能である。このようなサービスとしては買い手のプロファイルや購入履歴をもとに商品の推薦を行うリコメンダーシステム〔18〕がある。このように、今後のインターネットサービスでは利用者の個人情報は様々なサービスを行う上での重要な情報源となるであろう。そのような個人情報をいかに保護するかということは重要な課題となる。このような試みとしては匿名化器(anonymizer)や偽名化器(psudonimizer)の研究がある。匿名化器の一例としては利用者は取引の相手に応じて何種類もの電子メールアドレスを使い分けることを支援するシステムがある〔19〕。偽名化器の一例としては、多数のWWWプロキシサーバを連携させることにより誰が情報アクセスしているのかを判別不能にする技術が研究されている〔20〕。逆にこのような技術は不正利用されれば様々な社会問題を引き起こすことも考えられ、情報セキュリティの問題とも関連させてゆく必要がある。

参考文献

〔1〕 Mihai Barbuceanu and Mark S. Fox, Capturing and Modeling Coordination Knowledge for Multi-agent Systems, International Journal on Cooperative Information Systems, Vol. 5, No. 2-3, pp. 275-314 (1996).

〔2〕 Michael R. Genesereth: An Agent-Based Framework for Interoperability. In Jeffrey M. Bradshaw (ed.) Software Agents, AAAI Press (1997).

〔3〕 Tim Finin, Yannis Labrou, and James Mayfield: KQML as an Agent Communication Language. In Jeffrey M. Bradshaw (ed.) Software Agents, AAAI Press (1997).

〔4〕 http://fIPA.comtec.co.jp/

〔5〕 M. Huhns and M.P. Singh: Agents and Multiagent Systems: Themes, Approaches, and Challenges, Readings in Agents, Morgan Kaufman, pp. 1-23 (1998).

〔6〕 M. Gruninger and M.S. Fox, The Logic of Enterprise Modelling, Modelling and Methodologies for Enterprise Integration, pp. 140-157, Chapman & Hall (1996).

〔7〕 R.G. Smith: The contract net protocol: high-level communication and control in a distributed problem solver, IEEE Transactions on Computers, Vol. C-29, No. 12, pp. 1104-1113 (1980).

〔8〕 H. Van Dyke Parunak: Manufacturing Experience with the Contract Net, Distributed Artificial Intelligence, Morgan Kaufman Publishers, Inc., Los Altos, California, pp. 285-310 (1987).

〔9〕 横尾真他：分散制約充足による分散協調問題解決の定式化とその解法，電子情報通信学会論文誌，Vol. J75-D-I, No. 8, pp. 704-713 (1992).

〔10〕 M.N. Huhns and D.M. Bridgeland: Multiagent Truth Maintenance, IEEE Transactions on System, Man, and Cybernetics, Vol. 21, No. 6, pp. 1437-1445 (1991).

〔11〕 S.E. Conry et al.: Multistage Negotiation for Distributed Constraint Satisfaction, IEEE Transactions on System, Man, and Cybernetics, Vol. 21, No. 6, pp. 1462-1477 (1991).

〔12〕 E.H. Durfee and T.A. Montgomery: Coordination as Distributed Search in a Hierarchcal Behavior Space, IEEE Transactions on System, Man, and Cybernetics, Vol. 21, No. 6, pp. 1363-1378 (1991).

〔13〕 Toru Ishida, Les Gasser and Makoto Yokoo, ``Organization Self-Design of Distributed Production Systems,'' IEEE Transactions on Knowledge and Data Engineering, Vol. 4, No. 2, pp. 123-134 (1992).

〔14〕 Edmund H. Durfee and Victor R. Lesser: Partial Global Planning: A Coordination Framework for Distributed Hypothesis Formation, IEEE Transactions on Systems, Man, and Cybernetics, Vol. 21, No. 5, pp. 1167--1183 (1991).

〔15〕 横尾真:インターネットオークションの理論と応用, 人工知能学会誌, Vol. 15, No. 3 (2000).

〔16〕 Jeffrey S. Rosenschein and Gilad Zlotkin: Rules of Encounter, MIT Press (1994).

〔17〕 Michael P. Wellman: A Market-Oriented Programming Environment and ITS Application to Distributed Multicommodity Flow Problems. Journal of Artificial Intelligence Research, Vol. 1, pp. 1-23 (1993).

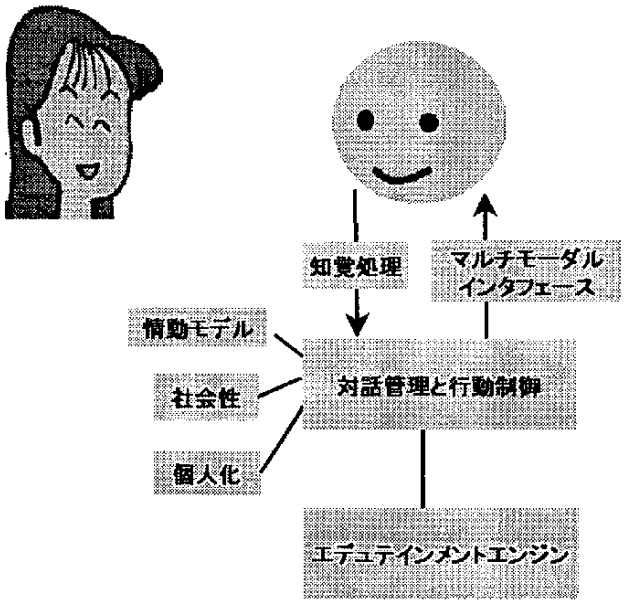
〔18〕 Balabanovic, M. and Shoham, Y.: Fab: content-based, collaborative recommendation, Communications of the ACM, Vol. 40, No. 3, pp. 66-72 (1997).

〔19〕 Eran Gabber, Phillip B. Gibbons, David M. Kristol, Yossi Matias, and Alain Mayer: Consistent, Yet Anonymous, Web Access with LPWA. Communications of the ACM, Vol. 42, No. 2, pp. 42-47 (1999).

〔20〕 Michael K. Reiter and Aviel D. Rubin: Anonymous Web Transactions with Crowds. Communications of the ACM, Vol. 42, No. 2, pp. 32-38 (1999).

3.3 インタフェースエージェント

ここでは，エデュティントメント（教育+エンターティンメント）を事例として、インタフェースエージェントに関する技術の現状について議論したい。参照モデルとして図表3.3-1のようなインタフェースエージェントを考える。

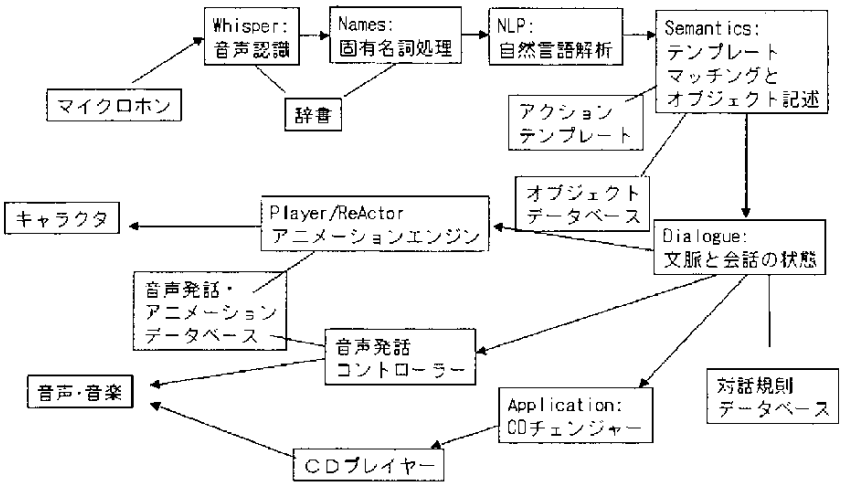


図表3.3-1　インタフェースエージェントのエデュテインメントへの適用

このモデルでは、インタフェースエージェントは、人間とエデュティンメントを行うエデュテインメントエンジンの間に介在し、対話管理と行動制御を中心に、知覚処理、マルチモーダルインタフェース、情動モデル、社会性に関する知識ベース、個人化のためのデータベースから構成される。場合によっては，学習コンポーネントも組み込まれる。

図表3.3-2はマイクロソフト研究所で開発されたPEEDYというシステムのアーキテクチャであり，ほぼ図表3.3-1に対応している。

基本的には、人間や環境に関する知覚入力によってエデュテインメントエンジンを駆動し、エデュテインメントエンジンの出力を言語的なメディア(発話)と非言語的なメディア(表情、身ぶり、手振りなど）に変換して伝える。この他にエデュテインメントでは、教材



図表3.3-2　典型的なインタフェースエージェントの構成[Ball 1997]

の認識や提示のための機構が必要である。

基本的な技術には次のようなものがある。

●個人化
- プロファイルを利用したインタラクション
- 適応学習

●心のモデル
- 感情モデル
- BDI
- 社会的インタラクション：代理性の生成など
- アイデンティティ

●入出力
- 対話制御（注意を引く、割り込む、黙っている、自発的行動）
- 認識（音声言語、顔情報認識、ジェスチャ認識、生理情報、環境情報）
- アニメーション(音声、表情、ジェスチャの合成)

●デザインメソッド
- 社会的文脈への組み込み

残念ながら、現時点でのインタフェースエージェントの技術では、図表 3.3-1 のシステムを素朴に作ってもうまく動かない。これまで構築されているシステムは、うまく動かないところを巧みなデザインでかわして、実用化にこぎつけている。

インタフェースエージェントの技術は、個人化と擬人化に大別される。個人化は、ユーザ個人のプロファイルを利用して個人に特化したサービスを提供することである。擬人化は、人間や動物のキャラクタを画面上に表示して、マルチモーダル対話を可能にする技術である。

これまでのところ、擬人化に対する批判が根強く残っている。第一の原因は、ユーザから見ると擬人化されたエージェントの背後には高い知能が控えていることが予測されるにもかかわらず、まだそのような高度の人工知能が実現されていないことである。第二の原因が、インタフェースだけ見てもまだまだ音声会話がぎこちなく、認識ミスや意図伝達失敗の発生や、長すぎる認識時間などである。

状況認識についても同様の傾向がある。エージェントは自分の周囲の状況を認識できることが要請されることが多い。例えば、エデュテインメントでも、学習者や学習者のいる環境の状況をエージェントが察知して、行動を調整することが望まれる(例えば、学習者がつかれたり、飽きたりしてきたらエンターティンメントに切り替える)。しかし、現段階では状況認識はまだまだ途上の技術であり、実世界に対しては不完全である。そのため、実世界にタグを埋め込んだり、人間がエージェントに状況を教えるなどによって状況認識の困難さを迂回することが必要であるといえる。

現状としては、擬人性を強く主張しなければ、実用レベルに達していると言える。つまり、

●個人性だけを主張する。

●擬人的な存在としてよりもペットとして位置付ける。

●テレフォニーサービスのように限られたメディアだけにする。

などのアプローチがある。

換言すると、発話と状況理解・行為のプランニング・発話生成という「まともな」アプローチをとると種々のフェーズで困難が生じ、エージェントの不適切な振舞いや応答拒絶といった破滅的状況が起こる可能性が高い。

しかし、会話に関する種々の手法（たとえば、ELIZA で用いられていた、内容の浅い理解に基づく応答生成）の有効性はすでに確認されつつあり、Extempo 社の擬人化エージェントやゲームソフト「シーマン」などに採用され、効果をあげはじめている。また、擬人化のレベルもかなり「ビリーバビリティ」（人間にとって、相手が人工システムであると言う違和感を忘れさせて、会話に引き込む）が高まってきている。インタフェースエージェントを構築するためのツールキットも広く利用可能になってきているので、キラーアプリケーションが生まれ、インタフェースエージェント作りに取り組む人たちの層が厚くなれば、急速にブレークスルーが生まれることも考えられる。

3.4 標準化

エージェントにおいて標準化のもつ意味は非常に大きい。標準化で目指しているものは、異なるベンダーが構成したエージェントシステムの相互運用性である。現在の情勢では、何らかの標準化にのっとらないソフトウェア製品はほとんど考えられないものである。例えば、電子商取引というアプリケーション領域を考えてみただけでも世界中のエージェントを簡単に接続できるようにすることによってもたらされるメリットは計り知れないものがある。

エージェントは知的なソフトであるから本来は標準化などしなくても、互いに通信可能な状態にしておけば自力で共通言語を作り出して、インタラクションができるようにすべきであるという意見もある。これは極めて妥当な意見であり、そうなってこそエージェントの真価が発揮されるものである。事実、このような言語の創出を目指した取り組みも研究では行なわれている。しかし、ソフトウェアシステムとしての現在のエージェントの実力を見る限りにおいてまだその域には達してはいない。現状では、標準化という作業はエージェントに活動の場を与えるという意味でも不可欠なものである。

エージェント標準化に関する取り組みは、FIPA(Foundation for Intelligent Physical Agents)（本編 2.4.1 節参照）、 OMG(Object Management Group)における ADSS DSIG(本編 2.4.2 節参照)、OMG Agent Working Group などで行なわれている。こうした活動は参加企業のメンバーのボランティアに支えられた NPO(Non-Profit Organization)という形で進行している。わが国企業からの発言も貢献もかなり大きい。FIPA には、日本から Director(役員)として常時複数名参加しているし、OMG ADSS DSIG は日本企業が中心的役割を果たしている。これらの活動は世界からの関心も高く、参加はしなくても動静を見守っているウォッチャーの数は多い。

現状では、エージェント標準化への取り組みで一番進んでいるものは、1997 年 10 月付けの仕様案(FIPA97)を公表した FIPA であるように思われる。しかしながら、その後提出された仕様案は FIPA97 と整合性が取れたものになっていなかったため、FIPA は仕様案の再

検討と仕様策定のプロセスの再構成を強いられ、表面上では一歩後退することになった。2000年1月のロンドン会議で新しい仕様策定プロセスに向けて大筋合意が取られ、4月から新方式による成果が出始めることが期待されている。ADSS DSIGは1997年11月にWhite Paper、OMGのAgent Working Groupは1999年12月24日にGreen Paper(version 0.9)を出したところであるが、これらの活動の間の人の行き来も活発であるので、FIPAの経験が別の標準化に直接生かされる可能性も高い。

エージェント技術がまだ若いものであるだけに、標準化の活動はまだ不安定な感がある。エージェント標準化の重要性を信じ、積極的に貢献する人や企業は存在するが、まだまだ十分な集積度には達していないように思われる。そのため、これらの標準化活動において、問題認識が共有され、標準化項目はたくさんあげられたが、なかなか成果に結びついていないのが現状であると考えられる。これは、エージェント技術を使った大型製品ないしはキラーアプリケーションがまだ出現していないのに対応しているように思われる。

エージェントの電子商取引への応用は大変魅力的である。それを現実化するためには、標準化において決済や信用・信頼性・安全性の確保まで含めた取り組みをきちんとすることが急務であろう。現在の標準化で成果になりそうなところは、まだエージェントシステムの枠組みと相互運用性やエージェント通信言語にとどまっており、これらだけでは本格的なビジネス応用は困難であろう。しかし、標準化活動における信用・信頼性・安全性への取り組みはまだ本格化していない。このあたりが、ビジネス応用におけるボトルネックになっているように思われる。

そうしてみると、電子商取引において信用・信頼性・安全性の問題が生じてこない領域や、既存のネットワークセキュリティ技術などで間に合わせることのできる領域などがもう一つの可能性として浮上する。例えば、生産加工の領域では、ネットワークはかなり独立性の高いものであり、セキュリティ的な不安要素が少ないため、標準化活動をうまく進めることによって面白い展開が期待できそうな領域であるといえる。

標準化活動の展開を妨げる大きな要因となっているものは人材不足である。特に、アカデミアからの貢献が少ないように思われる。よく見れば、標準化活動はアカデミアの研究者にとっても大変魅力が大きい。第一に、標準化活動に貢献することによって、実問題と接することができる。第二に、標準化活動でなかなか手が出せないフィールドトライアルやコンフォーマンステスト(与えられた製品が標準化に適合していることの検証)などは、アカデミアにとって格好の研究材料になり得る、第三に、標準化は最先端の研究領域よりも後を走るものであるものの、研究者の研究成果を標準のなかに残し、定着させる機会を与える。これらの問題については、アカデミアの側の意識改革が必要である。

標準化活動への貢献を促すためには、精神的な要因だけでは不十分である。FIPAの場合では、年4回の会議に加えて、その間のInterim会議、メール討論など相当の労力を集中的に注ぎ込まないと、標準化への貢献をものにすることは困難である。これからエージェント標準化が本格化すると、研究開発者が片手間で標準化活動に参加するだけではほとんど意味がないであろう。FIPAなどでは、エージェントの標準化が進展するにつれて、当該技術だけではなく標準化活動に専門的な技能をもつ専門家が強い発言力をもつようになってきている。まじめに参加していると言うだけでは通用しない。標準化の専門家を養成し、その活動をさまざまな側面から支援する体制を確立することが急務である。

3.5 ネットワークエージェント開発方法論

3.5.1 設計開発支援の現状

現在、様々なエージェントの設計開発が進められている。そこで利用される言語や開発環境はもちろんのこと、その設計手順や手法も多種多様である。これは、各々が扱う問題や処理の違いを反映しているのみならず、開発対象となるエージェントそのものの概念や機能が異なることから、エージェントやエージェント指向アプリケーションの設計手法も独自のものを採用する結果となっていることを示している。

一般に、エージェント／エージェント指向アプリケーションの設計開発を系統的に進めるための標準的な手法や方法論の確立が望まれるが、現時点では、そのための研究が徐々に立ち上がりつつある段階といえる。例えば、ここ５年間の関連研究の動向を、エージェント関係の国際会議などの論文から拾ってみると、次のような傾向が見られる。

・1995 前後

エージェント関連技術の開発と応用が活発化してきた時期で、エージェントシステムの仕様化手法などに関する研究〔1、2〕が僅かに散見される程度であった。

・1996～1997 年

BDI エージェントのモデル化手法〔3〕、エージェント指向分析・設計法〔4〕、マルチエージェント設計法〔5〕、マルチエージェント設計のデザインパターン〔6〕、エージェント指向ソフトウェア工学〔7〕などの研究がみられるようになった。

・1998～1999 年

エージェント設計のメタモデル〔8〕、合成概念に基づく設計・検証手法〔9、10〕、デザインパターン〔11、12〕、エージェント指向開発に関する考察〔13〕やエージェント指向分析・設計法〔14〕などの研究が現れてきている。また、これらの研究と並行して、エージェント言語／処理系、開発環境（フレームワーク）などに関する研究開発が、種々のアプリケーション構築と同時に推進されている（本節では、これらに関する議論は割愛する）。

上述した研究では、特にネットワークエージェントに焦点をあてている訳ではなく、一般的な立場から、エージェント／エージェント指向システムの設計法／設計技法について検討を進めている。その意味で、ネットワークエージェントを対象とした具体的な設計法や設計支援環境を与えることは今後の課題といえるが、その芽は着実に育ってきている。例えば、最近では、モバイルエージェント（ネットワークエージェント）の処理系やアプリケーションに関する研究開発事例が徐々に蓄積されつつあり、これらをもとに、ネットワーク環境上で稼動するエージェントやそのアプリケーションを構築する新たな手法の出現も期待できる状況となっている。

以下、本節では、ネットワークエージェント開発方法論の確立にむけた議論の参考として、ネットワーク分野におけるエージェント指向アプリケーションの構築事例を概観する。

3.5.2 アプリケーション事例にみるネットワークエージェントの設計開発

ここで取り上げる事例は、データ転送の量や品質が貧弱な狭帯域ネットワーク環境において、画像、音声などのメディアを用いた利用者指向のビデオ会議サービスの提供を目的として設計された「やわらかいビデオ会議システム」である(15、16)。このシステムでは、

a) 利用者からの要求（利用機能や要求品質など）が時間的に変化する、

b) システムの動作環境（ネットワークやプラットフォームの特性／性能）が動的に変動する、

という状況のもとで、システムがこれらの変動に対して能動的に対処することにより、利用者からの要求機能／要求品質を最大限に維持する機能の実現を目指している。

本システムの設計開発は概ね次のような手順に従って行われている。

1) 要求抽出／仕様定義

本システムのサービスの機能や特性に関する要求仕様について、特に、利用者から投入されるサービス品質(QoS)とその設定パラメータなどを中心として収集・分析・整理する。また、利用者サイドのQoSを、システムが処理するレベルでのQoSにマッピングし、基盤となるネットワークを含むシステム全体の設計条件／実装条件を分析・整理し、その設計仕様を定義した。

2) システムアーキテクチャ設計（エージェントとその組織構成の設計）

設計目標となるシステムをエージェント指向システムとして実現するためのアーキテクチャを設計する。具体的には、設計仕様に基づいて、エージェントとして実現する機能の切り出し、エージェントの相互関係と組織構造、組織構成／再構成の方法などが定義される。更に、システムを構成するエージェントのモデル、エージェントに組み込んで制御する（エージェント化される）マルチメディア通信ソフトウェア／アプリケーション（主として既存のもの）の選定、構築ツール（エージェントフレームワーク）なども併せて選択・決定される。

3) 機能設計／知識獲得（エージェント設計、ドメイン指向協調プロトコル設計）

次の段階として、各種のエージェントが設計される。本システムでは、エージェント指向分散システム構築用フレームワーク〔17〕を利用するため、これらのエージェントの設計は、同フレームワークが提供するエージェントモデルに基づいて、その動作知識を獲得・記述することにより行われる。また、フレームワークが提供するエージェント通信／協調のプロトコルでは対処できないドメイン指向の協調プロトコル（例えば、ビデオ会議サービスの統括・管理のために利用者毎に定義されるシステム管理エージェント相互間での協調動作、或いは、利用者のプラットフォーム上でのシステム管理エージェントと状況監視エージェントとの協調動作などで利用される専用プロトコル）を定義するための知識の収集・獲得とプロトコル設計なども行われる。

4) プロトタイピングとチューニング（エージェント実装）

機能設計に基づいて、実際に種々のエージェントを実装し、システムを試作する。本システムの場合、エージェントの設計結果（エージェントプログラム）をフレームワークが提供するリポジトリに登録する、また、エージェント組織を形成して動作させる環境を利用者のプラットフォーム上に生成することにより、システムが実装される。その後、各エ

ージェントの個別動作や、利用要求に則したエージェント組織（ビデオ会議サービス）の構成／再構成動作の実験を通して、エージェント動作や協調プロトコル処理の中核となっているエージェント知識をチューニングする。

5) 試験運用と実験・評価・改良

実際に利用者からビデオ会議サービスの利用要求（QoS 条件を含む）を投入し、ネットワーク環境上でビデオ会議を実行し、システム全体としての動作実験を行う。また、システム動作中に新たな利用者要求を投入したり、ネットワークやプラットフォームの動作条件を変更したりするなどして、利用者の視点からやわらかいビデオ会議サービスの機能と動作特性を評価する。更に、その結果に基づいて、エージェントの機能／知識、システム構成の改良を行う。

上述したように、このシステムでは、設計開発の基盤として、既存のエージェントフレームワークを利用しており、スクラッチからエージェントを組み立てる方法はとっていない。これまでの経験によれば、フレームワークそのものの機能が不充分な点は否めないが、本システムを含む小～中規模のエージェント指向アプリケーションの構築を系統的かつ効率的に進める上で、エージェントフレームワークの利用が極めて有用であることを確認している。

本節で述べたように、現在、エージェントフレームワークを含むエージェント／エージェント指向アプリケーションの開発ツールや環境に関する研究開発が増えつつある。これらをもとに、今後、ネットワークエージェントやそのアプリケーションの構築を系統的に支援する設計方法論や設計技法が確立される可能性は高まっている。モバイルエージェントなどの標準化の動きなども視野に入れつつ、精力的に研究開発を推進することが重要であろう。

参考文献

〔1〕 F. Brazier, B. Dunin-Keplicz, N. R. Jennings, J. Treur: "Formal Specification of Multi-Agent Systems: a Real-World Case", in: Proc. of the 1st Int. Conf. on Multi-Agent Systems (ICMAS'95), AAAI Press/MIT Press, pp. 25 ～ 32, (1995).

〔2〕 B. Dunin-Keplicz, J. Treur: "Compositional Formal Specification of Multi-Agent Systems", in: M. J. Wooldridge, N. R. Jennings (eds.), Intelligent Agents (ATAL'94), LNAI 890, Springer-Verlag, pp. 102 ～ 117, (1995).

〔3〕 D. Kinny, M. Georgeff, A. Rao: "A Methodology and Modelling Technique for Systems of BDI Agents", in: W. Van de Velde, J. W. Perram (eds.), Agents Breaking Away (MAAMAW'96), LNAI 1038, Springer-Verlag, pp. 56 ～ 71, (1996).

〔4〕 B. Burmeister: "Models and Methodology for Agent-Oriented Analysis and Design", in: K. Fischer (ed.): Working Notes of the KI'96 Workshop on Agent-Oriented Programming and Distributed Systems, DFKI Document D-96-06, (1996).

〔5〕 N. Glaser: "The CoMoMAS Methodology and Environment for Multi-Agent System Development", in: C. Zhang, D. Lukose (eds.), Multi-Agent Systems * Methodologies and Applications, LNAI 1286, Springer-Verlag, pp. 1 ~ 16, (1997).

〔6〕 E.A. Kendall, M.T. Malkoun: "Design Patterns for the Development of Multi-Agent Systems", in: C. Zhang, D. Lukose (eds.), Multi-Agent Systems * Methodologies and Applications, LNAI 1286, Springer-Verlag, pp. 17 ~ 31, (1997).

〔7〕 M.J. Wooldridge: "Issues in Agent-Based Software Engineering", in: P. Kandzia, M. Klusch (eds.), Cooperative Information Agents (CIA97), LNAI 1202, Springer-Verlag, pp. 1 ~ 18, (1997).

〔8〕 J. Ferber, O. Gutknecht: "A Meta-Model for the Analysis and Design of Organizations in Multi-Agent Systems", in: Proc. of the 3rd Int. Conf. on Multi-Agent Systems (ICMAS'98), IEEE Computer Society, pp. 128 ~ 135, (1998).

〔9〕 F. Brazier, F. Cornelissen, R. Gustavsson, C. Jonker, O. Lindeberg, B. Polak, J. Treur: "Compositional Design and Verification of a Multi-Agent System for One-to-Many Negotiation", in: Proc. of the 3rd Int. Conf. on Multi-Agent Systems (ICMAS'98), IEEE Computer Societ, pp. 49 ~ 56, (1998).

〔10〕 J. Engelfriet, C.M. Jonker, J. Treur: "Compositional verification of multi-agent systems in temporal multi-epistemic logic", in: J.P. Muller, M.P. Singh, A.S. Rao (eds.), Pre-Proc. of the 5th Int. Workshop on Agent Theories, Architectures, and Languages, pp. 91 ~ 105, (1998).

〔11〕 Yariv Adior, Danny B. Lange, "Agent Design Patterns: Elements of Agent Application Design", Proc. Autonomous Agents 98, (1998).

〔12〕 Y. Tahara, A. Ohsuga, S. Honiden: "Agent System Development Method Based on Agent Patterns", in Proc. ISADS99, pp. 261-263, IEEE, (1999).

〔13〕 M. Wooldridge, N. Jennings: "Pitfalls of Agent-oriented Development", in Proc. Autonomous Agent '98, pp. 385-391, ACM, (1998).

〔14〕 M. Wooldridge, N. Jennings, D. Kinny: "A Methodology for Agent-Oriented Analysis and Design", in Proc. Autonomous Agents '99, pp. 69-76, ACM, (1999).

〔15〕 菅沼拓夫, 藤田茂, 菅原研次, 木下哲男, 白鳥則郎, "マルチエージェントに基づくやわらかいビデオ会議システムの設計と実装", 情処学論, 38(6), pp. 1214-1224, (1997).

〔16〕 Suganuma T., Kinoshita T., Sugawara K., Shiratori N., "Flexible Videoconference System based on ADIPS Framework", Proc. Third Int. Conf. on Practical Application of Intelligent Agent and Multi-agent Technology (PAAM98), pp. 83-98, (1998).

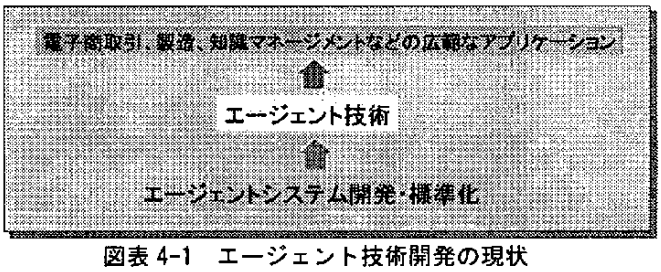
〔17〕 Kinoshita T., Sugawara K., "ADIPS Framework for Flexible Distributed Systems", in T. Ishida (ed.) "Multiagent Platforms", Lecture Notes in Artificial Intelligence 1599, Springer-Verlag, pp. 18-32, (1999).

Ⅱ．本編

4．まとめ

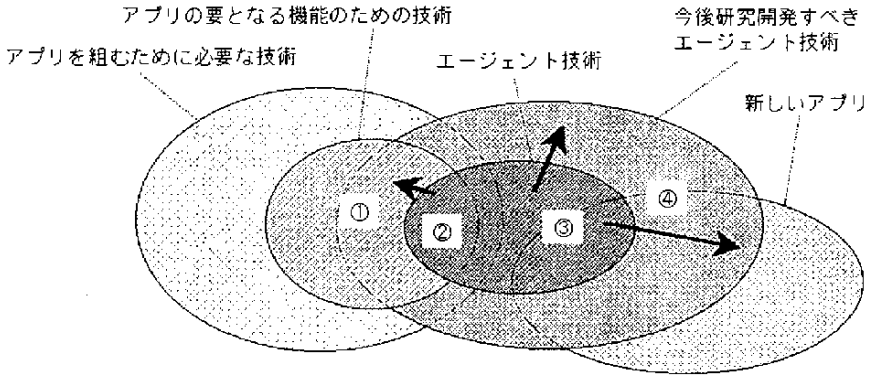
4. まとめ

本調査の結果、エージェント技術について、図表 4-1 のように、エージェント技術を本格適用したアプリケーション、エージェント技術のコア、エージェントシステムを構築するためのプラットフォームと標準化という縦の線が明確になったと言える。



図表 4-1　エージェント技術開発の現状

現在の技術マップは俯瞰するために図表 4-2 のような図式を描いてみた。



①既存アプリの要として使われているエージェント技術
②既存アプリの要となるために今後開発すべきエージェント技術
③新規アプリの要となりそうな既存エージェント技術
④今後開発すべきエージェント技術

図表 4-2　現在のエージェント技術の状況

①は既存アプリの要として使われているエージェント技術である。②は、既存アプリの要となるために今後開発すべきエージェント技術である。③は、新規アプリの要となりそうな既存エージェント技術である。④は今後開発すべきエージェント技術である。

①の既存アプリの要として使われているエージェント技術としては、次のものを挙げることができる。

第一に、エージェントのコンセプトとアーキテクチャである。これは技術そのものとは言えないが、技術開発を進めるために不可欠である。 CORBA や Java のアーキテクチャはエージェントシステム構築基盤としてエージェント技術と密接な関わりがある。

第二に、エージェント通信言語をあげることができる。エージェント通信言語は、ブロ

ーカリング、ファシリテーション、マッチメーキングなどの付加的な機能によってエージェントらしさをかもし出している。内容記述のための技術としてXMLが、内容目録定義技術としてオントロジー技術が密接な関わり合いがある。さらに進んだ技術として、エージェント間の交渉やオークションを実現する研究開発が急速に巣寸手要る。

第三に、モバイルエージェント技術がある。モバイルエージェントはエージェント通信とよばれる技術をベースにしており、最初に述べたエージェント一般のコンセプトとアーキテクチャにエージェント移動（マイグレーション）のための技術を付加した独自のコンセプトとアーキテクチャをもっている。行動に知的な側面をもたせるために、プランニングなど人工知能技術との関わりも深い。

第四に、インタフェースエージェント技術がある。インタフェースエージェント技術は、間接操作を実現するインタフェース技術であり、個人化と擬人化に特色がある。

②の既存アプリの要となるために今後開発すべきエージェント技術としては、次のようなものがある。

●QoS保証
- ・性能保証
- ・信用保証

●セキュリティとプライバシー
- ・個人情報の活用：個人化のメリットを情報発信や交渉に生かす

●オントロジー技術との融合
- ・個人オントロジー vs グループオントロジー／グローバルオントロジー

●ビリーバブルエージェント
- ・不信を払拭するレベルの実現

●実世界認識との融合
- ・環境の認識
- ・ユーザの生理的状態・心的状態・個人プロファイルの認識

●複雑なインタラクションの実現
- ・交渉（例：電子商取引における）
- ・感情的なインタラクション

●エージェントシステム開発方法論
- ・エージェントシステムのアプリを開発するための考え方

③は、新規アプリの要となりそうな既存エージェント技術である。マルチエージェント技術は、研究の歴史も長く、次のような項目についてかなりの蓄積がある。

●マーケット指向プログラミング

●交渉アルゴリズム
- ・オークション

●マルチエージェント学習

●協調・分散アルゴリズム

しかし、現状の開発状況を見る限り、上記の成果は十分に取り入れられているとは言えない。マルチエージェント技術を取り込むことによってエージェントらしいアプリケーシ

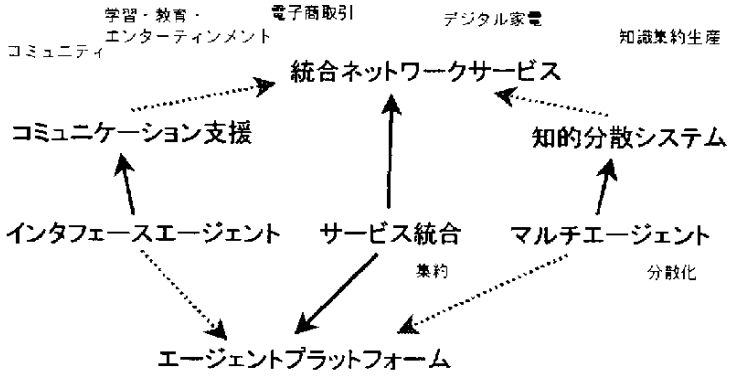
ョンが期待される。

④の今後開発すべきエージェント技術については、エージェントを用いた人間のコミュニケーションの支援が考えられる。社会的エージェントと呼ばれる、人間の社会に入って会話ができるエージェントについての研究開発とその成果の利用、また、コミュニティ支援や協同学習支援という文脈におけるエージェント技術の展開などが期待される。

これによる将来のエージェントテクノロジーの新しい応用として次のものが考えられる。

- ●代行サービス、派遣サービス、業務支援エージェント（早期）。人間の知的作業を肩代わり。税金対策や節税アドバイザ、旅行アドバイザ、新製品ウォッチャ、放送番組ウォッチャ、機器の監視、ライバルウォッチャ、消費者ウォッチャ、ユーザサポート、出張サービス、出張広告、業務知識伝承、イベント企画・事務代行、バーチャル窓口、秘書エージェントなど。
- ●仲介サービス（早期）。ニーズ側とシーズ側を引き合わせる。商品紹介、書籍紹介、不動産仲介、就職斡旋、結婚相手紹介など。また、電話自動取り次ぎ（発信者と受信者の都合の良い時間をみつけて回線を接続する）なども含まれる。エージェント単体よりも専門会社の業務支援として使われるのではないか。
- ●キャラクタ（中期以降、ただし単純なものは早期から）。誰かのイミテーションや自分で育てる。それ自体におもしろさ。話し相手エージェント、英会話の先生エージェント、バーチャルタレント、有名人キャラクタなど。
- ●介護エージェント、ケアエージェント（中、後期）。ロボットとセンサー・アクチェータ埋め込み環境が一体化して、ネットワークに接続。高齢者世帯や、乳幼児がいる共働き世帯など。また、悪徳商法や詐欺から弱者を保護するエージェントも需要が高いだろう。

以上をまとめると、図表 4-3 のようになる。



図表 4-3　エージェント技術開発の展望

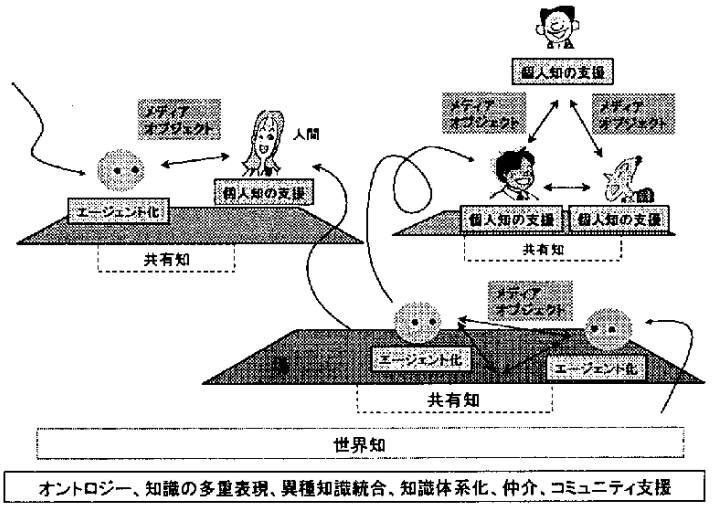
資料編 A. 研究開発への取り組み事例

A.1　東京大学・大学院工学系研究科･電子情報工学専攻・西田研究室

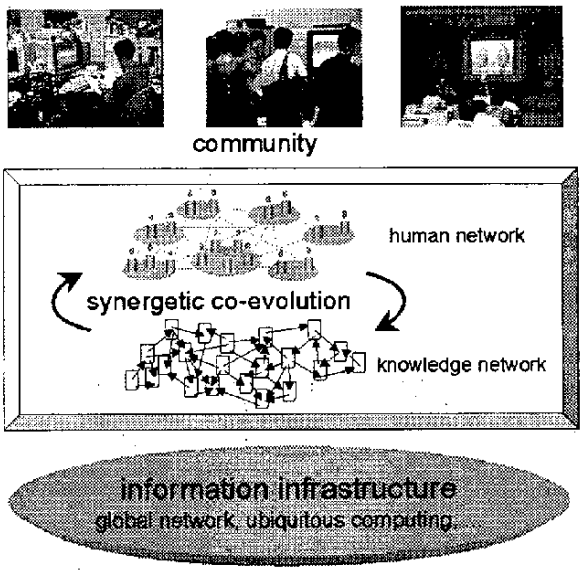
http://www.kc.t.u-tokyo.ac.jp

A.1.1　知識コミュニティの研究

仮想世界と物理世界での人々の知的活動（情報収集・整理・体系化・共有・創出）を支援する枠組み―知識コミュニティ（図表 A.1.1-1）―の実現を研究目標に掲げている。共有された場における人間同士、人間とエージェント（自律的に行動する人工システム）の間、あるいはエージェント同士のインタラクションによる個人知識とコミュニティ知識の相乗的発展をねらう（図表 A.1.1-2）。



図表 A.1.1-1　知識コミュニティの枠組み



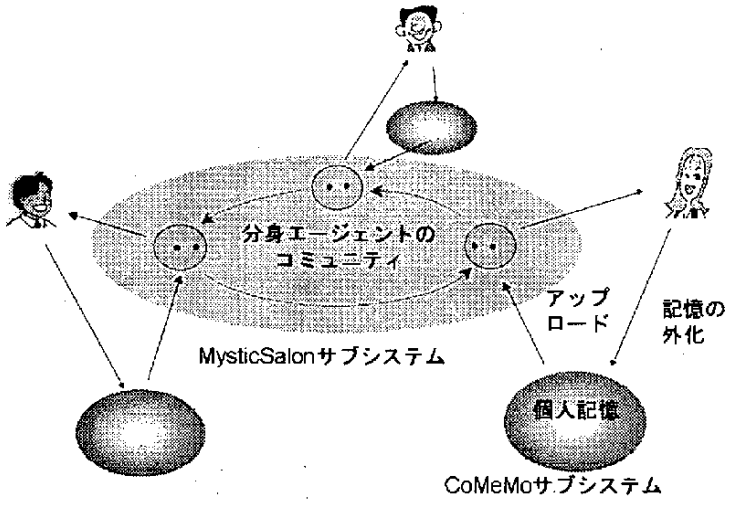
図表 A.1.1-2　ヒューマンネットワークと知識ネットワークが相乗的に発展する場としての知識コミュニティの位置付け

A.1.2 研究課題

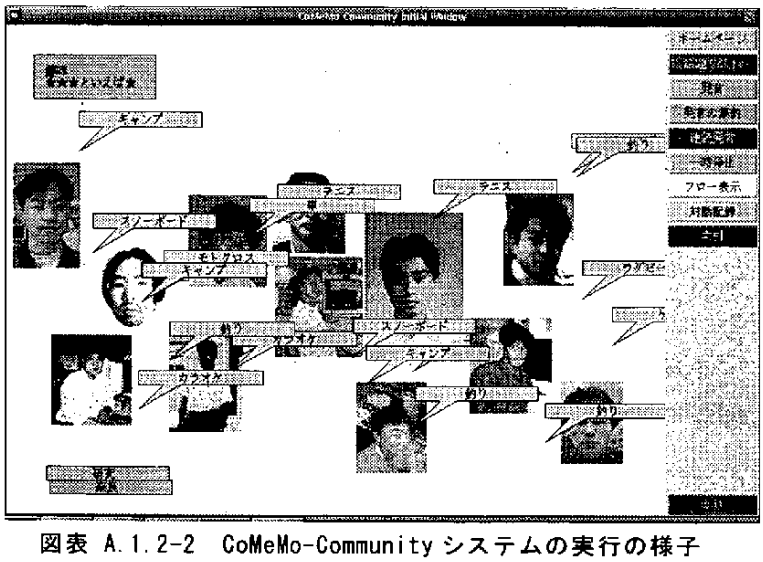
知識コミュニティを実現するためには、文科系・理科系をまたいだ多角的な視点からの学際的な研究と、実世界への適用を前提とした実証的研究が不可欠であると考えられる。現在の研究課題は次の通り。

(1) CoMeMo-Community

知識化される前段階の未組織・文脈依存の情報という意味での「記憶」の果たす役割は大きい。個人・コミュニティの日常記憶マネージメントによって、コミュニティにおけるメンバー間の相互理解と知識共有の支援をめざしたシステムの研究開発を進めている。ユーザの外化された記憶を保持し、ユーザに代わって情報発信する機能をもつ分身エージェントという概念を用いて知識の流通と共有を図る。コミュニティにおける各メンバーの公開記憶データの相互関連を分身エージェントの間の擬似的な会話として表示する。それを閲覧することにより、ユーザは自分の属するコミュニティの他のメンバーの関心や知識の概要と相互関係を把握できる（図表 A.1.2-1、図表 A.1.2-2)。



図表 A.1.2-1　CoMeMo-Community のアーキテクチャ

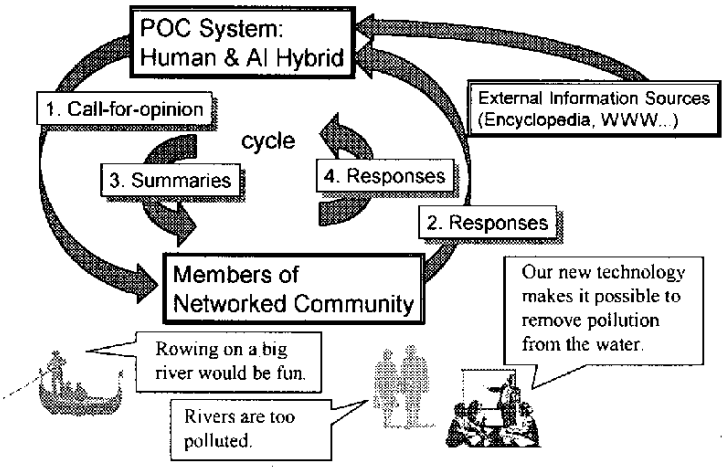


図表 A.1.2-2　CoMeMo-Community システムの実行の様子

(2) パブリック・オピニオンチャンネル

人々がコミュニティに向けて自分の意見を自由に発信できるデジタル・デモクラシーの実現を目指した新しい放送システムの概念である。コミュニティメンバーの意見を「物語」として編集し、オン・エアーする。人間のキャスターのアシストや掛け合いなどができる人工知能キャスターの実現もめざす。

研究面では、(a) 大量・不均質な意見の公平な集約メカニズムの実現、(b) 与えられた材料を視聴者が楽しみながら理解できるようにするための連続物語ストリーム構成メカニズムの実現、(c) 人間の大量な不均質情報の要約と理解のメカニズムの解明、(d) パブリックオピニオンチャンネルにより結合されたコミュニティにおける情報のループバックによる少数者影響の解明や情報操作の防止のための方策の解明など、人工知能、認知心理学、社会心理学において従来研究されてこなかった新規な研究フィールドの創出をめざす（図表 A.1.2-3)。

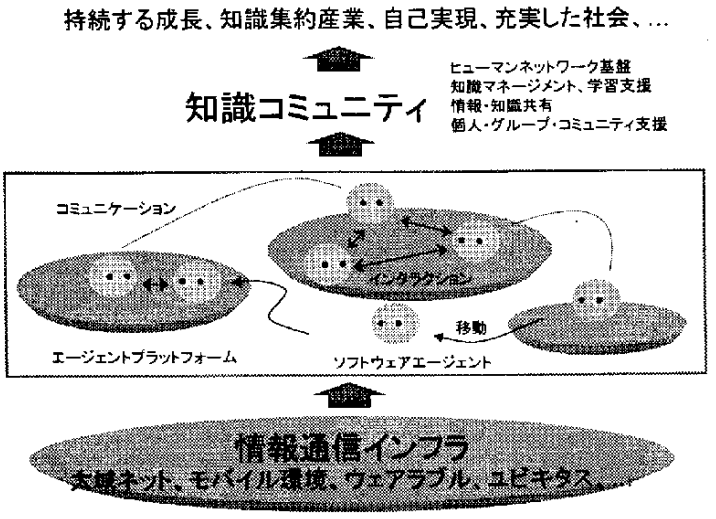


図表 A.1.2-3 パブリック・オピニオン・チャンネル

異なる分野からのアプローチを統合するための枠組として提案している〔Nishida 1999、 畦地 2000〕。コミュニティにおいて、人々からのメッセージをシームレスに集め、編集されたメッセージを人々にフィードバックする社会的情報処理システムである。

(3) エージェントプラットフォーム

ソフトウェアエージェント（ネットワーク上でユーザの委嘱を受けて作業を行う自律ソフトウェアコンポーネント）という概念を中心に構成されるミドルウェアである（図表 A.1.2-4)。知識コミュニティを情報ネットワーク基盤の上に構築するために必要となる。エージェントプラットフォームのデザインに取り組むとともに、相互運用性確立のための共通的枠組みの提言を行う。



図表 A.1.2-4　エージェントプラットフォーム

A.1.3　研究体制

郵政省情報通信ブレークスルー基礎研究 21 西田結集型プロジェクト、スタンフォード大学日本センター、NHK 放送技術研究所と連携する他、産学共同・国際共同研究を進めている。

主要発表文献

〔1〕 平田高志，村上晴美，西田豊明．連想表現を用いたコミュニティにおける知識の視覚化とその 評価実験．システム制御情報学会論文誌，12(7)，pp. 428-436，1999.

〔2〕 Toyoaki Nishida, Nobuhiko Fujihara, Shintaro Azechi, Kaoru Sumi, and Hiroyuki Yano: Public Opinion Channel for Communities in the Information Age, New Generation Computing", Vol. 14, No. 4, pp. 417-427, 1999

〔3〕 Toyoaki Nishida: Facilitating Community Knowledge Evolution by Talking Vitrualized Egos, In: Hans-Joerg Bullinger and Juegen Ziegler (eds.): Human-Computer Interaction VOLUME 2, Lawrence Erlbaum Associates, Pub., pp. 437-441, 1999

〔4〕 西田：分身エージェントに基づくコミュニティコミュニケーション支援，bit, Vol. 31, No. 7, pp. 103-108, 1999

〔5〕 西田，冨山，桐山，武田（編著）：工学知識のマネージメ ント，朝倉書店，1998.

連絡先:nishida@kc.t.u-tokyo.ac.jp

A.2 東北大学 電気通信研究所 情報通信システム研究分野

－やわらかい分散システムとエージェント技術－

木下哲男

A.2.1 概要

東北大学電気通信研究所の情報通信システム研究分野（白鳥則郎教授、木下哲男助教授）では、情報通信システムとそのアーキテクチャに関する基礎研究、及び、人間・機械系コミュニケーションと情報通信アプリケーションに関する先端的研究を担当している。９０年代初頭より、次世代情報処理の基本となる「Flexible Computing」の概念を掲げ、人間からみた人工システムの「やわらかさ」について考究し、人間と人工システムの共生に基づく新しい情報処理を実現するために様々な研究を進めている。そこでは、エージェント技術をFlexible Computingを支える重要な技術と捉え、エージェントモデル、マルチエージェント応用システム、その開発ツールとなるエージェントフレームワークなどに関する研究を精力的に推進している。

A.2.2 わらかい分散システムとやわらかいネットワーク

Flexible Computing コンセプトに基づくネットワークシステムを「やわらかいネットワーク」と呼び、その上で構成される分散システムを「やわらかい分散システム」と呼ぶ。これらのシステムでは、利用者の視点と設計者／システムの視点の両側面からみたシステムの「やわらかさ」を提供し、様々な利用要求とシステム動作環境に柔軟に対処でき、非専門家ユーザでも容易に安心して使える機能の実現を目指している。

やわらかさを実現するレベルは、(a) システムの構成要素、(b) 複数の構成要素によって構成される集団（サブシステム）、(c) 多数のシステムからなる社会などが考えられる。種々のレベルのやわらかさの基準や尺度は、利用要求とその変化、システム動作環境の変動、システムが提供するサービスの品質のゆらぎなど、着目するレベルの機能や特性に応じて定式化され、システムの機能や動作の観測・判断・制御の基本情報として利用される。

更に、こうしたやわらかさを実現するしくみの設計や応用システムの構築においては、自律的に組織を形成して、協調的に問題解決を実行するエージェント／マルチエージェントの枠組みが極めて適していることから、現在、幾つかの具体的な応用事例を対象として、エージェント技術を効果的に活用したやわらかい分散システム／やわらかいネットワークの設計方法論や構築技術に関する研究を進めている。

A.2.3 エージェント技術とその応用に関する研究

エージェント技術に基づくやわらかい分散システム／やわらかいネットワークの研究の概要を述べる。本研究の枠組みは、以下の４つのサブテーマからなる。

(1) エージェント指向コンピューティング基盤（エージェントフレームワーク）
(2) ネットワークコンピューティング基盤（やわらかいネットワーク）
(3) ソーシャルコンピューティング基盤（ソーシャルコンピューティングサービス）
(4) 人間－エージェント共生空間（やわらかい分散システム）

図表 A.2.3-1 に示すように、サブテーマ（2）から（4）では、現行のネットワーク環境上にそれぞれ新しい機能階層を実現し、人間・エージェント共生空間において、利用者指向のやわらかい分散システム（アプリケーションサービス）を提供する。これらの各階層は、様々なエージェントによって構成されるマルチエージェントシステムとその集合体であり、その設計開発を支援する環境がサブテーマ（1）のエージェント指向コンピューティング基盤として与えられる。以下、各サブテーマについて概説する。

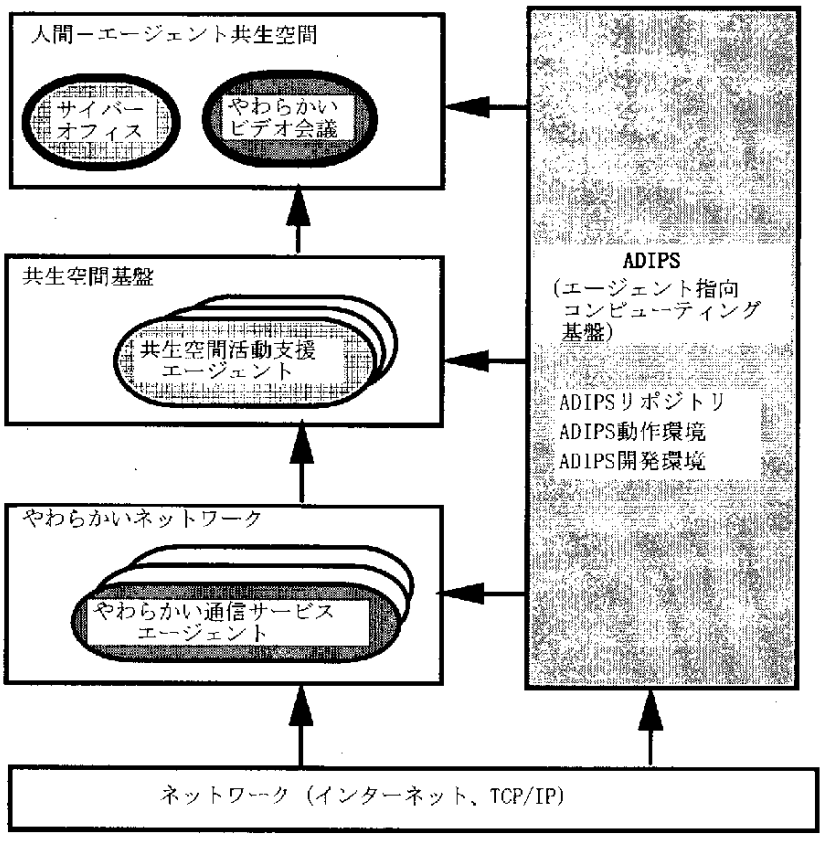
（1）エージェント指向コンピューティング基盤：エージェント／マルチエージェントシステムの設計開発支援環境としてのエージェントフレームワークについて研究する。現在、ADIPS(Agent-based Distributed Information Processing System)フレームワークと呼ばれる知的エージェント構築環境を開発している。本フレームワークでは、従来型ソフトウェアのエージェント化機能、要求指向のエージェント組織化機能、エージェント組織の再構成機能、エージェント動作モニタ機能などが提供され、エージェント指向のやわらかい分散システムの構築を効果的に支援する（図表 A.2.3-2)。

（2）ネットワークコンピューティング基盤：やわらかいネットワークのコンセプトに基づくエージェント指向ネットワーク環境について研究する。エージェントフレームワークに基づいて、現行のネットワークシステムが提供する種々のサービスをエージェント化し、上位層のエージェントとの連携動作を支援・促進することにより、アプリケーション主導のやわらかいネットワーク管理／ネットワークオペレーションを実現する。

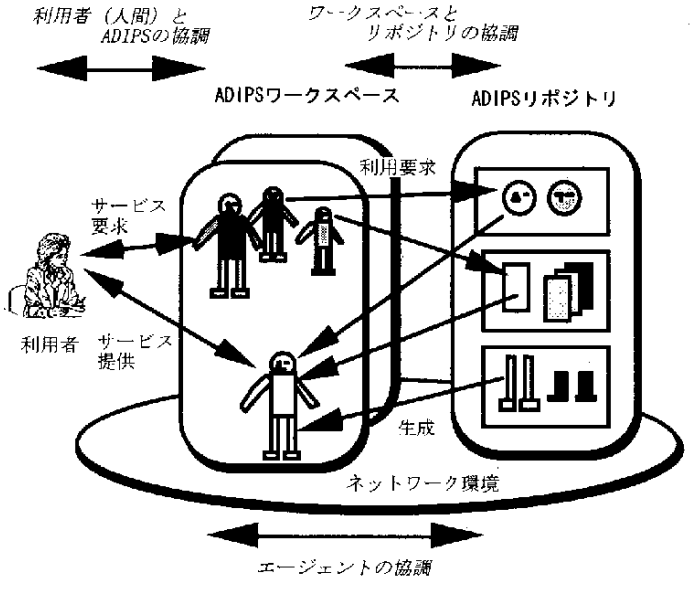
（3）ソーシャルコンピューティング基盤：ネットワーク環境／分散環境（人間－エージェント共生空間）における人々の多様な活動を支援する共通的サービスの構成法について研究する。人間－エージェント共生空間の基盤となる機能層であり、コミュニケーション／協調支援機能（ソーシャルウェア）や能動的情報資源構成機能などのサービスを提供する。

（4）人間－エージェント共生空間：やわらかい分散システムとして実現されるアプリケーションの構成法、及び、その利用環境について研究する。多様なエージェントによって提供されるアプリケーションが、広域分散環境上での人々の活動を支援することにより、Flexible Computing コンセプトに基づく人間とシステム（エージェント）の協調と共生の実現を目指す。これまで、利用者指向のやわらかい分散システムの応用として、利用要求や動作環境の変化に柔軟に対処するやわらかいビデオ会議システム、非同期メッセージングシステム、研究活動支援システム、サイバーオフィスなどのマルチエージェントアプリケーションを構築し、その有効性を検証している。

現在、上述した研究と並行して、エージェント技術を適用した未来開拓プロジェクト（文部省）やギガビットネットワークプロジェクト（郵政省）なども推進している。今後も継続して、広域ネットワーク環境をベースとしたエージェント技術とその応用に関する研究開発に幅広く取り組んでゆく予定である。



図表 A.2.3-1　研究の枠組み



図表 A.2.3-2　ADIPS フレームワーク

A.3　大阪市立大学工学部情報工学科知識情報工学研究室における取り組み

—WWW 情報収集エージェント MetaCommander—

北村泰彦

インターネットはわれわれの日常生活に不可欠に感じるほど、現在社会に浸透しつつある。WWW はその中でも最も広く普及している技術であるといえる。WWW は単なる個人情報の発信にとどまらず、ビジネスや研究活動、コミュニティ形成など様々な目的のために利用されている。現在の WWW は利用者がブラウザを用いてページ単位で情報閲覧することが基本的な利用法であった。しかしながら関連する情報が複数のサイトに分散して存在し、またその内容も非同期に更新されることから、関連情報をブラウザを介して手作業で収集、統合、更新することは結構骨の折れることである。このような問題に対して、MetaCommander〔1、2〕は利用者の代わりに複数の WWW サーバにアクセスし、自動的に情報を抽出、再合成してくれるシステムである。利用者はスクリプトを記述することにより、MetaCommander に命令を与えることができる。図表 A.3-1 に示すように、MetaCommander はスクリプトを解釈し、インターネットを介して情報を収集し、その結果を HTML 文書として出力する。利用者は WWW ブラウザを介してその結果を表示できる。スクリプトには図表 A.3-3 に示すような 40 以上のコマンドが利用可能であり、情報の検索、抽出、出力の整形、表示、保存、簡単な計算が可能である。

図表 A.3-2 は朝日新聞と読売新聞のサイトからトップ記事を収集する簡単なスクリプトの一例である。まず MetaCommander は朝日新聞のトップページを取り込む（3行目）。トップページは table タグを用いて整形されているが、まず3番目の table を取り出し（4、5行目）、さらにその中の4番目の table の部分を抽出している（6、7行目）。次に、読売新聞のトップページを取り出し（9行目）、その中の2番目の table の部分を抽出し（10、11行目）、先に抽出した朝日新聞の記事と統合している。

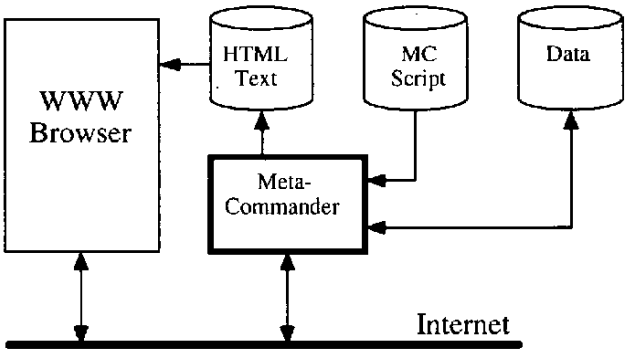
MetaCommander は Java で記述されており、http://www.kdel.info.eng.osaka-cu.ac.jp/~mc よりフリーソフトウェアとして一般に公開されており、これまで、電子新聞のトップ記事の収集〔2〕、バーチャルショップからの価格情報の収集〔3〕、生物データベースの統合〔1〕など様々な用途のために利用されている。

参考文献

〔1〕 北村，野崎，辰巳：スクリプトに基づく WWW 情報統合支援システムとゲノムデータベースへの応用，電子情報学会論文誌，J81-D-I(5):451-459 (1998).

〔2〕 北村，野崎：Java による WWW 情報収集支援ツール「MetaCommander」，JavaWorld, 2(3):72-77, 1998.

〔3〕 Yasuhiko Kitamura, Tomoya Noda, and Shoji Tatsumi. Single-agent and Multi-agent Approaches to WWW Information Integration, T. Ishida (Ed.), Multiagent Platforms, Lecture Notes in Artificial Intelligence, Vol. 1599, Berlin et al.: Springer-Verlag, 133-147, 1999.



図表 A.3-1　MetaCommander

```
1: tag("BODY"){
2: print("ASAHI")
3: getURL( "http://www.asahi.com") {
4:         searchTag("table") {
5:                 if($COUNT == 3) {
6:                         searchTag("table") {
7:                         if($COUNT == 4) { print }}}}}
8: print("YOMIURI")
9: getURL( "http://www.yomiuri.co.jp/main/main.htm") {
10:        searchTag("table") {
11:                if($COUNT == 2) {print}}}}
```

図表 A.3-2　スクリプトの一例

類別	関数名	機能
HTTP	getURL	GET でページの取得
	postURL	POST でページの取得
検索	searchString	文字列検索
	searchTag	タグによる検索
抽出	getAnchor	アンカーの抽出
	getString	文字列の抽出
	cutString	文字列による切りだし
整形	print	文字列や HTML の出力
	tag	タグの出力
入出力	file	ファイルへ保存
	open	ファイルオープン
	getline	1 行読み込み
	putline	1 行書き込み
変数	set	変数へ代入
	calc	演算処理
	strcat	文字列の追加
制御	if	条件分岐
	while	条件ループ
GUI	inputLine	1 行入力
	dialog	確認ダイアログ

図表 A.3-3　主な MetaCommander 関数

A.4　キヤノン(株)情報技術研究所におけるエージェント技術への取り組み

キヤノン(株)情報技術研究所
上田 隆也
takaya@cis.canon.co.jp

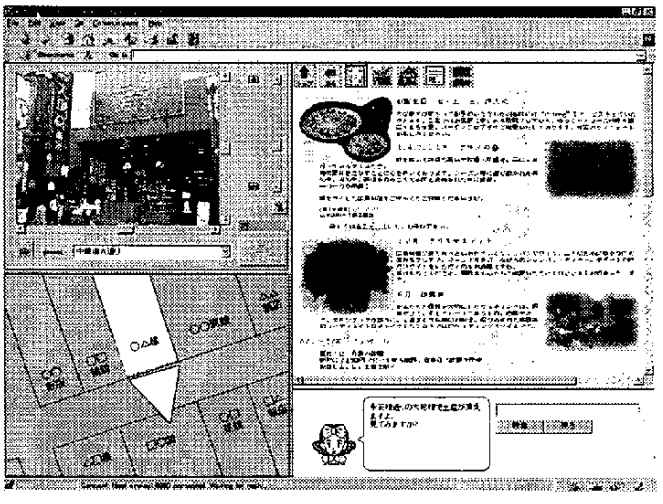
A.4.1　概要

キヤノン(株)情報技術研究所では、インターネット上の情報の新しい利用方法やそれを具現化するシステムに関する研究を進めており、その一つとして状況に応じて情報を能動的に提示する情報エージェントについて研究している。特に、今後モバイル/ウェアラブル機器上の情報エージェントが重要になるという考えから、実世界に密着した情報提示に重点を置いている。ここでは、その研究事例である実空間と情報空間の双方向ナビゲーションシステム Canopy(Context-aware navigation and operation yard)の概要を中心に、情報技術研究所のエージェント技術への取り組みを紹介する。

A.4.2　ナビゲーションシステム Canopy

Canopy は、インターネットカメラにより実空間の各地点を容易に移動できることを利用し、Web の情報空間と実空間とを速やかに行き来できるナビゲーションを実現したシステムである〔1〕。

図表 A.4.2-1 は Canopy のユーザインタフェースの表示例である。左上のフレームには、カメラアングルとズームが操作可能なインターネットカメラ(弊社製品 WebView Livescope[1])から送信される映像が表示される。左下のフレームには地図、右上のフレームには Web ページ[2]が表示される。右下のフレームでは、ユーザに対する能動的なナビゲーションや検索の際の対話を行なう。映像と地図が実空間を表現し、Web ページが情報空間を表現している。



図表 A.4.2-1　Canopy のユーザインタフェース

1 http://www.x-zone.canon.co.jp/WebView/参照。
2 http://www.daichin.co.jp/からの引用。

Canopy では下記の機能を実現している。

- ●実空間/情報空間のいずれかにおいてユーザが注目対象を別の建物に切り替えた場合、それに連動してもう一方の空間で、その建物が表示されるようにカメラ及びアングルを切り替えたり(実空間)、その建物の情報を記述しているWebページに切り替えたりする(情報空間)。例えば、カメラアングルを変化させ、ある店の映像を表示させたときには、その店のホームページを表示する。
- ●情報空間で検索を行なう場合に、現在注目している建物のみを対象にして検索を行なう。これにより検索対象の自然な絞り込みを実現している。例えば、ある通りをカメラで撮影しているときには、その通りにある建物のみを検索対象にする。
- ●ユーザの操作が暫時途切れたときには、ユーザが情報を求めている状態と捉え、能動的にアドバイスを提供する。ユーザの注意を喚起するため、キャラクタイメージを動かし、合成音声を用いて近くの店の紹介などを行なう。

上記の機能を実現するためには、映像・地図・Web ページの各メディアで表現される同一の建物を対応付けることが必要である。Canopy では、映像と地図、地図と Web ページをそれぞれ対応付けることにより、地図を介して三者の対応関係を自動的に求めている。映像と地図は、カメラ座標系と地図座標系の相互変換により対応付ける。また、地図と Web ページは、Web ページ中の HTML テーブルによって暗黙に表現されているオントロジーを抽出し、これを地図のオントロジーと統合することにより自動的に対応付ける。

ここで，オントロジーの統合には、我々が開発した事例に基づくフレームマッピング技術〔2〕を使用している。我々は、XML などの普及により今後オントロジー技術の重要性が増すなかで、分散情報源においてはオントロジーが統一されるのは困難であると考え、異なるオントロジーの統合技術に取り組んでいる。

A.4.3 その他の研究事例

このほかに、インターネットからの情報の収集・整理・活用を一貫して支援するシステムとして、利用者の視点の変化に適応できる支援方式についても研究を行なっている〔3〕。本システムでは、エージェントが利用者の視点を事例から学習し、それに合わせて情報を収集する。さらに、収集された情報を、その時点での利用者の視点に応じて整理して提示する。

参考文献

〔1〕 伊藤，大谷，上田，池田：属性オントロジーの抽出と統合を用いた実空間と情報空間のナビゲーションシステム，人工知能学会誌，Vol. 14, No. 6, pp. 1001-1009, 1999.

〔2〕 伊藤，上田，池田：分散情報源に対する情報エージェントのための事例に基づくフレームマッピング，電子情報通信学会論文誌 D-I, Vol. J-81-D-1, No.5, pp. 433-442, 1998.

〔3〕 Otani, Itoh, Shibata, Ueda, Ikeda: An Information Retrieval System based on Personal Viewpoints for Everyday Use, Second International Conference on Knowledge-based Intelligent Electronic Systems, 1998.

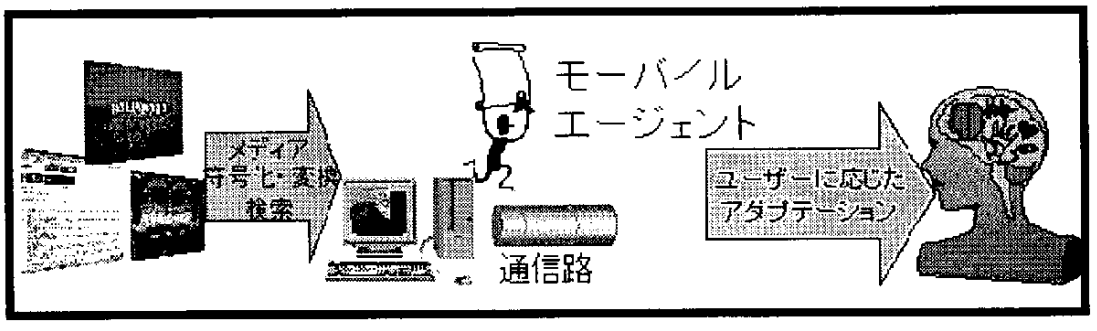
A.5 松下電器産業株式会社 東京通信システム研究所

－Content-Agents－

栄藤 稔、三浦 康史

A.5.1 情報流通のためのモバイルエージェント

我々はコミュニケーションを実現するメディア符号化伝送、通信路制御の研究を個々の最適化問題として扱うだけでなく、インタフェースを含めた系として実現していく研究を行っている。その中でモバイルエージェント技術は実現手段研究の一つである。メディア処理・ネットワーク制御に特化したモバイルエージェントを、我々はContent-Agentsと呼び、現在、基本プロトタイピングを終了し、実証実験中である。



図表 A.5.1-1

A.5.2 Content-Agents、what?

我々は、現時点で“エージェント”を知的な振る舞いをするプログラム構築技術とは捉えていない。ソフトウェア工学の文脈で単に身体性を持つ通信ソフトウェア、あるいは、オブジェクト指向分散ソフトウェアのコンポーネントであると考える。開発したContent-Agentsは、一般的なJAVAベースのモバイルエージェントであり、プラットフォームそのもを訴求するものではない。我々のアプローチの特徴は以下である。

(1) 応用を分散メディア変換処理や分散ワークフローに特化

(2) ドキュメントオブジェクト化されたメッセージング（P3P in W3C、 MPEG7）の利用

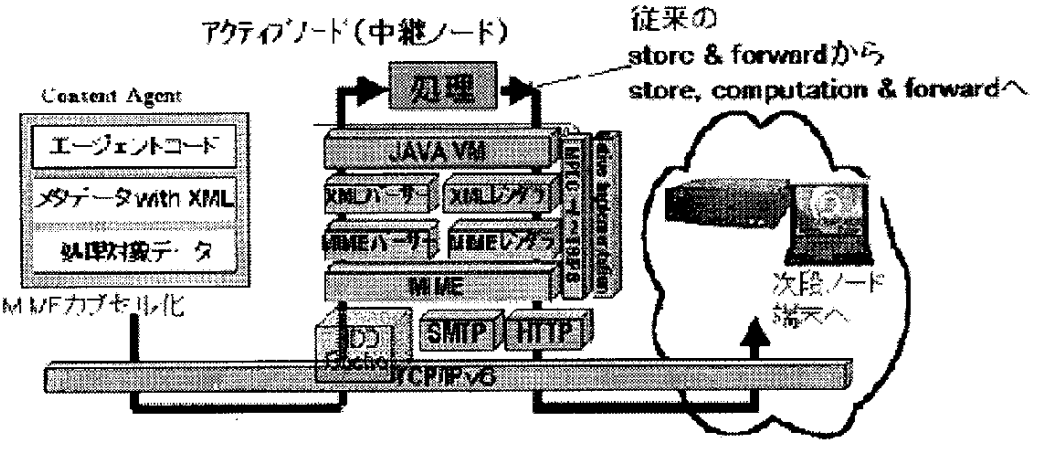
(3) 実時間・ネットワークQoS制約に対する実行、マイグレーションスケジューリング

(4) モーバイルエージェントによる負荷分散→スケーラビリティのある処理エンジン

コンテンツ流通のため、MPEG-7〔1〕やW3Cにおけるメタデータ標準の利用エンジンであることを意識しており、特にMPEG7で“Universal Multimedia Access”〔2〕と呼ばれる

コンテンツアダプテーション機能を任意のサイトで実現することを目標とする。

図表 A.5.2-1 は、Content-Agents の実行環境を示し、実体は SMTP/HTTP 上を MIME でパッケージングされて移動する。状態は、運ばれるコンテンツ自身と XML 記述により保持され、Weak Migration が実現されている。Content-Agents 間の通信はサイト上にレポジトリを設け、HTTP を介して行われる。ネットワーク QoS(Quality of Service)パラメータ、ユーザー・端末記述も、レポジトリを介して Content-Agent 自身が収集する。



図表　A.5.2-1

A.5.3　Active Network との対比

Content-Agent の移動をネットワークの文脈で理解すると、“Layer-7 Routing”となる。これに対して、ネットワーク層のプロトコルをプログラマブルとして、従来の固定的なプロトコル(例:TCP/IP、RTP)の頸城から逃れようとする Active Network の研究がある〔3〕。Active Network では、Layer-3 レベルのプロトコルが可変となり、store&forward の処理を行う従来のルーターに対して、store、compute & forward の処理が、プログラマブルなルーターで行われる。Content-Agents の実行環境では、下位プロトコルは固定とし(SMTP/HTTP)、アプリケーションレベルで compute に相当する処理を行い、ペイロード(MIME カプセル化されたコンテント)を変換、ルーティングを下位プロトコルにマップする点が異なる。以上の構成により、端末記述、CC/PP (Composite Capability/Preference Profiles)、ユーザ記述、P3P (Platform for Privacy Preferences)により、ペイロードであるコンテンツ表現記述(例えば、HTML、 SMIL+MPEG、 MPEG7) の表現を可変とするコンテンツ配送が可能となっている。

A.5.4　エージェントセキュリティ

モバイルエージェントの本質であるプログラムの移動は、セキュリティ管理を伴う。エージェントのホストに対する攻撃には SandBox モデル等の研究事例があるが、逆にホストからエージェントに対する攻撃に関しては研究余地が残る。特に Content-Agents では、エージェントの保持しているデータの盗用、改竄、意図しない経路変更は避けなければならない. 我々はマルチホップエージェントの動作環境において、そのエージェントにより何らかの問題が発生した場合に、移動経路をたどることによりその問題を起こしたホストの特定可能なアルゴリズムを提案している〔4〕。

参考文献

〔1〕 http://drogo.cselt.stet.it/mpeg/

〔2〕 http://www.research.ibm.com/networked_data_systems/transcoding/

〔3〕 A Survey of Active Network Research, IEEE Communications Magazine, Vol. 35, No. 1, pp80-86. January 1997.
(http://www.tns.lcs.mit.edu/publications/ieeecomms97.html)

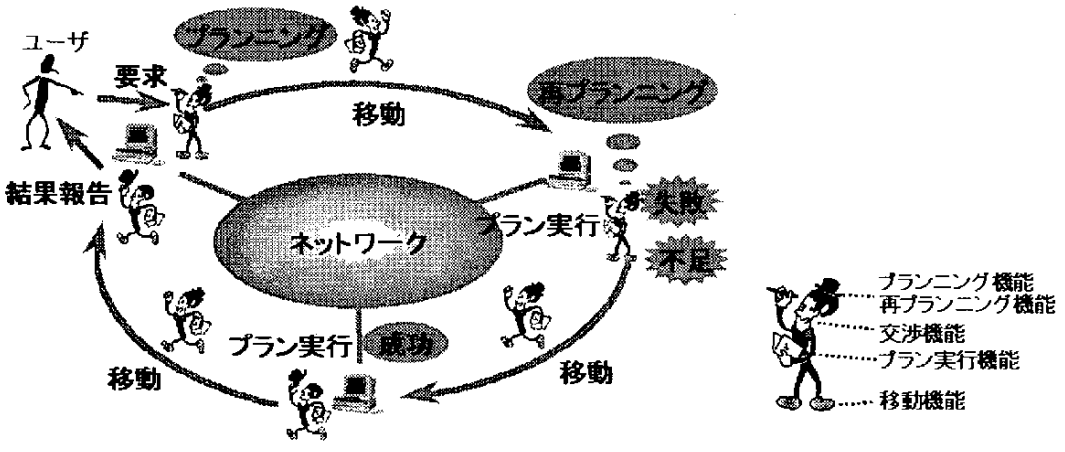
〔4〕 三浦, 栄藤: "モバイルエージェントのマルチホップ認証についての考察", 暗号と情報セキュリティシンポジウム (SCIS2000), D-19, (Jan 2000).

A.6 東芝研究開発センターにおける取り組み

（株）東芝　研究開発センター　コンピュータ・ネットワークラボラトリー
大須賀 昭彦
akihiko.ohsuga@toshiba.co.jp

A.6.1 知的ネットワークエージェント Plangent の開発

Plangent (Planning Agent) は、プランニングという推論機構を持つモバイルエージェントが、目標達成の手段を自分で考えながら、ネットワーク上を自律的に動きまわって処理を行なうエージェントシステムである。ユーザからの要求を受けとった Plangent エージェントは次のように行動する。(1) どこで何をするかといった行動計画をプランニングによって立案する。(2) その行動計画に基づいて必要な情報やサービスのある場所まで移動する。(3) 移動先の情報やサービスを活用して処理を実行する。(4) 予期せぬ事態などによって計画の実行が失敗した場合、再プランニングによって状況に合った行動計画を作りなおす。(5) 新たな行動計画に基づいて移動と実行を行なう。(6) 目標が達成されるまで



図表 A.6.1-1 Plangent の動作概要

これらを繰り返し、最後にユーザのところへ戻って結果を報告する〔1～5、17〕

A.6.2 マルチエージェントフレームワーク Bee-gent の開発

Bee-gent (Bonding and Encapsulation Enhancement Agent) は、レガシーシステムやデータベース、WWW サーバなどの各種アプリケーションをネットワークを介して柔軟に連携するためのマルチエージェントフレームワークである。Bee-gent は、エージェントラッパーと呼ばれるモジュールと、仲介役となるモバイルエージェント群から構成される。エージェントラッパーが接続対象となるアプリケーションをエージェント化し、仲介エージェントがその間の調整役となって各システム間の連携を実現する。エージェントラッパーと仲介エージェントとの会話には XML/ACL を用いる。XML/ACL とは、エージェント間の共通言語として FIPA (Foundation for Intelligent Physical Agents) が定める ACL (Agent Communication Language) を XML 上に実装したものである〔6、18〕。

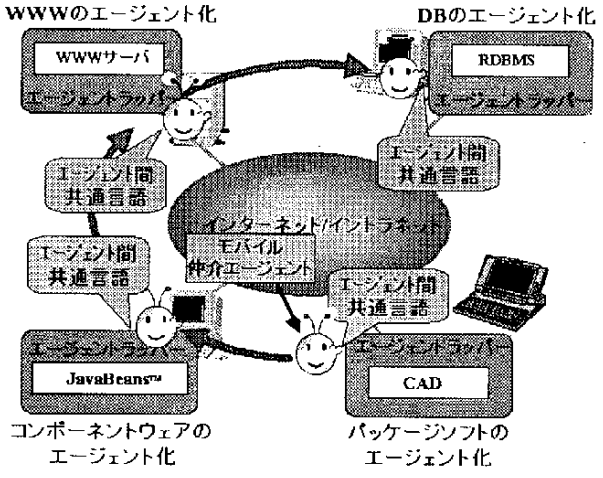


図 A.6.2-1 Bee-gent の動作概要

A.6.3 アプリケーションの開発、開発方法論や支援環境の整備、エージェント技術の標準化など

Plangent と Bee-gent を用いたアプリケーションとして、ドライバー情報支援エージェント〔7〕、ヒューマンナビゲーションエージェント〔8〕、組込み機器巡回・点検モバイルエージェント〔9〕、製品環境情報システム〔10〕、化学プラント診断保全エージェント〔11〕、モバイル環境向け検索エージェント〔12〕などを開発中である。また、エージェント開発を支援する研究として、エージェントパターンなる考えに基づく開発方法論〔13、14〕やエージェントアプリケーション開発のための支援環境の整備〔15〕などを進めている。さらに、FIPA などに対するエージェント技術の標準化提案、移動エージェント相互運用技術の開発〔16〕やエージェント相互接続実験も行なっている。

参考文献

〔1〕 本位田, 大須賀: オブジェクト指向からエージェント指向へ, ソフトバンク (1998).
〔2〕 本位田, 飯島, 大須賀: エージェント技術, 共立出版 (1999).

〔3〕 Ohsuga, A., Nagai, Y., Irie, Y., Hattori, M., and Honiden, S.: PLANGENT: An Approach to Making Mobile Agents Intelligent, *IEEE Internet Computing*, Vol. 1, No. 4 (1997), pp. 50-57.

〔4〕 大須賀: プランニングモバイルエージェント, bit, Vol. 31, No. 3 (1999), pp. 88-95.

〔5〕 大須賀ほか:知的モバイルエージェント Plangent プログラミング, bit, Vol. 31, No. 8, 9, 10 (1999).

〔6〕 川村, 長谷川, 大須賀, 本位田: Bee-gent:移動型仲介エージェントによる既存システムの柔軟な活用を目的としたマルチエージェントフレームワーク, 信学論, Vol. J82-D-I, No. 9 (1999), pp. 1165-1180.

〔7〕 Hattori, M., Kase, N., Ohsuga, A., and Honiden, S.: Agent-Based Drivers' Information Assistance System, *New Generation Computing*, Vol. 17, No. 4 (1999), pp. 359-367.

〔8〕 池谷, 加瀬, 大須賀: エージェント技術を適用したヒューマンナビゲーションシステム, 情処 ITS 研究会, 99-ITS-3 (1999), pp. 97-104.

〔9〕 田中, 小泉, 前田, 大須賀, 武脇, 関口: エージェント技術の応用－電力系統保護制御における整定エージェントと解析エージェントの試作－, 計測自動制御学会第 23 回制御技術部会研究会 (1999).

〔10〕 吉岡, 川村, 長谷川, 大須賀, 春木: マルチエージェントフレームワーク Bee-gent による環境調和型製品設計支援システムの開発, 第 59 回情処全国大会 (1999).

〔11〕 長野, 川村, 長谷川, 大須賀: マルチエージェントフレームワーク Bee-gent を利用した設備診断と計画保全の連携システム, 第６０回情処全国大会 (2000).

〔12〕 吉岡, 川村, 長谷川, 大須賀: マルチエージェントフレームワーク Bee-gent による Web 検索システムの開発, 第６０回情処全国大会 (2000).

〔13〕 Tahara, Y., Ohsuga, A., and Honiden, S.: Agent System Development Method Based on Agent Patterns, in Proc. *20th International Conference on Software Engineering* (1999), pp. 356-367.

〔14〕 田原, 大須賀, 本位田: ビヘイビアパターンに基づくモバイルエージェントシステム開発手法, 情処論文誌, Vol. 40, No. 12 (1999. 12), pp. 4319-4332.

〔15〕 田原, 長野, 吉岡, 川村, 長谷川, 大須賀, 本位田: マルチエージェントフレームワーク Bee-gent のためのビジュアル開発支援ツール, 第 59 回情処全国大会 (1999).

〔16〕 長谷川ほか: 移動エージェント相互運用の実現, 第 59 回情処全国大会 (1999).

〔17〕 http://www2.toshiba.co.jp/plangent/

〔18〕 http://www2.toshiba.co.jp/beegent/

A.7 富士通研究所ネットメディア研究センター

ーエージェント技術に関するネットメディア研究センターの取り組みー

岡田 誠
okadamkt@flab.fujitsu.co.jp

本稿では、富士通研究所ネットメディア研究センターにおけるエージェント技術への取り組みを、研究の位置付けおよびアクテビティという視点から紹介する。

A.7.1 研究の位置付け

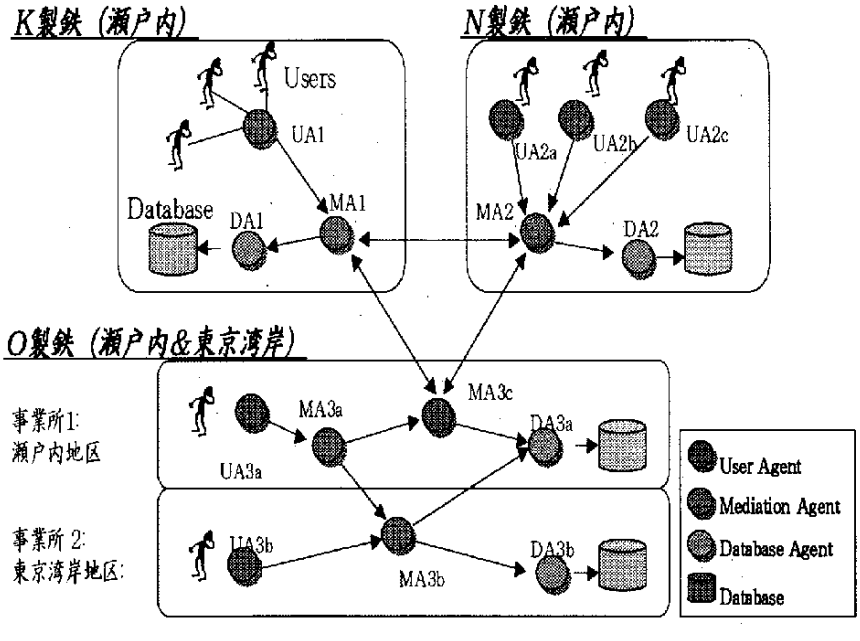
富士通研究所ネットメディア研究センターでは、エージェント技術という視点から、分散オブジェクト環境をサポートするアーキテクチャにかかわる研究を進めている。いわゆるミドルウェアとよばれるレイヤーを対象とし、インターネットに代表されるような動的でオープンな環境の中で、いかにサービスとサービスを柔軟かつ動的に連携させていくかに関して、プラットフォーム要素技術の視点から研究開発を進めている。

A.7.2 アクテビティ

以下ではネットメディア研究センターにおける代表的なアクテビティを紹介する。

(1) 分散メディエータ

鉄鋼設備 CALS 向けのエージェントシステムとして、分散した複数組織(特に企業)間での情報の共有システムを開発した〔1〕。仲介(Brokerage)機能を持つ分散配置されたメディエータエージェントの連携により、複数のデータベースへの効率的で柔軟なアクセスを行うための基盤技術を提供している。図 A.7.2-1 はシステムの構成例である。分散配置されたメディエータエージェントがユーザーからの要求とデータベース側からのアドバタイズ情



図表 A.7.2-1 分散メディエータ

報の両方をパターンマッチを用いながら適切な対象へと仲介していく。

分散メディエータによるサービス仲介により、企業間に開いたサービス連携をクライアント側・データベース側から独立して行うことが可能であり、また各企業間のサービスポリシーの違いをその部分で吸収することができる。

Web-Computing 環境への整合性を考え、エージェントはサーブレットとしても実装している。エージェントの追加・削除がメディエータ間のアドバタイズとして行われることにより、ユーザーに対してあたかも自動的にサービスが拡張しているかのようにみせることも可能である。またこのような枠組みは分散メディエータにサービス連携ゲートウェイとしての役割を果たさせるものともいえる。

現在 Brokerage に関してはエージェントに関する標準化団体である FIPA〔2〕においても標準化の議論が進行中である〔3〕。ネットメディア研究センターは提案者〔4〕としてこの議論に中心的に関わっている。

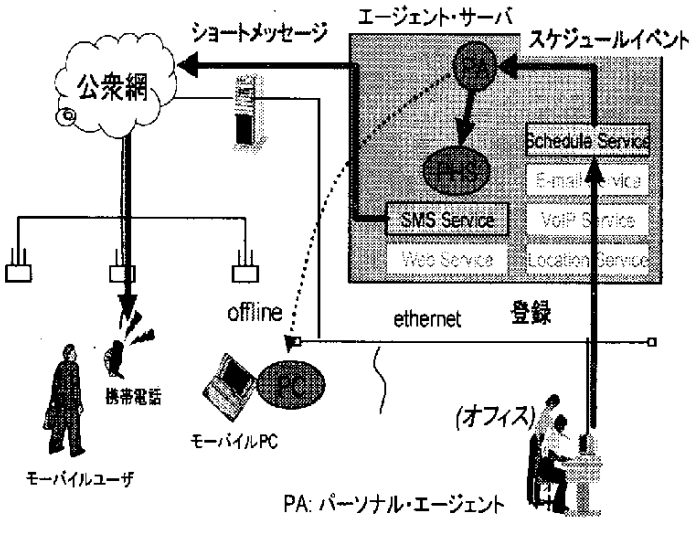
(2) エージェント連携機構

Agent 連携のフレームワークとして、Agent 間コミュニケーションバスとエージェント・インターフェイスの研究開発をおこなっている。Agent 間コミュニケーションバスは、無線、LAN、共有メモリなどを共通で単純なインターフェイスを持つ通信メディアとして扱うためのものであり、エージェントインターフェイスはエージェント間の連携をメッセージとインターフェイスのパターンマッチにより構築していくというものである。

コミュニケーションバスとパターンマッチ型のエージェントインターフェイスによって、より柔軟なエージェント間の協調を構築することが可能となる〔5,6〕。

(3) パーソナルエージェント

ユーザの属性や状態をプロファイルとして管理し、ユーザ毎にカスタマイズされたサービスを提供するためのものである。ユーザが異なる通信デバイスを使っても、端末能力に応じたサービスが提供されるための要素技術の開発を行っている。3GPP〔7〕では現在 VHE (Virtual Home Environment)として標準化が進められているが、ネットメディア研究センターでは VHE のプロトタイプ〔8〕を作成し、TELECOM99 においてデモンストレーションを



図表 A.7.2-2　パーソナルエージェント、ネットワークエージェント

行った。図表 A.7.2-2 はそのときのサービス構成図である。

(4) ネットワークエージェント

モーバイル端末で使われるネットワークメディアの違いによる接続手順、帯域、安定性の差異を隠蔽し、ユーザに対してシームレスなネットワークアクセスを提供するためのものである。メディアの自動検出・最適設定（network plug & play）の方式に関する研究を進めている。FIPA でも Nomadic Application 仕様の一環としてプロトコルの標準化が進められている。

(5) エージェントライブラリの開発

Agent 開発の Library として、非同期メッセージ配信、Agent 間コミュニケーションバス、モジュールインターフェイス、メッセージのパターンマッチ機能、内部コードの置き換え機能等に関して Java ベースのライブラリの開発と整備をおこなっている[9, 10, 11]。

(6) FLA (Fujitsu Laboratories of America)

ネットメディア研究センターとは組織が異なるが富士通研究所に所属しエージェントの研究をおこなっている組織として FLA の NAR (Network AgentResearch)[12]がある。 NAR は 1997 に設立され、Agent Programining 言語の April や April ベースの非同期通信モデル ICM 等の研究を行っている。ネットメディア研究センターの Java 非同期メッセージ配信ライブラリの開発も NAR の April や ICM と関連づけながら進めている。

NAR はまた FIPA や OMG の Agent Working Group においても積極的に活動し、組織運営や仕様書作成に強く関わっている。

参考文献

〔1〕 毛利他: 企業間情報共有のためのマルチエージェントプラットフォーム、CALS/EC Japan 1998 論文集, pp363-370

〔2〕 FIPA: http://www.fipa.org

〔3〕 FIPA: TCC Baseline document, http://fipa.comtec.co.jp/fipa/Common_docs/fipa9122.doc

〔4〕 市来他: A proposal on Agent Brokerage http://fipa.comtec.co.jp/fipa/Common_docs/cfp5_008.html

〔5〕 岩尾他:Flexible Mult-Agent Collaboration using Pattern Directed Message Collaboration of Field Reactor Model, PRIMA99 to be published

〔6〕 岡田他:Awareness アナロジーによる分散オブジェクトの緩やかな連携, DPS Workshop, IPSJ Symp. Ser. Vol.98, No.14, pp227-234

〔7〕 3GPP: http://www.3gpp.org/

〔8〕 藤野他: Mobile Communication Middleware Suporting Virtual Home Environment, ICCC'99, I-1-03

〔9〕 牛嶋他: Pathwalker: Message based Process Oriented Programming Library for Java, INAP 98, pp.137- 143

〔10〕 Pathwalker Library: http://www.fujitsu.co.jp/hypertext/flab/ free/paw/
〔11〕 岩尾他: Field Reactor Library を用いた分散オブジェクト連携, DICOMO 99, IPSJ Symp. Ser. Vol.99, No.7, pp525 - pp530
〔12〕 NAR: http://www.nar.fla.com/

A.8 三菱電機におけるエージェント技術研究開発への取り組み

情報技術総合研究所　木野茂徳

三菱電機におけるエージェント技術研究開発への取り組みについて、主なものを以下にご紹介する。

（1）移動型エージェントプラットフォーム DiaConcord（米国商標 Concordia）の開発

マルチエージェントによる協調分散作業環境の実現を目的とした Java ベースの移動型エージェントプラットフォームの開発。後述するいくつかのエージェントシステムですでに利用が始まっており、また、エージェント単位でのセキュリティ設定を可能にするなど、実用化のための技術開発が続けられている。

（2）ユーザインタフェース型エージェント構築ミドルウェア Collagen (Collaborative Interface Agents) の研究開発

対話理論に基づく双方主導 (mixed initiative) による問題解決エージェント構築用ミドルウェアの開発。対話型の利用者支援システム、例えば、人による誤操作・誤指示の可能性を指摘し利用者とともにその改善を図るシステムの構築などへの利用が期待されている。

（3）EAI (Enterprise Application Integration) 向け MELBA (Multi-Enterprise Links By Agents) の研究開発

本報告書第３章にご紹介しているが、MELBA Broker と呼ぶ EAI（業務アプリケーション連携）を対象としたルールベースのエージェント実用化開発を行っている。実装面では、上記移動型エージェントプラットフォーム"DiaConcord"を利用した「インターネット経由のアプリケーション連携」が進んでいる。

（4）資材融通調整を行う交渉エージェント（CO²PLANNER）の研究開発

資材の融通調整を行うエージェントを例に、調整作業に参加する組織自身の新規参加・離脱を許容し、また、調整中においても各組織の評価基準や調整案の変更を許容する、多段階交渉を特長とした交渉エージェントの研究開発を、東京電力（株）殿と

共同で行っている。

（5）DiaConcord 応用：電力・水道など公共事業（Utility）分野での設備監視・情報収集システムへの適用技術開発

マルチエージェントの並列実行による設備情報収集や、監視エージェントによるシステム管理･保守といった分野から導入が進んでいる。

エージェント技術の研究領域として、

a) エージェントが行う知識処理やマルチエージェント処理におけるプランニングアルゴリズムの研究と

b) これらエージェント自身や知識処理部のソフト生成方式の研究

c) 対話型エージェントを利用した Human-Computer 協調作業の研究

d) エージェント間、ならびに　既存ソフトウェアとの間の標準的なエージェントインタフェース技術に関する研究開発

e) エージェント移動やエージェント間情報伝達におけるセキュリティ技術に関する研究

などがあげられる。

参考文献、ホームページ：

〔1〕 Mitsubishi Electric Horizon Systems Lab. (Waltham, MA)
Concordia Home Page： http://www.meitca.com/HSL/Projects/Concordia/

〔2〕 Mitsubishi Electric Research Lab (Cambridge, MA)
Human-Computer Interaction Home Page：
http://www.merl.com/threads/hci/index.html
C. Rich and C. L. Sidner: COLLAGEN A collaboration manager for software interface agents User Modeling and User-Adapted Interaction Vol. 8, No 3/4

〔3〕 Mitsubishi Electric America Business Internet Division (Sunnyvale, CA)
MELBA Home Page： http://www.bid.mea.com/melba.html
および、本報告書 2.1.2 項

〔4〕 小野，萩原，秋吉：リサイクル推進のための建設副産物リサイクル融通調整支援システム；電気学会論文誌，Vol. 118-C, No. 10, pp. 1418-1423 (1998)
阿部，秋吉，小野，樟：マルチエージェントに基づくリサイクル融通計画システムにおける耐ネットワーク障害機能の実現，人工知能学会研究会 SIG-KBS-9803-4, pp. 19-24 (1999)

〔5〕 久山，川田，Wong，野村，前川：モバイルエージェントを応用した分散データベース検索システム　情報処理学会第 58 回全国大会　4Q—07　1999

A.9　ＮＨＫにおける放送用エージェントの研究開発への取り組み

NHK は、エージェントの技術を放送分野へ応用する研究を積極的に進めている。ここでは、主に「番組受信」と「番組制作」分野でのエージェントの研究・開発状況を中心に述べる。また、放送用エージェントの要素技術についても説明する。

番組受信分野：放送のデジタル化、多チャンネル化に伴い、テレビ受信機に大量の番組や付加情報が集中することが予想されている。具体的には、ISDB (Integrated Services Digital Broadcasting) チャンネルや双方向ネットワーク、衛星、ケーブルなどを利用して、多くの情報を得ることができる。従って、今後のテレビ受信機は、大量の番組を個々の視聴者の嗜好に応じて容易に選択して提示できる高度な情報処理能力が求められている。このような機能を実現する技術として、NHK ではエージェント技術を用いた次世代型テレビ（エージェントテレビと呼ぶ）の研究を行っている。エージェントテレビは、受信した番組や関連データなどを蓄積するホームサーバーを備えたシステムで、視聴者の好みや要求に従って、自動的に番組を記録したり提示したりする機能を有する。

このような機能を実現するには、エージェントテレビは、放送されてくる番組の内容と各視聴者の嗜好や趣味に関する様々な情報や知識を知る必要がある。特に各放送番組の内容に関する情報は、視聴者の趣味に合う番組を選択ために、不可欠である。このような番組の内容に関する情報を記述したものを EPG (Electronic Program Guide：電子番組案内) と呼び、視聴者に関する知識を記述したものをユーザープロファイルと呼ぶ。現在、EPG のフォーマットに関する標準化が MPEG、DVB、ARIB などの各標準化機関によって国際的に活発に行われている。番組の選択は、放送局から提供される EPG と視聴者に関するユーザプロファイルの情報をマッチングさせることで行われる。また、エージェントは EPG を常時参照することで番組のスケジュールや放送内容の変更にも柔軟に対応できる。

NHK は、エージェント技術の国際仕様 (FIPA Specification) をもとに、エージェントテレビシステムを試作した。このシステムの研究・開発は、ヨーロッパとの共同研究である FACTS プロジェクト (FIPA Agent Communication Technologies and Services) で行い、標準仕様への検証点、変更点を提起した。このプロジェクトには、NHK の他に CSELT 社(伊)、ロンドン大学 (英)、IRST (伊) が参加している。エージェントテレビシステムの実装は基本的に FIPA の第１次標準である FIPA97 と FIPA98 に基づいて進めている。

番組制作分野：本格的なデジタル放送時代には、高品質な番組と並行して多様な放送用コンテンツをより効率的に制作する仕組みが求められる。従来の番組制作は、（１）発想、（２）企画・台本、（３）取材・素材収録、（４）素材加工、（５）編集の５つのプロセスから成り立っており、番組制作の全プロセスを通して情報の蓄積や活用を図るなど、制作作業をトータルに支援する「トータル的な番組制作システム」が求められる。
我々は、このような制作システムの構築にエージェント技術を適用した新たな制作システムの研究を行っている。番組制作においては、基本的に制作者の感性や意図が重要であり、番組制作でのエージェントの利用は、全自動番組制作システムの実現ではなく、制作者の

ニーズに合った様々な情報を多角的に発見・提供し、番組制作用機器などを自動的に制御することで番組制作時間を短縮し、品質の高い番組の制作を支援することを目的としている。ＮＨＫでは、インターネットや番組アーカイブを検索して常時適切な情報や話題、または素材映像を制作者に随時提供して番組の発想から制作を支援する、ネットワークを基盤としたエージェントシステムの構築を始めている。番組制作における実験的な応用例としてＭＩＴのメディアラボが提案したインテリジェント・スタジオ・プロジェクトがある。インテリジェント・スタジオでは、複数のスマートカムス（SmartCams）と呼ばれるカメラがあり、番組ディレクターの言葉による指示にしたがってスタジオの中の人間や物体を自律的に撮影するカメラシステムである。これらのスマートカムスは、カメラレンズから得た視覚データをそのシーンや物体に対する記号で表現された知識と関連させることで、被写体の撮影を行う。また、複数のスマートカムスがある特定の撮影知識を共有しながら協調することで、多様な角度からの撮影が可能になる。ロボットカメラ以外にも、エージェントによる他の放送機器の自動制御に関する研究も活発に行っている。このような技術は、制作プロセスの効率化を図ることができ、制作コストの負荷を軽減することができる。一方、人工生命のような動物型のエージェントや擬人化されたエージェントは、視聴者と仮想現実の世界の中で対話的にやり取りできる参加シアター（Participatory Theater）の仮想キャラクタとして、番組への利用が期待できる。

以上、ＮＨＫにおける放送用エージェントの研究・開発の取り組みについて述べた。次に、このような放送用エージェントシステムを構築する際、必要になる要素技術について述べる。ここでは、エージェントプラットフォーム、エージェント通信言語、プロトコールなど、一般的なエージェント技術ではなく放送分野への応用に特化した要素技術をリストアップし、簡単に説明する。

- **ユーザインターフェース**：放送は幅広い視聴者層を対象にしているのが特徴である。そのため、例えば、エージェントテレビのユーザインターフェースを構築する場合、年寄りから子供までの幅広い年齢層や障害者などを考慮して設計する必要がある。例えば、リモコン、キーボード、自然言語、GUI、ジェスチャーなどを用いた様々なユーザインターフェースが重要になる。
- **番組自動選択**：番組選択はエージェントテレビの最も中心な機能である。番組選択の場合、問い合わせ（Query）による番組検索の支援だけでなく、視聴者の視聴履歴や嗜好の変化などを考慮した番組選択能力が求められる。また、プロファイルに記述されていない新番組でも、視聴者にとって重要であると思われるものが選択できる検索（Innovative Search）機能は極めて重要である。
- **プレゼンテーション機能**：選択された番組を視聴者に見やすい形態で提示することもシステムの重要な機能である。従って、視聴者の嗜好やテレビの視聴時間に合わせて番組を提示するような柔軟性が必要となる。例えば、番組ブラウジングや１つの画面に複数の生放送を提示するマルチスクリーン、時間がない視聴者には、番組の重要な部分だけを提示する番組要約などが挙げられる。さらに、緊急放送や重要なイベント（例えば、スポーツの場合、ゴールやホームランのシーンなど）を検出したとき、それを視聴者に知らせる機能が求められている。

A.10 NTT コミュニケーション科学基礎研究所におけるエージェント技術への取り組み

小暮潔　　桑原和宏

我々はエージェントに関する基礎理論から、サイバー社会での社会活動のための応用まで幅広く研究を行っている。具体的には、エージェントの設計理論に関する研究、インターネットオークションに関する研究、「ソーシャルウェア」〔4〕の研究である。

A.10.1 エージェント設計理論

エージェント技術の応用分野の中には電子商取引などのようにインターネット上の多種多様な環境下で高い信頼性が要求されるものがある。このような分野では、エージェントの動作を保証すること、すなわち、エージェントのプログラムが与えられたとき、エージェントが望ましい動作を行うこと（たとえば、モバイルエージェントが必ず戻ってくること）を保証することが望まれる。このような保証を与えるための技術として形式的検証がある。我々は形式的検証機能を持ったエージェントプログラミング環境の設計・構築を目指している。

形式的検証を可能にするには、エージェントプログラミングに適した理論的基盤が必要である。我々はそのような基盤として、エージェントの持つ重要な２つの性質を捕らえるのに適した形式的基盤を採用している。すなわち、エージェントの知的な振る舞いをとらえるのに適した知識ベースプログラム（knowledge-based program）〔3〕と、エージェント間のインタラクションをとらえるのに適したプロセス代数〔11〕である。

(1) 知識ベースプログラムに基づくアプローチ

我々は、モデルチェッキング〔10〕による論理的検証機能を備えたモバイルエージェントシステム ERDOS を開発している〔1,2〕。ERDOS はプログラムの基本単位として知識ベースプログラムで導入された、テストと動作の対を使用する。ここで、テストは様相論理などの知識を取り扱う論理式で表現され、テストが成功するとき、すなわち、この論理式が証明できるとき、動作を実行するという明快な意味を持つ。ERDOS システムでは、エージェントシステムのプログラムが与えられたとき、これを有限状態遷移モデルに変換し、モデルチェッキングによる論理的検証を行い、エージェントシステムが望みの動作を行うことを判定することができる。

(2) プロセス代数に基づくアプローチ

プロセス代数は通信しながら並行に動作するプロセスから構成されるシステムの振る舞いをとらえるために提案された理論的枠組であり、個々のプロセスを表すために、プロセス間通信の送受信、プロセス呼出し、条件分岐などを表すプリミティブから構成されるプロセス式を用いる。モバイルエージェントによる電子商取引などエージェントの動的生成が必要な応用を考えると、通信路の動的生成や通信に関する情報の交換を行う機能が望

まれるが、このような機能を持つプロセス代数にπ-計算がある。我々は、このπ-計算に基づくネットワークプログラミング言語として Nepi² を設計・実装し〔5〕、エージェントプログラムの基礎としている〔7〕。プロセス代数では、プロセスの実行状態に関する情報がすべてプロセス式の中に含まれる。この特徴には、プロセスの実行状態を含めたエージェントの移動を容易に実現することができるという利点がある。このような利点を生かし、我々はモバイルエージェントの機能を実装するのに有用な Nepi² の拡張を行っている〔8〕。

A.10.2 インターネットオークション

インターネット環境のような個人の同定が実際的に不可能である環境では、オークションにおいて架空名義入札の可能性を排除することができない。我々は、このような状況下で、複数財のオークションプロトコルとして理論的に最も優れているとされている一般化 Vickrey オークションでも誘引両立性、パレート効率性、個人合理性を同時に満たすことがないこと、および、どのようなオークションプロトコルもこれらを同時に満たすことがないことを証明している〔12-14〕。

A.10.3 ソーシャルウェア〔4〕

我々は、サイバー社会における様々な社会活動を支援する「ソーシャルウェア」の研究を進めている。この一環としてネットワーク上のコミュニティの形成を支援する Community Organizer〔6〕や、議論を支援する CommunityBoard〔9〕などを提案している。さらに、各種ソーシャルウェアを構築するためのエージェントプラットフォーム Shine〔15〕の研究も進めている（Shine に関しては 2.3.4 節で紹介している）。

参考文献

〔1〕 櫟：知識と信念の推論に基づくモバイルエージェントシステム，第 13 回人工知能学会全国大会，1999.

〔2〕 櫟：エージェントプログラミングとその形式的検証，信学技報 AI99-47，1999.

〔3〕 Fagin, R., Halpern, J. Y., Moses, Y., and Vardi, M. Y.: Reasoning About Knowledge, MIT Press, 1995.

〔4〕 Hattori, F., Ohguro, T., Yokoo, M., Matsubara, S., and Yoshida, S.: Socialware: multiagent systems for supporting network communities, Comm. ACM, Vol. 42, No. 3, pp. 55-61, 1999.

〔5〕 Horita, E., Mano, K.: Nepi²: a two-level calculus for network programming based on the π-calculus, ASIAN-97, 1997.

〔6〕 亀井，吉田，大黒，服部：Community Organizer：ネットワークコミュニティの形成支援，ヒューマンインタフェースシンポジウム '99，pp. 333-336，1999.

〔7〕 河辺，真野，小暮：アクタモデルのπ計算に基づく意味づけ—エージェントの形式化に向けて—，情報処理学会第 57 回全国大会，1998.

〔8〕 河辺，真野，小暮：π-計算に基づくモバイルエージェントの形式化，信学技報 AI98-43，1998.

〔9〕 Matsubara, S., Ohguro, T., and Hattori, F.: CommunityBoard 2: mediating between speakers and an audience in computer network discussions, Agents-99, pp. 370-371, 1999.
〔10〕 McMillan, K. L.: Symbolic Model Checking, Kluwer Academic Publishers, 1993.
〔11〕 Milner, R.: Communication and Concurrency, Prentice Hall, 1989.
〔12〕 Sakurai, Y., Yokoo, M., Matsubara, S.: A limitation of generalized Vickrey auction in electronic commerce: robustness against false-name bids, AAAI-99, 1999.
〔13〕 櫻井, 横尾, 松原: 電子商取引における一般化 Vickrey オークションの問題点: 架空名義入札に対する頑健性, コンピュータソフトウェア, to appear.
〔14〕 Yokoo, M., Sakurai, Y., Matsubara, S.: The effect of false-name declarations in mechanism design: toward collective decision making on the internet, 20th IEEE International Conference on Distributed Computing Systems, to appear.
〔15〕 吉田, 大黒, 亀井, 船越, 桑原: サイバーコミュニティのアプリケーションのためのプラットフォーム Shine の提案, 第 13 回人工知能学会全国大会, pp. 461-462, 1999.

A.11 株式会社コムテックにおける研究開発への取り組み事例

A.11.1 概要

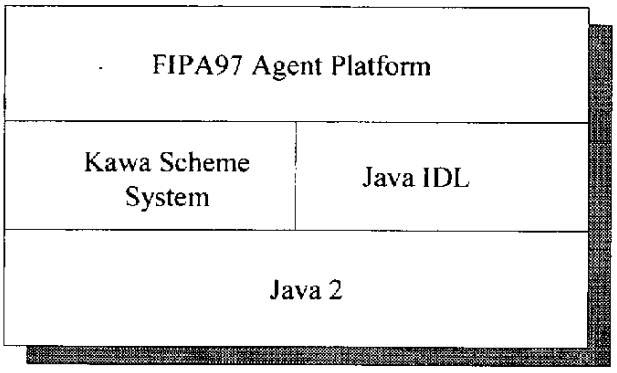
我々は FIPA に基づいた利用環境基盤の構築及びプラットフォームの開発を行っている。本開発によって実装されたエージェント及びソフトウェアモジュールを用いることにより、FIPA に準拠した複数のエージェントが協調動作する環境を構築することができる。

これまでに開発した要素は、エージェントの管理、エージェント通信言語、ソフトウェアインターフェース、オントロジーサービスである。本開発からのフィードバックはよりよい標準を策定するために寄与している。

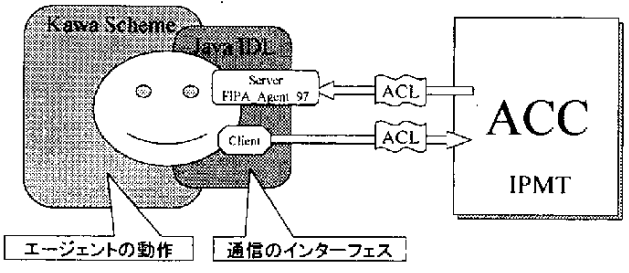
A.11.2 エージェントプラットフォーム

FIPA では全てのエージェントがエージェントプラットフォーム (AP) と呼ばれる共通基盤の上で動作するというエージェント管理参照モデルを持つ。この AP にはディレクトリーファシリテーター (DF)、エージェント管理システム (AMS)、エージェント通信路 (ACC) というエージェントが最低一つ存在し、これらのエージェントが AP 上のエージェントの管理及び AP 間でのメッセージ転送を行う。

図表 A.11.2-1 および図表 A.11.2-2 に概要を示す。Java の上に、LISP の一方言である Scheme のインタープリターである Kawa をのせ、エージェントの動作を Kawa で記述してい



図表 A.11.2-1　システムの構成



図表 A.11.2-2　エージェントの構成

る。一方、エージェント間通信のために、JavaIDL を用い、IIOP によるコミュニケーションをサポートする。

（1）ディレクトリーファシリテーター（DF）

エージェントが提供するサービスの登録及び検索を行うエージェントで、エージェントのイエローページ情報を他のエージェントに提供する。

（2）エージェント管理システム（AMS）

AP 上に存在する全てのエージェントの識別子（GUID）や通信アドレスの登録を行うエージェントで、エージェントのホワイトページ情報を他のエージェントに提供する。

（3）エージェント通信路（ACC）

IIOP を使用して AP 間のメッセージ転送を行うエージェントで、AP 内部と外部のゲートウェイを提供する。

（4）ラッパーエージェント（Wrapper）

エージェント以外のソフトウェアが提供するサービス（非エージェントソフトウェアサービス）に接続し、エージェントに対してこのサービスを提供するエージェントである。

（5）エージェント資源ブローカー（ARB）

ラッパーエージェントが提供する非エージェントソフトウェアサービスの登録及び検

索を行うエージェントである。

A.11.3　エージェント通信言語とコンテント言語

FIPA ACL のパーサーとジェネレーターを開発した。ACL には動詞に相当するコミュニカティブアクト、通信のパラメーターを表現するメッセージパラメーター、メッセージの内容を記述したコンテントが含まれる。

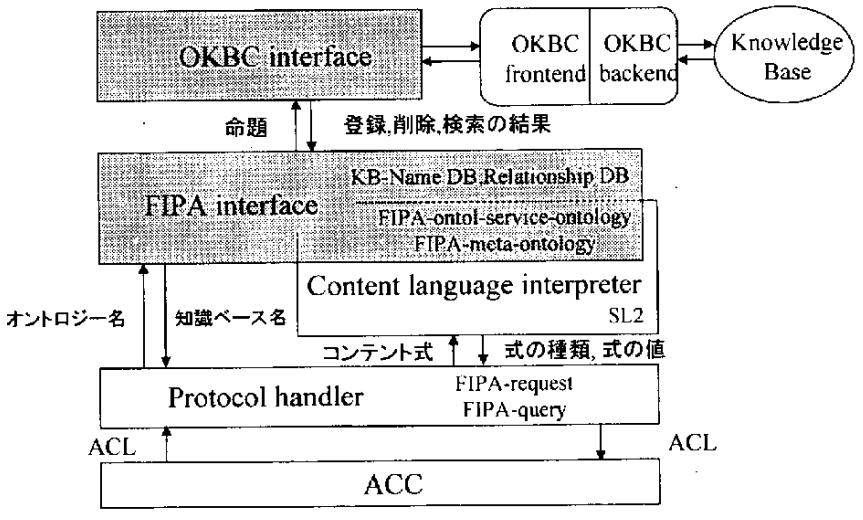
また、コンテント言語 SL0 および SL2 のインタープリターを開発した。

A.11.4　オントロジーエージェント（OA）

OKBC に接続し、オントロジーサービスを ACL で行うためのエージェントを開発した。図表 A.11.4-1 に概要を示す。

A.11.5　今後の予定

最新の仕様である FIPA 99 準拠となるように、プラットフォームおよび ACL の機能拡張を行う予定である。



図表 A.11.4-1　オントロジーサービス

A.12　日立製作所におけるエージェント技術への取り組み

日立製作所システム開発研究所
豊内順一

A.12.1　概要

エージェント技術に関して、日立製作所が取り組んでいる三つの主な活動について概説

する。

（1）エージェント基盤製品化

エージェントアーキテクチャに基づく、アプリケーション非依存なソフトウェア共通基盤である「エージェント基盤」の製品化と、実アプリケーションへの適用。

（2）自律分散サービスシステム（ADSS*）技術研究開発

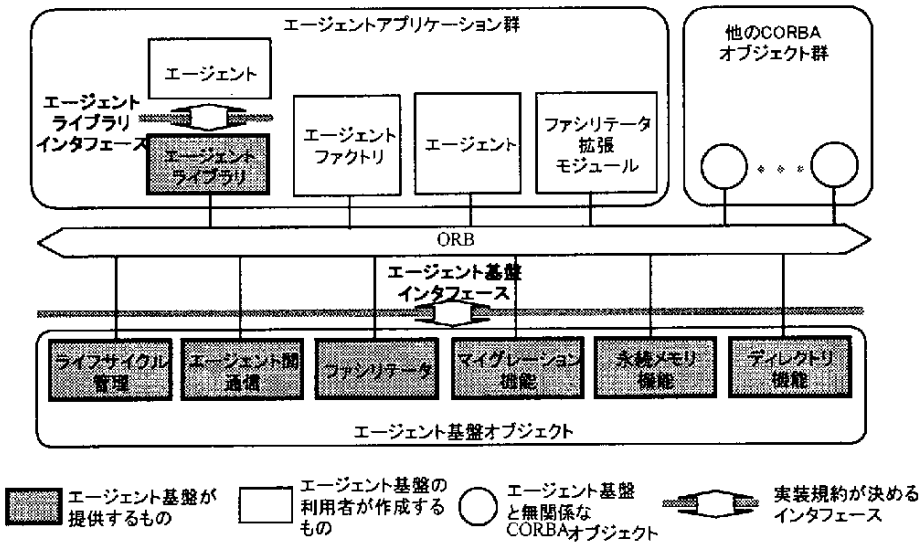
複数の情報システムの連携やサービスの統合を目的とし、エージェント概念に基づくADSSアーキテクチャの確立と基本技術の研究および、ミドルウェアの開発。

（3）技術の普及、標準化、および各種研究プロジェクトへの提案

上記の研究開発成果のOMG**, TINA-C***などの国際標準化団体への提案、および標準化推進。

A.12.2 エージェント基盤

エージェントを用いたアプリケーションの共通基盤として、図表 A.12.2-1 のような機能を実現したソフトウェアを開発している。他のエージェントシステムとのインタオペラビリティを確保するため、FIPA97 仕様に準拠したエージェント基盤間インタフェースとACLをサポートする。



図表 A.12.2-1 エージェント基盤の構成

A.12.3 自律分散サービスシステム技術

ADSSの概念では、連携するシステムやコンポーネントを、サービスを要求する「リクエスタ」、サービスを提供する「プロバイダ」、サービス情報の要求と提供とを仲介する「メディエータ」の3種のエージェントとしてモデル化し、これらから構成されるコミュニティの管理技法を提供する。この概念に基づいたアプリケーションの開発を支援するミドルウェアを開発するとともに、テレコム分野、公共情報サービス分野を中心としたシステムへの適用を進めている。

A.12.4 技術の普及、標準化および各種研究プロジェクトへの提案

OMG において、1997 年 9 月に日立主導で ADSS 技術を議論するためのワーキンググループ(ADSS DSIG)を設立し、ADSS 概念の普及、技術の標準化を推進している。OMG での提案技術のうち、テレコム分野への適用技術に関しては、TINA-C においても提案、標準化を行っていく。

* ADSS: Autonomous Decentralized Service System,
** OMG: Object Management Group,
*** TINA-C: Telecommunications Information Networking Architecture Consortium

---

A.13 オムロン株式会社におけるインタフェース・エージェントへの取り組みについて

オムロン株式会社技術本部ＩＴ研究所ファジィ・ヒューマンメディア研究室
中嶋　宏

A.13.1 はじめに

弊社ＩＴ研究所においては、「人に優しいシステム機器」に関する研究開発を進めている。その中から、特に人と機械のインタフェース技術に関する研究開発事例を選んで紹介する。

A.13.2 心のモデルとその応用事例

(1) MaC モデルの開発

我々は、人と機械のコミュニケーションを自然で豊かなものにするためには、心のシミュレーションが必要であると考えている。この豊かなコミュニケーションの実現に向けて、MaC モデル (Mind and Consciousness model) と呼ぶ心のモデルを開発し、これを次に示すようなアプリケーションに応用し、その効果を検証中である。

(2) サイバーキャラクタ “クリーチャ・ワールド”〔1〕

コンピュータアニメーションで表現された仮想世界中を、3 体のクリーチャと呼ぶキャラクタが自律移動するシステムである。クリーチャ達は、相互にコミュニケーションすることに加えて、触覚センサやマウスを通じて、利用者とインタラクションすることができる。この事例では、MaC モデルを利用して、感情表出、自律的行動、個性表出などの機能を実現している (図表 A.13-1)。

(3) ペットロボット “たま”〔2〕

ペットロボットの開発動機は、従来の産業用ロボットとは異なった価値を与えるということにあった。産業用ロボットでは、その機能的な側面に価値を求めるが、ペットロボットは、その存在自体に価値に与えることを目論んでいる。具体的には、人とペットロボットのインタラクションを通じて、人に親近感や安心感などの精神的価値を与えるということである。人がペットロボットを触り、ペットロボットがそれに反応する。その反応を人が見て、さらにペットロボットを触りたくなる。このようなユーザとペットロボットの対話ループから、相互理解や愛着が生まれるとの仮説を持っている。こうした対話を実現するため、MaC モデルを活用した自律的行動の実現に加え、見た目のデザインや触感をできるだけ動物に近付けるなどの工夫を加えている（図表 A.13-2)。

(4) 擬人化エージェント *“Stellar”*〔3〕

“Stellar” は人間の容姿を持ち、音声対話を行うエージェントである。顔画像の識別およびトラッキング、音声認識、音位置検出などの認識系の機能に加えて、3D グラフィクスによって表情や身ぶりなどを表出する機能を統合したシステムである。ここでは、MaC モデルを対話制御に活用している。認識エラー発生時のリカバーや、話題の重要性に基づく柔軟な対話制御を実現している。加えて、表情や身振りを通じて MaC モデルで生成した Stellar の感情を利用者に伝えることができる（図表 A.13-3)。

A.13.3 テレフォニー・エージェント

電話という音声メディアのみを利用したインタフェース・エージェントの開発を進めている。応用として、電話転送、資料請求、各種照会業務などのコールセンタ業務の自動化を行っている（図表 A.13-4)。〔4〕。

a) 音声認識

認識エンジンは、米国 Nuance Communications 社と共同で日本語化を行った〔5〕。認識語彙については、数値や固有名詞を中心に文法をパッケージ化し取り揃えている。

b) 音声合成

合成エンジンは、ATR〔6〕から波形接続型音声合成 CHATR〔7〕を技術導入した。CHATR の最大の特徴は人間の肉声を再現できる点である。この特徴を活かして、録音音声と CHATR による合成音声の自然な結合によるシステム発話を実現している。

C) 音声対話

対話制御を容易に行うために、先に述べた MaC モデルを応用した対話エンジンの開発を進めている。一方で、VUI（Voice User Interface）というデザインコンセプトを設定して、人の振る舞い特性を最大限に利用した設計原理を追求している。これは、テレフォニー・エージェントの設計指針とも言える。

参考文献

〔1〕 Ushida, H., Hirayama, Y., and Nakajima, H.: Emotion Model for Life-like Agent and Its Evaluation, Proceedings of the 15th National Conference on Artificial Intelligence (AAAI-98), pp.62-69 (1998).

〔2〕 牛田，平山，中嶋，田島，齋藤：心のモデルに基づくインタラクティブエージェント，第4回知能情報メディアシンポジウム講演論文集，pp.165-172 (1998).

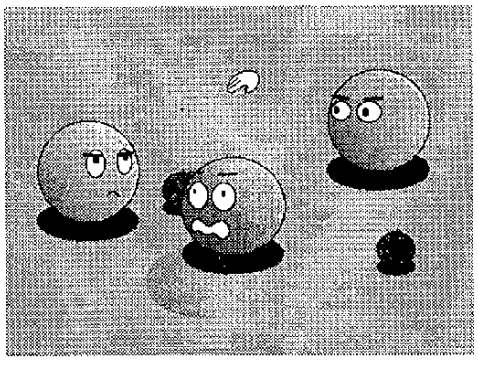
〔3〕 牛田，平山，中嶋：自律的行動決定モデルに基づくインタラクティブキャラクタ，インタラクション99論文集，情報処理学会シンポジウムシリーズ，Vol.99，No.4，pp.51-52 (1999).

〔4〕 http://www.omron.co.jp/cma/

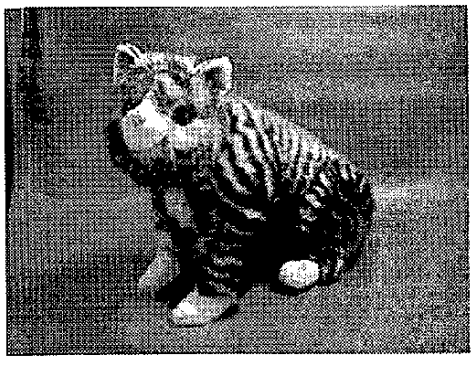
〔5〕 http://www.nuance.com/

〔6〕 http://www.itl.atr.co.jp/Japanese/index_j.html

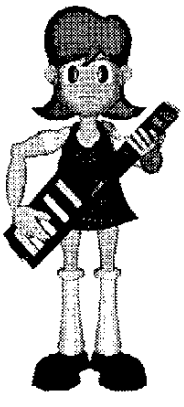
〔7〕 芦村和幸，“マルチメディア時代の音声合成…多言語音声合成システムCHATR”，インタフェース，Vol.24，No.8，pp.109-118(1998) .



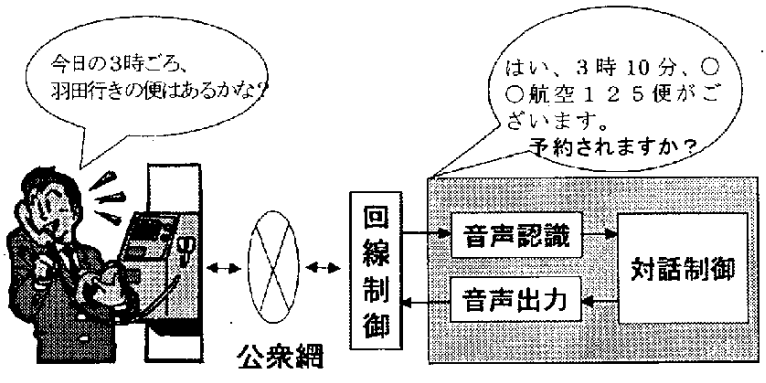
図表 A.13-1 サイバーキャラクタ“クリーチャ・ワールド”



図表 A.13-2 ペットロボット“たま”



図表 A.13-3 擬人化エージェント“Stellar”



図表 A.13-4 テレフォニー・エージェント

※ CHATRはエイ・ティ・アール音声翻訳通信研究所の登録商標です。

A.14 日本アイ・ビー・エム株式会社東京基礎研究所における エージェント技術への取り組み

日本アイ・ビー・エム（株）
中村 祐一

(1) Aglets

Aglets は Java ベースの移動エージェントシステムであり、1996 年夏に最初のバージョンを公開した（www.trl.ibm.co.jp/aglets/）。公開以来、３万件近いパッケージがダウンロードされ、このエリアではもっともポピュラーであり、今でも高い知名度を持っている。開発者間での意見交換も活発で、Aglets 専用のメーリングリストには 500 人以上の参加がある。現状では、後述するたび Can 以外には、リアル・アプリケーションはないが、大学などの研究機関を中心に、数多くのプロトタイプシステムが試作されている。さらに、エージェントというエリアを超えて、移動エージェントの存在をアピールした点での功績も大きく、「日経 BP 部門賞」「JAVA 大賞の準大賞」などを受賞している。

(2) たび Can

Aglets を実際の世界に適用するための活動として、エージェント・マーケットプレースのフレームワーク e-Marketplace を Aglets 上に開発した。これは顧客エージェントと店舗エージェントがやり取りをするための場所を実現したものであり、この枠組みを利用してたび Can という商用の旅行情報サービスを提供している（www.tabican.ne.jp）。e-Marketplace の特徴は現実にマルチエージェントシステムを構築するために、高いスケーラビリティ実現した点と、今後の EC を見据えて XML に基づく対話プロトコルを実現した点である。

(3) Caribbean

Aglets 上に e-Marketplace を実現することには、パフォーマンス上の限界が見えてきたので、新たにビジネス・エージェント・サーバー Caribbean として構築しなおしている。e-Marketplace と比べると、約 10 倍以上のパフォーマンスが得られており、高いスケーラビリティも確保されている。さらに、業務用のエージェント・サーバーということを念頭において、リカバリーの機構も実装しており、高いアベーラビリティの提供も期待できる。また、EJB（Enterprise Java Beans）との共存も考慮に入れて、EJB-Compliant になるような API を提供することも目指している。こうすることにより、将来の Java によるサーバー技術の構成要素の一つになることが期待できる。

(4) リアル・アプリケーション

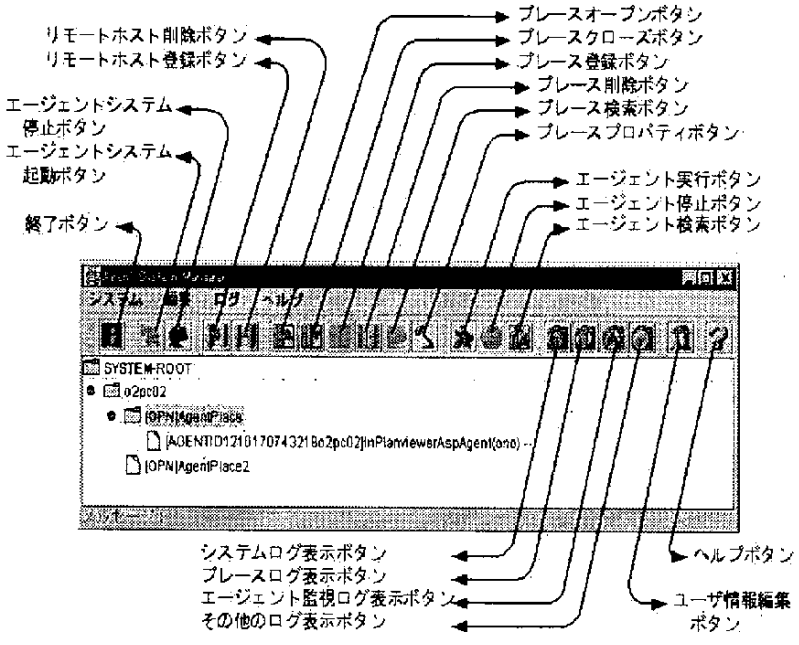
エージェント技術のインパクトを目に見える形にするにはアプリケーションが重要であることは言うまでもない。この際、未来を予感させるようなコンセプトを示し、それに沿ってアプリケーションを作っていくことが重要である。たび Can の構築にあたっては、複数のコマースサイトが移動エージェントで結ばれる構想を示し、そのスタートがたび Can であるという位置づけで大きな反響を得た。現在、同じ EC のエリアを中心に、通知サービスに関する実験をしたり、B2B にフォーカスした構想を検討中であり、いかにしてエージェント技術により世の中にインパクトを与えられるかについて考えている。

A.15　ＮＥＣにおける移動エージェント技術への取り組み

A.15-1　はじめに

NEC では、ネットワーク上の分散システムの開発を容易にするために、移動エージェント技術を用いた開発環境の構築を進めております。ここでは、Java 上の移動エージェントシステムの構築を目指す MIPLACE と、移動かつ分散可能なエージェントの構築を可能にする Mobidget について紹介致します。

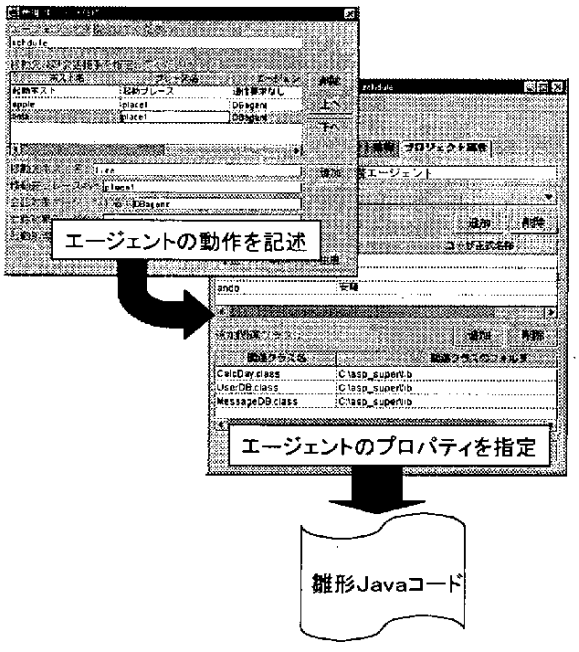
A.15-2　MIPLACE:Java 上の移動エージェントアプリケーション構築プラットフォーム



図表 A.15.2-1　MIPLACE のエージェント管理 GUI

MIPLACE (Mobile and Intelligent PLatform for Agent Communication Environment) は、Java を用いて移動エージェントシステムを構築することを目的に開発されました。図表 15.2-1 に示すようなシステム管理 GUI を備え、プレースやエージェントへの各種制御が一元的に、かつ容易に操作可能であることを特徴としています。提供するサービスは以下の通りです。

プレース管理 ····· プレースの生成、登録、削除等
エージェント管理 ····· エージェントの生成、削除等
モニタ機能 ····· 各種ログ情報の表示/蓄積
ユーザ管理 ····· 管理者や実行者の登録・管理
セキュリティ管理 ····· 起動/登録時のユーザ認証
障害管理 ····· ホストダウン等の障害検知
通信管理 ····· 簡便で安全なエージェント間通信
永続エージェント管理 · 状態保存、復元、検索等



図表 A.15.2-2　EasyMaker による雛型コード自動生成

また、移動ホストや通信する相手を GUI 上から指定するだけでエージェントコードの雛型を自動生成できる EasyMaker というツールを装備しています(図表 15.2-2 参照)。EasyMaker が、移動と通信の処理を記述した Java コードを生成しますので、あとはアプリケーション固有のクラスを追加することで、MIPLACE のエージェントを作成することができます。この他、移動エージェントの標準仕様である MASIF に準拠したエージェント操作インタフェースを装備するなど、システムインテグレーションに適したエージェント開発環境を提供しております。

A.15.3　Mobidget: 移動と分散機能を利用できるネットワークのためのプログラム言語

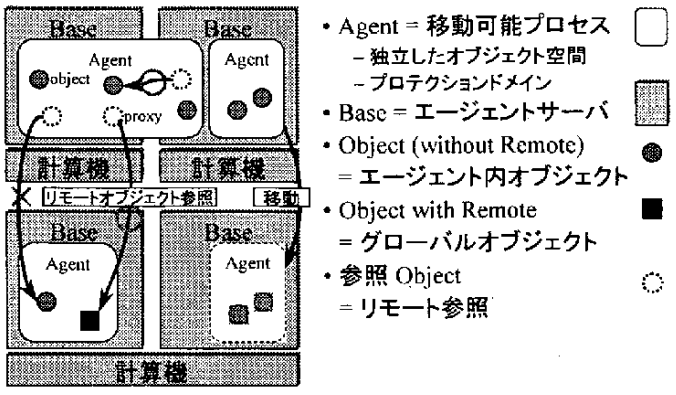
分散システムの開発を容易にするための新しいプログラム言語として Mobidget の研究開発を進めております。Mobidget の特長とするところは、エージェントの移動性と分散性にあります。

(1)移動性:

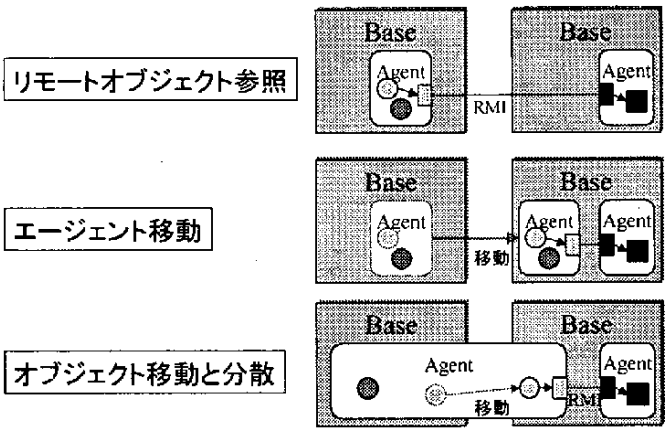
プログラム中の任意の位置に移動命令を挿入でき、移動命令実行直後の命令から、移動先で実行を再開できます。移動前後においては、実行場所以外に関するプログラムの内容に変更はなく、例えば、マルチスレッドで実行していたエージェントは、マルチスレッドのまま実行再開されます。

(2)分散性:

Mobidget における「エージェント」とは、論理的なオブジェクト空間を形成する単位であって、計算機という物理的な境界に依存しません。リモートオブジェクトの生成などを契機に、複数の計算機に分散して存在するエージェントを作成できます。分散エージェント内の通信(メソッドコール、フィールド参照等)は、同一計算機にあるオブジェクト間の通信と同様に一貫性を保って実行できます。



図表 A.15.3-1　Mobidget の基本構成要素



図表 A.15.3-2　Mobidgetにおける遠隔サーバとの３つの通信手段

エージェント間の通信は、特別なインタフェースを用いたリモートメソッド呼出で実現します。Mobidget では、呼出先/呼出元のエージェントが突然移動してしまうことにも備えて、移動前後にも矛盾なく通信が成立するエージェント間通信を提供しております。また、効率的な不要メモリ回収を行う分散 GC も装備して、ネットワークにまたがる複雑なシステムを容易に構築することができます。

図表 A.15.3-1 に Mobidget の基本オブジェクト間の関係を示します。エージェントは、複数計算機に分散することも、移動することもできます。これら機能を用いて遠隔サーバの資源にアクセスする３つの実装方法の例を図表 A.15.3-2 に示します。リモートオブジェクト参照、エージェント移動、部分的なエージェント移動などを選択できることで、実行条件に合わせた通信性能の最適化などに有効です。

Mobidget は、以上に述べたような機能を備えることによって、計算機というハードウェアに依存しない論理的な実体であるエージェントを作成することが可能になり、柔軟なネットワークエージェントシステム構築に寄与できると考えております。

A.16　日本ユニシスにおけるエージェント技術への取り組み

日本ユニシス株式会社

A.16-1　概要

日本ユニシスにおいてのエージェント技術の適用例として，大規模広域分散システムを安全かつ容易に構築・運用するためのプラットフォームとして開発された CORBA 製品である「System ν」及び，次世代電子図書館研究開発事業の中の「移動エージェント機能」について紹介します。

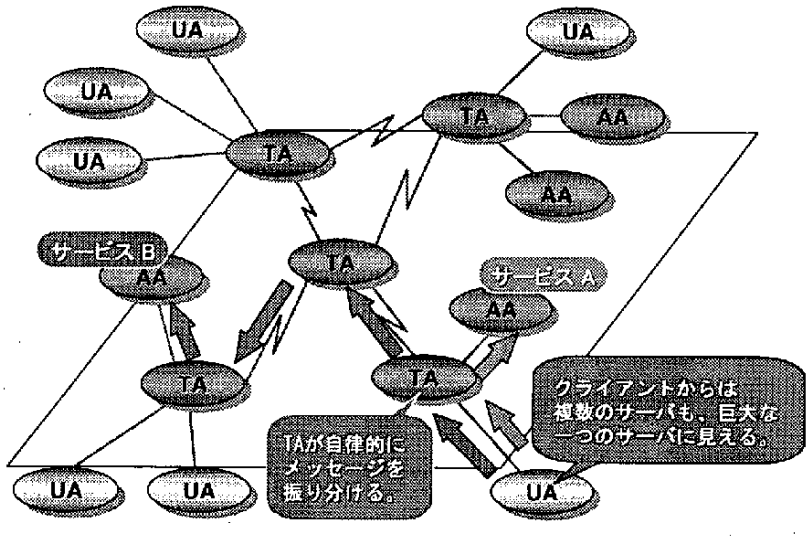
また，今後のエージェント製品開発のための調査研究として，FIPA やスタンフォード大学での活動を紹介します。

A.16-2　System ν （図表 A.16.2-1）

弊社が開発した Transactional CORBA 開発・実行環境です。分散オブジェクト環境上で発せられる処理要求やデータなどの通信制御や、分散オブジェクトの管理を行うために、エージェント機能を搭載しています。エージェントとは、データとそれを処理する手続きと、カプセル化したオブジェクト指向におけるオブジェクトをさらに機能拡張し、データや手続きを状況に応じて適切に処理する自律的なプロセス（実行可能プログラム）のことと定義しています。

SYSTEM νのエージェントは次の 3 種類が存在します。

1) TA (Transfer Agent　トランスファ・エージェント)
   SYSTEMνにおけるオブジェクトの管理とデータ／処理要求の制御を行うエージェント
2) UA (User Agent　ユーザ・エージェント)
   ユーザ・インタフェースを実現するエージェント
3) AA (Application Agent　アプリケーション・エージェント)
   処理（メソッド）を担当するエージェント



図表 A.16.2-1　System νのエージェントモデル

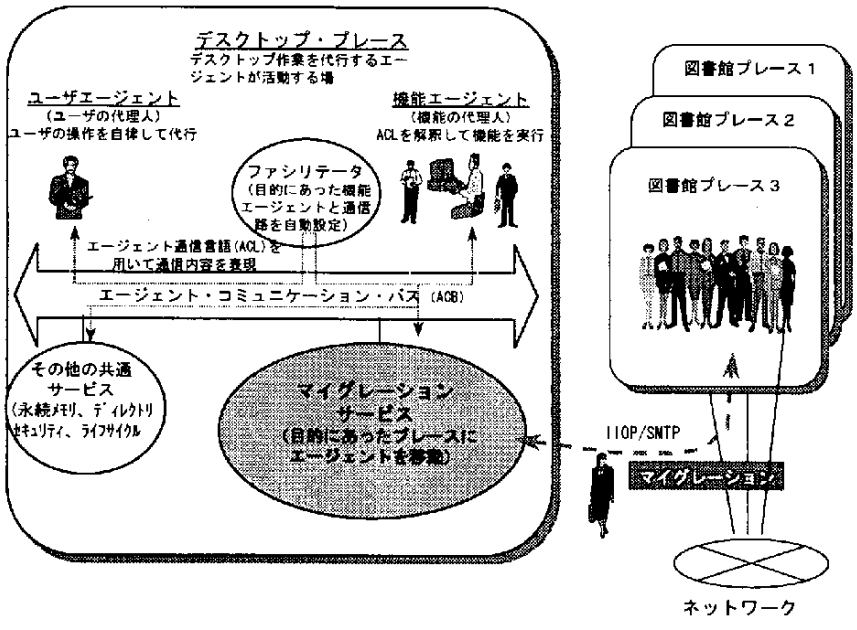
アプリケーションと利用者のフロントエンドとなる UA と、アプリケーションの本体となる AA は、TA を介して通信を行い、通信プロトコルには CORBA2.0 で定める IIOP(Internet Inter-ORB Protocol) を使用しています。これらのエージェントが協調動作することで、ミッションクリティカルな大規模分散システムの実現を容易化しています。

A.16-3 次世代電子図書館システム研究開発事業

日本情報処理開発協会（JIPDEC）で推進されている「次世代電子図書館システム 研究開発事業」(http://www.dlib.jipdec.or.jp/) において移動エージェント機能に関する研究開発を行っています。本移動エージェント機能の特徴は、

- ●CORBA 基盤上で構築（各サービスは CORBA オブジェクトとして実装）
- ●エージェント移動に使用するプロトコルは IIOP、SMTP を採用
- ●Java 言語によるエージェントの実装
- ●セキュリティフレームワークの構築

となっており、LAN、WAN 上で，セキュリティを考慮に入れたエージェント移動の実現を目指しています（図表 A.16.3-1)。



図表 A.16.3-1 エージェント基盤参照モデル

A.16.4 スタンフォード大学

米国スタンフォード大学へ要員を派遣し、EC システムの調査及び共同研究開発を行っています。活動の一つとして、スタンフォード大学でのエージェント研究の調査、エージェントの EC 分野における適用があげられます。

A.16.5 FIPA への参加

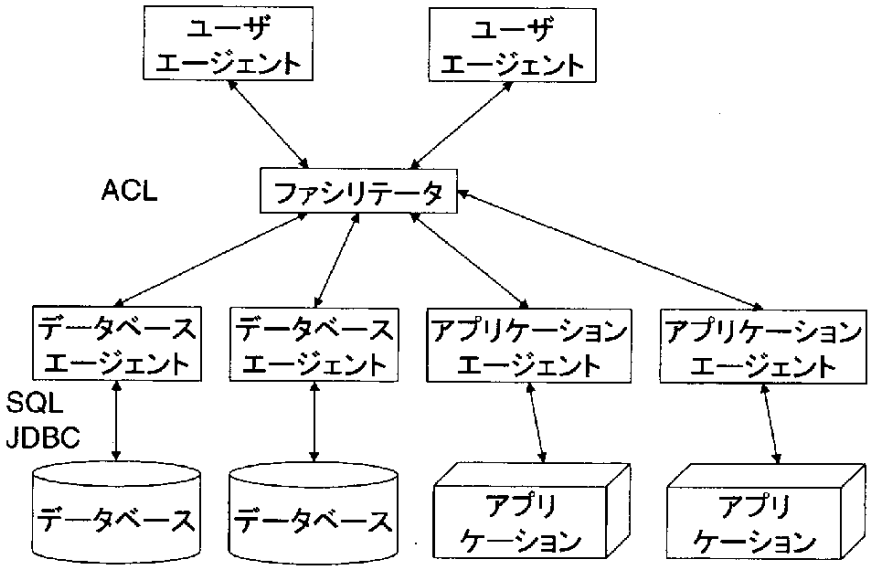
FIPA(The Foundation for Intelligent Physical Agents http://www.fipa.org)へ参画し、FIPA のアーキテクチャ調査と FIPA 対応製品開発を検討中です。

A.17 株式会社 富士通研究所 知能システム研究部

- 会話型エージェントシステム SAGE と INTERSTAGE AGENTPRO -

富士通研究所 知能システム研究部では、会話型エージェントシステム SAGE(Smart AGent Environment) [1] の研究開発を進めている。研究開発では SAGE を主に散在するデータベースやアプリケーション等の異種情報源の仮想統合への応用に適用してきた。この研究成果をもとに 1999 年 8 月に富士通がソフトウェア製品 INTERSTAGE AGENTPRO (以下 AGENTPRO と呼ぶ) として製品化し、出荷した。以下に AGENTPRO [2] の概要、および富士通研究所 知能システム研究部のその他のエージェント技術への取り組みを紹介する。

A.17.1 AGENTPRO の概要



図表 A.17.1-1 AGENTPRO の構成

ユーザエージェント、ファシリテータ (仲介エージェント) 、データベースエージェント、アプリケーションエージェントからなる。 これらのエージェントたちが ACL メッセージをやり取りしてサービスを提供する。

この会話型エージェントシステム AGENTPRO におけるエージェントたちは富士通が提供する CORBA 対応のミドルウェア製品 INTERSTAGE 上で動作し、ACL (Agent Communication Language) メッセージを用いて互いにやり取り (会話) する。

主な適用分野は企業内、企業間に分散して存在しているデータベースやアプリケーションの仮想情報統合である。製品では基本的に図表 A.17.1-1 のような構成をとる。

この AGENTPRO は会話型エージェントシステムの利点を生かしつつ、以下のような特徴を持つ。

a) 既存資産であるデータベースやアプリケーションを情報源として、容易にエージェ

ント化するためのデータベースエージェントやアプリケーションエージェントを持つ。

b) エージェント系へのユーザインタフェースを構築するためのAPIを持つ。

c) 情報源のエージェントはアドバタイズメッセージなどを送ることにより、ファシリテータに情報源の状態、メタデータを知らせることができる。

d) ファシリテータはオントロジ[3]やアドバタイズメッセージの情報に基づいて、各エージェントの違いを考慮したメッセージの仲介を行う。

e) 通信基盤であるCORBAシステムと連携してネーミングサービスやセキュリティ機能などを使用、提供する。

ただ単に物理的に分散しているものを統合するのではなく、もともと別々の起源を持つ情報源、あるいは別々に運用管理されている情報源を統合するための各種機能を有している。これは変化の激しい企業環境などで、各部署、各企業が情報源を独立に運用管理しながらも、変化に対応した形で統合した情報提供が短い時間で可能になるという点で重要である。

AGENTPROの具体的な適用分野は以下の分野を想定している。

a) **企業内のナレッジマネジメント**:多数のDB・Webサーバ・文書サーバの統合による情報・知識の共有

b) **既存システム統合**:新規業務と既存業務の一体化

c) **企業間EC /サプライチェーンマネジメント**:企業間の購買、調達、情報知識の統合と共有

d) **消費者向けのパーソナライズ**:各消費者の嗜好に合わせたタイムリな情報提供、消費者の購買傾向による推奨

実際の事例としては、現在富士通内で展開している「富士通全社SE情報システム(FIND Future)」がある。FIND Futureは富士通のSE、4万人に対し、SE向けの技術情報、製品情報、保守情報を提供するシステムである。利用者はキーワードとともに、カテゴリー/グループを指定でき、エージェントが束ねた数十のサーバを高速にアクセスできる。既に1998年10月から本運用に入っており、現在は1日に数千人が利用している。その大きな評価は、高速性(80%の要求が2秒以内に完了)と、各サーバが独立して運用可能なことによる運用性の高さである。

A.17.2 今後の研究開発-オントロジ

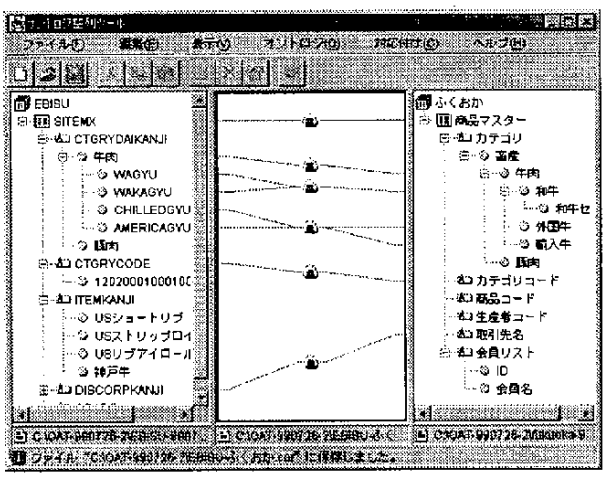
今後のエージェントの研究開発としては、現在のエージェントシステムの機能、性能、運用性などの更なる向上を考えている。

その中でもオントロジは重要であると考えている。別々の起源の情報源に関して、オントロジを明確化し、異なるオントロジの間の変換を実現することは、仮想情報統合にとって本質的である。

現在異なるオントロジを対応づけるビジュアルなツールOAT(Ontology Alignment Tool)を作成しており(図表A.17.2-1)、このツールは次期バージョンでAGENTPROに含まれる

[3]エージェントが使う語彙体系のことである。

予定である。



図表 A.17.2-1 OAT(Ontology Alignment Tool)
ファイル、データベースなどから読み込んだオントロジを左右に二つ並べて、そのオントロジの要素を互いに GUI で対応づけることができる。

A.17.3 FIPA 活動

エージェントに関する標準化団体 FIPA に関連した活動としては、以下のような項目を行っている。

a) **標準への提案**:オントロジに関する提案や KIF の FIPA Content 言語への採用の提案などを行なっている。KIF に関連した提案は FIPA 99 の Content Language Library に採用される予定である。

b) **相互運用性テスト**:AGENTPRO を FIPA 対応に変更したものと Comtec 社の AP (Agent Platform) との間で相互運用性テストを進めている。AP 間の通信の実現と Comtec AP への登録の確認という第一段階が終わっている。

A.17.4 まとめ

富士通研究所 知能システム研究部のエージェント技術への取り組みを、研究成果をもとに製品となった AGENTPRO、オントロジを中心にした今後の研究開発、FIPA 活動を簡単に紹介した。

富士通研究所 知能システム研究部では今後も各種の有用なソリューションを提供する基盤としてのエージェント技術の研究開発へ取り組んでいく予定である。

参考文献

〔1〕 T. Sugasaka, R. Masuoka, A. Sato, H. Kitajima, F. Maruyama, "An application of multi-agent technology to electronic commerce - SAGE:Francis-", Proceedings of I'MEDIAT98, September, 1998.

〔2〕 "INTERSTAGE" (AGENTPRO の概説書などへのリンクがある),
http://www.fujitsu.co.jp/hypertext/softinfo/product/net/INTERSTAGE

A.18　ATR 知能映像通信研究所におけるエージェント技術への取り組み

ATR 知能映像通信研究所
間瀬健二

A.18.1　概要

応用分野としてのエージェントインタフェース技術の研究と、基礎的分野としてのエージェントの社会性の研究を行なっている。

ATR 知能映像通信研究所は、映像を中心としたマルチメディア情報を駆使することにより、21 世紀の新しいコミュニケーション文化の創出に貢献するコミュニケーション技術の研究開発を目標にかかげている。そこでは、距離と時間を越えながらいかにも実際に会話をしているようなリアルな感覚を与えてくれるコミュニケーションと、現実にはありえないが人の感性を刺激して活性化する超リアルなコミュニケーションを実現するための技術が研究課題となっている。現在、研究所ではテーマを、(1) コミュニケーション環境生成技術、(2) エージェントインタフェース技術、(3) イメージコミュニケーション技術、(4) 人のコミュニケーション過程の分析、(5) バーチャルリアリティ技術、(6) アート&テクノロジーの融合、などのサブテーマに分けて研究をすすめている。

A.18.2　コミュニケーションの目的とその支援

人間同士のコミュニケーションにおいては、協同で作業をしたり作業の依頼や命令という目的のタスクを実行する際に、創造的発想、相互理解、自己理解、話題の整理および話題の展開などの重要な側面がある。我々は、コミュニケーションにコンピュータが介在して，これまで人間同士では達成できなかった密でクリエイティブなコミュニケーションを実現するために、このような側面の活動を支援するコンピュータシステムを開発したいと願っている。そして、そのアプローチとして、使われるツールやインタフェースを独立に考えるのではなく、インタフェースにタスクを取り込む形で，そこに自律性（外部からのコントロールではなく、自分の立てた規範にしたがって行動すること）や擬人性（人のように見えても、実際はそうでないコンピュータなどを人に見立てること）を持たせることによって、ただの自動プログラムではなく、エージェントとして捉えることにしている。これをインタフェース・エージェントとよんでいる。

A.18.3　インタフェースエージェントの擬人性

エージェントが有するあるいは作業している各種リソース（データ、知識ベース、対象物など）をパターン化して表示したり、またエージェントの活動状況（煩雑度、困難度、結果の評価など）を表示することは、人間がエージェントを理解するのに手助けとなる。その際にインタフェースエージェントのジェスチャや表情で表すことができる対象として、忙しくデータを検索しているエージェントが走り回ったり、目がまわる表情をつけることなどが考えられる。我々が物事を記憶したり考えたりするときに自分の体の全体や一部をリファレンスにすることがあるように、エージェントを擬人化することはその理解の補助にもなると考えられる。

A.18.4 主なプロジェクト

(1) C-MAP (Context-aware Mobile Assistant Project)

知識や興味を共有する人の間のコミュニケーションを促進する環境を構築することを目標として、エージェント技術や情報可視化技術を研究している。博物館、美術館、研究所公開、ショールームといった、人や知識が集まる場所を想定し、見学者一人一人の状況や興味に応じた展示ガイドをする個人アシスタントシステムを構築することを研究のテストベッドとしてプロジェクトを進めている。アプリケーションとしてメタミュージアムと呼ぶコンセプトも提案している。

(http://www.mic.atr.co.jp/dept2/c-map/index-jp.html)

[主なサブゴール]

- 物理的文脈および意味的文脈を認識して、個人の状況に合わせた適応的情報サービスの提供
- 擬人化キャラクタによる親密性の向上
- 実空間と仮想空間の相互強化
- 人と人のネットワーク構築支援（コミュニティ支援）

[関連技術]

- モバイルコンピューティング
- 情報可視化
- ユーザインタフェース
- 文脈認識（位置検出，興味獲得）
- システムインテグレーション
- モバイルエージェント

[実験サービス]

- 研究所公開ガイドシステムとして
  a) 1997年11月ＡＴＲ研究発表会で来訪者に端末を貸し出して実験を実施。
     サブノート型携帯パソコンを30台用意し、のべ170名の来訪者が利用。
  b) 1999年11月の同じ研究発表会では，PalmGuideと呼ぶ，IBM WorkPad型の携帯端末を70台用意して実験。
- 学会サービスとして
  人工知能学会の２０００年度全国大会では研究発表の、オフィシャルガイドサービスとして実施する予定。

(2) 人間コンピュータ間のインタラクションにおける社会心理学的要因の分析

コンピュータ（あるいは，ソフトウェア，ロボット）が人間社会に入ってくるときに、礼儀や個性といった社会的要因がインタラクションに及ぼす影響をおもに心理的実験により分析し、エージェントやインタフェースの設計に反映する。次のような仮説が実験的に確かめられている。

- コンピュータとの互恵性の成立（と日米文化的差異）
- 「人らしさ」の帰属は社会的環境による

(3) その他

- 画像処理，音声処理による，ユーザの状況，動作，感情認識技術
- コンピュータグラフィックスによる，アバター（化身）の表示と制御，仮想変身
- ネットワーク上の仮想遠隔会議における仮想人物の生成と会議進行
- 仮想ダンスなど協調空間の制御

A.19　技術研究組合新情報処理開発機構　マルチモーダル機能シャープ研究室

－円滑な対話機能を持つインタフェースエージェント－

シャープ(株)　システム開発センター内
三吉　秀夫

シャープ（株）は通産省が推進するリアルワールドコンピューティングプロジェクトに参画し、社内に設置されたRWCPシャープ分散研究室（「マルチモーダル機能シャープ研究室」）において円滑な対話機能を持つマルチモーダルインタフェースエージェント：MAICO（Multimodal Agent Interface for COmmunication）の研究開発を進めている。本稿では、その概要を紹介する。

A.19.1　はじめに

デジタル情報家電機器の普及に伴い、より使いやすいユーザインタフェースが望まれる。現在のコンピュータのユーザインタフェースはマウス・キーボードを用いたGUIが主流であるが、上記のような将来の利用場面では必ずしも適していない。

マルチモーダルインタフェースは、人間が普段のコミュニケーションに用いる手段（音声、表情、身振り・手振り）をマンマシンインタフェースに応用しようというものであるため、訓練が不要で誰にでも使いやすい次世代ユーザインタフェースとして有力視されている。特にインタフェースに擬人化されたキャラクタを置く擬人化エージェントは更に親近感を与えるユーザインタフェースとして各所で研究開発が行われている。

我々が開発しているMAICO〔1,2〕もこの流れを汲むものであるが、他所では言語理解に重点を置くのに対し、我々は非言語情報処理に重点を置くのが特徴である。この種のインタフェースエージェントは音声対話が中心であるため、言語処理の重要性はいうまでもないが、我々が敢えて非言語情報に注目するのは、それらが持っている対話を円滑化する機能（対話促進機能）に注目しているからである。我々は、力学系を用いて音声のパワー・ピッチ情報をもとに相づちを打たせるマルチモーダルインタフェースエージェントMAICOを試作した。本稿ではそれについて紹介する。

A.19.2　非言語情報と対話促進機能

人間同士のコミュニケーション、特に話がはずむ会話においては、双方の対話者は音声言語を用いて意図を伝達するのはもちろんであるが、更に身振り手振りを交えたり、伝達される意味内容によって表情を変えたり、適切なタイミングで相づちを打つことによって対話を円滑に進めている。これらの非言語情報は、対話で交わされる意味内容には直接関連してはいないが、対話を活発化させ円滑に進めていくために重要な役割を担っている。

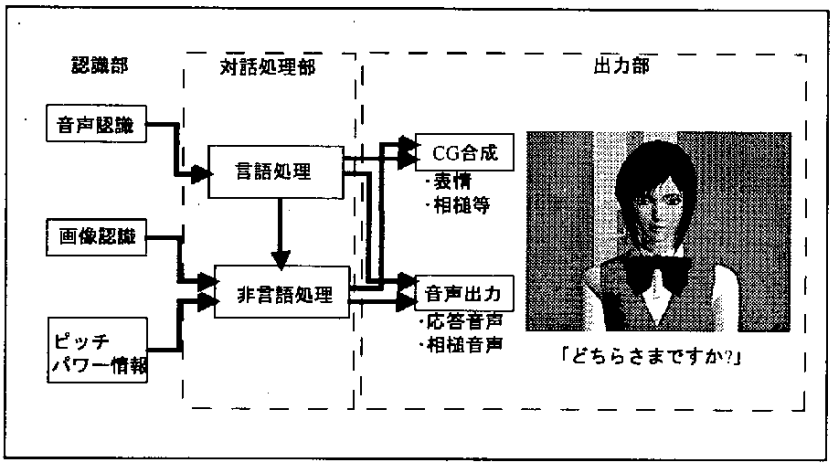
一般にマルチモーダルインタフェースは音声やジェスチャを用いたインタフェースであると言われている。しかし単に音声認識機能とジェスチャ認識機能を組み合わせただけでは現在のＧＵＩに代わる次世代のユーザインタフェースになるとは思われない。たとえ音声認識やジェスチャ認識を使っても、応答の仕方が悪ければキーボードで入力するのと何ら変わらず、却って使い勝手が悪くなる。我々はマルチモーダル情報をインタフェース

に利用することの本質は、この非言語情報が持つ対話促進機能を利用することにあると考える。

これを積極的に利用することにより、あたかも親しい友人とお喋りをするような要領で、機械と対話が出来るようになる。これによってユーザインタフェースの重要な要件である、快適性、親近感を備えた、誰にでも使い易いユーザインタフェースが実現できる。我々が非言語情報の注目するのは以上の背景による。

A.19.3 マルチモーダルインタフェースエージェント MAICO

本節では、我々が開発中の MAICO について述べる。MAICO は音声対話型のユーザインタフェースを指向した擬人化エージェントであり、音声のパワー・ピッチ情報を利用して相づちを打つのが特徴である。図表 A.19.3-1 にシステム構成図を示す。対話処理部における非言語処理部が相づちを生成するモジュールである。



図表 A.19.3-1 システム構成図

ここでは「相槌をうちたい」という気持ちに対して力学系対話モデル〔3〕を適用する。これはある時刻 t におけるシステムの状態を x、ユーザから得られる状態を y とするとき、$fint$ をシステム自身の状態、$fext$ をユーザの影響とみなし、システムの状態の変化 q を(1)－(4)式で表すものである。

$$\frac{dx}{dt} = q \qquad (1)$$

$$\frac{dq}{dt} = f_{\mathrm{int}}(x) + f_{ext}(y) \qquad (2)$$

$$f_{\mathrm{int}}(x) = -\omega^2 x \qquad (3)$$

$$f_{ext}(y) = \begin{cases} b(y-a)^2 + c & if \quad y < 0 \\ -b(y+a)^2 - c & otherwise \end{cases} \qquad (4)$$

ここでωは周波数、a、b は正の定数である。これらの感情は時間によって変化する。ある時刻 t におけるシステムの状態の変化は、システムの状態の変化量とユーザの気持ちの

状態変化量の和として定義されることを示している。これはシステムの内部状態だけでなく、ユーザの振る舞い（外部パラメータ）からも影響を受けることを示す。即ち、ユーザに対する反射的な反応しか示さない受動的なシステムではなく、自律的な反応を持ち、且つ、ユーザの気持ちの変化に応じた反応を示し、自分自身で対話に関わることのできる能動的なシステムとなっている。

MAICO はユーザが語り掛けると「はい、えぇ」と相づちを打ち表情豊かに応答するとともに、あらかじめ設定したタスクを実行する。現在までに、将来の応用例として「留守番テレビ電話」のエージェントとして、伝言を受け付けるデモシステムを試作した。

現在は、非言語情報として音声のパワー・ピッチのみを利用しているが、発話速度や各身体部位の動き情報などもユーザの何らかの内部状態（感情）を表出していると考えられるので、今後扱っていきたいと考える。

参考文献

〔1〕 S. Seki, et al, Multimodal Agent Interface for Communication, 2000 Real World Computing Symposiumum Proceedings, 2000.

〔2〕 向井理朗ほか、感情を持つマルチモーダル対話エージェントシステムの提案、人工知能学会、SIG-SLUD, 1999 年 10 月。

〔3〕 関進ほか、力学系対話モデルを用いた相槌システム、情報処理学会全国大会、1999 年 3 月.

A.20 横河電機におけるエージェント技術への取り組み

横河電機（株） 情報技術開発センター

和田英彦

A.20.1 はじめに

横河電機とエージェント技術との関わりは、OAA (Open Agent Architecture) を導入して評価用アプリケーションを試作したのが最初である〔1〕。この段階では評価用アプリケーションの試作と評価をするに留まっていた。

その後、モーバイルエージェントを利用して、柔軟性に優れた次世代の生産情報制御システムをターゲットとした研究開発が開始され、エージェントとの本格的な関わりが始まった〔2〕。本稿では、次世代の生産制御システムへのエージェントの適用についての概略を紹介する。

A.20.2 背景とアプローチ

将来の生産制御システムを考える場合、

1) 高品質、低コスト、タイムリーな生産への要求
2) 市場が急変したり、嗜好が突然変わる不確定性
3) 納入から保守、改良、廃棄までのライフサイクルサポート

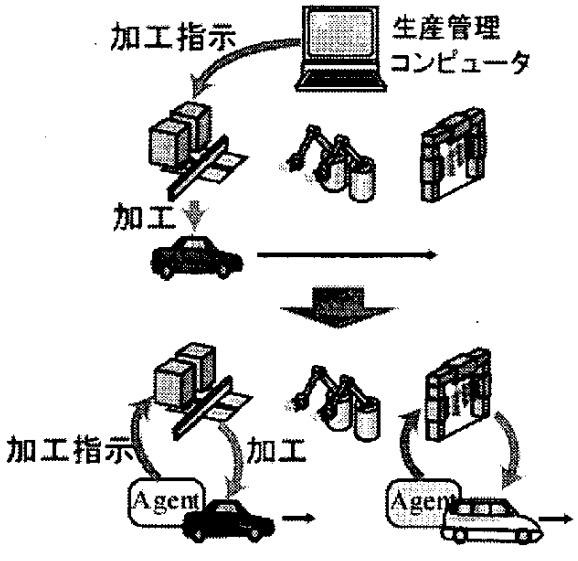
について考慮する必要がある。これらの背景を元に、次世代の生産制御システムとして実現すべき項目の一つとして、「要求に即応できる柔軟な生産ライン」の構築が求められている。横河電機ではこの要求を満たすべく、変種変量生産に対応できる生産ラインを実現することを目指した研究開発を行っている。

従来の生産システムで使われている階層型のシステムでは、生産ラインの構成を変えて別の製品を生産しようとする場合、システム内の多くの箇所でソフトウェアを修正する必要があったり、生産設備を拡張していった場合に、管理用コンピュータで負荷の集中が起こる、などの柔軟性に関わる問題点があった。変種変量生産を実現できる柔軟性を持つシステムでは、

1) 生産ラインの変更を行なうための装置の追加、変更や削除に対して、システム全体への影響を極力与えずに対応できること
2) 製品の改良や特注に対しても個別に生産可能であること

が必要であり、従来方式の階層型システムでは十分に対応することはできなかった。

上述の要求を実現するために、我々は個々の生産物と、生産ラインに設置された装置に自律性を持たせたシステムを提案している。生産物と装置それぞれにエージェントを対応させることにより、自律性を持った実行主体としてとらえ、エージェントを個々の実行主体を実現する基本構成要素として利用することで、協調性および自律性を持たせた分散型システムの実現を目指している。図表 A.20.2-1 にその概念を示す。

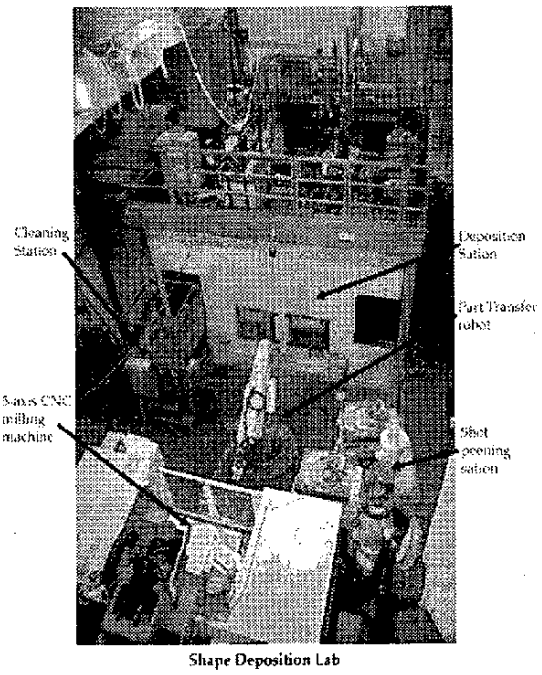


図表 A.20.2-1　概念図

A.20.3　プロトタイプシステム

我々は、エージェントを用いた生産制御システムの実現性を示すためにカーネギーメロン大学(CMU) Shape Deposition Manufacturing(SDM)実験室と共同で、実験室の機械加工システムを使ってプロトタイプシステムを構築した。[3]

実験室では、工作物はパレットと呼ばれる金属製の板の上に作られていく。パレットは、パレット移動用のロボットによって、パレット受け取り機構を持った複数の加工装置にセットされる。現在のSDMテストベッドは、CNC、Deposition、Shot-peening、Cleaningの4つの装置から構成される。Deposition装置も複数の工程をまとめるために別のロボット装置を使用している。実験室の概観を図表A.20.3-1に示す。



図表A.20.3-1　概念図

SDM実験室のシステムは、長年に渡って学生によって改良や拡張が行なわれてきている。そのようなad hocな拡張の結果、システム構成が複雑となってきており、システムの維持管理や新しい装置を付け加えることが難しくなっている。また、システム全体がひとつのコントローラにより制御される典型的な集中型制御システムであり、二つの工作装置を同時に使用することができない、などいくつかの改善すべき点が発生している。これらの問題を解決するために、我々が提案しているコンセプトをアーキテクチャを用いて、実験室の加工装置制御システムを開発した。

プロトタイプシステムでは、開発言語として Java を使用し、移動エージェントのプラットフォームとして、IBMが開発しているAgletsを用いた。開発したプロトタイプシステムでは、工作機械の並行利用やソフトウェアの修正なしにシステム構成を変更することを実現し、我々が提案しているコンセプトを実証した。

参考文献

〔1〕 井上，根岸，日野，秋本，“連邦アーキテクチャを用いた会議予約エージェントシステムの試作と評価”,情報処理学会ソフトウェア工学研究会資料 112-8,pp.57-64, November 1996.

〔2〕 星，“エージェントによる次世代生産システムの課題と展望”，人工知能学会研究会，SIG-J-9801-2，pp.5-10，December 1998.

〔3〕 H. Wada, Y. Sakuraba, M .Negishi, T. Yamakawa, K. Kubo, Y. Kashiyama, “A Machinery Control System Using Mobile Agents’”, Proc. of the 4th International Symposium on Autonomous Decentralized Systems, pp.124 - 131, March 1999.

資料編B　エージェントシステム
開発プロジェクト一覧

資料編 B. エージェントシステム
開発プロジェクト一覧

B1. 北　米
B2. 欧　州
B3. 日　本
B4. アジア
B5. オセアニアその他

B1　北米

北米　アメリカ

プロジェクト名	制作グループ
Agent Web	University of Maryland Baltimore County
Amalthaea	The Software Agents group of the MIT Media Laboratory
Butterfly	The Software Agents group of the MIT Media Laboratory
BUZZwatch	The Software Agents group of the MIT Media Laboratory
Expert Finder	The Software Agents group of the MIT Media Laboratory
HeretoThere	The Software Agents group of the MIT Media Laboratory
HIVA : TTT Toolkit	The Software Agents group of the MIT Media Laboratory
Impulse	The Software Agents group of the MIT Media Laboratory
Kasbah	The Software Agents group of the MIT Media Laboratory
Let's Browse	The Software Agents group of the MIT Media Laboratory
Letizia	The Software Agents group of the MIT Media Laboratory
Market Maker	The Software Agents group of the MIT Media Laboratory
MindShare	The Software Agents group of the MIT Media Laboratory
Mobile Agents for Routing Discoverry	The Software Agents group of the MIT Media Laboratory
Remembrance Agent	The Software Agents group of the MIT Media Laboratory
Straum	The Software Agents group of the MIT Media Laboratory
Tete-a-Tete	The Software Agents group of the MIT Media Laboratory
Trafficopter	The Software Agents group of the MIT Media Laboratory
Yenta	The Software Agents group of the MIT Media Laboratory
Calendar Agent	The Software Agents group of the MIT Media Laboratory
Challenger	The Software Agents group of the MIT Media Laboratory
Friend of a Friend Finder	The Software Agents group of the MIT Media Laboratory
Maxims	The Software Agents group of the MIT Media Laboratory

	コメント	ホームページ
	エージェント全般に関する研究へのリンク集．カテゴリ別に分類してある．	http://www.cs.umbc.edu/agents/
	情報リソースの発見・監視・フィルタリングを行うシステム	http://agents.www.media.mit.edu/groups/agents/
	個人プロファイルを元にユーザが興味のあるIRCチャンネルを推薦するシステム	http://agents.www.media.mit.edu/groups/agents/
	テキストのコレクションの中から特定のトピックやテーマを抜き出して追跡するシステム	http://agents.www.media.mit.edu/groups/agents/
	問題を解決するためにその分野の専門家を探してくれるエージェント	http://agents.www.media.mit.edu/groups/agents/
	エージェントにロケーションベースのタスクを割り振ることによってマーケットベースの戦略のシミュレーションを行うことが出来る	http://agents.www.media.mit.edu/groups/agents/
	インターネット上のアプリケーションの調和を図るシステム	http://agents.www.media.mit.edu/groups/agents/
	マルチパラメータ交渉が可能な売買エージェント	http://agents.www.media.mit.edu/groups/agents/
	商品の取り扱いに関する基礎的な変化を認識することを助けるシステム	http://agents.www.media.mit.edu/groups/agents/
	共通に興味がありそうな材料を提示することによりグループブラウジングをアシストする	http://agents.www.media.mit.edu/groups/agents/
	ユーザの興味を学びユーザが興味を持ちそうなウェブページを見つける	http://agents.www.media.mit.edu/groups/agents/
	エージェントベースの動的なマーケット創造のための電子商取引インフラ	http://agents.www.media.mit.edu/groups/agents/
	グループ内での知識共有をサポートする	http://agents.www.media.mit.edu/groups/agents/
	モバイルエージェントを用いてネットワークのマッピングをおこなう	http://agents.www.media.mit.edu/groups/agents/
	ユーザの現在の状況から判断し随時有用な情報を提供する	http://agents.www.media.mit.edu/groups/agents/
	マルチエージェント環境を用いてインターネット上で人々の存在を示すアプリケーション	http://agents.www.media.mit.edu/groups/agents/
	エージェントの仲介で売り手と買い手が交渉を行う	http://agents.www.media.mit.edu/groups/agents/
	分散したネットワークを用いて道路状況の収集と提供をおこなう	http://agents.www.media.mit.edu/groups/agents/
	共通の興味を持つ人々のクラスターを見つけるエージェントベースのシステム	http://agents.www.media.mit.edu/groups/agents/
	ユーザのアクションを監視することによりユーザの日程・スケジューリングルールなどを学習する	http://agents.www.media.mit.edu/groups/agents/
	分散されたCPU時間配分を完全に実行するマルチエージェントシステム	http://agents.www.media.mit.edu/groups/agents/
	個人的な質問の答えを得るためにソーシャルネットワークを用いることが可能なエージェントのネットワーク	http://agents.www.media.mit.edu/groups/agents/
	白紙の状態から学習する問題を解決するために協力するエージェントを用いたE-mailフィルタリングアプリケーション	http://agents.www.media.mit.edu/groups/agents

Mondrian	The Software Agents group of the MIT Media Laboratory
NewT	The Software Agents group of the MIT Media Laboratory
PDA@Shop	The Software Agents group of the MIT Media Laboratory
AmEC Infrastructure	The Software Agents group of the MIT Media Laboratory
Aircraft Maintenance System	The Software Agents group at Carnegie Mellon University
Joccasta	The Software Agents group at Carnegie Mellon University
Larks	The Software Agents group at Carnegie Mellon University
Matchmaker	The Software Agents group at Carnegie Mellon University
MokSAF	The Software Agents group at Carnegie Mellon University
Pleiades	The Software Agents group at Carnegie Mellon University
RETSINA	The Software Agents group at Carnegie Mellon University
Warren	The Software Agents group at Carnegie Mellon University
WebMate	The Software Agents group at Carnegie Mellon University
Multiagent System Research Group	Multiagent Systems Research Group at Washington University St. Louis
Multi-Agent Systems Laboratory	Multi-Agent Systems Laboratory at University of Massachusetts Amherst
Database Laboratory	Database Laboratory at North Carolina State University
Autonomous Networked Tactical Sentries	Center for Information Technology at University of South Carolina
D'Agent	ActComm Project at Dartmouth College
Market-based resource control	ActComm Project at Dartmouth College

	ビデオを用いプログラマでないドメインエキスパートにオブジェクトや手続きの構成を表現させる良く学習するグラフィカルエディタ	http://agents.www.media.mit.edu/groups/agents/
	Usenetニュースのプレゼンテーションをパーソナライズするためにキーワードベースのフィルタリングや遺伝アルゴリズムを用いる	http://agents.www.media.mit.edu/groups/agents/
	買い物を対象としたポイントオブセールスのためのハンドヘルドコンピューターとモバイルエージェントを統合する	http://agents.www.media.mit.edu/groups/agents/
	エージェントの仲介による電子商取引の研究	http://ecommerce.media.mit.edu/
	航空機の修理や文書化の補助を行うためにRETSINAエージェントを用いるウェアラブルソフトウェア	http://www.cs.cmu.edu/~softagents/
	共同の計画を立てているタスクにおいてチームの意思決定の効力を増加させるように設計されたプロジェクト	http://www.cs.cmu.edu/~softagents/
	プロバイダ・リクエスタ・マッチメーカエージェント間のマッチメーキングサービスのための一般的なエージェント能力記述言語	http://www.cs.cmu.edu/~softagents/
	サービスを提供するエージェントとサービスを提供するエージェントのコネクションを仲介するミドルエージェント	http://www.cs.cmu.edu/~softagents/
	チームの意思決定やルートプランニングのための仮想環境やチームコーディネーションをサポートするソフトウェアシステム	http://www.cs.cmu.edu/~softagents/
	複雑化するマルチユーザのタスクに分散されたRETSINAフレームワークを適応することをねらったプロジェクト	http://www.cs.cmu.edu/~softagents/
	分散インテリジェントソフトウェアエージェントを開発するためのアーキテクチャ	http://www.cs.cmu.edu/~softagents/
	インターネット上ですでに利用可能なリソースにアクセスするために分散エージェントアーキテクチャを用いる投資運用アプリケーション	http://www.cs.cmu.edu/~softagents/
	WWWブラウジングのために検索を改良しユーザの興味を学習するパーソナルエージェント	http://www.cs.cmu.edu/~softagents/
	電子商取引における交渉の自動化や契約パラメータの最適化などの研究	http://siesta.cs.wustl.edu/~mas/
	洗練されたAI問題解決やマルチエージェントの研究	http://mas.cs.umass.edu/index.shtml
	マルチエージェントシステムを用いた遠距離教育やヘルスケア情報システム等の研究	http://www.csc.ncsu.edu/info/dblab/www/
	音を聞いたり物を見たりするセンサーとソフトウェアエージェントを結合したインテリジェントセンサーの開発やそれらの能力や応用の研究	http://www.engr.sc.edu/research/CIT/
	複数言語・セキュリティ・フォールトトレランスなどに焦点を当てたモバイルエージェントシステム	http://actcomm.dartmouth.edu/
	モバイルエージェントのためのリソース配分制御のためのマーケットベースの制御手法	http://actcomm.dartmouth.edu/

Network sensing and planning 2	ActComm Project at Dartmouth College
Cooperation among self-interested agent	DAI-Hards group at University of Tulsa
Evolving Cooperative Groups	DAI-Hards group at University of Tulsa
Evolving coordination strategies	DAI -Hards group at University of Tulsa
An automated intelligent meeting scheduling system	DAI -Hards group at University of Tulsa
Multiagent Case-Based Learning	DAI -Hards group at University of Tulsa
Multiagent Soccer	DAI -Hards group at University of Tulsa
Softbots	Softbot group at University of Washington
Consortium for Integrated Intelligent Manufacturing Planning and Execution	Laboratory for Advanced Information Technology at University of Maryland Baltimore County
Extended Enterprise Coalition for Integrated Collaborative Manufacturing Systems	Laboratory for Advanced Information Technology at University of Maryland Baltimore County
Evolver	Laboratory for Advanced Information Technology at University of Maryland Baltimore County
Secure Agent Communication Languages	Laboratory for Advanced Information Technology at University of Maryland Baltimore County
Soar Intelligent Forces progect	Soar at University of Southrn California
TEAMCORE	Soar at University of Southrn California
Virtual Environments for Training	Soar at University of Southrn California
Business Rules Markup Language (BRML)	OASIS
Agent Society	Agent Society
Knowledge Sharing Technology Project	Stanford University Knowledge System Laboratory
Dynamics of Multiagent Systems	Xerox Palo Alto Research Center
HP Labs Bristol Agent Technology Groupe	HEWLETT PACKARD

	モバイルエージェントを用いてネットワークに強いアプリケーションを実現する	http://actcomm.dartmouth.edu/
	利己的なエージェントが協調しあう動機を持つための領域や環境の研究	http://euler.mcs.utulsa.edu/~research/
	マルチエージェントを用いて一般的な問題を解決するために効果的に協力できる複数の遺伝アルゴリズムを用いる	http://euler.mcs.utulsa.edu/~research/
	遺伝アルゴリズムを用いてエージェントの行動戦略を進化させる研究	http://euler.mcs.utulsa.edu/~research/
	関連する人々のミーティングスケジューリングが可能な自動スケジューリングエージェントによる分散交渉メカニズム	http://euler.mcs.utulsa.edu/~research/
	全体のパフォーマンスがますように個々に学習するマルチエージェントシステムの研究	http://euler.mcs.utulsa.edu/~research/
	協力した行動が要求され制限されたフォーメーションで戦略を立てるサッカーを通してエージェントの行動を評価する	http://euler.mcs.utulsa.edu/~research/
	人の代理でソフトウェアツールおよびサービスを利用するインテリジェントエージェント	http://www.cs.washington.edu/research/projects/softbots/www/softbots.html
	プランニングシステムがManufacturing exection Systemsによって実時間ファクトリフロアな豊富な情報にリンクするための基本的なアルゴリズムやテクノロジーの開発	http://www.cs.umbc.edu/www/lait/
	人々・アプリケーション・ソフトウェアエージェントが共同作業をするための新しいフレームワーク	http://www.cs.umbc.edu/www/lait/
	ソフトウェアの発展の手助けになるであろうデザインインフォメーションウェブの形における統合開発環境であるcapability packageの開発	http://www.cs.umbc.edu/www/lait/
	セキュリティーの概念を適用できるソフトウェアエージェント定義のためのセキュリティーアーキテクチャおよびそれらの設計開発	http://www.cs.umbc.edu/www/lait/
	実時間におけるエージェントの振る舞いのモデリングを行うためにSoarのアプリケーションの調査を行う	http://www.isi.edu/soar/
	分散された異種のエージェントから動的で強固なチームを構築する	http://www.isi.edu/soar/
	バーチャルリアリティとインテリジェントチュータリングテクノロジーを統合したトレーニングシステム	http://www.isi.edu/soar/
	エージェントコミュニケーション言語の研究	http://www.oasis-open.org/cover/sgml-xml.html
	Agentを研究している企業・グループなどが参加してエージェントの仕様の標準化を図る	http://www.agent.org/
	知識ベースの共有と再利用を可能にするであろうテクノロジーや方法の研究	http://www-ksl.stanford.edu/kst/kst-overview.html
	マルチエージェントシステムのダイナミクスに関する研究と電子商取引などへの応用	http://www.parc.xerox.com/spl/groups/dynamics/www/multiagent.html
	マルチエージェントコミュニティのデザイン・開発	http://agents.hpl.hp.com/overview.htm

Business Rules for Electronic Commerce	Intelligent Agents Project at IBM T.J. Watson Research
Information Economies	Intelligent Agents Project at IBM T.J. Watson Research
MailCat	Intelligent Agents Project at IBM T.J. Watson Research
Agent Building Environment (ABE)	Intelligent Agents Project at IBM T.J. Watson Research
Interaction Lab	University of Southern California
Exception Handling in Multi- Agent Systems	Adaptive Systems and Evolutionary Software Group at Center for Coordination Science Massachusetts Institute of Technology
Dynamic Domain Architectures	Adaptive Systems and Evolutionary Software Group at Center for Coordination Science Massachusetts Institute of Technology
Coordination&Control of Agent Based Systems	SRI International
Agent-Based Computational Economics(ACE)	Iowa State University
Computational Laboratories Group	Computational Laboratories Group at University of California Santa Barbara
Hyper-Economy Development Group (HEDG)	Hyper-Economy Development Group
SiFAs and their Negotiation Behaviour in Expert Systems	Worcester Polytechnic Institute Single Function Agents Research
Sail Boat design Project	Worcester Polytechnic Institute Single Function Agents Research
Conflicts and Negotiations in SiFA Based Design Systems	Worcester Polytechnic Institute Single Function Agents Research
SiFA Knowledge Acquisition Tool	Worcester Polytechnic Institute Single Function Agents Research
Knowledge Acquisition for SiFAs	Worcester Polytechnic Institute Single Function Agents Research
Learning in Multi-Agent System	Worcester Polytechnic Institute Single Function Agents Research
Cmattie	Conscious Software Research Group at University of Memphis
Intelligent Distribution Agent	Conscious Software Research Group at University of Memphis
Animate Agent Project	Artificial Intelligence Laboratory at University of CHICAGO
Prototype for an Open Intelligent Information System Project	University of Nebraska-Lincoln
Aruna's Software Agent World	Aruna (Brandeis University)
Agent Dynamic Load Balancing	Mobile Autonomous Software Agent Computing Group at University of Connecticut
AIRbots	AIRbots Inc

	電子商取引におけるインテリジェントルールベースエージェントの基礎的なコアテクノロジーとパイロットアプリケーション	http://www.research.ibm.com/iagents/
	複数のインテリジェントエージェントによって構成されるマーケットの調査	http://www.research.ibm.com/iagents/
	電子メールを分類するための学習アシスタント	http://www.research.ibm.com/iagents/
	エージェント開発環境	http://www.research.ibm.com/iagents/
	エージェントを用いたマルチロボットの制御	http://www-robotics.usc.edu/~agents/
	動的で複雑で誤りを生じやすい環境に適応できるマルチエージェントシステムの開発	http://ccs.mit.edu/ases/
	変化する環境や好ましくない環境に適応可能なソフトウェアシステムを開発するためのツールや方法の研究	http://ccs.mit.edu/ases/
	情報の収集や配布のような命令や制御機能のためのエージェントベースの技術の開発	http://www.erg.sri.com/projects/coabs/
	コンピュータを用いたエージェントベースのエコノミクスに関する研究	http://www.econ.iastate.edu/faculty/tesfatsion/ace.htm
	エージェントベースの財政学	http://www.econ.ucsb.edu/~cath/complab/
	エージェント間での関連付けられた値や知識の交換のモデル	http://www.lucifer.com/~sasha/HEDG/HEDG.html
	ひとつの機能・ひとつの観点しか持たないエージェントであるシングルファンクションエージェント同士の交渉言語等の研究	http://www.cs.wpi.edu/Research/aidg/SiFA/sifa.html
	エージェントを用いた帆船のデザインシステム	http://www.cs.wpi.edu/Research/aidg/SiFA/sifa.html
	SiFAs を用いたシステムの競合解消手法の研究	http://www.cs.wpi.edu/Research/aidg/SiFA/sifa.html
	SiFAs に知識を与えるためのツールの研究	http://www.cs.wpi.edu/Research/aidg/SiFA/sifa.html
	SiFAs エージェントの知識獲得にかんする研究	http://www.cs.wpi.edu/Research/aidg/SiFA/sifa.html
	マルチエージェントシステムにおける学習機能の研究	http://www.cs.wpi.edu/Research/aidg/SiFA/sifa.html
	グローバルワークスペース理論に基づくフレームワーク上のエージェント設計と実装	http://www.msci.memphis.edu/~csrg/
	知的なマルチエージェントシステムの研究	http://www.msci.memphis.edu/~csrg/
	人間と共に働くことの出来る知的なソフトウェア・ハードウェアエージェントの研究	http://people.cs.uchicago.edu/~firby/aap/
	コンセプトベースの情報検索ツールとユーザインタフェース・インテリジェントエージェントの統合に関する研究	http://www.cse.unl.edu/~jsd/
	マルチエージェントベースのスケジュール調整システムの研究	http://www.cs.brandeis.edu/~aruna/sa.html
	エージェントの動的な負荷分散に関する研究	http://www.engr.uconn.edu/masa/
	特定の職務のためにインターネット・イントラネットを動き回るモバイルエージェントの研究と応用	http://www.airbots.com/

Responsible Agent for Product-Process Integrated Design	Center for Electronic Commerce at The Environmental Research Institute of Michigan
Intelligent Assistants	Mindmaker Inc.
AgentBuilder	Reticular Systems Inc.
Agent Development Environment(ADE)	Gensys Corporation
Agentx	International Knowledge Systems
Gossip	Tryllian
iGENTM	CHI Software Inc.
Intelligent Agent Factory	Bits&Pixels
Customer Intelligence Agency	Bits&Pixels
Electronic Pharmacy System	Bits&Pixels
Search Sugar	Bits&Pixels
Intelligent Agent Library	Bits&Pixels
Intelligent Routing	Bits&Pixels
Microsoft Agent	Microsoft
Versatile Intelligent Agents	Kinetoscope
Agent TCL	ActComm Project at Dartmouth College
Bond	BOND Lab at Purdue University
Co-operative Agent Building Environment (CABLE)	Logica Corporation
Distributed Environment-Centerd Agent Framework	DECAF Group at University of Delaware
InfoSpiders	CASTOR Labotatories at University of Iowa
Intelligent query agents for e-commerce	CASTOR Labotatories at University of Iowa
Evolutionary agents to model societies and organizations	CASTOR Labotatories at University of Iowa
Java-based Agent Framework for Multi-Agent Systems	(Alan Galan) University of Chincinnati
Java Agent Template Lite (JATLite)	CDR Project Stanford University
Knowbot Systm Software	Corporation for National Research Initiatives
Mobiware	Mobiware Groupe at Columbia University in the City of New York
Open Agent Architecture	SRI International Artificial Intelligence Center
Agent Link	Agent Link

	異なった機能とプロダクトライフサイクルにおける異なるステージの製品の特徴を管理している人間のデザイナーを手助けするエージェント	http://www.erim.org/cec/rappid/index.htm
	スケジューリングや電子メールの監視などを行うインテリジェントアシスタントの研究・開発	http://www.mindmaker.com/
	エージェント開発環境の研究開発	http://www.agentbuilder.com/
	エージェント開発環境の研究開発と応用ツール	http://www.gensym.com/
	エージェント開発環境の研究開発	http://www.iks.com/
	モバイル秘書エージェントの研究開発	http://www.tryllian.com/
	Congnitive Agent ツールキットの研究開発	http://www.cognitiveagent.com/
	エージェント開発環境の研究開発	http://www.bitpix.com/
	消費者の活動の情報を元に消費者に会いに行くことの出来るエージェント	http://www.bitpix.com/
	エージェントを用いた薬学システム	http://www.bitpix.com/
	ユーザの嗜好を学習する検索システムの研究開発	http://www.bitpix.com/
	エージェントライブラリの研究開発	http://www.bitpix.com/
	インテリジェントエージェントを利用した知的なルーティングの研究	http://www.bitpix.com/
	Windows のインタフェースと相互作用するエージェント	http://msdn.microsoft.com/msagent/default.asp
	エージェントの活動を管理するエージェントサーバの研究開発	http://www.kinetoscope.com/via/default.htm
	モバイルエージェント記述言語の研究	http://actcomm.dartmouth.edu/
	Java ベースの分散オブジェクトシステムとエージェントフレームワークの研究	http://bond.cs.purdue.edu/
	マルチエージェント開発環境の研究	http://public.logica.com/~grace/Architecture/Cable/public/
	マルチエージェントフレームワークの研究	http://www.cis.udel.edu/~graham/DECAF/
	ユーザの要求する情報を取ってくるためにユーザの代わりに Web ブラウズするインテリジェントエージェント	http://dollar.biz.uiowa.edu/
	消費者の活動からユーザの嗜好を抽出して商品カタログのブラウズをするなどの補助を行うショッピングエージェント	http://dollar.biz.uiowa.edu/
	社会や組織の進化のエージェントベースモデル	http://dollar.biz.uiowa.edu/
	Java ベースのマルチエージェントシステムとそのフレームワークの研究	http://www.ececs.uc.edu/~abaker/JAFMAS/
	ユーザが迅速にソフトウェアエージェントを作ることを可能にする Java プログラムパッケージ	http://java.stanford.edu/java_agent/html/
	モバイルエージェントの実行環境の研究	http://www.cnri.reston.va.us/home/koe/
	モバイルミドルウェアツールキットの研究	http://comet.columbia.edu/mobiware/.index.html
	マルチエージェントアーキテクチャに関する研究	http://www.ai.sri.com/~oaa/
	ヨーロッパのエージェントプロジェクトのリンク集	http://www.agentlink.org/

Autonomous Agent at Rock Island Arsenal	Rock Island Arsenal
SodaBot	Artificail Intelligence Lab. Massachusetts Institute of Technology
Jackal	Jackal Project at University of Maryland Baltimore County
Concordia	HorizonSystems Laboratory MITSUBISHI ELECTRIC Information Technology Center America
SciAgents	(Anupam Joshi) University of Maryland Baltimore County
InfoSleuth	Microelectronics and Computer Technology Corporation
Jinni	BinNet Cooperation
BinNet Internet	BinNet Cooperation
BinProlog	BinNet Cooperation
Fraud Prevention	Authentix Inc.
Agent-Based Engineering	Stanford University's Center for Design Research
Multi-Agent Systems Group	University of Texas
Multiagent Systems	University of South Carolina
Open Systems Laboratory	University of Illinois at Urbana-Champaign
Mobile Robot Laboratory	Multiagent Robotic Systems, Georgia Institute of Technology
D'Agents	Dartmouth College
Multi-Agent Systems Laboratory	University of Massachusetts
Interaction Agent	Information Discovery
AgentOS	University of California, Irvine
SodaBot/SodaBotL	MIT Artificial Intelligence Lab
MASS	University of Massachusetts
Concordia	Mitsubishi Electric Information Technology Center America
Swarm	Swarm Development Group (SDG)

北米 カナダ

Enterprise Integration Laboratory (EIL)	University of Toronto
Prototyping an Intelligent Agent through Wizard of Oz	GroupLab The University of Calgary
Intelligent Agents	Dave Orchard
Agent Awareness Application for The Collaborative Agent Architecture	Stephen Lam
DataLink	Paul Roberts

	エージェントベースの工場スケジューリングおよびシミュレーション	http://www.aaria.uc.edu/
	エージェント開発環境の研究	http://www.ai.mit.edu/people/sodabot/sodabot.html
	Javaベースのエージェントのためのコミュニケーションインフラストラクチャ	http://jackal.cs.umbc.edu/
	いつ・どこでも情報にアクセスするためのモバイルエージェントアプリケーションの研究開発	http://www.meitca.com/HSL/Projects/Concordia/
	複数の学問領域にまたがる問題解決環境構築のエージェントベースアプローチ	http://www.cs.umbc.edu/~ajoshi/sciag/
	アプリケーションのために情報の発見・統合・プレゼンを行う協調エージェントのコミュニティの実装	http://www.mcc.com/infosleuth/
	Javaベースのインテリジェントモバイルエージェントライブラリ	http://www.binnetcorp.com/
	エージェントを用いたウェブサイトの検索	http://www.binnetcorp.com/
	エージェント構築ツール	http://www.binnetcorp.com/
	エージェントを用いてコストを削減するシステムの研究開発	http://www.authentix.net/
	エージェントをベースとした空間デザイン技術の研究	http://www-cdr.stanford.edu/ABE/
	分散システムにおけるエージェントの協調の研究	http://dali.uta.edu/
	分散人工知能に関する研究	http://www.engr.sc.edu/research/CIT/projects/multiagent.html
	並列・分散リアルタイムコンピューティングに関する研究	http://osl.cs.uiuc.edu/
	エージェントに基づくロボットシステムの開発	http://www.cc.gatech.edu/ai/robot-lab/research/multi-agent.html
	モバイルエージェントによる応用システムの開発	http://agent.cs.dartmouth.edu/
	エージェント間通信・協調に関する研究	http://mas.cs.umass.edu/index.shtml
	売り主と顧客の間で知識・情報をやり取りするエージェントシステム	http://www.datamining.com/inagent.htm
	モバイルエージェントを用いた分散OSの開発	http://bolero.ics.uci.edu/agentos/
	エージェントシステムの総合開発環境	http://www.ai.mit.edu/people/sodabot/sodabot.html
	マルチエージェント協調のシミュレーションシステム	http://mas.cs.umass.edu/research/mass/index.html
	協調エージェントシステム構築のフレームワーク	http://www.meitca.com/HSL/Projects/Concordia/
	マルチエージェント協調のシミュレーションシステム	http://www.swarm.org/index.html

	マルチエージェントシステム開発のための環境	http://www.eil.utoronto.ca/ABS-page/ABS-intro.html
	ユーザとインタラクションするシュミレーションエージェント	http://www.grailsearch.com/Intelligent%20Agents.htm
	エージェント全般に関するテキスト	http://www.pacificspirit.com/Courses/Agents/index.htm
	協調的エージェントアーキテクチャのためのエージェント知覚システム	http://www.cpsc.ucalgary.ca/~lamsh/SENG/691/Presentation1/tsld001.htm
	インテリジェントエージェント技術に基づくDatalink社の製品の紹介	http://www.brokerxpress.com/company/story10.htm

Research Publications	Douglas H. Norrie
NEXIR	Bhasha
Intelligent Agents - Current Investigations	Andrew Tunnicliffe
Computer Science 856	Reagan Penner
Network Management	David C Blight
The Virtual Body	Marie-Lauren
ELECTRONIC COMMERCE MAJOR PROJECT	Nicolas D. Georganas(CITR)
Thanh-Trang Dang's Home Page	Thanh-Trang Dang
Andre Vellino's Research Interest	Andre Vellino
Elicitation of constructs on Intelligent Agent design	David Dattner
The Learning Web: A System View and an Agent-Oriented Model	Douglas H. Norrie and Brian R. Gaines University of Calgary
TeleLearning Project	Université de Montréal HERON Laboratory
Intelligent Agents - Introduction	Andrew Tunnicliffe
Intelligent Agents…	AssistiveTechnologyCenter
Intelligent Agents	Mark Bentley
Agents of Today and Tommorow	Rob Parkhill
Agent-Based Systems for Intelligent Manufacturing: A State-of-the-Art Survey	Weiming Shen and Douglas H. Norrie
R E C E N T P U B L I C A T I O N S	Dr. Stan Szpakowicz
Research	John Anderson
A Mediator Architecture for Intelligent Manufacturing Systems	D.H. Norrie
Multi-Agent Co-ordination and Dynamic Clustering	D.H. Norrie
Agent-Based Concurrent Design Environment (ABCDE)	D.H. Norrie
Distributed Objects & Components: Mobile Agents	Manfred Schneider
IMS Group Publications by year	IMS Group
Intelligent Scheduling Agent Framework	Tong Zheng
Internet Based Comparative Shopping Agent	Sunil Kamath
Intelligent Agent Software	Wallace Hannum
Planning from action and change	Wuilbert Jaramillo
Thesis Project: An Intelligent Multi-Agent System Architecture	YUE ZHANG
TALK TO MY AGENT	HI-TECH CULTURE

	マルチエージェントシステムに関する文献集	http://www.acs.ucalgary.ca/~undseth/moredata/donepub.html
	インテリジェントエージェント技術に基づく次世代情報収集ツール	http://www.bhasha.com/bhasha-products.htm
	インテリジェントエージェントの調査報告	http://www.cyberus.ca/~vellino/hrlw2.ag.html
	ソフトウェアおよび知識工学に関するリンク集	http://www.cs.usask.ca/grads/rtp130/Cmpt856/index.html
	ネットワーク応用に関する研究紹介およびリンク	http://www.ee.umanitoba.ca/~blight/telecommunications/nm.html
	仮想現実に関する研究	http://instruct.uwo.ca/writing/207/margaret/gregoire1.htm
	インテリジェントエージェントを用いた電子商取引に関する研究	http://www.mcrlab.uottawa.ca/research/CITR_98.htm
	インテリジェントエージェントに関する研究	http://www.eng.uwaterloo.ca/Gradstudent/tt3dang/index.html
	インテリジェントエージェントに関する研究	http://ai.iit.nrc.ca/~andre/research.html
	インテリジェントエージェントに関する研究	http://ksi.cpsc.ucalgary.ca/courses/547-96/dattner/proj3/proj3.html
	エージェントを用いてネットワーク上でユーザに学習環境を提供しようとする研究	http://ksi.cpsc.ucalgary.ca/articles/LearnWeb/EM95J/
	本研究室で行われているエージェントに関するプロジェクトの紹介	http://www.iro.umontreal.ca/labs/HERON/TeleLearning/telelearning.html
	インテリジェントエージェントに関する研究	http://www.cyberus.ca/~vellino/hrlw.ag.html
	インテリジェントエージェントに関する研究	http://www.tinman.org/Frontiers/Pavilions/AssistiveTechnologyCenter/IntelligentAgents.html
	インテリジェントエージェントに関する研究	http://ksi.cpsc.ucalgary.ca/courses/547-95/bentley/547talk.html
	エージェントの今後の展望について述べているページ	http://ksi.cpsc.ucalgary.ca/courses/547-95/parkhill/Agents.html
	製造のためのエージェントベースシステムの開発の調査	http://sern.ucalgary.ca/CAG/publications/abm.htm
	環境の変化に対応する自律的エージェントのモデリングに関する研究	http://www.site.uottawa.ca/~szpak/recent_publ.html
	複雑な領域で作用する自律的エージェントに関する研究：	http://www.cs.umanitoba.ca/~andersj/research.html
	分散製造システムでの仮想エージェントの生成と協調のための調整アーキテクチャの研究	http://www.acs.ucalgary.ca/~norrie/moredata/paper1.html#sec1
	マルチエージェントシステムの協調に関する研究	http://www.acs.ucalgary.ca/~norrie/moredata/paper1.html#sec2
	マルチエージェントシステムの設計に関する研究	http://www.acs.ucalgary.ca/~norrie/moredata/paperball.html#sec1
	モバイルエージェントに関する研究及びリンク集	http://www.csioo.com/cetusen/oo_mobile_agents.html
	マルチエージェントシステムのアーキテクチャに関する研究	http://imsg.enme.ucalgary.ca/pub_year.html
	インテリジェントスケジューリングエージェントフレームワークに関する研究	http://www.cs.ualberta.ca/~jonathan/Courses/622/Proposal/tongz/tongz.final.html
	オンラインショッピングエージェントの研究	http://www.cs.ualberta.ca/~sunil/thesis.htm
	インテリジェントエージェントソフトウェアに関する研究	http://www.cquest.utoronto.ca/env/aera/aera-lists/aera-a/99-01/0044.html
	エージェントのプランニングについての研究	http://www.ensc.sfu.ca/people/grad/wjaramil/personal/cmpt882/proj.html
	インテリジェントマルチエージェントシステムアーキテクチャに関する研究	http://www.cas.mcmaster.ca/~yzhang/Thesis.html
	インティジェントエージェントの紹介	http://204.174.18.3/htc/4_2t.html

A Hybrid Agent-Oriented Infrastructure for Modeling Manufacturing Enterprises	Weiming Shen and Douglas H. Norrie
Intelligent Agents	Telecommunications Information
Perspectives on Representation and Analysis of Negotiation	Gregory E. Kersten and David Cray
Lin Zeng's Research	Lin Zeng
Curriculum Vitae Jiawei Han	Jiawei Han
MSISArea Faculity	Ali R. Montazemi
Proposal by Tong Zheng	Tong Zheng
Human-Computer Interface Design	Weidong Sun
Recent Publications by Yves Lesperance and Students	Yves Lespérance
Department of Mechanical Engineering	D.H. Norrie
DAMAS	DAMAS
NetSA	DAMAS
FuzzyCLIPS	Bob Orchard
Multiagent Systems Research at EIL	EIL
The Agent Building Shell: Programming Cooperative Enterprise Agents	EIL
Conversational Agent Coordination Language	EIL
WWW Interface to the Agent Environment	EIL
Behavior Description Language	EIL
Coordination Theory Based on Constraint Exchange Amongst Agents	EIL
Teamwork Coordination in a Virtual Supply Chain	EIL
Multiagent Simulation for Dealing With Perturbation in Supply Chains	EIL
Customized Service Provisioning in Telecommunications	EIL
ISA'2000	ISA
Software Agents-based Workflow	Thirteenth Canadian Conference
Agents et Systemes Multiagents	Brahim Chaib-draa
SoftwareAgentCourse	BabakEsfandiari
A gentle introduction to agents and their applications	Michael Weiss
Ontobroker: Or How to Enable Intelligent Access to the WWW	Dieter Fense

	エージェントを用いた決定・交渉支援システムに関する研究	http://interneg.carleton.ca/interneg/research/misc/boston_gdn/
	マルチエージェントセキュリティシステムの研究	http://www.cs.toronto.edu/~zeng/research.html
	データマイニングおよびインテリジェントエージェントに関する研究	http://www.cs.sfu.ca/~han/personal/99.html
	適応型インタフェースエージェントに関する研究	http://www.business.mcmaster.ca/msis/profs/montaz/
	スケジューリングエージェントに関する研究	http://www.cs.ualberta.ca/~jonathan/Courses/622/Proposal/tongz/tongz.r1.html
	エージェントの定義・設計論・分類	http://www.cs.concordia.ca/~grad/wsun/comp675/research5.html
	マルチエージェントシステムの開発およびエージェントのモデリング・プログラミングに関する研究	http://www.cs.yorku.ca/~lesperan/publications.html
	インテリジェントエージェント開発システムに関する研究	http://www.enme.ucalgary.ca/personnel/mech.htm
	データマイニング・エージェント・エージェントシステムに関する研究	http://auguste.ift.ulaval.ca/
	ネットワークソフトウェアエージェントの開発	http://auguste.ift.ulaval.ca/~netsa/
	ファジーな推論システムの開発	http://ai.iit.nrc.ca/IR_public/fuzzy/fuzzyClips/fuzzyCLIPSIndex.html
	本大学で行われているマルチエージェントシステム研究の紹介	http://www.eil.utoronto.ca/ABS-page/ABS-intro.html
	エージェント設計開発に関する研究	http://www.eil.utoronto.ca/ABS-page/ABS-overview.html
	会話的エージェント間協調言語に関する研究	http://www.eil.utoronto.ca/ABS-page/ABS-intro.html#HDR3
	エージェント環境へのWWWインタフェースに関する研究	http://www.eil.utoronto.ca/ABS-page/ABS-intro.html#HDR5
	ビヘイビア記述言語に関する研究	http://www.eil.utoronto.ca/ABS-page/ABS-intro.html#HDR7
	エージェント間の制約されたやりとりに基づく協調方法の研究	http://www.eil.utoronto.ca/ABS-page/ABS-intro.html#HDR9
	バーチャルサプライチェーンでの協調的チームワークに関する研究	http://www.eil.utoronto.ca/ABS-page/ABS-intro.html#HDR12
	マルチエージェントシステムのシミュレーションの研究	http://www.eil.utoronto.ca/ABS-page/ABS-intro.html#HDR14
	エージェント通信のカスタマイズサービスに関する研究	http://www.eil.utoronto.ca/ABS-page/ABS-intro.html#HDR16
	ISA2000（国際会議）	http://www.icsc.ab.ca/isa2000.htm
	ソフトウェアエージェントおよびワークフローに関するワークショップ	http://www.grad.ift.ulaval.ca/~maamar/workshop1.htm
	マルチエージェントシステムに関する研究	http://auguste.ift.ulaval.ca/~chaib/
	エージェントについて学習するページ	http://mystic.sce.carleton.ca:8080/cgi-bin/agentcourse.cgi
	エージェントとエージェントアプリケーションの紹介	http://www.magma.ca/~mrw/agents/
	インテリジェントWWWアクセスエージェントに関する研究	http://ksi.cpsc.ucalgary.ca/KAW/KAW98/fensell/
	ハイブリッドエージェント機構に関する研究	http://ksi.cpsc.ucalgary.ca/KAW/KAW98/shen/
	インティジェントエージェントに関するリンク集	http://www.ee.umanitoba.ca/~blight/telecommunications/ia.html

B2 欧州

ヨーロッパ イギリス

プロジェクト名	制作グループ
The Cognition and Affect Project	School of computer Science, The University of Birmingham
Luigi	KMI, The Open University
O-Plan - An Intelligent Planning Agent on the Superhighway	Artificial Intelligence Applications Institute, University of Edinburgh
METATEM	CARD, Manchester Metropolitan University
CARD project	CARD, Manchester Metropolitan University
My Research	Chris Harrison
Mobility Mailing List	Loughborough University
MAVIS 2	MMRG, ECS, University of Southampton
MEMOIR	PAC, University of Southampton
FIRM	MMRG, ECS, University of Southampton
Distributed Artificial Intelligence Research Unit	DAI Reserch Unit
Market Based Control for the Management of Telecommunications Networks	Market Based Control for the Management of Telecommunications Networks
Agents for Telecommunications Systems	Agents for Telecommunications Systems
Agent-Based Software Engineering	Agent-Based Software Engineering
Modelling Team Problem Soilving Using Joint Intentions	Modelling Team Problem Soilving Using Joint Intentions
Negotiation by Dialectic Argumentation	Negotiation by Dialectic Argumentation
Socially Rational Agents	Socially Rational Agents
Brokerage in an Information Economy	Brokerage in an Information Economy
JATLiteBean	JATLiteBean
JATLite	JATLite
PAAM	Practical Aapplications Company
Logic Programming Section	Department of Computing, Imperial College

	コメント	ホームページ
	自然、人工物に関わらず、様々な機能を統合した知的エージェント全体のアーキテクチャのデザインを研究、また、様々なツールを開発。THE SIM_AGENT TOOLKIT など。	http://www.cs.bham.ac.uk/~axs/cogaff.html
	最少の共通因子の電子メールと Web ツールを使って、ユーザの代理としてミーティングを取り決める心理上の原則に基づいたエージェント	http://kmi.open.ac.uk/projects/current.html
	英国の情報スーパーハイウェー上で利用可能であるアシスタントエージェント	http://www.aiai.ed.ac.uk/~oplan/innov.html
	マルチエージェントのプログラミング言語	http://www.card.mmu.ac.uk/
	METAMEM を使ったエージェント環境の実装および、エージェントの行動の研究	http://www.card.mmu.ac.uk/
	エージェントのリンク集	http://www-icparc.doc.ic.ac.uk/~cah1/Research.html
	可動的なコード、オブジェクト、エージェントと同種のものに関してあらゆることを論じるためのメーリングリスト	http://mobility.lboro.ac.uk/
	ビデオ、イメージと音声のためにマルチメディアのアーキテクチャを開発する研究	http://www.mmrg.ecs.soton.ac.uk/projects/mavis2.html
	オープンハイパーメディアシステムに統合された分散型オブジェクト指向データベーステクノロジーを使った企業情報システムのためのフレームワーク	http://www.pac.soton.ac.uk/memoir/
	製造業環境、特にメンテナンスと欠陥診断問題を緩和する、統合型情報管理の新しいテクニックを研究	http://www.mmrg.ecs.soton.ac.uk/projects/firm.html
	分散環境下で、エージェントを用いた研究するグループの集まり	http://www.elec.qmw.ac.uk/dai/
	マーケットベース理論を用いた、電話ネットワークの管理についての研究	http://www.elec.qmw.ac.uk/dai/projects/agentCAC/index.html
	電話のネットワークの管理を例に、エージェント交渉などについて研究	http://ada.econ.ucl.ac.uk/~nirvulkan/EPSRC.html
	複雑なソフトウェアを構築するための、方法として、マルチエージェントシステムの応用について研究している	http://www.elec.qmw.ac.uk/dai/projects/abse.html
	マルチエージェントシステムがある問題に対処するときの、問題解決の方法などについて研究	http://www.elec.qmw.ac.uk/dai/projects/cooperation.html
	マルチエージェントシステムにおけるエージェントの交渉のモデル等について	http://www.elec.qmw.ac.uk/dai/projects/
	マルチエージェントのシステムの中での行動の合理性についての研究	http://www.elec.qmw.ac.uk/dai/projects/soc-rat.html
	他のエージェントへの情報のブロードキャストの方法について研究	http://www.elec.qmw.ac.uk/dai/projects/brokerage/index.html
	ソフトウェアエージェントを生成するプログラムをパッケージ化した、JATLite を拡張したもの	http://kmi.open.ac.uk/people/emanuela/JATLiteBean/
	ソフトウェアエージェントを生成するプログラムのための Java パッケージ	http://java.stanford.edu/java_agent/html/
	知的エージェントの実務的アプリケーションとマルチエージェントのカンファレンス	http://www.practical-applications.co.uk/PAAM/index.html
	CSCW に基づいたマルチエージェントシステムを含む、分散型シンボリックプログラミング言語とアプリケーション	http://www-lp.doc.ic.ac.uk/~klc/

A Formal Framework for Agency and Autonomy	The Agebt-based Systems Group, Department of Computer Science, The University of Warwick
Agent Systems Specification	The Agebt-based Systems Group, Department of Computer Science, The University of Warwick
The Development of Intelligent Software Agents for Genome Analysis and Protein Structure Prediction	The Agebt-based Systems Group, Department of Computer Science, The University of Warwick
Motivated Cooperation in Multi-Agent Systems	The Agebt-based Systems Group, Department of Computer Science, The University of Warwick
FOMAS	Department of Computer Science, The University of Warwick
Mark d'Inverno's Research	Mark d'Inverno, Cavendish School of Computer Science, University of Westminster
INFORMER	School of Psychology, University of Nottingham
ELMA	School of Psychology, University of Nottingham
Autonomous Vehicle	School of Psychology, University of Nottingham
Lego Agents	School of Psychology, University of Nottingham
Wacbots	School of Psychology, University of Nottingham
Intelligent Software Agents	Computer Science Department, University of Aberdeen
Remora	LAS, Computer Science Department, University of Aberdeen
Knowledge Refinement in Multi-Agent Systems	LAS, Computer Science Department, University of Aberdeen
MAGI	LAS, Computer Science Department, University of Aberdeen
IAN	LAS, Computer Science Department, University of Aberdeen
LAW	LAS, Computer Science Department, University of Aberdeen
VIS, EVIS & ELVIS	LAS, Computer Science Department, University of Aberdeen
MAVA	LAS, Computer Science Department, University of Aberdeen
Agent-K	Computer Science Department, University of Aberdeen
PROJECTS PAGE	University of Durham
ILOG Rules	ILOG
Andy Wood	Birmingham 大学
ANSA and APM	ANSA
ASG	The Faculty of Information Technology, University of Brighton

	エージェントの正式なフレームワークの研究	http://www.dcs.warwick.ac.uk/absg/
	エージェント理論のとプラクティスを統合し、エージェントシステムのアーキテクチャとオペレーションを正式化する	http://www.dcs.warwick.ac.uk/absg/
	ゲノム分析とタンパク質構造の推定のための知的ソフトウェアエージェントの開発	http://www.dcs.warwick.ac.uk/absg/
	マルチエージェントシステムにおける協調法	http://www.dcs.warwick.ac.uk/absg/
	UKベースの研究とマルチエージェントのシステムの開発を促進	http://www.dcs.warwick.ac.uk/~mikeluck/fomas/
	アプリケーションの正式化手法	http://www.wmin.ac.uk/~dinverm/
	マルチエージェントシステムの正式記述テクニックの定義	http://www.psychology.nottingham.ac.uk/research/ai/themes/agents/informer.html
	マルチエージェントシステムの問題解決のための効率的なコンピュータアーキテクチャの開発	http://www.psychology.nottingham.ac.uk/research/ai/themes/agents/elma.html
	安価な自律エージェント環境の構築、エコロジカルなビヘイビアの探求、クラシックとリアクティブなプランニングの統合の基礎の提供	http://www.psychology.nottingham.ac.uk/research/ai/themes/agents/vehicle.html
	LEGOによるロボットエージェント	http://www.psychology.nottingham.ac.uk/research/ai/themes/agents/lego.html
	LEGOによるロボットエージェント	http://www.psychology.nottingham.ac.uk/research/ai/themes/agents/wacbot.html
	エージェントベースのシステムの能力を拡張する学習テクニックの研究	http://www.csd.abdn.ac.uk/research/intelligent_agents.html
	クライアント／サーバアプローチのメタ検索エンジン	http://www.csd.abdn.ac.uk/~pedwards/res/remora.html
	マルチエージェントのシステムでの知識精製	http://www.csd.abdn.ac.uk/~pedwards/res/drama.html
	電子メールのフィルタ学習	http://www.csd.abdn.ac.uk/~pedwards/res/PREVIOUS/magi.html
	USENETのニュースのフィルタリング	http://www.csd.abdn.ac.uk/~pedwards/res/PREVIOUS/ian.html
	WWW用の学習エージェント	http://www.csd.abdn.ac.uk/~pedwards/res/PREVIOUS/law.html
	個人化パンフレットの作成	http://www.csd.abdn.ac.uk/~pedwards/res/brochure.html
	訪問者情報を含んでいるデータベース（ホテル、レストラン、パブ）のグループへの知的なアクセス。	http://www.csd.abdn.ac.uk/~pedwards/res/mava.html
	エージェント指向プログラミングとKQMLの統合	http://www.csd.abdn.ac.uk/~pedwards/publs/agentk.html
	プロジェクトページへのリンク集	http://www.dur.ac.uk/~dcs3sjg/projects.html
	ビジネスシステムにやわらかさを持たせる、ILOGというルールに基づくエージェントの開発	http://www.ilog.co.uk/products/rules/intelligent.cfm
	インターフェースエージェントに関して研究している	http://www.cs.bham.ac.uk/~amw/
	オブジェクトベースの分散システムのためのアーキテクチャおよびソフトウェアの開発	http://www.ansa.co.uk/
	自律システムの研究	http://www.it.bton.ac.uk/research/asg.html

Socially Intelligent Agents	Reading 大学
Agents Metapage	Durham 大学
SimJava	エジンバラ大学

ヨーロッパ　ドイツ

Intelligent and mobile agents and their applicability to service and network management	EURESCOM
Communications Management Process Integration Using Software Agents	EURESCOM
MESSAGE	EURESCOM
Mobile Service Agents	ECRC
Ara	Universitat Kaiserslautern
AIDA	Mole, University of Stuttgart
ASAP	Mole, University of Stuttgart
ATOMAS	Mole, University of Stuttgart
FESTIVAL	Mole, University of Stuttgart
HAWK	Mole, University of Stuttgart
Security in Mobile Agent Systems	Mole, University of Stuttgart
Cooperative Agents supporting information exchange	TUM Informatik XI
PRESENCE	DFKI
CoMMA-MAPS	DFKI
MAS-FORKS	DFKI
MAS-MARS, TeleTruck	DFKI
Magic Lounge	DFKI
WWW-Persona	DFKI
ASSISI	DFKI

	エージェントが社会的、文化的な制約の中でどのように行動するのかを研究	http://www2.cyber.rdg.ac.uk/Cybernetics/
	リンク集	http://www.dur.ac.uk/~dcs3sjg/agents.html
	Java をベースとした、エージェントベースシュミレーター	http://www.dcs.ed.ac.uk/home/hase/simjava/

	知的モバイルエージェントと、ネットワーク管理やサービスへのそれらの適応性についての研究	http://www.eurescom.de/Public/Projects/P700-series/P712/P712.HTM
	ソフトウェアエージェントを使った通信管理プロセスの統合	http://www.eurescom.de/Public/Projects/P800-series/P815/p815.htm
	オブジェクト指向ソフトウェアからエージェント指向のアプリケーション開発のための方法論の拡張	http://www.eurescom.de/Public/Projects/p900-series/P907/P907.htm
	情報ハイウェイを想定したモバイルサービスエージェント	http://www.ecrc.de/research/dc/msa/
	異種ネットワーク上におけるモバイルエージェントのために、携帯性や実行安全性を提供するプラットフォームの開発	http://www.uni-kl.de/AG-Nehmer/Projekte/Ara/index_e.html
	モバイルエージェントのインフラの提案	http://mole.informatik.uni-stuttgart.de/
	巨大な分散型アプリケーションの主要な問題の解決や、モバイルエージェントを使ったコンピュータネットワークの運営など	http://mole.informatik.uni-stuttgart.de/
	電子商取引のためのエージェント技術	http://mole.informatik.uni-stuttgart.de/
	MOLE エージェントと CORBA により電子マーケットアプリケーションや仮想オフィスを構築	http://mole.informatik.uni-stuttgart.de/
	モバイルエージェントによる情報検索	http://mole.informatik.uni-stuttgart.de/
	モバイルエージェントシステムのセキュリティについての文献目録	http://www.informatik.uni-stuttgart.de/ipvr/vs/projekte/mole/security.html
	WWW の上に種々の分散プロバイダによって申し出られるフレキシブルにアクセスしている外部のサービス（探索サービス、ホワイトページサービス、カタログサービス）の革新的な機構を供給しする協力的なエージェントの提案	http://www11.informatik.tu-muenchen.de/proj/ca/ca.html
	生きたような振る舞いをするコンピュータキャラクタを動かすために用いられる技術の促進	http://www.dfki.de/pas/f2w.cgi?dmasp/presence-e
	問題分析で識別されたシステムコンポーネントを、自律的で対話型の概念的なエンティティー（エージェント）として設計する技術。デザインとマルチエージェントシステムの実装に専門的知識およびツールとテクニックを提供。	http://www.dfki.de/pas/f2w.cgi?dmasc/maps-e
	各エージェントが自律した輸送システムであるマルチエージェントシステムとして自動化ローディングドックを設計	http://www.dfki.de/pas/f2w.cgi?dmass/forks-e
	広大な領域におけるリソース管理	http://www.dfki.de/pas/f2w.cgi?dmass/mars-e
	知的なコミュニケーションサービスを持った情報スペース	http://www.dfki.de/pas/f2w.cgi?iuip/mlounge-e
	WWW 上でのプレゼンテーションエージェント	http://www.dfki.de/pas/f2w.cgi?iuip/webp-e
	コンテンツベースの情報アシスタントの設計	http://www.dfki.de/pas/f2w.cgi?daimc/assisi-e

COSMA	DFKI
CoMMA-COGs	DFKI
Knowbots	INM
Graduiertenkolleg Kooperation und Ressourcenmanagement in verteilten Systemen	Graduiertenkolleg Kooperation und Ressourcenmanagement in verteilten Systemen
InteRRaP	InteRRaP
GrassHopper	GrassHopper
TAXT	TAXT(The Agent eXpert Team)
EXCALIBUR	EXCALIBUR
Social Interaction Framework(SIF)	Social Interaction Framework(SIF)
JIAC Site	JIAC
PageSpace	FLP, Fachbereich (13) Informatik, Technische Universitat Berlin
AARFTA	The DAI-Labor, Department of Computer Science, Universsity of Technology Berlin
ABROSE	The DAI-Labor, Department of Computer Science, Universsity of Technology Berlin
AMASE	The DAI-Labor, Department of Computer Science, Universsity of Technology Berlin
JavaTOn	The DAI-Labor, Department of Computer Science, Universsity of Technology Berlin
MIATA	The DAI-Labor, Department of Computer Science, Universsity of Technology Berlin
SIATAI	The DAI-Labor, Department of Computer Science, Universsity of Technology Berlin
Traffic Telematics	The DAI-Labor, Department of Computer Science, Universsity of Technology Berlin
Multi-Agent-Soccer	DAI Research Unit, Technische Universitat, Chemnitz
Chemnitz's Agent system	DAI Research Unit, Technische Universitat, Chemnitz
Mobile Agents reseach	IPVR, University of Stuttgart
Ara	AG Systemsoftware, Fachbereich Informatik, Universitat Kaiserslautern
Follow Me	Unsere Project
AMETAS	Johann wolfgang geothe 大学
MAGNA	GMD FOKUS
MASIF	GMD FOKUS
The IN & Mobility Sub-Cluster	GMD FOKUS

	電子メールによる自動予約スケジューリング	http://www.dfki.de/pas/f2w.cgi?lts/cosma-e
	マルチエージェントシステムにおいてモチベーショナルな問題を扱うためのリソース認知アーキテクチャの開発	http://www.dfki.de/mas/projects/cogs.html
	オントロジーと知識表現システム	http://www.inm.de/kip/general/guide.html
	協調して、資源を分配、管理する分散環境システムについての研究	http://hp0.informatik.tu-muenchen.de/gkolleg/gkolleg.html
	エージェント間の相互干渉について研究	http://www.dfki.uni-sb.de/mas/interrap/
	IKV++によって開発されたエージェント開発環境 GrassHopper	http://www.ikv.de/products/grasshopper/
	エージェントの一般的なモデル化について研究している	http://samuel.cs.uni-potsdam.de/soft/taxt/research/research_frameset.html
	エージェント自立的な行動についてアーキテクチャの観点から研究している	http://www.first.gmd.de/concorde/excalibur/goals.html
	社会的な相互作用を考えたフレームワーク SIF について研究している	http://www.dfki.de/~sif/
	EC アプリケーション、電話サービスなどのためのアーキテクチャ	http://dai.cs.tu-berlin.de/english/forschung/projekte/JIAC/
	分散ソフトウェアコンポーネント間の交渉や動的処理を必要とするネットワークアプリケーションのサポートについての研究	http://flp.cs.tu-berlin.de/pagespc/
	未来の電気通信アプリケーション実現のためのエージェントアーキテクチャ	http://dai.cs.tu-berlin.de/english/forschung/projekte/AARFTA/main.html
	電子商取引のためのエージェントベースのブローカーサービス	http://dai.cs.tu-berlin.de/english/forschung/projekte/ABROSE/main.html
	マルチメディア情報サービスへのエージェント型モバイルアクセス	http://dai.cs.tu-berlin.de/english/forschung/projekte/AMASE/main.html
	Java ベースの T-オンラインクライアント	http://dai.cs.tu-berlin.de/english/forschung/projekte/javaTOn/main.html
	エージェント型電気通信アプリケーションのための管理基盤	http://dai.cs.tu-berlin.de/english/forschung/projekte/MIATA/main.html
	エージェント型電気通信アプリケーションのためのセキュリティ基盤	http://dai.cs.tu-berlin.de/english/forschung/projekte/SIATA/main.html
	現代的な交通量遠隔操作サービスを実現する知的エージェント	http://dai.cs.tu-berlin.de/english/forschung/projekte/trafficTele/main.html
	プレイヤーの動きを協調的に調整するマルチエージェントサッカー	http://www.tu-chemnitz.de/~hla/soccer/
	マシン学習、データマイニング、知的エージェント、マルチエージェント、スマートホーム、ロボットなどの研究（ドイツ語）	http://www.tu-chemnitz.de/informatik/HomePages/KI/forschung.html
	モバイルエージェントの研究	http://www.informatik.uni-stuttgart.de/ipvr/vs/projekte/mole/agents.html
	モバイルエージェントのためのリモートアクセスプラットフォーム	http://www.uni-kl.de/AG-Nehmer/Projekte/Ara/index_e.html
	様々なデバイスの存在するネットワーク環境において、ユーザーが恒久的な存在として存在できるようなアーキテクチャの開発	http://hyperwave.fast.de/FollowMe
	エージェント型システム、エージェント指向プログラミング環境について研究している	http://www.vsb.cs.uni-frankfurt.de/ametas/index_en.html
	クライアント／サーバモデルにエージェントコンセプトを補完する次世代電気通信アプリケーションのフレームワーク	http://www.fokus.gmd.de/ice/projects/magna.html
	モバイルエージェントシステムの互換性を検証する施設	http://www.fokus.gmd.de/research/cc/ecco/masif/index.html
	知的モバイルエージェントのアプリケーションの探求	http://www.fokus.gmd.de/research/cc/ecco/climate/sc-in-mobility/entry.html

The Communications Management Sub-Cluster	GMD FOKUS
End-to-End Agent Systems Sub-cluster	GMD FOKUS
The Agent Platforms Sub-Cluster	GMD FOKUS
The WAVE Page	Ira, Universitat Karlsruhe
Learning coordination skills in Multi-Agent Systems	Ira, Universitat Karlsruhe
Learning to support communcation in Multi-Agent Systems	Ira, Universitat Karlsruhe
Agents for Remote Access	Kaiserslautern 大学

ヨーロッパ スウエーデン

PAWS	Media Lab, Ericsson Telecom
Intelligent Systems Laboratory	Intelligent Systems Laboratory
DECIDE's Publications	DECIDE, DSV, Stockholm University
PEFNA	DSV, Stockholm University
DSV RoboCup team	DSV, Stockholm University
SHAMAN	SoK, Hogskolan i Karlskrona/Ronneby
QASE	SoK, Hogskolan i Karlskrona/Ronneby
MAPPA	Intelligent Systems Laboratory, Swedish Institute of Computer Science
AMP	Intelligent Systems Laboratory, Swedish Institute of Computer Science
AKL	Intelligent Systems Laboratory, Swedish Institute of Computer Science
INTAGE	Intelligent Systems Laboratory, Swedish Institute of Computer Science
Coord	Intelligent Systems Laboratory, Swedish Institute of Computer Science
AgentLib	Intelligent Systems Laboratory, Swedish Institute of Computer Science
Agent-Based Systems	Intelligent Systems Laboratory, Swedish Institute of Computer Science
Perdio	Intelligent Systems Laboratory, Swedish Institute of Computer Science
MECARCA	Intelligent Systems Laboratory, Swedish Institute of Computer Science

	サービスのデザインとネットワーク管理サービスにおける知的モバイルエージェントの研究	http://www.fokus.gmd.de/research/cc/ecco/climate/sc-com-mgmt/entry.html
	端末システムにおけるエージェントソフトウェアデザインの研究	http://www.fokus.gmd.de/research/cc/ecco/climate/sc-end-to-end/entry.html
	エージェントプラットフォームにおける情報交換の場を提供	http://www.fokus.gmd.de/research/cc/ecco/climate/sc-platforms/entry.html
	大規模なシステムにおける、複雑な要因からの自己組織化、自己修復などについて、インテリジェントエージェントを用いることで研究を進めている	http://www-zorn.ira.uka.de/wave/wave.html
	マルチエージェントのシステムでの学習調整技能の研究	http://wwwipr.ira.uka.de/~friedric/multi/multicoord.html
	マルチエージェントシステムにおける通信サポートの学習	http://wwwipr.ira.uka.de/~friedric/multi/multicomm.html
	エージェントのリモートアクセスについて、ワイヤレスな環境環境での研究を行っている	http://www.uni-kl.de/AG-Nehmer/Projekte/Ara/index.html

	新しい情報に自動アップデートしてくれる個人用 Web 監視エージェント	http://www.fek.su.se/forskar/program/imorg/dok/1996-08-28_1/erimedlab/paws/NewPAAM.doc.html
	エージェントカーネル言語 AKL や、AKL によって記述されたシステム AGENTS などを開発している	http://www.sics.se/isl/akl/
	DECIDE のメンバーによる出版物リンク	http://www.dsv.su.se/~mab/DECIDE/Publ.html
	ニュースの個人的情報をフィルタリングするエージェント	http://www.dsv.su.se/~fk/if_Doc/IntFilter.html
	DSV の RoboCup チーム	http://www.dsv.su.se/~robocup/
	分散コンピューティングの柔軟かつ安定したモデルの設計・実装についての研究及び発展	http://www.soc.hk-r.se/
	コンピュータゲーム Quake2 のエージェントシミュレート環境	http://www.soc.hk-r.se/lab/projects/Qase/qase.html
	持続的個人エージェントを通したマルチメディアの無線アクセス	http://www.sics.se/mappa/
	スウェーデンの Telecom, Telia と共同で作業しているエージェントベースの電子商取引プロジェクト	http://www.sics.se/~market/
	同時発生制約型プログラミング言語	http://www.sics.se/isl/akl/
	コミュニケーションと知的サービスのための知的エージェントの研究	http://www.sics.se/isl/intage/
	市場取引の交渉モデルにおいてエージェントの行動を調整して分散化する方法を研究する	http://www.sics.se/isl/coord/
	エージェントベースのディジタル図書館	http://www.sics.se/isl/diglib/
	エージェントのリンク集	http://www.sics.se/isl/abc/survey.html
	オープン・分散型・フォールトトレラント・反応的・知識集中的なソフトウェアシステムの効率的な構築を支援する	http://www.sics.se/isl/perdio/
	協調型エージェントアーキテクチャの研究	http://www.siti.se/projects/mecarca.html

TUFF	Intelligent Systems Laboratory, Swedish Institute of Computer Science
COPROCO	Intelligent Systems Laboratory, Swedish Institute of Computer Science
Agent-Based System	SICS

ヨーロッパ　イタリア

PSCS	IP-CNR, Italian National Research Council - Institute of Psychology
FIPA	CSELT
FARGO	FARGO
TuCSoN	TuCSoN
KLAIM-a Kernel Language for Agents Interaction and Mobility	KLAIM-a Kernel Language for Agents Interaction and Mobility
ICCS&SS	IP-CNR
Map	Multimedia & Distributed System Laboratory
The Agents Projects	IP-CNR
AGENTS and INTELLIGENCE	Dipartimento Energia, ENEA
MUSTER	ERG-ING-TISGI, C.R.CASACCIA, ENEA
IDA Project	ERG-ING-TISGI, C.R.CASACCIA, ENEA

ヨーロッパ　オーストリア

Distributed Objects & Components: Mobile Agents	The Distributed Systems Group in the Information Systems Institute of the Technical University of Vienna
D'Agents	The Distributed Systems Group in the Information Systems Institute of the Technical University of Vienna
Gypsy Project	Gypsy Project
Asgaard / Asbru	IfS, Technischen Universitat Wien
Patient Advocate	IfS, Technischen Universitat Wien
JACK Intelligent Agents	Agent Oriented Software Pty. Ltd.
Emotional Software Agents	OFAI, University of Vienna
A distributed intelligent multimedia information system architecture	OFAI, University of Vienna

	スケジューリングと資源配分による輸送会社の計画プロセスの開発	http://www.sics.se/~tuff/
	複雑な操作でスケジューリングするための構築技術	http://www.sics.se/col/projects/coproco/
	リンク集	http://www.sics.se/ps/abc/survey.html

	社会の依存と協調が特別強いマルチエージェント相互作用の正式化モデル	http://pscs2.irmkant.rm.cnr.it/users/amedeo/www/pscs.html
	エージェントベースのアプリケーションの中で互換を持たせる一般的なエージェント技術の開発を促進する団体	http://drogo.cselt.it/fipa/
	Javaをベースとしたモバイルエージェント環境を開発している	http://www.dsg.technion.ac.il/fargo/overview.php3
	インターネットエージェントのモデル	http://www.lia.deis.unibo.it/Research/TuCSoN/
	エージェントがマシンからどのマシンへ移動できるのかということが判断できる言語	http://rap.dsi.unifi.it/
	コンピュータシミュレーション及び社会科学についての国際会議	http://pscs2.irmkant.rm.cnr.it/users/rosaria/www/iccs.html
	モバイルエージェントを管理、開発するソフトウェアのパッケージを開発している	http://sun195.iit.unict.it/MAP/
	ユーザの表情に、しぐさに対してタスクを課すインテリジェントユーザーインターフェースエージェントに関して研究している	http://airone.fub.it:8080/projects/agents.html
	エージェントについてのモデルと定義	http://wwwerg.casaccia.enea.it/ing/tispi/gadomski/gad-ag.html
	マルチユーザ緊急管理サポート方法、またその方法論にマッチする情報技術システムサポートの詳細を発展させる	http://tisgi.casaccia.enea.it/projects/muster/
	知的な決定支援システムの開発	http://wwwerg.casaccia.enea.it/ing/tispi/gadomski/gad-mica.htm

	モバイルエージェントのリンク集	http://www.infosys.tuwien.ac.at/cetus/oo_mobile_agents.html
	大規模なナビゲーションとコミュニケーションサービス、保全機構とデバッギングと追跡が可能なモバイルエージェントシステム。Agent TCL。	http://www.infosys.tuwien.ac.at/Research/Agents/sp-dagents.html
	コンポーネントベースのモバイルエージェント環境	http://www.infosys.tuwien.ac.at/Gypsy/
	時間指向骨格プランの設計と実行をサポートするタスク指向問題解決手法の設計およびモデリング言語の開発	http://www.ifs.tuwien.ac.at/~silvia/projects/asgaard/
	医療分野における患者中心の要求をサポートする協調エージェント	http://www.ifs.tuwien.ac.at/~silvia/projects/pa/pa.html
	コンポーネントベースの商用マルチエージェント環境	http://www.agent-software.com.au/jack.html
	エージェントの感情的なプロセスを統一されたモデルによって表すことを研究している	http://www.ai.univie.ac.at/oefai/agents/esa.html
	インテリジェントマルチメディアインフォメーションシステムに対するアーキテクチャのフレームワークを定義している	http://www.ai.univie.ac.at/oefai/agents/werkl.html

ヨーロッパ　オランダ

ToKen 2000	Department of Computer Science, University of Limburg
Information and Probability	Department of Computer Science, University of Limburg
Conflicting Arguments	Department of Computer Science, University of Limburg
Repair-Based Scheduling	Department of Computer Science, University of Limburg
Knowledge Update in Agents	Department of Computer Science, University of Limburg
Information Reuse in Agents	Department of Computer Science, University of Limburg
The ARCHIMEDES project	Department of Computer Science, University of Limburg, Maastricht
KADS-II	CommonKADS, Universiteit Van Amsterdam
Multi-Agent Systems for Information Discovery	Computing Science Institute, Katholieke Universiteit Nijmegen

ヨーロッパ　ベルギー

Model Age	ESPRIT, Computer Science Department, University of Namur
VUB AI-lab team	VUB AI-lab
COMRIS	VUB AI-lab
The Development of Communication and Coordination	VUB AI-lab
MAGICA	VUB AI-lab
GEOMED	VUB AI-lab
CONSTRUCT	VUB AI-lab
SACEA	VUB AI-lab
NMS	VUB AI-lab
Learning hypertext	Johan Bollen, VUB AI-lab

ヨーロッパ　フランス

Arcadia	IRIT
COBALT	IRIT

	知識の開示とアクセスの方法の研究	http://www.cs.rulimburg.nl/research/neural%20networks/projects/ToKeN2000/
	知識表現と推論のための情報間の関係と主観的な可能性及び客観的定義の研究	http://www.cs.rulimburg.nl/research/agent%20technology/projects/information-and-probability.htm
	議論間の競合の解消法	http://www.cs.rulimburg.nl/research/agent%20technology/projects/conflicting-arguments.htm
	スケジューリングの修復及びその方法をスケジュール生成に応用する研究	http://www.cs.rulimburg.nl/research/agent%20technology/projects/repair-based%20scheduling.htm
	原因の完全性制約という環境でエージェント知識ベースをアップデートする研究	http://www.cs.rulimburg.nl/research/agent%20technology/projects/knowledge-update-in-agents.htm
	エージェントが新たな問題を解決するときの追加リソースとして、コンピュータの歴史の概念を紹介する	http://www.cs.rulimburg.nl/research/agent%20technology/projects/information-reuse-in-agents.htm
	ディジタル文書検索用のマルチメディア知識ベースエキスパートシステムのアーキテクチャ	http://www.cs.rulimburg.nl/~wiesman/archimedes/
	統合型KBS開発のための進化型包括的方法論	http://swi.psy.uva.nl/projects/CommonKADS/home.html
	情報検索のためのマルチエージェント	http://www.cs.kun.nl/~bernd/MAS/mas.html

	情報工学フィールドに有用なエージェントの概念の普通の正式のモデルを確立する	http://www.info.fundp.ac.be/~pys/ModelAge/
	ロボカップを題材に、インテリジェントエージェントについて研究	http://arti.vub.ac.be/RoboCup/index.html
	人間とソフトウェアエージェントとの同居を促す情報空間を提供する	http://arti.vub.ac.be/~comris/
	エージェント間の行動の調整と通信言語の進展	http://arti.vub.ac.be/~edwin/
	広域ネットワーク、インターネットとWWW上での遠隔通信サービスを通してマルチメディア電子カタログの生産とそれらの配達をするフレームワークの作成	http://arti.vub.ac.be/magica/magica.html
	局所的な情報資源とソフトウェアエージェントを用いて、全体をプランニングし調停することについて研究している	http://arti.vub.ac.be/geomed/geomed.html
	協調的な処理についてのソフトウェアエージェントアーキテクチャとアプリケーションの研究	http://arti.vub.ac.be/construct/construct.html
	遠隔通信技術とソフトウェアエージェントに基づいた分散環境の管理プロシージャの改良の研究	http://arti.vub.ac.be/milieu/milieu.html
	新しいマルチメディアネットワークとサービスのためのユーザニーズと技術要求と、既存のネットワークアーキテクチャ上でのその影響を研究する	http://arti.vub.ac.be/nms/nms.html
	学習ハイパーテキストネットワーク、連合の分散知識表現、動的知識開発への進化のアプローチ	http://pespmc1.vub.ac.be/papers/namurart.html

	分散リソースネットワークにおいて関係ある情報を見つける協調モバイルエージェント	http://www.info.unicaen.fr/~serge/3wia/workshop/papers/paper30.html
	KQML と CORBA に基づいたエージェントコミュニケーションツール	http://www.irit.fr/~Dominique.Benech/cobalt/index.html

Chicoisne 99	MAGMA, Laboratoire LEIBNIZ, IMAG
Occello 98	MAGMA, Laboratoire LEIBNIZ, IMAG
Van Aeken 99	MAGMA, Laboratoire LEIBNIZ, IMAG
Agent Technologies	Agent Technologies
Agents' World	Laboratoire d'Informatique de Paris Nord, Institut Galilee, Universite Paris 13
ACACIA	Inria
Agent Technologies	MBA

ヨーロッパ スペイン

SMASH	IIIA, CSIC
COMRIS(スペイン)	IIIA, CSIC
Dialogic Frameworks	IIIA, CSIC
Federated Learning	IIIA, CSIC
Jim	IIIA, CSIC
Fishmarket	IIIA, CSIC
Agents2000	IIIA, CSIC
MAST	Grupo de Sistemas Inteligentes, dit, UPM
Agentes Moviles	Oviedo 大学

ヨーロッパ アイルランド

A Survey of the State of the Art in Mobile Agent Systems	Department of Computer Science, National University of Ireland, Maynooth
Agent Links(アイルランド)	Agent Links(アイルランド)
Distributed Problem Solving in Multi- Agent Systems	GET SMART, Department of Computer Science, Trinity College Dublin
Intelligent Mobile Agents in Mobile Computing	GET SMART, Department of Computer Science, Trinity College Dublin
Intelligent Interface Agents	GET SMART, Department of Computer Science, Trinity College Dublin

	エージェント間の交渉のモデル	http://cosmos.imag.fr/MAGMA/publications/abstracts/chicoisne99.html
	リアルタイムに特化したASTROエージェントモデルにおいて、外部の組織とその交渉をまとめることでうまく協調する方法を提案する	http://cosmos.imag.fr/MAGMA/publications/abstracts/occello98b.html
	基礎的なレベルにおいてエージェントの組織化を研究することを考慮した最小マルチエージェントシステム(MMAS)を提案する	http://cosmos.imag.fr/MAGMA/publications/abstracts/vanaeken99.html
	エージェントテクノロジーのリンク集	http://www.insead.fr/CALT/Encyclopedia/ComputerSciences/Agents/
	エージェントとマルチエージェントが集まる重要なイベントのフォーラム	http://goma.univ-paris13.fr/AgentsWorld/
	マルチエージェントシステムにおける知識獲得支援ツール	http://www-sop.inria.fr/acacia/page-acacia-anglais.html
	リンク集	http://www.insead.fr/Encyclopedia/ComputerSciences/Agents/

	病院でのメディカルサービスのためのマルチエージェントシステム	http://www.iiia.csic.es/Projects/smash/
	人間とソフトウェアエージェントとの同居を促す情報空間を提供する	http://www.iiia.csic.es/Projects/comris/
	人工知能エージェントと人間のエージェントを含む構造化されたエージェント交渉の正式な記述としてDialogicalフレームワークを開発	http://www.iiia.csic.es/Projects/agents/agents.html
	エージェント間の協調に関する研究	http://www.iiia.csic.es/Projects/FedLearn/fed-learn.html
	高度なプログラミング可能な交渉と調整サービスをエージェントに対して提供するインターエージェント	http://www.iiia.csic.es/Projects/Jim/Jim.html
	インターネット上でエージェントを使った入札と競売の研究	http://www.iiia.csic.es/Projects/fishmarket/
	広範囲な、予測できない環境においても活動できるエージェントAGENT2000について研究	http://www.iiia.csic.es/agents2000/
	エージェント間通信のための汎用分散フレームワークとして考案されたマルチエージェントシステムツール	http://www.gsi.dit.upm.es/~mast/
	エージェントの移動性について、マルチエージェントテクノロジーに基づいて研究している	http://www.uniovi.es/~oviedo3/arturop/MobAgentsSpa.html

	リンク集	http://csa8.cs.may.ie/projects/
	エージェントテクノロジーのリンク集	http://odyssey.ucc.ie/www/user-dirs/colmor/jwz/agents_links.html
	マルチエージェントシステムにおける分散問題解決	http://www.cs.tcd.ie/research_groups/aig/iag/areab.html
	モバイルコンピューティングにおける知的モバイルエージェントの研究	http://www.cs.tcd.ie/research_groups/aig/iag/area2.html
	知的なインタフェースエージェントの研究	http://www.cs.tcd.ie/research_groups/aig/iag/area3.html

KIMSAC	Broadcom

ヨーロッパ　スイス

Projects at the Artificial Intelligence Lab	Artificial Intelligence Laboratory, Department of Information Technology, University of Zurich University of Zurich
Olivier's Pointers	Olivier Baujard
DISA	Olivier Baujard
PLATFORM	IDSIA

ヨーロッパ　ノルウェイ

TACOMA	Department of Computer Science, Faculty of Science, University of Tromso
The Virtual Secretary	Gunnar Hartivigsen Group

ヨーロッパ　ハンガリー

Multi-Agent Modeling Language(MAML)	Multi-Agent Modeling Language(MAML)
HUNABC	Central European University

ヨーロッパ　フィンランド

Monads	ヘルシンキ大学
ACADEMIC RESERCH PROJECTS	ヘルシンキ大学

ヨーロッパ　ポルトガル

COORDINATION 2000	GLOC, Computer Science Department, Faculty of Science and Technology, New University of Lisbon
Coordina	GLOC, Computer Science Department, Faculty of Science and Technology, New University of Lisbon

	新技術のアプリケーションを通してキオスクベースの情報サービスの有用性を拡張する	http://www.broadcom.ie/kimsac/

	形態学と制御の間の相互依存におけるエージェントへの進化のアプローチデザイン、およびバイオロボットによる検証	http://www.ifi.unizh.ch/ailab/
	マルチエージェントのリンク集	http://www.expasy.ch/sgaico/html/olb/Sources/Access/Pointers.html
	病院組織や製造の分野で実用的な知識ベーススケジューリングシステムの開発	http://www.expasy.ch/sgaico/html/olb/Sources/Disa/Disa.html
	タイムテーブル型架線運送システムのためのコンピュータで制御された貨物プラットフォーム	http://www.idsia.ch/PLATFORM/

	エージェントをサポートしたＯＳに焦点をあて、また、クライアント／サーバーモデルのような他の分散型システムによって伝統的にアドレスされた問題を解決するためのエージェントの使い方を研究する。	http://www.tacoma.cs.uit.no/
	秘書エージェントのセキュリティ環境について研究	http://www.vise.cs.uit.no/vise/

	エージェントをベースとしたモデルを構築するためのプログラミング言語	http://www.agentbuilder.com/AgentTools/
	エージェントとマルチエージェントのためのフォーラム	http://www.ceu.hu/hunabc/

	適合型のインテリジェントエージェントをモバイルという観点から、そのアーキテクチャの構築を目的として、研究している	http://www.cs.helsinki.fi/research/monads/
	リンク集	http://www.cs.helsinki.fi/%7ehhelin/agents/research.html

	協調モデルと言語についてのカンファレンス	http://www-gloc.di.fct.unl.pt/coord00/
	アプリケーションの協調モデル	http://www-gloc.di.fct.unl.pt/activity/coordina/

B3　日本

アジア　日本

プロジェクト名	制作グループ
エージェント指向パラダイムによる超並列分散知能処理システムに関する研究	九州大学
マルチエージェントシステムを用いた生態系のモデルに関する研究	島根医科大学
エージェントグループ	名古屋大学
モバイルコミュニケーショングループ	名古屋大学
	名古屋大学
未来開拓学術研究プロジェクト	慶應義塾大学理工学部
エージェント指向コンピューティング	慶應義塾大学理工学部
未来開拓学術研究プロジェクト	慶應義塾大学理工学部
未来開拓学術研究プロジェクト	慶應義塾大学理工学部
未来開拓学術研究プロジェクト	慶應義塾大学理工学部
所真理雄研究室	慶應義塾大学理工学部
情報学部貝原俊也（個人）	流通大学
石田研究室	京都大学
MarketBase QOS Control Project	京都大学
Internet Analysis プロジェクト	京都大学
Data Mining プロジェクト	京都大学
Perceptual Information Infrastructure プロジェクト	京都大学
Mobile Assistant プロジェクト	京都大学
Agent Network プロジェクト	京都大学
Real-Time Search プロジェクト	京都大学
Production System プロジェクト	京都大学
ISDNETプロジェクト	ISDNET
マーキュリー	ISDNET
耐シンパエージェント調査	ISDNET
サイバーエージェント「えみちゃん」プロジェクト	日本工業大学

	コメント	ホームページ
	エージェント指向パラダイムによる超並列分散知能処理システムに関する研究	http://www_al.is.kyushu-u.ac.jp/AGENT/index.html
	マルチエージェントシステムを用いた生態系のモデリングに関する研究	http://www.shimane-med.ac.jp/jcmi97/paper/075-254.htm
	コミュニケーションを取る過程におけるエージェントの決定に関する研究	http://www.inagaki.nuie.nagoya-u.ac.jp/research/AGNT/index.html
	モバイルコミュニケーションに関する研究を進めている	http://www.inagaki.nuie.nagoya-u.ac.jp/research/MOBILE/index.html
	モバイルコミュニケーション、マルチモーダルインターフェースについて研究を進めている	http://www.inagaki.nuie.nagoya-u.ac.jp/person/kawaguti/index.jap.html
	エージェント指向コンピューティング」ならびに複雑系物理モデルによる生命情報システムの研究を行っている	http://www.future.st.keio.ac.jp/
	巨大で複雑な開放型システムの構築のための方法論，理論，プログラミング手法，実行技法，などの開発	http://www.future.st.keio.ac.jp/agent/index.html
	マルチエージェントに関する研究	http://www.mt.cs.keio.ac.jp/person/kei.html
	進化・遺伝の過程を模倣した遺伝アルゴリズムを用いて、ダーウィン型とラマルク型の進化の過程を再現	http://www.mt.cs.keio.ac.jp/person/sasaki/sasaki_j.html
	モーバイルエージェントを中心的に研究	http://www.mt.cs.keio.ac.jp/person/maeda.html
	慶應義塾大学理工学部未来開拓学術研究プロジェクトを推進している	http://www.mt.cs.keio.ac.jp/
	博物館用自己想起型インターフェースエージェントについて研究	http://majesty.umds.ac.jp/staff/kaihara/kaihara.html
	主にエージェント関係の研究を進めている	http://www.lab7.kuis.kyoto-u.ac.jp/j-index.html
	アプリケーション QoS 制御システム QoS の設計と実装	http://www.lab7.kuis.kyoto-u.ac.jp/research/japanese/multiagent/qos.html
	ネットワーク統計データからユーザの集団的行動の予測に関する研究	http://www.lab7.kuis.kyoto-u.ac.jp/research/japanese/multiagent/internet.html
	大量の情報から有用な知識を得るための研究	http://www.lab7.kuis.kyoto-u.ac.jp/research/japanese/multiagent/datamining.html
	マルチエージェントシステム 視覚エージェントの実環境における様々な行動のモデル	http://www.lab7.kuis.kyoto-u.ac.jp/research/japanese/distributed-vision/pii.html
	モバイルエージェントを利用した実世界における人々の日々の行動を支援、過去のプロジェクト	http://www.lab7.kuis.kyoto-u.ac.jp/research/japanese/community-networking/mobileassist.html
	協調プロトコルエージェント記述言語 AgenTalk の研究開発、過去のプロジェクト	http://www.lab7.kuis.kyoto-u.ac.jp/research/japanese/multiagent/agentnet.html
	自律エージェント、自律移動ロボットや実時間システムのためのプランニング、過去のプロジェクト	http://www.lab7.kuis.kyoto-u.ac.jp/research/japanese/multiagent/rtsearch.html
	マルチエージェント・プロダクションシステムの構築、過去のプロジェクト	http://www.lab7.kuis.kyoto-u.ac.jp/research/japanese/multiagent/pdsystem.html
	耐シンパエージェント、モバイルエージェントの実装や、モバイルエージェントのセキュリティーの調査	http://www.isdnet.co.jp/
	Java による移動エージェント言語	http://www.isdnet.co.jp/product/index.htm
	モバイルエージェントに関するセキュリティーについて	http://www.isdnet.co.jp/IPA/
	仮想空間におけるアバタについて研究	http://www3.nit.ac.jp/~shindo/research.html

慶応大学　安西研究室	慶応大学
PRIME PRJECT	慶応大学
Computer Assisted Language Learning Group	徳島大学
WIDE	WIDE
ITS 高次推論プロジェクト	電気通信大学
次世代マイクロカーネルプロジェクト	慶應義塾大学
慶応大学徳田研究室 組み込みシステム用基盤ソフトウェア SMAF の研究開発	慶應義塾大学
モバイルエージェント、エージェントプラットフォームプロジェクト	慶応義塾大学
長尾研究室	東京工業大学
原 章(個人)	東京工業大学
上林研究室	京都大学
北橋研究室	大阪大学産業科学研究所
サイバーメディア研究会	情報処理学会
適応ビジョンシステムに関する研究（RWC）	電子技術総合研究所
マルチモーダル対話システム	電子技術総合研究所
長谷川修（個人）	電子技術総合研究所

	センサネットワークを利用する知的なソフトウェア及びそのためのプラットフォームの研究	http://www.aa.cs.keio.ac.jp/index-j.html
	センサネットワークを利用する知的なソフトウェア及びそのためのプラットフォームの研究	http://www.ayu.ics.keio.ac.jp/prime.html
	エージェント指向漢字学習支援システムの研究	http://www-b4.is.tokushima-u.ac.jp/index-jp.html
	コンピュータネットワークに関する諸問題を解決するための研究	http://www.wide.ad.jp/index-j.html
	ITS.高次推論プロジェクトなどで、マルチエージェントシステム、知的インターフェースなどを扱っている	http://www.ai.is.uec.ac.jp/
	分散実時間マイクロカーネル(RT-Mach)実時間ネットワークプロトコル，統合メディアサーバ，連続メディアベースなどの技術の開発	http://www.mkg.sfc.keio.ac.jp/
	エージェントの階層的構成とエージェント間移動の二つの概念を提案し、有効性を検討する	http://www.mkg.sfc.keio.ac.jp/smaf/
	創造的デジタルメディア環境の主要な構成要素を研究対象とし、RAVIO (Real-time Audio & Video Intelligent Operation)をテーマとする	http://www.ht.sfc.keio.ac.jp/COE98-RAVIO/
	視覚情報処理を含む高度な機械知能を実現することに関して、進化的計算論，神経回路網，画像工学を主に研究している	http://nagao-www.isl.titech.ac.jp/index_j.html
	マルチエージェントにおける自動グループ化についての研究	http://nagao-www.isl.titech.ac.jp/~ahara/my_work.html
	研究内容は、データベースの基礎、グループウェア技術、コンピュータによる遠隔教育支援、地理データベース 、論理設計手法の改良とデータベースとの統合等である	http://www.isse.kuis.kyoto-u.ac.jp/
	情報の伝達・蓄積においてその内容に適合したメディアの選択と統合方式，さらには表現すべき情報に自らが自在に適合させるメディアの研究開発	http://www.am.sanken.osaka-u.ac.jp/home.html
	マルチメディア・ハイパーメディア・バーチャルリアリティ・サイバースペースの有機的な統合技術、あるいはサイバースペースへのインタフェースとしてのマルチモーダル・メディア処理技術、さらにまた創生期にあるモーバイルコンピューティングやエージェントコンピューティングとメディアとの関連技術	http://www.am.sanken.osaka-u.ac.jp/cybermedia/
	画像を媒介とした人間とコンピュータとの高度な対話の実現のために、リアルワールドのシーンを理解し、人間を含む外界の多様な変化に適応できるビジョンシステムを開発	http://web.etl.go.jp/~7234/
	笑顔で話しかけるコンピュータ、すなわちマルチモーダル対話システムのプロトタイプの試作	http://www.etl.go.jp/etl/gazo/mm-press/mmpress.html
	マルチモーダルシステムにおける擬人化エージェントについて	http://www.etl.go.jp:8080/etl/gazo/people/hasegawa/hasegawa.html

情報統合対話ラボ	電子技術総合研究所
山本 吉伸（個人）	電子技術総合研究所
創発大域ダイナミクス・ラボ	電子技術総合研究所
ヒューマンメディアラボ	電子技術総合研究所
大沢一郎〔個人〕	電子技術総合研究所
和泉 潔（個人）	電子技術総合研究所
電気通信フロンティア研究開発	郵政省
西本一志（個人）	北陸先端科学技術大学院大学
第二研究室	ART
C-MAP	ART
櫻井 彰人（個人）	北陸先端科学技術大学院大学
河野小夜子（個人）	北陸先端科学技術大学院大学
Flage	情報処理振興事業協会
IDA	情報処理振興事業協会
新田研究室	東京工業大学
増塩智宏（個人）	東京工業大学
渥美研究室	創価大学
岡田中道(個人)	創価大学
前沢力 （個人）	創価大学
人間情報研究部情報基礎理論研究グループ	NTT コミュニケーション科学基礎研究所
磯崎 秀樹(個人)	NTT コミュニケーション
渡辺研究室	電気通信大学情報工学科情報メディア学講座

	音声と動画像の認識・合成・再生系の統合によって、人間と人工物が自然な対話を行うための情報統合技術の確立	http://www.etl.go.jp/~7233/intmod/
	ヒューマンインターフェースの研究などを中心に幅広く研究している	http://www.etl.go.jp/%7Eyoshinov/
	目標の効率的達成や環境変化への追従を行えるように、エージェントの進化学習機構とマクロな拘束条件の制御を統合したマルチエージェント計算モデルの実現	http://www.etl.go.jp/%7E6641/
	エージェントを用いて、データベースの自動情報収集などを行っている	http://www.etl.go.jp/%7Esakamoto/
	RASCAL: 知能エージェントシステムの関する研究	http://media.etl.go.jp/~osawa/
	人工市場に関する研究	http://www.etl.go.jp/~kiyoshi/index-j.html
	主に超高速通信技術分野、バイオ・知的通信技術分野、高機能ネットワーク技術分野についての郵政省の研究開発プロジェクト	http://www.iijnet.or.jp/scat/frontier/frontier.html
	科学・技術における創造性のみならず、芸術的な創造性も対象とし、人をもっと創造的にしてくれるようなメディアの研究	http://www.jaist.ac.jp/~knishi/index-j.shtml
	エージェント介在型のコミュニケーション形態を模索し、人間の創造性を支援するためのコンピュータシステムを研究	http://www.mic.atr.co.jp/organization/dept2/index.html
	展示会場を想定し、見学者一人一人の状況や興味に応じた展示ガイドをする個人アシスタントシステムを構築することを研究	http://www.mic.atr.co.jp/dept2/c-map/index-jp.html
	言語処理機能のモデリングの研究	http://www.jaist.ac.jp/~kouhou/FP/j/ks/a-sakurai.html
	環境問題のモデル作り（環境問題を経済、政策を合わせて考える統合モデル）＆（エージェント）シミュレーション	http://www.jaist.ac.jp/~skawano/
	自律的に適応するソフトウェアを構築するためのプログラミング言語	http://www.ipa.go.jp/NEWSOFT/public/Flage/index.html
	モバイルエージェントによる侵入追跡機能および痕跡による侵入検出手法に基づいた、システムに負荷のかけない、大規模分散環境でも適用可能なネットワーク侵入検出システム	http://www.ipa.go.jp/STC/IDA/jp/index.html
	人工知能の基礎と応用に関する研究	http://www.ntt.dis.titech.ac.jp/
	グループ間交渉について	http://www.ntt.dis.titech.ac.jp/~math/index.html
	マルチエージェント関連の研究	http://www.intlab.soka.ac.jp/~matsumi/
	マルチエージェント学習支援のための協調形態の事例ベース化に関する研究	http://www.intlab.soka.ac.jp/~kamarass/search/sotsuken/sotsuron.html
	エージェント間提携の進化的学習に関する研究	http://www.intlab.soka.ac.jp/~maezawa/
		http://www.brl.ntt.co.jp/cs/ninri-g/index-j.html
	マルチエージェント環境で他者の持つ情報を推定する高速アルゴリズム	http://www.brl.ntt.co.jp/people/isozaki/index-j.html
	エージェント分野ではマルチエージェントシステムで、情報の量と質が創発現象などマクロな特性にどう関連するかを実験的に・論理的に調べている	http://www.sw.cas.uec.ac.jp/swlab/

シンセシスの科学	日本学術振興会
創造的シンセシス支援のための開放型エージェント	日本学術振興会
自律学習機能富士通研究室	自律学習機能富士通研究室
岸本研究室	立命館大学
美濃研究室	京都大学
小倉研究室	福井大学
小川研究室	立命館大学
情報研究所	電力中央研究所
上田研究室	神戸大学
二村研究室	早稲田大学理工学部
木戸出研究室	奈良先端科学技術大学院
関研究室	奈良先端科学技術大学院
伊藤研究室	奈良先端科学技術大学院
千原研究室	奈良先端科学技術大学院
片井研究室	京都大学
イメージ情報科学研究室	イメージ情報科学研究室
自立的加工支援システム	IMS
溝口研究室	大阪大学産業科学研究所
前田恭伸（個人）	個人

	人間と科学技術とが密接に関わる分野である設計、製造，環境保全、リサイクル、安全などでは、人間指向、人間尊重を理念としたシンセシスの方法論を、基盤技術の体系化を通して科学	http://www.sugano.mech.waseda.ac.jp/synthesis/HOSright.html
	創造的シンセシス支援のための開放型エージェント	http://www.ina.sd.keio.ac.jp/~synth/
	学習型マルチエージェントシステムの研究を行っている	http://www.rwcp.or.jp/lab/al-fujitsu/al_fujitsu-j.htm
	エージェント型量子コンピューターについて研究している	http://www.falcon.cs.ritsumei.ac.jp/
	ヒューマンインターフェースなどについて研究している	http://www.imel1.kuis.kyoto-u.ac.jp/index.html
	RoboCup シミュレーションリーグの公式サーバであるサッカーサーバを用いたマルチエージェントの研究	http://bishop.fukui-u.ac.jp/
	マルチエージェントシステムを基盤技術とし，心理学的 ・認知科学的見地に基づく感情モデルや，デジタル信号処理，自然 言語処理，画像処理，CG などのメディアフュージョン（融合）の研 究	http://www.airlab.cs.ritsumei.ac.jp/
	文章のエージェント化などについて研究している	http://criepi.denken.or.jp/CRIEPI/cirl/cirl-j.html
	日本学術振興会　未来開拓学術研究推進事業「シンセサンスの科学」プロジェクト等に加わっている。中心的研究としては、創発的システムの開発などを行っている	http://www.mi-2.mech.kobe-u.ac.jp/index-j.html
	アルゴリズムの設計・解析・評価法、計算と論理に関する理論と応用、計算機のプログラムの動生成、推論機能などを中心に研究している	http://www.futamura.info.waseda.ac.jp/
	人工物と情報の氾濫する環境の把握、すなわちこのような環境を記述し利用する技術について研究している	http://ai-www.aist-nara.ac.jp/index-j.html
	計算機同士のインタラクションについての研究において、多エージェント系プロトコルの設計と検証の問題に取り組んでいる	http://narayama.aist-nara.ac.jp/index-j.html
	脳・複雑系の研究では、マルチエージェント強化学習等を行っている	http://mimi.aist-nara.ac.jp/Japanese/home-page.html
	インターフェースエージェント関連として、いたわりエージェントというものについて、研究	http://chihara.aist-nara.ac.jp/index-j.html
	共生システム論システム論的基盤の普遍性・合理性に留意しつつ新しいシステム知能化のあり方について研究している	http://www.symlab.sys.i.kyoto-u.ac.jp/
	委託開発事業として感性擬人化エージェントのための顔情報処理システムの開発などを取り扱っている	http://www.tokyo.image-lab.or.jp/
	自律加工支援システム(AMSS)の基本ソフトの開発	http://www.ims.mstc.or.jp/project/project_ims/ims9923.html
	オントロジー高額に関して、未来開拓学術進行事業のシンセサンスの科学に貢献している	http://www.ei.sanken.osaka-u.ac.jp/japanese/
	研究の題材としては，社会システム・環境システムを取り上げ、その中の情報の流れを研究対象としている	http://www.sys.eng.shizuoka.ac.jp/hello/maeda.html

吉田 順（個人）	個人
高橋　浩之(個人)	東京工業大学
石川・結城研究室	電気通信大学知能機械工学科
千石講師（個人）	金沢工業大学
山川研究室	早稲田大学
勅使河原研究室	創価大学
西田結集型プロジェクト	郵政省通信総合研究所関西先端研究センター
西田研究室	東京大学
畦地真太郎(個人)	個人
知識処理研究室	郵政省
MACC マルチ・エージェントと協調計算研究会	情報科学会
伊藤研究室	名古屋工業大学
IKYO	IKYO
丹沢安治教授（個人）	個人
電子図書館システム開発事業	電子図書館システム開発事業
ＷＧ３　複合コンテンツ対応技術（エージェント機能）調査	ECOM
エレクトロニック・コマース実験における 複合コンテンツサービス 開発・提供プロジェクト	ECOM

	感情や信念の変化が起きやすいディスカッションを例とし，マルチエージェントモデルを用いて相互作用を説明	http://www.valdes.titech.ac.jp/~yoshida/
	コストレス・シグナリング理論への数理的アプローチ	http://vs.valdes.titech.ac.jp/~takaha/
	エージェント技術を採用した設計支援システム	http://www.ishikawa.mce.uec.ac.jp/
	ネットワーク上の資源を有効活用するため、安全で高性能なモバイルエージェント用プラットフォームの機能を考案し、実装する	http://www.infor.kanazawa-it.ac.jp/~sengoku/index-j.html
	エージェントによる機械構造設計の最適化及び満足化	http://www.yamakawa.mech.waseda.ac.jp/index-jp.html
	線 LAN、衛星通信、モバイル環境下でのトラフィックマネージメント、集団意志決定支援	http://netlab.t.soka.ac.jp/teshiga/index-j.html
	情報ネットワークが人間の社会生活に与える影響を、社会心理学的アプローチ、認知心理学的アプローチ、情報科学的アプローチを総合して解明	http://www-karc.crl.go.jp/synsophy/index-j.html
	仮想世界と物理世界での人々の知的活動（情報収集・整理・体系化・共有・創出）を支援する枠組み：知識コミュニティを提唱し、実証システムの研究開発	http://www.kc.t.u-tokyo.ac.jp/
	エージェントの個性について、ネットワーク社会におけるコンフリクトの研究	http://www-karc.crl.go.jp/synsophy/azechi/japan/index.html
	話し言葉対話システムの設計と構築に関する研究　自律エージェントによるコミュニケーション支援　前言語的コミュニケーションの認知科学　意味の進化のシミュレーション	http://www-karc.crl.go.jp/kss/indexj.html
	ソフトウェア学会のマルチ・エージェントと協調計算研究会	http://cape.etl.go.jp/macc//
	エージェントのフォーメーション維持に関する研究　生物の模擬によるマルチエージェントの振舞いに関する研究　遺伝アルゴリズムによるエージェント動作の研究　ゲーム理論に基づいたエージェント間の秩序維持の研究	http://www-itolab.ics.nitech.ac.jp/
	エージェントに教師の役割を置き換え	http://www.voc.or.jp/i-kyo/
	情報テクノロジーの発達とともに現れてきた新しい組織形態（共生・ネットワーク組織、モジュラー組織、バーチャル・コーポレーション）、取引形態（電子市場、戦略的提携）を新制度派の経済学（プロパティー・ライツ理論、取引費用理論、プリンシパル・エージェント理論）を用いて分析する	http://www.senshu-u.ac.jp/~thm0175/sensei.htm
	電子図書館に代表される情報流通基盤の確立を目的	http://www.dlib.jipdec.or.jp/
	“エージェントサービス”は、学術的なエージェントの定義にとらわれず、個人の要求に対して（静的でなく）パーソナライズされた何らかのサービスを実施しているサイトに対して、評価を行っている	http://www.ecom.or.jp/about_wg/wg03/lsagent.htm
	日本型のＥＣについての考察などが行われている	http://www.ecom.or.jp/19pr/pia/index.htm

岡田 孝	関西学院大学情報メディア教育センター
石塚伊庭研	東京大学
西尾研究室	大阪大学
山田研究室	東京工業大学
市川研究室	千葉大学
國藤 進（個人）	個人
富田章夫（個人）	個人
瀧本　周平（個人）	個人
ロボット班	関西大学情報工学研究室
宮永 喜一（個人）	個人
吉田順（個人）	個人
今井研究室	東京大学
小林研究室	東京工業大学
程 子学（個人）	会津大学
佐藤 和彦（個人）	会津大学
小暮 晶子（個人）	会津大学
山口研究室	静岡大学
北村泰彦	大阪市立大学
人工知能学会人工知能基礎研究会	人工知能学会
プランジェント	東芝

	人間の思考能力と知識やパターンの認識、検索能力を計算機上に実現し、人間が主導権を取りながら、なめらかに共同作業のできる知的エージェントの開発が基本的な研究目標	http://www.clab.kwansei.ac.jp/~okada/
	マルチモーダル擬人化インタフェースとその感性基盤機能	http://www.miv.t.u-tokyo.ac.jp/index.htm
	(1)モーバイルコンピューティング技術、(2)コンピュータネットワーク技術、(3)知識処理技術の三つの要素技術との融合による高度マルチメディア情報システムの構築手法に関する研究	http://www-nishio.ise.eng.osaka-u.ac.jp/index-jp.html
	「適応システムの構築」をテーマとして，主に人工知能を基盤とした適応・学習の理論的展開と応用を行っている	http://www.ymd.dis.titech.ac.jp/index.html
	音声対話理解、福祉情報工学 、音楽情報処理等を中心にして研究を進めている	http://www.icsd4.tj.chiba-u.ac.jp/
	発想支援システム、グループウェア、および知識システムの研究	http://www.jaist.ac.jp/~kouhou/FP/j/ks/kunifuji.html
	エージェントによるグループ意思決定を支援するWebアウェアネス環境の構築	http://www.jaist.ac.jp/~atomita/
	複数のエージェントがいかにして協調し学習して協調関係をうまく構築するかという問題を、遺伝的アルゴリズムを用いて解く可能性を示し、その過程を研究する	http://www.jaist.ac.jp/~tkmt/
	エージェントの自律的な学習について研究	http://joho.densi.kansai-u.ac.jp/~robot/index-j.html
	協調型エージェントシステムによる並列信号処理	http://csw.ice.eng.hokudai.ac.jp/SigPro/serv02.htm
	マルチエージェントによる認知モデル	http://www.valdes.titech.ac.jp/~yoshida/
	情報の符号化という立場から，情報・通信システムの効率向上，信頼性向上，情報セキュリティ向上等に取り組んでいる	http://suri.iis.u-tokyo.ac.jp/imailab-j.html
	本研究室では，DAIのベンチマークとして知られる追跡問題を対象にマルチエージェント強化学習の方法論について検討するとともに，クレーン群制御問題などの実問題への適用を通じて，工学的有用性を明らかにしている	http://www.fe.dis.titech.ac.jp/
	ネットワークエージェン トを用いる複数の委員会の会議日程調整法人間や集団心理を考慮した支援	http://www.u-aizu.ac.jp/~z-cheng/
	配属問題を例に挙げて、エージェントシステムを用いてこれらの問題にとりくんでいる	http://himalayas.u-aizu.ac.jp/~king/
	程調整の１つである複数の会議間における会議日程調整について考える	http://www.u-aizu.ac.jp/~s1041103/
	オントロジー工学を核にした インテリジェントシステムの研究開発に取り組んでいる	http://panda.cs.inf.shizuoka.ac.jp/japanese/
	マルチエージェント問題解決、マルチエージェントの実時間探索	http://www.kdel.info.eng.osaka-cu.ac.jp/~kitamura/
	エージェントに関する基礎的研究	http://www.ar.sanken.osaka-u.ac.jp/SIG-FAI/
	Plangentのエージェントは、移動機能，プランニング機能を備えることで，ネットワーク上を移動し，その場その時の最適な処理を自ら判断し，行動することができます	http://www2.toshiba.co.jp/plangent/

Beegent	東芝
mobile space	お茶の水大学
agent space	お茶の水大学
MOBA	MOBA
音楽音響信号に対するビートトラッキングシステム	電子技術総合研究所
Planet	筑波大学
大沢英一	個人
Aglets	日本 IBM
日本ユニシス　システム”ニュー”	日本ユニシス
FIPA インテリジェントエージェント研究会	FIPA
FVCS	東北大学
FAMES	東北大学
ADIPS	千葉工業大学
未来開拓プロジェクト	東北大学
菅原研究室	千葉工業大学

	Bee-gent は、"100%ピュアエージェントシステム"を構築するための新しいエージェントフレームワークです。コミュニケーションのエージェント化という新しい概念を採用して、既存アプリケーションを活かして柔軟で開放的な分散システムを構築します	http://www2.toshiba.co.jp/beegent/index_j.htm
	Java ベースの高階の Mobile Agent システム。階層的にエージェントを構成している	http://islab.is.ocha.ac.jp/ipa/index.html
	Java ベースのシンプルな Agent システム	http://islab.is.ocha.ac.jp/agent/
	Java ベースで、移動コードを意識せずに記述	http://www.shudo.net/moba/index-j.html
	音楽音響信号に対してリアルタイムにビートを認識・予測するビートトラッキングシステム（BTS）を実現する。BTS では、複数のエージェントが並列にビートを解釈することで、あるエージェントの解釈がはずれても、ビートを見失わずに正しい解釈を出力できる。また、主要なドラム音の出現位置を検出することにより、各ビートが強拍か弱拍かを判断する。	http://www.etl.go.jp/~goto/PROJ/bts-j.html
	ネイティブコードを使ったモバイルエージェントシステム．タプルスペースと、サンドボックスを利用している	http://www.osss.is.tsukuba.ac.jp/~planet/
	人工知能研究における合理的エージェントとそれらが構成する社会(マルチエージェント・システム)により，開放型分散システムや認知機構をモデル化/実装することを目標としてる	http://www.csl.sony.co.jp/person/osawa.j.html
	Java 移動エージェント技術 Aglets	http://www.trl.ibm.co.jp/aglets/index-j.html
	ネットワークコンピューティング技術、分散トランザクション技術、更にはエージェント技術を駆使して開発された次世代の企業情報システム基盤の研究	http://www.unisys.co.jp/
	研究調査、翻訳など	http://ias.comtec.co.jp/
	やわらかいビデオ会議システムは，既存のビデオ会議システムをエージェント化することにより，利用者指向／環境指向の動作を実現する	http://www.shiratori.riec.tohoku.ac.jp/~fumi/lab/index-j.html
	FAMES は，既存メールソフトに変更を加えずエージェント化し，ユーザー要求に従いさまざまな機能拡張エージェントを瞬時に再構成することにより，メールメッセージ等の知的配送処理を実現するシステムである	http://www.shiratori.riec.tohoku.ac.jp/projects/flexible-gw/index-j.html
	FASE プロジェクトを実現するためのエージェントエージェント記述言語	http://ww.adips.suga.net.it.chiba.ac.jp
	次世代のネットワーク環境のために、アプリケーションなどがネットワーク層の変化を感じることなく使用できる環境の設計を目指している	http://www.shiratori.riec.tohoku.ac.jp/projectNew/miraiKaitaku-j.html
	エージェントエージェント記述言語 ADIPS の開発を行っている	http://www.suga.net.it-chiba.ac.jp/

B4 アジア

アジア シンガポール

プロジェクト名	制作グループ
Intelligent Agent-Mediated Internet Trading	Assoc Prof David Siew
Intelligent Agents group	JSAIC
Intelligent Systems Lab, DISCS, NUS	National University od Singapore
Dr. Gary Soon Huat TAN	Dr. Gary Soon Huat TAN
CENTRE FOR SYSTEMS SECURITY	CENTRE FOR SYSTEMS システムセキュリティセンター
Mr David Siew	Mr David Siew
PRICAI'98	Kent Ridge Digital Labs
Dr. LAM Kwok-Yan	Dr. LAM Kwok-Yan
Agents	T. Rex
Publications	gtan

アジア 中国

プロジェクト名	制作グループ
Multi-agent System on manufaturing	Quianli

アジア 香港

プロジェクト名	制作グループ
Jerome's Homepage	Jerome Yen
Raymond Chung's Home Page (Reseach)	Raymond Chung
ADL 1995: McLean, Virginia, USA	ADL
CIA 1997: Kiel, Germany	CIA
AAAI 1991: Anaheim, California	AAAI
PRICAI 1996: Cairns, Australia	PRICAI
The ETO Intelligent Agent -IAgent-	ETO
NewsAgent	PCM Magazine
Joseph Lee's Home Page	Joseph K.W. Lee
The UNTPDC-RMIT Cooperation	RMIT
Intelligent Agent Technology	IAT'99

アジア インド

プロジェクト名	制作グループ
Internet - Current Cites	Andrews Rinne

アジア 韓国

プロジェクト名	制作グループ
Intelligent Agents	J. Choi
Published Papers	Jae Kyu Lee
Intelligent Software Agent Bibliography	jjyoo

	コメント	ホームページ
	電子商取引を扱うインテリジェントエージェントの研究	http://www.icis.ntu.edu.sg:8000/R&D/Intel-agent.htm
	関連組織（マルチエージェント環境）	http://jsaic.krdl.org.sg/projects/agents.html
	エージェントによるスケジューリング	http://www.iscs.nus.edu.sg/labs/intelligent/agents.html
	シミュレーションエージェント	http://www.comp.nus.edu.sg/~gtan/
		http://www.comp.nus.edu.sg/~lamky/css.html
	コミュニケーションソフトウェアシステムとネットワークに関する研究	http://www.icis.ntu.edu.sg:8000/Staff/david.html
	インテリジェントシステム（国際会議）http://jsaic.krdl.org.sg/pricai98/pricai-main.htmq	http://jsaic.krdl.org.sg/pricai98/pricai-prg-paperposter.htm
	分散システムに関する研究	http://www.comp.nus.edu.sg/~lamky/
	エージェントについて研究している組織	http://jsaic.krdl.org.sg/tralvex/pub/agents.htm
	Intelligent Agent に関する研究	http://www.comp.nus.edu.sg/~gtan/pub.html

	コメント	ホームページ
	製造（manufacturing）のマルチエージェントシステムに関するリンク	http://www.lib.tsinghua.edu.cn/english/virtual/cam/cam_agen.html

	コメント	ホームページ
	Intelligent Agent に関する研究	http://www.csis.hku.hk/~jyen/main.html
	モバイルエージェントに関する研究	http://www.cse.cuhk.edu.hk/~lmchung/research.html
	ADL95 の公演内容（Intelligent Agent の発表を含む）	http://ftp.ust.hk/dblp/db/conf/adl/adl95.html
	CIA97 の公演内容（Intelligent Agent の発表を含む）	http://sunsite.ust.hk/dblp/db/conf/cia/cia97.html
	AAAI1991 の公演内容（Intelligent Agent の発表を含む）	http://sunsite.ust.hk/dblp/db/conf/aaai/aaai91-1. html
	PRICAI 1996 の公演内容（Intelligent Agent の発表を含む）	http://sunsite.ust.hk/dblp/db/conf/pricai/pricai96.html
	情報検索エージェント	http://sunsite.ust.hk/untpdc/eto/etoagent/etoagent.html
	ニュースフィルタリングエージェント	http://stlinux.ouhk.edu.hk/mirror/nonags-w31/news
	インターネット Ag に関する研究	http://www.csis.hku.hk/~jkwlee/research.html
	協調に関する研究	http://sunsite.ust.hk/untpdc/rtpdc/rmitmenu.html
	インテリジェントエージェントテクノロジーに関する国際会議	http://www.comp.hkbu.edu.hk/IAT99/

	コメント	ホームページ
	インターネットに関する文献	http://ces.iisc.ernet.in/hpg/envis/citdoc711.html

	コメント	ホームページ
	Intelligent Agent に関する研究	http://cse.hanyang.ac.kr/~jmchoi/agents/
	Intelligent Agent に関する研究	http://iis.kaist.ac.kr/~jklee/papers.html
	Intelligent Agent に関する文献集（アブストラクト、キーワードつき）	http://atom.kjist.ac.kr/~jjyoo/agentpprIdx.html

WWW Homepage of Woongkyu Lee	Woongkyu Lee
Brad Appleton's WWW Links	Brad Appleton
Prof. Choong-Ho CHO?	Choong-Ho CHO
Introduction to Professor	Cha, Sang Kyun
Jeong, Jun Home Page	Jun Jeong
APNOMS'98 Sumitted Paper List (Korea)	APNOMS
Network Management Pointers	Luisa Rossari
Welcome To Dr. Shin's Homepage	Dr. Shin
References by Authors(Hayes-Roth, Barbara)	Hayes-Roth
AIF Projects	AIF
Jong Woo Kim's Home Page	Jong Woo Kim
Kyoung-Soo Han	Kyoung-Soo Han
Home Page for Bo-Hyun Yun	Bo-Hyun Yun
Yanghee's Backgrounds	Yanghee
Career Opportunity	Hyundai Information Technology Co., Ltd
Reactive Scheduling	winnie
Presentation Materials on Agents	jmchoi

アジア　マレーシア

プロジェクト名	制作グループ
fatimah	fatimah

アジア　ロシア

プロジェクト名	制作グループ
Audio, CD	Sovam Teleport
LAN Magazine Buyers Guide/Russian edition - Network Management	Open Systems Publishing House

アジア　台湾

プロジェクト名	制作グループ
Intelligent Agent	Yuh-Jong Hu
Intelligent Agent Systems & Applications	Yuh-Jong Hu
Intelligent Agent and its potential applications in Business	kuo
Intelligent Agents: Overview	sdwang
Internet Search Tools & Techniques	Hsiao-Tieh Pu
Kiyoko F. Aoki, Ph.D.	Kiyoko F. Aoki, Ph.D.
Jason Chi-Shun Hung	Jason Chi-Shun Hung
Jason C. Hung	Jason C. Hung
Shiow-yang Wu's Academic Publications	Shiow-yang Wu
YIH, WEN-TAU	YIH, WEN-TAU
Sheng-Yuan Yang	Sheng-Yuan Yang

	電子商取引を扱うインテリジェントエージェントの研究	http://msd.kaist.ac.kr/~lwk/
	Software Agent に関するリンク集	http://kmh.yeungnam-c.ac.kr/comScience/appleton/index.htm#Sw_Agents
	Intelligent Agent に関する研究	http://tigerking.korea.ac.kr/~chcho/main.html
	Intelligent Agent に関する研究	http://ee.snu.ac.kr/english/faculty/cha-sangkyun.htm
	Intelligent Agent に関する研究	http://inhavision.inha.ac.kr/~g1991263/English.html
	APNOMS'98 の公演リスト（Intelligent Agent の発表を含む）	http://amazon.postech.ac.kr/knom/apnoms98/paper-list-korea.html
	ネットワーク管理に関するリンク	http://amazon.postech.ac.kr/knom/apnoms98/paper-list-korea.html
	インテリジェントシステムに関する研究	http://mm.ewha.ac.kr/~ksshin/intel_home1.htm
	Agent に関する研究	http://csone.kaist.ac.kr/~sykim/BB/Reference_by_author/Hayes-RothB.html
	Intelligent Agent に関する研究	http://fuzzy.kaist.ac.kr/project/project.html
	Intelligent Agent に関する研究	http://islab.chungnam.ac.kr/~jwkim/index.html
	Intelligent Agent に関する研究	http://nlp.korea.ac.kr/~leganza/index_e.html
	Intelligent Agent に関する研究	http://nlp.korea.ac.kr/~ybh/
	Intelligent Agent に関する研究	http://vr.kaist.ac.kr/~nyh/career.html
	電子商取引におけるインテリジェントエージェントの研究	http://www.hit.co.kr/hitwww/htdocs/e_exp.htm
	インテリジェントエージェントを用いたスケジューリングの研究（アブストラクト）	http://red.hnc.co.kr/~winnie/papers.htm
	Agent に関する文献集	http://cse.hanyang.ac.kr/~jmchoi/agents/presentation.html

	コメント	ホームページ
	電子商取引に関する研究	http://www.econ.upm.edu.my/~peta/fatimah.html

	コメント	ホームページ
	電子商取引エージェント	http://alt.www.online.ru/emain/eshop/ecd.html
	ネットワークエージェント	http://www.osp.raid.ru/lan/bg97/40.htm

	コメント	ホームページ
	Intelligent Agent に関する研究	http://www.cs.nccu.edu.tw/~jong/agent/IA/ia.html
	Intelligent Agent System に関するリンク	http://www.cs.nccu.edu.tw/~jong/agent/IA/ias.html
	Intelligent Agent に関する研究	http://www.cis.scu.edu.tw/~kuo/agent/sld001.htm
	Intelligent Agent の概要	http://hpc.ee.ntu.edu.tw/sdwang/AI/ai-ia.html
	インターネットに関する研究	http://www.stic.gov.tw/stic/eng/asca/chap4.htm
	エージェントを用いた分散システムの研究	http://pubweb.nwu.edu/~kaoki/resume.html
	Intelligent Agent に関する研究	http://www.mine.tku.edu.tw/home/isiah/index.htm
	Intelligent Agent に関する研究	http://www.mine.tku.edu.tw/home/isiah/index.htm
	Intelligent Agent に関する文献集	http://ravel.csie.ndhu.edu.tw/publications.html
	WEB 情報をあつかうエージェントに関する研究	http://www.cc.nctu.edu.tw/~mjtsai/post_2/Hof_new.html
	Intelligent Agent に関する研究	http://ailab2.et.ntust.edu.tw/~ysy/ewho.html

Projects	Hsiangchu Lai
Publication	Rey-Long Liu
Artificial Intelligence: A Modern Approach	Stuart Russell
Shiow-yang Wu's Academic Publications	Shiow-yang Wu
Commercial Scenarios for the Web: Opportunities and Challenges	Donna L. Hoffman
Recommand Books	Java Center
B. CONFERENCE PAPER	Nian-Shing Chen
Chang, Chia-Hui	Chang, Chia-Hui
Is it an Agent, or just a Program?: A Taxonomy for Autonomous Agents	Stan Franklin

アジア　タイ

プロジェクト名	制作グループ
Introduction	The National Computer Science and Engineering Conference
Selected Publications	Lee Spector
Intruder Alert	AXENT Technologies, Inc

	インテリジェントエージェントによるカスタマイズされたインターネットマーケットシステムに関する研究	http://www.mis.nsysu.edu.tw/~hclai/projecten.htm
	Intelligent Agent に関する研究	http://www.mi.chu.edu.tw/~rlliu/Publication.htm
	Intelligent Agent に関する本の紹介	http://www.csie.ntu.edu.tw/~ai/online/aima.html
	電子商取引を扱うインテリジェントエージェントの研究	http://ravel.csie.ndhu.edu.tw/publications.html
	Web でのコマーシャルで使われる検索エージェント	http://www.cc.nctu.edu.tw/~mjtsai/post_2/Hof_new.html#search
	Intelligent Agent に関する本の紹介	http://javacenter.cis.thu.edu.tw/jc2.3/resource/book/book.htm
	Intelligent Agent に関する研究	http://www.nsysu.edu.tw/~nschen/refer/ConferencePaper.html
	Intelligent Agent に関する研究	http://ails6.csie.ntu.edu.tw/~chia/index.html
	自律エージェントの分類を提案する論文	http://www.ccunix.ccu.edu.tw/~d8742021/agentprog.html

	コメント	ホームページ
	Intelligent Agent に関する研究	http://www.cpe.ku.ac.th/news/ncsec98/html/intro.html
	Intelligent Agent に関する研究	http://orange.cp.eng.chula.ac.th/Prabhas/spector_pub.html
	エージェントを用いた警告システム	http://www.oga.co.th/syncom/axent/product/ita/ita.htm

B5　オセアニアその他

オセアニア　ニュージーランド

プロジェクト名	制作グループ
SSMAM	Software Engineering and Collaborative Modelling Laboratory
SARG: Agent-Integrated Desktop Tools	Software Engineering and Collaborative Modelling Laboratory
Database & Software Engineering Research Group	Department of Computer Science

オセアニア　オーストラリア

プロジェクト名	制作グループ
JACK Intelligent Agents	
Aditya K. Ghose	University of Wollongong
Sattiraju Prabhakar	
Chengqi Zhang	Deakin University
David Reilly	Bond University
dMARS	dMARS Product
Multiagent Problem Solving	University of New South Wales
BDIM Agent Toolkit and TOMAS	University of Melbourne
A Methodology For Developing Agent Based Systems For Enterprise Integration	RMIT
Bob Kummerfeld	University of Sydney
Intelligent Agent Laboratory	University of Melbourne
ARPEGGIO	University of Melbourne
LogicWeb	University of Melbourne
CiFi	University of Melbourne
Classified Advertisement Search Agent (CASA)	University of Melbourne
ARIS	University of Melbourne

アフリカ　南アフリカ共和国

プロジェクト名	制作グループ
The High Speed Networks Team	University of Pretoria

アフリカ　エジプト

プロジェクト名	制作グループ
Khaled Shaalan	Cairo University

	コメント	ホームページ
	エージェントモデルによるデータ解析・シミュレーション	http://divcom.otago.ac.nz:800/infosci/ssmam/index.html
	エージェント指向プログラム開発のためのアーキテクチャの研究	http://divcom.otago.ac.nz:800/infosci/SECML/lab/ABSI.html
	データベース・知的エージェントシステム	http://www.cs.otago.ac.nz/DB/index.html

	コメント	ホームページ
	JAVA をベースとしたエージェントプログラミングフレームワーク	http://www.agent-software.com.au/
	エージェントシステムによる推論アプリケーションの開発	http://www.uow.edu.au/~aditya/
	人工知能における自律学習エージェントの研究	http://www-staff.mcs.uts.edu.au/~prabhaka/
	マルチエージェントシステムによる分散人工知能システム	http://www3.cm.deakin.edu.au/~chengqi/
	WWW リソース収集アプリケーションの開発	http://www.fan.net.au/~dodo/rdm/
	エージェント指向開発環境	http://www.aaii.oz.au/proj/dMARS-prod-brief.html
	マルチエージェントによる問題解決	http://www.cse.unsw.edu.au/~aishare/3.RESEARCH_THEMES/MULTI-AGENT.PROB.SOLVING/m-a.prob.solv.html
	JAVA による知的エージェントフレームワーク	http://www.cs.mu.oz.au/~paolo/TOMAS/
	エージェントシステム構築の方法論	http://www.cse.rmit.edu.au/~rdsek/
	エージェントをベースとしたフィルタリングシステム	http://www.cs.su.oz.au/People/bob.html
	マルチエージェントシステム・フレームワークの研究	http://www.cs.mu.oz.au/agentlab/
	エージェントをベースとしたプロトタイプ環境	http://www.cs.mu.oz.au/agentlab/projects/arpeggio.html
	WWW 上の論理プログラミング	http://www.cs.mu.oz.au/agentlab/
	WWW 上での引用の検索	http://www.cs.mu.oz.au/~swloke/cifi/cifi.html
	WWW 上での広告の検索	http://www.cs.mu.oz.au/~xga/casa.html
	Web エージェント作成ツール	http://www.cs.mu.oz.au/agentlab/projects/aris.html

	コメント	ホームページ
	知的エージェントフレームワーク・CORBA	http://www.up.ac.za/academic/skoolit/atm/intro.html

	コメント	ホームページ
	オブジェクト指向プログラミング・知識をベースとした開発環境	http://www.claes.sci.eg/claes/staff/shaalan.html

南アメリカ　アルゼンチン

プロジェクト名	制作グループ
Intelligent Agents Laboratory	Tandil University
The Brainstorm Project	Tandil University

南アメリカ　ブラジル

プロジェクト名	制作グループ
The Behavior Based Model for Intelligence in Software Agents	University of Sao Paulo
Universal Communication Language (UCL)	University of Sao Paulo
Geber Ramalho	Federal University of Pernambuco
A Cooperation Metodology for Cognitive Agents	Universidade Federal de Santa Catarina

中央アメリカ　メキシコ

プロジェクト名	制作グループ
Centro de Inteligencia Artificial	Tecnologico de Monterrey
Intelligent Agents Repository	Center for Artificial Intelligence

	コメント	ホームページ
	エージェントの各種研究･調査	http://www.exa.unicen.edu.ar/~aigroup/lai/
	論理プログラミングを支援するマルチエージェントシステム	http://www.exa.unicen.edu.ar/~amandi/brainstorm.html

	コメント	ホームページ
	協調作業	http://java.icmsc.sc.usp.br/research/
	エージェント間言語	http://java.icmsc.sc.usp.br/research/
	人工知能･分散システム	http://www.di.ufpe.br/~glr/
	協調作業	http://www.lcmi.ufsc.br/~loureiro/index-2.html

	コメント	ホームページ
	分散人工知能システム	http://www cia.mty.itesm.mx/~lgarrido/index.html
	知的エージェントに関する総合リンク集	http://www-cia.mty.itesm.mx/~lgarrido/Repositories/IA/

資料編 C. ワークショップ記録

ネットワークエージェント技術に関するワークショップ

1. 日時：平成 11 年 12 月 9 日（木） 9：30～16：40
2. 会場：芝パークホテル本館 3 階 牡丹の間

C. ワークショップ記録

セッション１：ビジネス・産業応用

（１）EC（電子商取引）への応用　服部文夫（ＮＴＴソフトウェア）

EC におけるエージェント技術の応用に関する概観が示された。まず日米の EC マーケットの比較と展望が行われた後、エージェント技術の役割についての解説があった。B-to-C におけるエージェントの重要な役割としては、購入商品の斡旋、購入先の斡旋、交渉などがあり、このための事例として、Amazon.com、ONSALE、Kasbah などが紹介された。また B-toB におけるエージェント技術として、電子カタログ、マッチメイキング、サプライチェーン管理などがある。今後はマルチエージェント技術やエージェント基盤やプロトコルが重要になる。

（２）エージェント技術の応用　木野茂徳（三菱電機）

Java ベースの移動型エージェントの開発支援環境 Concordia とその活用事例の紹介がなされた。その特徴として、セキュリティ、永続性、交渉機能、エージェントサーバの遠隔管理／監視が挙げられる。またその応用例としては、EAI (Enterprise Application Integratio)、設備情報管理システムが挙げられた。その中で、XML の導入により、Java コードの減少という利点が得られた。今後の課題としては、交渉機構を中心としたマルチエージェント処理機構が挙げられる。

（３）東芝のネットワークエージェント技術とその応用　大須賀昭彦（東芝）

東芝が開発した二つのネットワークエージェントの紹介がなされた。知的ネットワークエージェント Plangent は知識処理機能と学習機能をもったモバイルエージェントで、状況にあった処理を推論により動的に生成することで、状況によって変化する処理に対応できる。またマルチエージェントフレームワーク Bee-gent はエージェント間の連携機能を持つモバイルエージェントであり、エージェントラッパー、モバイル仲介エージェント、エージェント間コミュニケーション言語を活用することにより、アプリケーション、DB、WEB ページなどの既存のソフトウェア資産を緩やかに連携させる。応用事例としてドライバー情報支援エージェント InfoMirror、ヒューマンナビゲーションエージェント、電力会社用機器巡回／点検エージェントが紹介された。今後の課題として、相互運用性、標準化、柔軟性、信頼性、セキュリティなどが挙げられた。

（４）ビジネス・産業応用から見たエージェント技術 - 生産制御システムへの応用 -

和田英彦（横河電機）

現在の生産制御システムの課題に向けた試みとして移動エージェントを用いた横河電機における加工装置制御システムの紹介が行われた。本システムの特徴は、エージェントが自律的に動作し、柔軟なシステムの構築を行うことで集中管理コンピュータを必要としないこと、また個々のエージェントが製造手順情報を持つことで、製品の製造手順を容易に変更できることが挙げられる。基本的なコンセプトの実証のためにプロトタイプシステムが開発された。

〈質疑応答〉

Qマツウラ（松下電器）　ただ今の横河電機さんの発表で、実際の製品とコンピュータ上のエージェントがうまくマッチすると非常におもしろいなと思ったのですが、実際問題として、物の動きとエージェントの動きをどういう形で合わされているのか教えてほしいです。

A和田　先ほどのシステムは実験室レベルなので「見ていればわかる」と言って逃げていますが、実際にはそういうわけにはいかなくて、バーコードやIDタグを付けてトラッキングをし、物と情報の同期を取らなければいけないと考えています。

Qオカダ（富士通研究所）　服部さんにおうかがいします。ECへの応用を考えたときにXMLは外せないと思うのですが、特にB-to-B応用を考えたとき、ロゼッタネットなどはエージェントに対してどういうアプローチをしているのか、ご存じだったら教えてほしいのですが。

もう一つ、B-to-Cアプリケーションの場合、ご指摘のとおりレコメンディングシステム、要するにユーザプロファイルが非常に大事だと思います。これは大須賀さんにもおうかがいしたいことですが、複数の企業や自治体が作ったシステムでどうやってユーザプロファイルを共有するのでしょうか。と言うのも、私はamazon.comへ行ったときや紀伊国屋へ行ったときに自分のプリファレンスを何回も入れたくないわけです。そういうものをどういうふうに解決しようとしているのか、それにエージェントはどうやって絡んでいくのかを教えていただけますか。

A服部　ロゼッタネットそのものに関してはいま把握していません。全体の流れとしてはXMLという表記形式はいいのですが、当然そこにセマンティックスをきちんと入れていかなければいけないわけです。AIの分野でいくとオントロジーですけれども、それをこれから作っていかなければいけないわけです。どういうふうにやるかはなかなか難しいのですが、実際にどんどん作っていこうという立場で言いますと、業界ごとにそれが作られるということで、一部の分野ではたぶんそういう作業が進んでいるのではないかと思います。金融の分野とか化学の分野はそういうボトムアップの流れが積み重なっていって、それがさらに相互に流通していくという感じになるのではないかと思います。今それをどこかの団体がトップダウンでやるという形にはなかなかいかないのではないでしょうか。

個人情報のプロファイルの件に関しましては、一つの流れとして、そういうプロファイルを相互に提供するサービスを商売にすることがアメリカなんかでは出てきています。要するにいろいろな企業で収集しているプロファイル情報をどこかのセンターへ集めて、それを売るという実際のビジネスが出てきています。しかしプライバシー保護の問題があって、どこまでなら売っていいかとか、提供した人の了承をどこまで得たらいいかとか、そのへんの難しい問題がありますので、これからどうなるかわからないですね。

A大須賀　個人情報の話はわれわれにとっても課題でして、他でもいろいろ議論されていると思います。エージェントが得る個人情報は、名前、住所、IDカード番号以上に個人の好みなどが入っていて、たとえばドライバー情報支援ですと、その人はどういうところによく行くのかとか、行動に関する情報まで結構取れてしまうんです。当然個人の方としては見られたくない。だけど企業側からするとその人がどういう行動をしているか、どうい

う買い物をしているかというのは非常に欲しい情報なので、そこはけっこう凌ぎ合いがあるかなと思います。

現状はそういうインフラができていないので、われわれのアプローチでは個人情報は自分の端末側に置いて、外から見えない形にしています。そうすると、その端末が起きていないとその個人の情報が活用できなので、今はモバイルエージェントがそこまで飛んでいって個人情報を取るようにしています。

Qオノ（東京電力）　東芝の大須賀さんと三菱の木野さんのお二方に教えていただきたいと思います。使う側からしますと、エージェント技術の非常に気になる点がSecurityと標準化の二点です。Securityに関しましては、たとえばPlangentですと指紋認証、秘密鍵といった技術、あるいはConcordiaの方では公開鍵暗号化、といった話を聞いたことがありますが、そちらについてどういう取り組みをされているのか教えていただけますか。

A大須賀　Securityに関しては、ネットワーク自体がどういうポリシーでSecurityを管理しようとしているのかが非常に重要で、エージェントの立場としてはそれを素直に継承しています。たとえばIDパスワードとか認証の話、アクセスコントロールの話などはそのまま継承していて、その上でエージェントにも認証が必要ならそのコードを乗せたり、アクセスコントロールも同様に、エージェントができるアクションの権限を与えたりしています。エージェントの中身が盗まれない、見られないという意味では暗号化とか認証の話も入れています。ですから個人情報が特別といった以外はネットワークインフラの話をそのまま持ってきているのが現状です。

A木野　三菱も基本的にそうなのですが、お客様によっては下位のレイヤで何もしていないところもあって、エージェントのレベルで暗号化と認証をSSLライクなことでやっています。ご指摘のとおり公開鍵です。当社にMistyというのがありまして、これは公開鍵ではないのですが、そういうのをやったりしています。

Securityに関しては、モバイルエージェントならではのところがいくつかあります。たとえばパーシステンシーをお話ししましたけれども、それをキープするために一度ディスクにしまってしまいますが、そのディスクが誰かから覗かれたりしないように、そこにも暗号を掛けるか掛けないかということが問題になります。

あとはアクセスパーミッションですが、このアクセスパーミッションは、いまJava2のところでいろいろSecurityモデルが変わっていますので、その部分はユーザインタフェースを変えずに、メカニズムとして変えるようにしています。エージェントレベルではそれぞれがIDを取り、そのエージェント1個1個に対してアクセスパーミッションを設定するツールを備えています。

Q木下　標準化についてはいかがでしょうか。

A木野　アクセスのリストのところはJava2に揃えようと思っています。それからプロトコルのところも、SSLを使いたい方のためにそれに合わせるようにしておきたいと思います。できるだけ独自色はなくしていかないと、お互いにコミュニケーションできなくなりますから。

エージェント同士が直接コミュニケーションできなくてもJavaのPlatformがあれば、少なくともJavaのマシン間で動くので、1回ディスクに書き込んで相手のエージェントが読み出すとか、別のAPIを使うことができます。標準化できれば、エージェント同士が直

接行き着くようなメカニズムを用意するのはわれわれの責任かなと思っています。
Q木下　大須賀さんの方はいかがでしょうか。Platformとして何か標準化の方へのお考えは。
A大須賀　仕組みとかACLといったところで標準化はかなり意識していますが、Securityに関しては、エージェント側での標準化はあまり議論が進んでいないかなという認識でいます。
Q長尾（日本アイ・ビー・エム）　私個人はモバイルエージェントが非常に好きでおもしろいと思っているのですが、皆さんのお話を聞いていると、モビリティはいらないのではないかという気持ちになってきました（笑）。マルチエージェントにできる、エージェント同士が共同作業で何かやってくれる、従って個々のエージェントが単にどこかのサイトに行って何かやってくれる、というだけではあまりメリットがないのではないでしょうか。マルチエージェントにするとマルチエージェントなりの複雑な話がたくさん出てくるわけですが、それはモバイルエージェントでやると結構辛いわけです。かなりの能力を持っていなければいけない。そういうものが移動するというのはオーバーヘッドがかかりますから、動かないほうがよくて、その間のプロトコルで何とかしようという話に当然なりそうな気がします。

日本アイ・ビー・エムもモビリティをやめてよりマッシブなエージェントをやろうという話になっているわけですが、それは結構うまくいっていて、そのお陰で10万とか100万のオーダーのエージェントがちゃんと動く。信頼性が割と上がっている。モビリティをなくしたからオーバーヘッドがなくなって、そういうところを強化できたという事例が出てきているわけです。

レガシーのシステムをエージェントにしてしまうという話がありましたが、そこまでできるのだったらもう動かなくていいのはないでしょうか。皆さんの直観、要するに未来を見通して本当にモビリティは重要なのかということをお聞きしたいと思います。
A大須賀　実はわれわれもそういう認識を割と持っていたんです。最初は、モバイルエージェントは看板になっておもしろいかなという感じで始めたのですが、最近の感覚だと、モビリティはないよりはあった方がいいかなと。前面に出るのではなく、エージェント機能の中の一つの特徴として必要かなということぐらいにトーンダウンはしていますが、モビリティの必要性は感じています。

ネットワーク性能が非常に細いときには、一旦飛んで、相手側でローカルに処理した方が、トータルのコストでは安いというケースが実際にあります。

また、新しい機能をどんどん提供していきたいとかバージョンアップのときには、モバイル機能があると、それを使ってソフトウェアの最新の機能を提供するような話がしやすいわけで、そこでモバイルエージェント機能はまだ使っています。
A木野　長尾さんのご指摘の通り、今のわれわれの使い方は必ずしも全体的にフィットはしていません。先ほどの講演の中でお話ししましたように、ネゴシエーションしないとエージェントらしくならないと思っています。

IBMさんとターゲットがちょっと違っているのは、「いくつかのエージェント」という場合の数が、大型マシンを持っていらっしゃるIBMさんと持っていない三菱電機の違いというだけで（笑）、あとはほとんど同じで、「動かねばならない」という要求もなくなってき

ています。

しかし、アメリカでは会社が一緒になったり離れたりすることが頻繁に起こって、同じ会社の中なのにまったく違うシステムが組まれている例があります。そういうところでは、送り込んで何が変わったのかを見て帰ってきて、お客様に問い合わせて、次のステップに進んでいくというように、実際に動く意味は出てきています。

大須賀さんが仰った通り、動かないエージェント、動くエージェントそれぞれを配置する方法の一つとして、動く機能も持っているというような使い方になっていくと思っています。

Q木下　服部さん、モバイルエージェントのECなんかではどうでしょう。

A服部　私は昔からモバイルエージェントをやっていたのですが、今のアプリケーションを見る限りは、どう見てもモバイルの必要性は出てこないという感覚です。たとえば、プライバシーの関係でどうしてもそこで動かなければいけない、ということがあるのかもしれませんが、よくよく考えてみるとやっぱり動く必要はないという場合が大半ではないかという気がします。

Q木下　和田さんはいかがでしょうか。

A和田　4人とも長尾さんの意見に従っちゃうとおもしろくないのかもしれないですが（笑）、われわれのところも移動エージェントということで話はやっているのですが、物と一緒に情報も流れていくという意味で言うとすごくわかりやすい。ですが、あるところにいて、ただコマンドだけ送ってやるという方法ででもやろうと思えばできると思います。

こういう話を上の人にすると、必ずなぜ移動しなければいけないのかと聞かれるのですが、そこらへんは苦しいなりにいろいろ答えを用意していくわけです。「物に1個エージェントを付けたまま、製品エージェントは動かないという形にすると、ある程度製品の個数が多くなってくると負荷が増え、そのマシンが落ちてしまうこともある。移動できると、今やっている製造は続けられて、ある程度回復した後に元へ戻せばいい」というような言い方をします。

そういう意味で言うと、移動することは必須ではないのですが、危険度を分散する意味で、ある程度必要かなという考え方でいます。

Q和田　逆に長尾さんにお聞きしたいのですが、拡張現実感をやっている長尾さんから出る質問としてはちょっと変だと思うんです（笑）。違うところに行って、そこでコミュニケーションを開始して何かをする、そういう要素技術としてモバイルエージェントは役に立つと思いませんか。

A長尾　話がそれてしまってすみません。私は必要だと思います（笑）。わざと聞いてみたんですけど、今日あったような話だったらモビリティは本質ではなくなるような気がして、その点を言いたかっただけなんです。私は個人の情報をやっぱり自分で持ちたいですから、自分のエージェントはいつもいてほしいわけです。環境に対して何か働きかけをしなければいけないときには、どうしても自分の情報を流さなければいけないわけですが、そのときに情報だけ、つまりデータだけを流すと誰がそれを見るのかわからないから、データの監視役も一緒に投げるというのは割と自然な発想です。

だけどそれも、私の情報を変に開示しないという非常にトラスタブルなサーバが必ず間に立ってくれるのだったら、動かなくてもいいわけです。そのサーバにやってもらえばい

いわけですから。

パーソナライゼーションなんていうのはエージェントに合っていそうな気がするけれども、それもサーバサイドでできちゃうような話が出てきそうで、それでさえもできないようなことを考えるべきなのかなと、最近私は思っています。私は、モビリティには賛成なんです（笑）。

○木下　今日のお話の中で、現状の問題と今後の課題が出たと思います。今のモビリティに関する話も、実はどういう応用で、どんなエージェントが必要で、どんな使い方をされるのだろうか、誰が使うのか、どこで使うのか、全部そういうところに絡んでくる話で、これから考えていかなければいけない大事な問題の一つだろうと思います。

ここでセッション１は終了にしたいと思います。

セッション２：インタフェースエージェント

（１）パーソナルなモバイルアシスタントのインタフェースエージェント

間瀬健二（エイ・ティ・アール知能映像通信研究所）

いつも気軽に持ち歩くことができ、その場に応じた情報提供をしてくれたり、人と人とのコミュニケーションを支援・促進してくれるモバイル・アシスタント・エージェントの研究紹介が行われた。その事例としては実空間に埋め込まれた知識を活用することで利用者の出会いや仲介を行うC-MAPがある。この中でインタフェースの擬人化、パーソナライゼーション、モビリティ、ユーザセンシングなどの研究が行われた。また具体的な実装例についても紹介された。

（２）擬人化エージェントと個人化エージェント　長尾　確（日本アイ・ビー・エム）

新しいインタフェースとして人間とスムーズに対話できるようなインタフェースエージェントの研究についての紹介があった。さらに対話に加えて、擬人エージェントによるマルチモダリティ機能に関する考察が行われた。インタフェースエージェントの今後としては、テクノロジの進歩により高度なインタフェースが普及する、音声機能が補助的に利用される、エンターテイメントとして生き残る、などのシナリオが示された。

（３）ソーシャルインテリジェンスと心のモデル　中嶋　宏（オムロン）

人と機械の自然で豊かなコミュニケーションを実現する「心をもった機械」の開発を目標とした研究事例が紹介された。これには心理学・社会学の知見をヒューマンインタフェースに応用するソーシャルインテリジェンスとAIのアプローチが取られている。ソーシャルインテリジェンスのアプローチとして重要な概念はMedia Equationと「心のモデル」である。この応用事例として、仮想世界キャラクタ「クリーチャワールド」、ペットロボット「たま」、擬人化エージェント“Stellar”が紹介された。

＜質疑応答＞

Qナカニシ（日立製作所）　インタフェースエージェントの話が出るときに、エモーショ

ナルマシンとかアフェクティブコンピューティングみたいな言葉が出てきます。擬人化や個人化と言っているのは、あくまで自分の経験上の類推を基にしていると思うのですが、そのへんについてどういった考えがあるのか、お聞かせ願えればと思います。

A間瀬　難しい質問ですが、感情について言えば、エージェントがそれを持つことでどれぐらい嬉しいかというのはよくわかりません。

たとえば博物館に行ったときにずっと一緒に回ってくれるようなモバイルアシスタントを例にしますと、それは私のことをむしろビジネスライクにわかった上で説明してくれればよくて、感情をあらわに出してもらう必要はありません。ですから、いまは感情のところにそんなに踏み込む必要はないと思っています。

アフェクティブコンピューティングに関連する話では、個人の生理的な情報を感知できて、状況の理解のために使うことには非常に興味があって、どんなものが使えるのかということをいまウォッチしているという感じです。

A長尾　感情という話をするときにも擬人化と個人化の話という二つの視点があって、アフェクティブコンピューティングは感情を個人化に使おうという話だと考えられます。つまりユーザの感情をコンピュータがいかに知るか、それによって文脈を変えるべきだろうということで、それは結構当たっていると思います。昔から、人間の顔の表情を認識したり声の抑揚を認識したりすることで心理的状態を知ろうという話があります。最近は血圧や脈拍や発汗のセンサを体に付けて測ろうというのもあります。でも方向として、そういうことを個人化の要素として取り入れていこうという話はそんなに間違っていないと思います。

もう一つ、擬人化としての感情は、人間が対話する相手に感情を持たせようという話です。よく言われているのは「人間が感情を持って話すのだから、対話する相手も何らかの感情を示してくれた方がバランスが取れるではないか」ということです。それはよくはわかりませんが、一つ言えるのは、リアクションをモデル化することです。自発性という話がありましたが、マシンの方から何か語りかけてくるというところは、もしかしたら感情的なモデルがある方がいいのかもしれません。自分を駆動するようなメカニズムは情動等のメカニズムに非常に近いと言われていますから、リアクションや自発性のモデル化に効いてくるかもしれません。

アージ理論などがありますが、ルールで感情をモデル化するというのが本当にいいのか、それがうまくはまるのか、というところはよくわからないです。でも何らかの要素でリアクションをモデル化するときに、これは感情として説明した方がいいというようなやり方はあるかもしれません。とにかく、そういう二つの視点で感情をとらえるというのはありだと思います。ただそれが本質かという議論になると、今の段階では疑問です。

A中嶋　私たちはまともに感情にやっているわけですが、その理由の一つには自律的な行動生成に使いたいということがありました。これは私の個人的な推測ですが、特にミンスキーは今までのAI的な手法に限界を感じています。でも、正しい答えがいま見つかっているわけではないと思います。

それから、これを使って何が嬉しいかという観点ですが、目的によってぜんぜん違ってくると思います。ペットロボットに入れたら非常に面白い動きをしますけれども、電話応答装置にそれを入れて嬉しいことになるかというと、それはまだわからないというところ

だと思います。

Q木野（三菱電機）　これは分類の問題だと思いますが、話し手と向かい合ったものがインタフェースエージェントなんでしょうか。

たとえば自分の代わりにどこかへ行ってそれを見てきてくれる、聞いてきてくれる。それもたぶん、インタフェースという意味では自分の代わりになって何かをしてくれているエージェントのように思うんです。

インタフェースエージェントというのを皆さんがどういう定義で思われているのか、お伺いしたいと思います。

A長尾　インタフェースと言うからには人間と接点がある部分に存在する。そうすると、やっぱり人間と接する、だから対話するというふうに捉えるのが一番スムーズです。もちろん、人間の代わりに何かをするというのは一般的にソフトウェアエージェントと言われているものですから、区別して考えることもできますし、インタフェースエージェントがソフトエージェント的にどこかへ行って何か取ってくるというのもありですから、両方ありじゃないでしょうか？

私の話の中では、対話的なインタフェース、擬人的なインタフェースという話を中心にしましたけれども、もちろんそれにとらわれる必要はありません。

A間瀬　私自身はインタフェースのところからスタートしていますので、インタフェースをもう少し自律性・適用性のあるものと考えたら、エージェント的な考え方をそこに持ち込むのは面白いだろうと思います。

逆にエージェントがインタフェースをしたらどうなるでしょうか。私は、インタフェースエージェントと言うときは二つの側面をいつも考えていて、インタフェースそのものがユーザに合わせて変わるようなものがこれから考えられるだろうと思っているわけです。

それがモビリティを持ってあちこち回るのだったらそれでもいいですが、最後に報告してくれるのは人間のところであって、何をやってきたか、どこを見てどういう情報を仕入れてきたかというのをユーザにわかりやすく見せるためには、エージェントが情報変換をして、可視化してくれないといけません。その意味では、インタフェースとしてのタスクがあって、それをしてくれるのではないかということです。

Q金（ＮＨＫ技研）　間瀬さんに一つ質問したいと思います。C-MAP プロジェクトの中でモバイルエージェントを使って知識のモビリティを作るということですが、基本的にエージェントの場合、知識ベースとして考えることができて、時間によってその知識もだんだん大きく重くなっていくと思います。それを動かすことは結構エネルギーが必要でしょうが、そういうところに対してはどういうストラテジーがあるでしょうか。

A間瀬　ちょっと言葉の使い方がまずかったかもしれないですけど、エージェントが知識を持ち運ぶというよりは、偏在している知識にエージェントがどんどん線を張っていってネットワークを活性化していって、必要なときに必要な線が発火されてというような構造ができたら面白いということを狙っています。まだ具体的にできていないのが正直なところです。

Q西田　インタフェースエージェントというのは、いま受け入れられていると言えるのでしょうか。われわれ技術者や研究者としては、社会やユーザに、少なくともこれからエージェントシステムを作っていくようなエンジニアには受け入れられたいわけですが、今ど

のあたりまで受け入れられていると言えるのでしょうか。もし受け入れられていないとしたら、どこで、どんなふうにしたら受け入れられるのでしょう。

A間瀬　学生の世代は完全に受け入れてもらえると思っています。ただわれわれ、あるいはわれわれの上司（笑）、また社会に受け入れられるかどうかというと疑問がありますが、そういう時代に変わっていくだろうと思っています。

A長尾　受け入れられるというのはいろいろ意味があると思いますけれども、たとえば今のインタフェースに取って代わる、あるいは今のインタフェースに拮抗するというレベルで受け入れられるかと考えると、今の時点はそうではありません。あくまで補助的、あるいはぜんぜん違うドメインや方向性で、要するにインタフェースとして認めているかどうかわからないけれども、とりあえず使っているというのが現時点だと思います。でも、やっぱり受け入れられたいですよね。

社会的な側面は、技術者にはどうしようもないじゃないですか。ある程度精度を超えれば閾値を超えてくれて、そうしたら黙っていても受け入れられる。音声認識は今そういうフェーズに入ってきているような気がします。だから対話システム、対話型インタフェース、あるいはエージェントインタフェースも、もう少し時間を見ればありかなと、私はそう期待しています。

エージェントが受け入れられるときにはもう一つ、コンテンツというキーポイントがあるんです。よく対比するのですが、Web の検索エンジンはエージェントの形で説明されることがあります。あれはぜんぜんインテリジェンスというレベルに達していませんが、十分にコンテンツがあって、かつロバストだったので、みんなが使い出して、非常に受け入れられたわけです。

鍵は、コンテンツがエージェントにとってより適した形になってくれることです。コンテンツ自体のクオリティ、あるいは内部構造の整合性がもう少しアップすると、ちょっとしたインテリジェンスがものすごく役に立つことが絶対あると思います。だからキーポイントはコンテンツなんです。

A中嶋　毎日その人と話をするということになれば、これは受け入れられたことになると思うんです。たとえばペットロボットを毎日抱っこして撫でる、あるいは電話を介したパーソナルエージェントが毎朝自分のメールボックスをチェックするということになれば受け入れられたことになると思いますが、そういう兆候がいまあるのではないかと思っています。

セッション３：プラットフォームと標準化

（１）FIPAのプラットフォームと標準化　須栗裕樹（コムテック）

エージェントプラットフォーム国際標準化活動である FIPA の理念、組織、仕様についての解説と問題提起がされた。近年ではFIPA 準拠のプラットフォームが開発されてきており、Nortel のFIPA-OS、ブリティッシュテレコムの ZEUS、フランスの AEGIS、Agentscape の Simon、GMD 系の Grasshopper2、Comtec Agent Platform などがある。しかしながら現在のところ相互接続はできない状況であり、FIPA の仕様が未成熟であることを表している。

問題提起としては、普及方策、他の標準化との関係、国際化と日本語化の問題が挙げられた。

（２）自律分散サービスアーキテクチャの提案　豊内順一（日立製作所）

企業情報システムにおける個別システム群の柔軟な相互接続を目指した自律分散サービスシステムについての紹介がなされた。このシステムの特徴は、プログラムを出力をネットワークにブロードキャストするというデータフィールドの考え方である。またシステムの成長性の確保、大規模システムを実現するためにメディエータ機能が採用されている。メディエータには、コミュニティ内での情報の流れを制御するCommunity Management、他コミュニティとの連携をおこなう Federation、サービス要求と提供の結び付けを行うBrokerage の機能がある。実装には CORBA が利用され、ネットワーク設備サービス統合管理システムと高度交通情報サービスシステムへの応用が紹介された。最後に OMG における標準化活動についても言及があった。

（３）日本アイビーエム（株）におけるエージェントに対する取り組み

中村祐一（日本アイビーエム）

日本アイビーエム東京基礎研で開発された移動エージェントシステム Aglets、Aglets上のエージェントマーケットプレース、スケーラビリティとパフォーマンスに特化したエージェントシステム Caribbean の紹介があった。Aglets はマインドシェアを獲得することはできたが、マーケットシェアは獲得できなかった。そこで e-Marketplace というフレームワークを作り、「たび Can」というサイトを立ち上げた。最後に移動機能はなくし、スケーラビリティを高めた Caribbean を開発した。これは 10 万個のエージェントの動作が可能で、リカバリ機能を重視している。

（４）マルチエージェント協調プロトコル記述 AgenTalk と
ソーシャルウェア構築用プラットフォーム Shine

桑原和宏（ＮＴＴコミュニケーション科学基礎研究所）

NTT と京都大学で開発されたマルチエージェント協調プロトコル記述 AgenTalk とネットワーク上での人々のコミュニケーションをサポートするソーシャルウェアプラットフォーム Shine の紹介が行われた。AgenTalk では、状態遷移に基づいたプロトコル記述、継承を用いたプロトコルの段階的定義、エージェント独自のプロトコルのカスタマイズが可能である。ソーシャルウェアアプリケーションとしてはコミュニティの形成支援を行うCommunity Organizer、コミュニティにおける議論の可視化を行う Community Board などがあり、Shine はアプリケーションの連携、アプリケーション構築用部品の共有・再利用を支援する。

（５）日本ユニシスにおけるエージェントの取り組み　保科　剛（日本ユニシス）

日本ユニシスにおける二つの開発事例として System ν と次世代電子図書館プロジェクトが紹介された。System ν は CORBA を用いて開発されておりエージェント機能が組み込まれている。これにはユーザインタフェースとなるユーザ・エージェント、オブジェクト管

理、データ／処理要求の流れを制御するトランスファ・エージェント、業務処理を実行するアプリケーション・エージェントがある。これらのエージェントによりユーザとアプリケーションが柔軟にまた効率よく接続される。次世代電子図書館プロジェクトでも CORBA をベースとしたエージェント基盤が開発されている。

＜質疑応答＞

Q大須賀（東芝）　中村さんに伺いたいと思います。一時代を築いた Aglets が Caribbean に変わったわけですが、ビジネス的に見直されたというのは非常に興味があります。最終的に、Caribbean においてのエージェントは何者になったのか、どういう意味でのエージェントになったのか、Aglets のエージェントから何が残って何が足されたのかを教えていただけますか。

A中村　引かれたことで大きいのは、モビリティです。Security も、後で再び足すかもしれませんが今のところは取っています。というのは Security は、会社間でエージェントが移動する、あるいは知らない人が来るというときにはいるわけですが、今の使い方では、１個のシステムを作るのにエージェントの基盤を使っているわけです。たとえば、何かのサービスをやるためにお客様、エージェントをたくさんホスティングする場合、プログラム自体は中の人が作ってそこにホストするだけなので、Security は取っています。

Aglets はメッセージングとして少し複雑なものを持っていたのですが、すごくオーバーヘッドが多くてパフォーマンスが上がらなかったので、それを上げるためにシンプルに直接送るところだけにしています。だいたいそういう感じのことが足されています。

Q大須賀　そうするとエージェントとして残ったといえるのでしょうか。

A中村　もともと Aglets と言っていたエージェントは、移動するというのもあったのですが、各エージェントがスレッドを持っていて、互いにメッセージングで会話できるというところがエージェントっぽいところかなと思っていました。そのあたりは残っています。つまり、ホスティングされているエージェント同士が非同期のメッセージで通信するというところは残っています。

Q大須賀　皆さんが考えているエージェントの本質は何か、どう思われているか、それぞれオブジェクトと呼ばずにエージェントと呼んでいる所以は何か、教えていただければと思います。

A須栗　オブジェクトはオブジェクトだけど、オブジェクトプラスアルファでインテリジェンスを持っていたり自律性がある。メッセージを非同期でやり取りしたり、自分で心理状態を持っていてやり取りする。ぎちぎちにインタフェースを書くのではなくて、単なるバイトストリームでやり取りをすることによって、そのバイトストリームに載せられたメッセージが自分の心理状態を表すセマンティックを持ち運ぶ。先方でどう解釈されるかというのが自分にはコントロールできない。そういう状態の中でも分散協調処理ができるシステム。非常に大雑把に言って、そんなところだと思います。

A豊内　われわれの ADSS 基盤は、もともとあまりエージェント基盤だと思っていないんです。エージェント基盤は後ろに「各社の取り組み」というのがあって、私どもも、その名も「エージェント基盤」という、先ほど電子図書館のところでご説明のあったものをまた別途持っています。ですので、ADSS がエージェントだと思ってやってきたわけではなくて、

改めて見ると世の中でエージェントと言っているものに近いかなという感じです。

ADSS での要件としては、先ほども申しましたように、異種システムの統合のための方便だと思っています。別の言い方だとラッパーとアダプタでいいわけで、われわれの今の立場の要件は、同期・非同期のクッションになってくれたり、フォーマットとプロトコルの変換をやってくれることです。

あとは Security の問題もありますけれども、現実には CORBA Security を分散環境でいかにうまく定義するかです。CORBA Security を使いこなしている人はたぶんほとんどいないと思いますけれども、逆にわれわれはツールで解こうとしているわけです。こちらの Security はこういうポリシーで行く、こっちの会社はこういうポリシーで行く、あと間は何とかしましょう、というのはツールで解けるのではないでしょうか。だからそこは、ADSS プラットフォームの方ではがんばっていないわけです。

あとは状態やフローの管理をしなければいけないわけですが、上にフローのエンジンがあっても従来のラッパーやアダプタでは耐えきれないところがありますので、そこはアプリケーションとミドルを繋ぐエージェントのレイヤのがんばるところかなと思っています。

A中村　KQML や ACL などのコミュニティで言われているように、知的なエンティティ同士が何かのカンバセーションをしましょう、そのエンティティをエージェントと言いましょう、ということだと思います。ただそこへ行くために三階層ぐらいで考えた方がいいのかなと思っています。

一つはエージェントプラットフォームで、それは私たちの作っているものもそうですし、カンバセーションはまだ持っていないし知的でもないけれども、スレッドを持ったオブジェクト同士が非同期でメッセージングできるようなものもそうです。

二番目がカンバセーションで、一つ一つのメッセージをどう繋げばカンバセーションになるのかというのが定義されていて、そこでお話ししますというレベルです。

三番目は、カンバセーションが与えられたときに知的に振る舞おうということで、エージェントの中に知性を実装するというレベルです。

そういう三階層モデルでアプローチしていけばいいのかなと思っています。

A桑原　私の個人的な意見で定義になっていないのですが、エージェント通信言語を解釈できるソフトウェアがエージェントだと割り切って考えています。そうするとエージェント通信言語は何かというところになるのですが、それは先ほど中村さんの言われた二番目のレベルというのでしょうか、多少ハイレベルなメッセージのやり取りができるソフトウェアをもエージェントと考えています。

A保科　私も昔、人工知能を専門領域にしていたこともあって、そういう意味ではエージェントに対する知的という意味での期待はものすごく高いわけです。ただ、それを言っているとなかなか広まらないというところも感じていて、それは自分の専門領域としてひとまず置いてしまったというところがあります。分散環境でのシステムの自律性と協調性は自分のテーマだと思っていて、大規模に広がったときにちゃんと動くようにする状況を作れればいい、それを外から見て少し知的に見えればかなりのことがやれたことになるのではないかと思っています。

エージェントを使うユーザから見れば、ある API を使ったらいいじゃないか、この API を使ったらエージェントシステムだということでいいじゃないか、というぐらいに割り切

っちゃっていいと思います。そうやらないと前に進めない気がしています。

Qミウラ（松下電器）　エージェントの標準化はFIPAの他にMASIFがありますけれども、二つ標準組織があるとあまりよろしくないと思います。FIPAとMASIFの歩み寄りはどの程度行われているのでしょうか、またそれは簡単なことなのでしょうか。

Grasshopperはもともと MASIFに対応するものとして作られたのですけど、今回の話ではFIPAにも対応しているということですが。

A須栗　一応整合性が取れるようにはなっています。ただし、現実問題としてMASIF自体非常に使いにくいということと、FIPAのモビリティのスペックがあまり使いやすくないということで、現実にどれだけの効力を持ってその二つがうまく噛み合っているかというのは別問題です。

MASIFを作った人たち、IBMとGMD FOKUSと、あとどこでしたか、とにかくIBMとGMD FOKUSがFIPAのモビリティのスペックを作るときにも非常に活躍されまして、その意味でMASIFとFIPAのマッピングはちゃんと取れるようになっています。

具体的に言うと、MAF Agent Systemに対応するのがAMS、MAF Finderに対応するのがとディレクトリファシリテータ、ORBに対応するのがACC（エージェントコミッションチャネル）でしょうと辻褄が合うようにはなっているわけです。

Grasshopperですが、Grasshopper2になって、MASIF絡みの仕組み自体は変えずにFIPAのACLを喋れるようになったということです。今のところ、MASIFに忠実でFIPAのACLを喋るというのはおそらくこれしかないと思います。

東芝さんもやっていらっしゃると思いますが、どんな具合でしょうか。

A大須賀　MASIFとFIPAはレイヤは違うと思っています。MASIFの方がレイヤ的に下という意識を持っていまして、FIPAの方がACLとか上の方に力を入れていますので、不整合は出ていません。ですので、両方に準拠するという形で実装することは十分できています。午前中に紹介したものの中では、MASIFとFIPAと両方意識したようなシステムも実際に作ったりしています。

A保科　インプリメントを生業にしているベンダとしては、標準がいっぱい出てくるというのはとても辛いのですが、一つのエリアに複数出てくるのは自然ですし、それがプラスに働くことも多いと思います。たとえばCORBAは、分散オブジェクトにJavaやDCOMがいて闘うからいいものになると思いますし、それが市場に対するベンダのコミットとも取れると思います。

エージェント技術も確立していないからこそ複数あるわけです。それはみんながやるというコミットをしていることだと思うので、私はプラスに受け止めたいです。インターオペラビリティが重要で、それがあればユーザは耐えられると思います。

Qナカニシ（日立製作所）　先ほどどなたかが、ユーザ数でネット上のデファクトスタンダードを取ることを仰っていたと思いますけれども、プラットフォームの標準化を考える上でも二つの捉え方があると思います。物理的に成り立っている人間関係やコミュニティがあると思っているわけです。Securityの話一つとっても、行き着くところはシステムだ、プログラムだという話ではなくて、ユーザそのもののポリシーとかマナーといったところの方が強いと思っています。

そういったことも含めて、エージェントを扱うコミュニティのあり方、あるいは皆さん

が仰るエージェントはどういったコミュニティに対して適用できるのかというところをお聞きできればと思います。

A豊内　先ほど一つ、ADSSのエージェントの要件で、ポリシーというのを言い忘れていました。あるメッセージがそのコミュニティの中でマルチキャストされたものを、そもそも受け取るかどうか。Securityもポリシーの一つだと考えていますので、そのポリシーベースで、単なるラッパーとアダプタだけではない、コミュニティベースでのメッセージの送信・受信、あるいはリクエストの拒否が大前提だと思っています。

コミュニティの話ですが、FAとか制御の世界ではSecurityがほとんどないわけです。誰かがある工場のネットワークに入ってハッキングしたり情報を盗み見るなんて、誰も思っていないわけです。そんなことよりも、全部欲しい人に届く方が大事なわけで、そういう意味ではネットワークのセグメントがコミュニティになっているわけです。

これが何を意味しているかというと、一つはSecurityです。Securityの一番簡単な方便がコミュニティです。もう一つは、オントロジーとかセマンティックスというのもコミュニティを解く一番安易な方法だと思います。

たとえば「ATM」と言ったときに、交換機のことを思い浮かべる人もいれば銀行に並んでいるやつを思い浮かべる人もいるわけで、それは銀行のだよといちいち言うのはばかばかしい。XMLで書けばいいのかもしれないですけど、「ATM」と言ったら銀行のもの、と思いつく人をグループ分けしておけば、わざわざそれをやる必要はなくて、そういう類似の興味、同じ言葉が通じる仲間をまずコミュニティにしてしまうというのがやりやすい。

じゃあコミュニティ間の連携はどうするの？というと、またこれは問題を先送りした話になるのですが、今われわれがやっているのはB-to-Bのシステムが多いので、そういう解き方です。昨日までは敵だった銀行同士が合併するのでシステムを繋ぎますというときには、Securityに関しては今までやらなかったような接続でいいだろうということもあって、一個のコミュニティと考えて繋いでしまうということもあります。そういうことで、繰り返しますがSecurityと言葉の定義が一つの切り口だと思います。

A須栗　直接お答えになるとは思わないのですが、逆に言うと、コミュニティが変わると振る舞いが変わるソフトウェアエンティティをエージェントと言っていいのではないかと思います。つまり、親しい人にはそれなりの、親しくない人にはやっぱりそれなりのと。

そのとき重要なのは、別のインタフェースを持っているのではなくて、同じインタフェースで、状況によって振る舞いが変わるということです。

じゃあそれができて何が嬉しいかというと、そこが先ほどの私の問題提起になるのですが、よくわからない。そういうことができそうだということはわかっている、じゃあそれを現実のアプリケーションとして何に使うか。そこがいま自分の抱えている一番大きな問題です。

A中村　会社の中でディスカッションしている話としては、ECの世界、特にB-to-Bのことです。普通、サプライヤとバイヤがいるわけですが、どうやったらいいバイヤコミュニティが作れるか、いいサプライヤコミュニティが作れるかという観点で考えています。

そういうふうに考えていくと、それは単にエージェントの技術とか、その技術がそこに使える、使えないという話ではなくて、もっと別の要素が非常にたくさんあります。既にある資源、たとえばMSNの数百万の人を使って、そこでコミュニティを作っちゃえという

やり方もあるわけです。そこらあたりの話を含めてすべてが技術でやれるというと全然そうではなくて、どうやったらそういうコミュニティが作れるかという方が、私としては関心のあるところです。

A桑原　コミュニティという話になると、Securityというかプライバシーとか非常に重要な問題になると思います。そういったものをいかに実現するかというときに、エージェントという考え方を導入することによって、たとえば自分のプライバシーに関するものは自分のパーソナライズなエージェントに全部管理させる。そのエージェントは、この状況やコミュニティに対して働きかけているときにはここまで出していいといったことを実現することができるのではないか。そういう柔軟な形でSecurityというかプライバシーの保護を実現しようとすると、エージェントという考え方が非常に有効になるのではないかと思っています。

A保科　私は、日立さんの言っているセキュアなところが現状では適用領域ではないかと感じています。Securityを要求されるようなエリアはエージェント技術以前に解決しなければいけない問題が、技術的にもありますし、それ以外の法的なものも多くて、そういうところをいきなり新しめの技術で狙ってもなかなか成果が得られないのではないかと思います。

むしろセキュアな世界できちっとできることをやって、そこのレベルができてからそのエリアに出て行く方が手順としてはいいのではないかと感じています。だからまず、本当に情報が公開できるところでどれだけこのエージェントが活躍できるかというところに期待したいと思います。

Qマツウラ（松下電器）　会社などでエージェントを説明しますと、プラグ・アンド・プレイの話も一緒に出てくるわけです。コンフィグレーションがいらないという機能はそういったものに非常に近いので、位置づけを話すときに多少なりと触れられることがあるのですが、皆さん、Jiniとか他の有名になりつつあるものに対してどういうふうに見ておられるか、ご意見をお聞きしたいと思います。

A豊内　今のご質問の意図は、プラグ・アンド・プレイに関して、たとえばJiniやE-speakがこんなふうになっていて、それに比べて自分のところはこうだということをお伺いになっているんでしょうか。

Qマツウラ　位置づけとしてプラグ・アンド・プレイの技術とエージェントの技術がまったく違う次元なのか、あるいはユーザレベルから見たらできることはそんなに変わらないから、いっしょにして考えていいものなのか、見解的なところを知りたいと思ったんです。

A須栗　次元が違う、レイヤが違うというのが公式見解です。Jiniに対しても優れたディレクトリサービスだということはわかっていますので、それをFIPAのディクトリサービスの構成要素、機能部品として使うということは十分考えられると思います。

そういう話が一方にあって、もう片一方、エンドユーザから見てできることは同じじゃないかというのはまさしくその通りです。なぜかというと、まだまだ世の中のエージェントアプリケーションと呼ばれているものが、エージェント技術の持つポテンシャルを全部引き出していないからではないかと思っています。

ですから、そのポテンシャルを十分発揮させることが大事なわけです。そうするとこんなバラ色の世界が開けるんですよというビジョンを持ちたいと思っています。

A豊内　ちょっと切り口の違うコメントですけれども、ミドルウェアでがんばるとアプリが楽です。プラグ・アンド・プレイもできるかもしれませんが、そうするとミドルウェアのプラットフォームの交流が大変なんです。こんなのだったらアプリでガリガリが書いた方がいいと、うちのアプリケーション部隊にも言われたことがあります。だからコンフィグレーション、たとえば今のプラグ・アンド・プレイの話で、OMG のトレーディングサービスだったらトレーダに登録しなければいけないわけです。それを、もし最初に私がこんなサービスを持っていると言って、マルチキャストしてボーンとプラグ・アンド・プレイができるのだったら、それはハッピーなことだと思います。だけどいま須栗さんが仰ったことの別の言い方は、弊社も含めて全体的に技術も製品も見解もまだ出揃っていないという状況ではないかと思います。

A中村　最近のファイルとしてJiniとかE-speakとかあって、エージェントの話をすると、一般の人はパッと見て、あ、似てるじゃないか、何同じことをやっているんだと言われる、という指摘ではないかと思うのですが、そういう気もします。というよりもむしろ、世の中がアーキテクチャ的にちょっと変わろうとしていて、その変わろうとしている世界がわれわれがやっているエージェントの世界にけっこう向いてくる可能性があるのではないかと考えています。

世の中がどう変わろうとしているかというと、今は割とクライアントサーバっぽい話が多くて、たとえばブラウザで触るとき同期的に何かリクエストを投げて答えが返ってくるわけです。ところが次の世界はきっともう少し非同期的になるのではないでしょうか。E-speak みたいな話は結構非同期っぽい感じがするわけですが、要するにあれはパブリックサブスクライブのアーキテクチャなわけです。何かリクエストを投げておくけれども、後で答えが返ってくるかもしれないという。

答えが後で返ってくるなら、ダイヤルアップで繋ぐユーザはリクエストを投げた後に電話を切るわけです。そのときにエージェントを入れて溜めておいたりフィルタリングしてくれたりする世界があるのではないでしょうか。そういう意味で世の中が変わりつつあって、クライアントサーバから、先ほどサービス・オリエンテッド・システムがありましたけれども、要するにいろいろなサービスがあって、それをみんながある場所にポンポンポンと注ぎ込んでおいて、そこでできる人がやってくれるというアーキテクチャが次の時代に来るのではないでしょうか。それはエージェントにとって非常にいいベースになるのではないかと、ポジティブに考えています。

A桑原　私のバックグラウンドはどちらかというとマルチエージェントだったので、エージェント間のハイレベルな協調とか交渉に興味があるわけです。そういうものを考えますと、Jini とかそういったものでは繋がるのが非常に楽になった、やっと話ができる段階になっただけです。話ができることでもっと面白いこと、高度なことができるようにするためには、まだまだいろいろ考えなければいけないことがあるのではないかと思っています。

A保科　先ほどの私の説明で、νのネットワーク・プラグ・アンド・プレイの話をしました。あれは外側から新しいものが入ってきたときに系の中で認識して動く、まさにプラグ・アンド・プレイだと思います。私のイメージとしては、エージェント技術をそういう部分に応用したということで、同じエージェント技術を使うとこういうことも素直にできますということなのではないかと思います。

Jiniが出てきたときは、あ、来たなという感じでした。ああいうものが出てくることで、エージェント技術がこのエリアに使えることを他の人も考え始めて、そういうのに従う製品が出てきました。こういうのはcompetitorと思えるわけです。こういうことを皆も認めはじめているということだと思うので、大歓迎だなと思っています。エージェント技術が使われていてということで、こんなこともできるのだから、あれもできる、これもできると説明されればいいのではないかと思います。

Q北村　最後に一言ずつお聞きしたいと思います。須栗さんの発表の中でリレーションデータベースの話が出ました。いわゆるオーバーヘッドとメリットということです。エージェントのプラットフォームが世の中にそれほど普及していないというのは、まだオーバーヘッドの部分が大きくて、そのメリットを隠しているのではないかと思うわけです。ですから今後それを逆転させていくことによってエージェントが普及していくと思いますが、それぞれの方々の中で、オーバーヘッドの部分が今はどうであって、そしてそれをメリットでどう乗り越えていくのか、そのへんの認識をお聞きできたらと思います。

A須栗　実は私は「たびCan」を愛用してまして（笑）、あるところでそれを使うのをやめたんです。なぜかというと、仙台の地元のHISに電話した方が安くて対応もいいと気がついたんです（笑）。結局そのレベルになってくると、エージェントだ、オブジェクトだ、何とかというテクノロジーの話ではなくて、アプリケーション、あるいは魅力あるコンテンツをどういうふうにエンドユーザに見せるかという話になってくると思います。

エージェント技術を普及させるための方向はたぶん二つあります。インテリジェントなエージェントの特性を行かしたハイソサエティなアプリケーションを格好良く作るという方向が一つ。でもこれは難しいだろうということで、もう一つは、多少無理があるかもしれないけど、既存の企業情報システムにエージェントをぶっこんでしまえという方向があると思います。

A豊内　昨日ちょうど「ADSSはいったい何がいいの？　何が儲かるの？」という議論を社内でやったときに、さんざん叩かれたわけです。私は個人的には、たとえばエージェント技術を使ってこういうシステム連携を実現するメリットは、一つはフレキシビリティであり、プロファイルと言っているサービスの特性の情報をどんどん追加できることだと思っています。今まではこういうパラメータでマッチングしていたけれども、それに新しい項目が付く、あるいはマッチングのルールが変わりますというのが、これからどんどんサービスが増えたり変わったりしていくのに対して重要で、それがエージェントの売りだと思っているわけです。

ただ実際に日々明日の商売、明日の日銭を稼いで開発している人々から言わせると、そんなのはいらない。二つのシステム、たとえば金融系みたいな固いシステムを繋ぐというと、静的で、パフォーマンスとか固さの方が大事で、パラメータなんか決まっているという意見もある。その中でも、それでも柔軟性はあった方がいいというのがあって、われわれは少し先を行っているのかなと。

つまり須栗さんは、今あるシステムに無理やりにでもとにかくエージェントを適用していくべきだというわけです。もちろんそれはその通りで、会社が許してくれたらぜひそうしたいのですが、私個人としては、負け惜しみではないですが次の時代のニーズを追いかけていると思っていまして、フレキシビリティというところでどこまでがんばれるか。パ

フォーマンスとかスケーラビリティも踏まえてがんばっていきたい。所信表明みたいになってしまいましたけれども、ネクストフェーズのミドルをやっているつもりで頑張っています。

A中村　エージェントのオーバーヘッドと利点というのはまだあまりよくわからないのですが、やるべきことは何かというと、世の中の動きが非常に速くてどんどんいろいろなものが出てきているわけだから、エージェントも何とかしてその舞台に出られるはずだと思っているわけです。たとえば先ほど言いましたE-speakはパブサブなわけですけれども、パブサブなんてどう考えてもパフォーマンスは悪いわけで、しかしサービスレベルは逆に上がるわけです。だからその話だけを聞くと、現実に考えたらあまりいいとは思わないのですが、彼らの作っている次のこういう世界が来るんだという世界観を見ていると、あ、これは結構面白いと思えてくるんです。

だからエージェントの話も、エージェントという技術はたぶんそういう世界を描く上での一個のコンポーネント、要素技術だと思うので、そういうものを使ってどういう世界が描けるかをみんなが考えないと、単に技術的な細かいところだけやっていてもなかなか花が開かないのではないかという感じがします。

A桑原　一般的に皆さん言われていることですけれども、エージェントシステムの利点というか、オーバーヘッドを超えた利点というと、一つのシステムを作る上での柔軟性ということかなと思っています。

柔軟性ということでは二つ考えていまして、一つはいろいろなものがくっつけられるということです。たとえばエージェントラッパーを付けて、それが新システム同士を繋げられる。今まで単体で動いていたものが、連携を取ることによって1足す1が2以上になるのではないか。

もう一つはインタフェースエージェントで代表されるような形で、パーソナライズというか、各ユーザの特性にカスタマイズした機能が実現できるのではないか。そういった二つの面での柔軟性を実現するのにはエージェントという考え方が有効で、それはオーバーヘッドを補って余りあるものがあるのではないかと思っています。

A保科　卑近な話になりますけれども、たとえば自律・協調を実現しようとして私がぶつかったオーバーヘッドは、エージェントが実際にCPUを使ってネットワークの上で本来やらなくていい、要するにきちっとコンフィグレーションされていればやらなくていい情報を交換していることです。たとえば新しいものが加わるといったときに、誰に繋がっていいか決まっていればいいのに、決まっていないからそこでネゴシエーションする。あるいは、動いている最中にどこを使えばいいかということをお互い決めるためにいろいろなネゴシエーションをする。それが、CPUであったり、メモリであったり、ディスクであったり、ネットワークであったり、実質的に資源に対する重さだと思います。

実際にああいうシステムを使っていただいて、全部事前に定義するからそういう通信を一切止めてくれというのは、たとえば金融系のミッションクリティカルなんかはそうです。大事な線やCPUがわからない間に使われることは止めてほしい。それがオーバーヘッドとして認められる、認められないというのは、やはり目的に依存していると思います。だからエージェントが発展するのはそういうのがあまり関係ないエリアで、それよりもそれによって発生する、たとえば今回の場合は実行効率と闘っていたのは運用性なわけです。コ

ンフィグレーションをするという面倒くささ、あるいは新しい機械が来たときに実際にそれを設定する人に比べてCPUがちょっと増える、あるいはメモリがちょっと増えるというお金よりもよさそうだから、しかもテーマとしても認められるからということだと思います。

だからこういうところは、観念的にいいというよりも、具体的に何がオーバーヘッドになるのか、それは実際にいくらだということを示していくことが、テーマによっては重要なのではないかと思います。総論でなんとなくよさそうでしょうというのは、なかなか現場に入っていかないです。それが定着させられない一つのポイントなのではないかと思うので、われわれも実際にこれから入れていく方々が判断できるだけの数量的なデータといったものを蓄積して示していくことが必要なのではないかと思います。

資料編 D. 海外調査報告

D.1 米国調査報告

D.1.1 **出張者**： 北村泰彦、大阪市立大学工学部情報工学科、助教授

D.1.2 **出張先**：(1) ACM EC99 (ACM Conference on Electronic Commerce)
期間・開催地：1999年11月3日－5日、コロラド州デンバー
(2) ワシントン大学（ミズリー州セントルイス）
Tuomas W. Sandholm
Assistant Professor, Director of Multiagent Systems Resarch Group

D.1.3 **出張期間**：1999年11月2日～10日

D.1.4 **報告**：

ネットワークエージェントはインターネットを社会基盤とする高度情報化社会の基幹技術として期待されており、インターネット上での様々な応用分野が開拓されてゆくことが期待されている。このような応用分野の中でもインターネット上での商品の売買や取引を行う電子商取引(EC: Electronic Commerce)は生活に密接に関係し、産業界にも直接多くの利益を生み出すことから、もっとも大きな注目を浴びている。またすでに多くの電子商取引のプラットフォームが開発され、実用化が始まっている。その中でも米国は先導的な役割を果たしている。

今後電子商取引がさらに高度化され、発展してゆくためにはネットワークエージェント技術の果たす役割は大きいことが予想される。そのため、米国においてどのようなネットワークエージェントに関連する技術が研究されているのかを調査する必要があった。そこで1999年11月2日から11月10日にかけて行った今回の調査では、短期間で広範囲な情報を収集するために電子商取引に関する国際会議である ACM Conference on Electronic Commerce (EC-99)に出席し、さらに深く要素技術の詳細を知るために、マルチエージェント技術を電子商取引に積極的に導入しようとしている研究者である Wahington Universtiy の Tuomas Sandholm 助教授を訪問した。

D.1.4.1 国際会議 ACM EC99

ACM Conference on Electronic Commerce (ACM EC'99)は ACM(Association of Computing Machinery：米国計算機学会)主催による電子商取引を対象とした第1回目の国際会議である。ACM は IEEE と並び、米国における計算機科学の中心的学会である。近年では実務的な視点から電子商取引に関連する会議が開催されるようになっているが、電子商取引の将来を展望する上では ACM が開催する会議が実務家と研究者がうまくミックスされ、もっとも相応しいように思える。またプログラム委員長は University of Michigan の Michael Wellman であるが、かれはマルチエージェントの分野において市場指向プログラミングで大きな成果を上げており、その他 MIT の Pattie Maes や Wahington Universtiy の Tuomas Sandholm などエージェント分野の代表的な研究者がプログラム委員に名を連ね、エージェントと EC の関連を知る良い機会となることが期待された。

本会議には 72 件の論文投稿があり、査読の結果、米国、英国、スイス、ドイツ、イスラエル、日本からの 21 件の論文が採録され、発表された。論文はプライバシとセキュリティ、オークション、ユーザインタフェース、ミドルウェア、ネットワークプライシングとQoS、価格付けと契約の６つのカテゴリに分類されていた。

会議は1999年11月3日から5日までコロラド州デンバーのDenver Marriott City Centerで開催された。会議は一つのホールで一見約 300 名ぐらいの参加であり、シングルセッションで行われた。

プログラムは以下に示すように基調講演、招待講演、一般講演、パネル討論から構成されていた。

11 月 3 日

9:00 Welcome and Introduction to the program Michael Wellman, ACM EC-99 Program Chair

9:10 Welcome and Introduce keynote speaker Stuart Feldman, ACM EC-99 Conference Chair

9:20 Keynote address: Elliot Maxwell, US Dept of Commerce Emerging Policy Issues for Electronic Commerce

10:45 Paper Session 1: Privacy and Security

14:00 Paper Session 2: User Interfaces

15:45 Panel Discussion 1: Topic Mobility and Electronic Commerce

11 月 4 日

9:00 Paper Session 3: Network Pricing and QoS

10:45 Paper Session 4: Auctions

14:00 Invited Talk: Problems in Online Advertising and E-Commerce

15:45 Paper Session 5: Middleware

17:00 Invited Talk : Looking Behind the Interface: An Overview of Legal Issues Raised by E-Commerce

11 月 5 日

9:00 Paper Session 6: Pricing and Contracting

11:15 Panel Discussion 3: Electronic Commerce: Where are we as a field of research?

基調講演はElliot Maxwell (US Dept of Commerce) による Emerging Policy Issues for Electronic Commerce であり、米国商務省の視点から電子商取引の将来動向についての講演であった。今後の電子商取引の課題としてプライバシ（メタデータ、Anonymizer、Psudonymizer)、認証（パスワード)、有害コンテンツ（フィルタリング、ブロッキング、ラベリング)、知的所有権（電子すかし、暗号、電子マネー)、課税、消費者保護などを扱う技術が重要であることが述べられた。特に、プライバシ保護のための Anonymizer（匿名化器)、Psudonymizer（偽名化器）はエージェントとの関連もあり、面白いテーマであると思った。

招待講演は Dwight Merriman (Founder and Chief Technical Officer, DoubleClick)

による Problems in Online Advertising and E-Commerce であった。DoubleClick は米国におけるインターネット広告を扱うベンチャー企業であり、ホームページ上でのバナーをはじめとするインターネット上での広告の効果を最大限に生かす方策についての講演であった。もう一つの招待講演は Jane Winn (Southern Methodist Univ School of Law) と Carl Oppedahl (Oppedahl & Larson LLP) による Looking Behind the Interface: An Overview of Legal Issues Raised by E-Commerce であったが、これは技術的な内容よりもタイトルにもあるように主に電子商取引に関する法律に関するものであった。

パネル討論は二つ行われた。最初のものは、Mobility and Electronic Commerce でモデレータは Anupam Joshi (Univ Maryland Baltimore County) で、パネリストは Martin Griss (HP Labs)、Anant Jhingran (IBM Research)、Danny Lange (General Magic)、Pattie Maes (MIT) であった。Anupam Joshi は KQML を用いた Dynamic Negotiation Agents と呼ばれるアドホックコラボレーションについての発表であった。Martin Griss は Agent Mediated E-Commerce というタイトルで、Hewlett Packard におけるサービス管理のためのワークフローやエージェント技術に関する発表であった。Anant Jhingran は現在のモバイルネットワークがその端末(パーソナルコンピュータ、PDA (Personal Digital Assitant)、携帯電話) に応じて必要とする通信帯域幅が異なり、InfoPyramid と呼ばれる、帯域幅に応じた情報提供のあり方が必要であることを述べていた。Pattie Maes は IMPULSE と呼ばれるシステムの紹介であった。現在の電子商取引が単にインターネット上で安価な商品を探すだけの検索システムになっていることに対して、IMPULSE はモバイルデバイス、エージェント、e-profile などの技術を用いて、EC と実世界でのショッピングを融合しようとの試みである。ここではオークションよりも交渉の技術が重要になり、オントロジの整備も求められる。またエージェントは売り手と買い手にとってプロキシとして動作する。最後に、Danny Lange が発表を行った。Danny Lange はかつて IBM 東京基礎研でモバイルエージェントプラットフォーム Aglets の開発者であり、またモバイルエージェント技術で有名な General Magic 社へ転職したこともあり、電子商取引におけるモバイルエージェントの役割について議論がされるかと期待していたが、それに関する話は一切なく、音声ユーザインタフェース (Voice User Interface) についての発表であった。

もう一つのパネル討論は Electronic Commerce: Where are we as a field of research? で、モデレータはプログラム委員長の Michael Wellman (Univ Michigan)、パネリストは Stuart Feldman (IBM Research)、Yoav Shoham (Stanford Univ)、Doug Tygar (UC Berkeley) であった。この討論はプログラムの最後で会議全体のとりまとめや、次回の EC2000 に関する報告であった。ちなみに EC2000 は Doug Tygar (UC Berkeley) と Jeffrey K. MacKie-Mason(Univ Michigan)がプログラム委員長になり、ミネアポリスで 10 月 15 日の週に開催される運びとなった。

以下、一般講演発表をまとめておく。

Paper Session 1: Privacy and Security

Enhancing Privacy and Trust in Electronic Communities

Bernardo A. Huberman, Matt Franklin, Tad Hogg, Xerox PARC

推薦システムでは各個人の選好を明示する必要があるが、これはプライバシ保護の観点

から望まれない場合も多い。もう一つの方法は第三者機関を利用する方法であるが、必ずしも十分に信頼できない場合もある。この発表では暗号技術を用いることで個別の選好を明らかにすることなく、コミュニティに共通の選好を見出す手法を提案している。

Privacy in E-Commerce: Examining User Scenarios and Privacy Preferences

Mark S. Ackerman, UC Irvine; Lorrie Faith Cranor, AT&T Labs-Research; Joseph Reagle, MIT

電子商取引において利用者はできるだけ個人情報を開示したくないものである。この発表では381人の米国インターネット利用者を調査し、プライバシに関しての意識調査を行った結果を報告したものである。これは今後のインタフェース設計に役立てようとするものである。

Distributed Digital-Ticket Management for Rights Trading System

Kazuo Matsuyama, Ko Fujimura, NTT Information Sharing Platform Laboratories

これは日本からの唯一の発表で電子チケットを扱ったものである。電子チケットにはプロバイダがアカウントを管理するものと、スマートカードで利用者がアカウントを管理するものがある。後者に関しては複数のアプリケーションでどのようにスマートカードを共用するか、限られたメモリやCPUでどのように性能を向上させるか、非同期取引をどのように扱うかが問題となっていた。本発表ではスマートカードを認証にのみに用いることで、これらの問題に対処する手法を提案している。

Nark: Receiver-Based Multicast Non-Repudiation and Key Management

Bob Briscoe, Ian Fairman, BT Research

この発表は取り消しになった。

Paper Session 2: User Interfaces

Opportunistic Exploration of Large Consumer Product Spaces

Doug Bryan, Anatole Gershman, Anderson Consulting

この発表ではキーワード検索やハイパーテキストに代わる opportunistic exploration という利用者のふるまいに基づくインタフェースの提案が行われた。このインタフェースでは水族館のように画面上に12000個の中から商品が浮き上がっては消えてゆく。表示される商品の関係にはコーパスの類似性、プロファイルの類似性、購買層の類似性が用いられている。

VRCommerce - Electronic Commerce in Virtual Reality

Yosi Mass, Amir Herzberg, IBM Haifa Research Lab

3Dの仮想現実空間を利用した利用者インタフェースの提案である。この発表では通信や処理オーバヘッドの問題、ナビゲーションの問題、独立した空間（店）を連続的空間（モール）に相互接続する問題などを扱っている。

E-Commerce over Communicators: Challenges and Solutions for User Interfaces

Mona Singh and Anuj K. Jain, Ericsson Inc.; Munindar P. Singh, North Carolina Sate University

電子商取引の端末としてコミュニケータと呼ばれるPDAやパームトップコンピュータの特徴を組み合わせたハンドヘルド端末についての発表である。コミュニケータには表示画面が小さいこと、入力の困難さ、低帯域、断続的あるいは信頼性の低い通信などの問題がある。この問題に対処する利用者インタフェース、バックエンドシステムについて議論している。

Paper Session 3: Network Pricing and QoS

How Many QoS Classes are Optimal?

Kai Cieliebak, Beat Liver, IBM Research Zurich

この発表ではサービスプロバイダが提供すべきQoSクラスの数がいくつであるかの問題を扱っている。クラスの数が多ければ、それだけ収入が増加するが、その保証のためのリスクも大きくなる。本発表では二つのQoSクラスが最適であることを理論的に証明している。

Paris Metro Pricing for the Internet

Andrew Odlyzko, AT&T Labs

PMP (Paris Metro Pricing)と呼ばれる単純な手法がパケットネットワークにおけるサービスの差別化のために提案されている。この手法は過去のパリの地下鉄に一等車と二等車に区別されていたことから、パケットにも等級分けをして、それぞれの料金を変えることにある。高い料金の通信路はトラフィックが少なくなり、より良いサービスを提供できる。

Internet Demand under Different Pricing Schemes

Jörn Altmann, Björn Rupp, Pravin Varaiya, UC Berkeley

電子商取引では複数のQoSの必要性が強調されている。しかしながら利用者が多くの選択肢から選択可能になる時、この柔軟性は本当に望まれるものとなるのであろうか？この発表は統計調査により、異なる利用者ベースの価格付けモデルの受容度、動的価格付けのメリットを調査している。

Paper Session 4: Auctions

iBundle: An Efficient Ascending Price Bundle Auction

David C. Parkes, University of Pennsylvania

電子商取引のアプリケーションとして重要な位置を占めるのがオークションである。ここでは複数の商品を組み合わせてオークションを行うiBundleという手法を提案している。これは最適な組み合わせ割り当てを保証する価格上昇オークションである。実験的結果から困難な資源割り当て問題に対して効率よい割り当てを生成することが確かめられた。さらにこのためにエージェントの選好に対して完全な開示を必要としない。

Privacy Preserving Auctions and Mechanism Design

Moni Naor, Benny Pinkas, Reuben Sumner, Weizmann Institute of Science

本発表は参加者の入力のプライバシを保護するオークションプロトコルの提案である。このためにauction issuerと呼ばれる第三者機関が導入され、オークション計算のプログラムを生成する。auction issuerとauctioneerが共謀しなければ、どちらも入札に関するいかなる情報も得ることができないという性質が得られる。プロトコルはbidderとauction issuerとの間の通信も必要がないので、計算効率も優れている。

Truth Revelation in Rapid, Approximately Efficient Combinatorial Auctions

Daniel Lehmann, Liadan Ita O'Callaghan, Yoav Shoham, Stanford University

組み合わせオークションは多量の計算が必要なことが知られている。従来は最適解を求めることを前提として、真理表明の議論がなされていた。しかしながら現実には近似解しか得ることができない。ここでは近似解を求めるGVA(Generalized Vickrey Auction)においては真理表明性が満たせないことを示している。さらに制限された参加者クラスにおいて真理表明性を保証する手法を提案している。

Brokering Strategies in Electronic Commerce Markets

Arie Segev, Carrie Beam, UC Berkeley

インターネット上には情報ブローカと呼べるサービスが出現している。そのようなブローカにおいて、Yahoo!モデルでは広告主が費用を支払い、eBayでは売り手に手数料が課される。これらのモデルには理論的な根拠がなかった。本発表では、売り手がM人、買い手がN人いる場合において、ブローカの最適戦略の性能における検索コストの効果を調べている。また料金との関係についても解析している。

Paper Session 5: Middleware

Middleware for Software Leasing over the Internet

H.A. Jacobsen, O. Günther, Humboldt Universität zu Berlin

MMM (Middleware for Method Management) はWWWを介してアプリケーションサービスの設置、統合、配布、利用を管理するインフラストラクチャである。MMMではライセンスではなくリースの形態をとっている。MMMの実装にはWWW技術(HTML, XML, MetaHTML)、分散オブジェクト技術(CORBA)、データベース技術(ODBC)に基づいている。

A Methodology for Workload Characterization of E-Commerce Sites

Daniel A. Menascé, George Mason University; Virgilio A. F. Almeida, Rodrigo Foneseca, Marco A. Mendes, University Federal de Minas Gerais

電子商取引サイトの性能解析と容量プランニングを扱っている。これには従来の負荷解析の手段であった秒単位のアクセス回数は適切でなく、連続的なアクセスからなるセッションの概念が必要になる。最初に顧客のふるまいを記述するCustomer Behavior Model Graph (CBMG) と呼ばれる状態遷移グラフを導入する。次に、負荷モデルを定義している。

最後に負荷を特徴付けるクラスタリングアルゴリズムを示している。

Recommender Systems in E-Commerce

J. Ben Schafer, Joseph Konstan, John Riedl, Univ Minnesota

リコメンダシステムは電子商取引の世界を変化させる重要なビジネスツールとなっている。ここではいかにリコメンダシステムが売上を伸ばすかを示し、6 つのサイトを解析している。それを元に、インタフェース、推薦技術、顧客入力の観点からリコメンダシステムの分類を行っている。

Paper Session 6: Pricing and Contracting

A Declarative Approach to Business Rules in Contracts: Courteous Logic Programs in XML

Benjamin N. Grosof, Hoi Y. Chan, IBM TJ Watson Research Center; Yannis Labrou, University of Maryland

電子商取引契約におけるビジネスルールの表現法を述べる。次にそのような表現をするための要求項目を解析する。これには宣言的意味論、競合解消、解析の容易さ、WWW への統合、直接実行可能性、計算量などが含まれる。このために Courteous Logic Programs (CLP) と呼ばれる知識表現法、Business Rules Markup Language (BRML) と呼ばれる XML への変換法を提案している。

Contracting with Uncertain Level of Trust

Sviatoslav Braynov, Tuomas Sandholm, Washington University

契約における信用の問題を扱っている。売り手と買い手のゲーム理論による定式化が行われている。買い手の信用のレベルが不確かな場合を扱っている。もし売り手の信用が買い手のものに等しいなら社会厚生は最大化されることを示している。買い手の信用を低く見積もることは両者にとって危険である。信用を改善する前払い契約 F-contracts の役割を議論する。これは誘引両立的であり信用の非対称性による非効率性を除く。

Automated Strategy Searches in an Electronic Goods Market: Learning and Complex Price Schedules

Christopher H. Brooks, Scott Fay, Rajarshi Das, Jeffrey K. MacKie-Mason, Jeffrey O. Kephart, Edmund H. Durfee; Univ Michigan and IBM Research

電子財の市場は情報財の柔軟性、無視できる限界コストのために価格付け手法はより複雑になる。ここでは複雑さが変化する価格スケジュールの動的性能を比較する。現在の利益を最大化する熟考と将来の利益を改善する探求をバランスする戦略を実装した二つの学習機構をもつ独占生産者を提供する。価格スケジュールの複雑さは探求の必要量と利益の両方に影響を与える。一般的に単純な価格スケジュールは頑健で、少ない利益であきらめる。

Strategic Pricebot Dynamics

Strategic Pricebot Dynamics

Amy R. Greenwald, Jeffrey O. Kephart and Gerald J. Tesauro, IBM Institute for Advanced Commerce

Shopbot はインターネット上の複数の売り手に問い合わせを行い、財やサービスの価格や属性に関する情報を集める。Pricebot は利益を最大化するように価格設定するソフトウェアエージェントである。この発表は戦略的な pricebot の動作を理解しようとするものである。価格設定戦略として、ゲーム理論的価格付け、myoptimal 価格付け、derivative following, Q-学習を比較している。

D.1.4.2 大学訪問 Wahington Universtiy: Tuomas Sandholm 助教授

（米国ミズーリ州セントルイス）

Tuomas Sandholm 助教授はマルチエージェントにおける第一線の研究者であり、主要な国際会議に多くの論文を発表している。彼のアプローチはマルチエージェントシステムに経済学の概念を導入するもので、自動契約やオークションの研究を行っている。今回の訪問では自動契約の一つの方法である多段コミットメントの説明とデモを受けた。多段コミットメントは罰金を支払うことにより破棄することが可能な契約手段であり、罰金と契約金額に応じて自動的な契約と破棄することが可能となる。また契約を破棄させないための適切な罰金額についての考察も行われてる。またこれに関連する eExchangeHouse のデモでは、インターネット取引における商品の部分的な配送と決済に関する方法を示していた。すなわち商品が分割される場合に、それぞれの部分にどのような価格付けをすればよいかを考察している。これは例えば、ソフトウェアのモジュール毎の販売、分割可能な電子レポートの販売などに応用できる。最後にこれらの手法の実世界での取引への応用について議論した。

D.1.4.3 まとめ

今回の訪問を通して、ネットワークエージェント技術との関連においては、プライバシとセキュリティにおける Annonymizer（匿名生成器）と Psudonimizer（偽名生成器）、利用者インタフェースにおけるインタフェースエージェントや推薦システム、ネットワークの価格付けにおける経済モデル、ミドルウェアにおけるエージェントプラットフォーム、オークションや契約におけるマルチエージェントシステム技術などが重要であると思わされた。

また今回の訪問では米国におけるインターネットならびに電子商取引の社会への浸透を身をもって体験できた。特に滞在したホテルではインターネットへのアクセスポイントを備えており、低額の使用料で内線を通じて常時アクセスすることが可能であった。また、eBay と呼ばれるインターネットオークションサイトが注目を浴びていたが、その利用法が書かれた手引書が一般書店にも並べられており社会への EC の普及度をうかがい知ることができた。

D.2 欧州調査報告

D.2.1 出張者： 西田豊明、東京大学大学院工学系研究科、教授
木下哲男、東北大学電気通信研究所、助教授

D.2.2 出張先：FIPA London Meeting
Marlborough Hotel, Bloomsbury Street, London WC1B 3QD

D.2.3 出張期間：2000年1月23日～30日

D.2.4 報告：

D.2.4.1 FIPAロンドン会議におけるエージェント標準化の動向

会議は、24日月曜日9時から28日金曜日午後までほぼ休みなく行われた。会議の構造は、まず月曜日の最初に全体セッションを開催し、会議期間中の作業項目（work items）を確認する。その後、5つあるTC（Technical Committee）毎に分かれて個別に作業を行い、最終日の最終セッションで再び全体セッションを行って、作業結果報告・全体討論・決議（resolutions）を行う。ただし、途中でも必要に応じてジョイントセッションや全体セッションを行ったり、関係分野の研究者を招いて講演会を開いたり、システムデモを行う。

今回の、主要な事項を箇条書きにしたい。

(1) 背景

FIPA（Foundation for Intelligent Physical Agents, http://www.fipa.org/）は、1996年にスイスジュネーブで設立された、エージェントの標準仕様策定を目標とした非営利団体である。年4回（1月、4月、7月、10月）のペースで国際会議を開催し、仕様策定の作業を進めてきた。1999年10月の標準仕様公開を目指したFIPA '99の活動は、

- TCA：abstract architecture（エージェントシステムの抽象アーキテクチャ）
- TCB：agent management（エージェントのライフサイクル管理など）
- TCC：agent communication language（エージェント間の共通した通信言語）
- TCD：interoperability（異なるベンダー間のインターオペラビリティ）
- TCE：nomadic application（移動体通信環境下におけるエージェントの活用）

にわかれて進められてきた.

前回会議（1999年10月、FIPA川崎会議）では、1996年からの初代議長（President）のLeonardo Chiariglione氏が退任し、後任にDonald Steiner（Siemens）が選ばれた。今回はSteiner氏にとってはじめてのFIPA会議である。

川崎会議では、FIPA'99としての公開をめざして策定してきた仕様書案と、すでに公開されたFIPA'97、FIPA'98の仕様書の不整合性が問題になった。実装をしながら作り上げたFIPA'97仕様と、抽象アーキテクチャ(abstract architecture)の視点から、より計算機科学の基礎的な手法にのっとって仕様策定しようとする動きがぶつかったためである。

仕様策定をより信頼性・安定度が高いものにするため、仕様書の公開の方法をまず整えることからやり直すことになった。その結果、川崎会議後は、これまでのように仕様策定期間は10月会議に始まり、10月会議に終わる１年を単位とする同期型のものから、実質本位の非同期型のものになった。

現在の枠組みは、３フェーズからなる仕様化を行う。仕様策定提案者は、他のFIPAメンバーの同意を得られれば、workplanと称する仕様策定計画書をTCに提出し、認められれば任意の時期に仕様策定作業を開始できる。該当するTCがなければ、新たにTCを設立する。TC内で承認が得られれば、preliminaryという状態で仕様書案は公開され、意見が求められることになる（有効期間６ヶ月）。さらにTC間で整合がとれ、FIPA会員に承認されるとexperimentalというレベルになる(有効期間２年)。さらに、その仕様案に基づく実装が行われて、仕様案の実現可能性と整合性が確認され、FIPA会員に承認されると、その仕様案はstandardというレベルに置かれ、晴れて標準として認められる。standardレベルに達しなかった仕様案や使命を終えた仕様案は、obsoleteと表示される。

（２）今回の会議の体制と活動

上に述べた３フェースの仕様化を行うため、新たにarchitecture board（AB）が設置された。ABは理事会（Board of Directors）の下に置かれ、TCを統率する。ABの役割はTCから上がってきた仕様書案の技術項目を審査し、その技術レベルと中立性を確保することである。TCからの独立を確立するため、TC chairはABのメンバーとなることはできない。

ABのメンバーとして、

- chair：須栗弘樹氏　(Comtec)
- Fabio Bellifemine氏（CSELT)
- Phil Buckle氏（Nortel)
- Jonathan Dale氏（Fujitsu)
- David Levine氏（IBM)
- Thierry Bouron氏（フランステレコム)

の６名が選任された。

これに伴い、須栗氏がchairを務めていたTCD（interoperability）が解散し、そのミッションは、ABと新しくつくられたInteroperability SIG（SIG：Special Interest Group）に引き継がれた。Interoperability SIGのchairとして須栗氏が就任した。

今回の会議では、主として新しく制定されたworkplanに基づく仕様策定活動のシステムを動き出させることに重点が置かれていた。そのために、従来FIPAで公開してきた文書の整理と再体系化、TCを母体としてworkplan提案の準備が行われた。

（３）イベント

会期中行われた２つのイベントが興味深かった。一つは、システムデモであり、もう一つは、Imperial CollegeのProf. Morris Slomanによるポリシーに関する講演である.

システムデモ

- Using Agents to Support Nomadic Users（Mobile Computing Laboratory, University

of Helsinki)：運転者の状態をセンスし、次に取る行動を予測する。QoS 制御も行う。移動エージェントは使わない。

- Agent Workflow and SAMBA (BT)：サービス統合と（セキュリティを含めた）情報個人化をするシステムを JADE の上に開発。
- IDIoMS:Intelligent Distributed Information Management System (BT & Fujitsu)：富士通・BT：知識マネージメントへの適用。３つのコンポーネント。情報個人化、コンタクトファインダー、関連情報提示
- FACTS Personal Travel Market & ASL (Broadcom)：個人向き旅行支援への応用。
- FIPA-OS (Nortel)：FIPA 仕様準拠の実験的エージェントフレームワーク。
- JADE (CSELT)：Java ベースのエージェント開発環境（Java Agent Development Framework)、FIPA97 準拠。プラットフォーム間通信、FIPA プロトコル、プラットフォーム間でのエージェント移動、コンテンツ記述言語、推論機能（JESS）などを提供。
- ZEUS (BT)：Java ベースのエージェント開発ツールキット。種々の機能が開発者用インタフェースを介して利用可能。

Morris Sloman (Imperial College)：Ponder: A Language for Specifying Management & Security Policies の講演概要

ポリシー（policy）とはシステムが挙動するときにおこなう選択を決定するための規則の集まりである。明示的にグループ化されたオブジェクトの集合において、オブジェクトのグループに対してポリシーを規定する。自律エージェントのコードを書き換えることなく、ポリシーを規定し、変更できるようにする必要がある。ポリシーは永続的であるが、インプリメンテーションを書き換えることなく動的に変更可能である。

オーソライゼーション・ポリシーは、ターゲットに対して何が許されていて何が許されていないかを規定する。例えば、人間に対して危害を加えてはいけないということはポリシーとして実現される。オブリゲーション・ポリシーは、行動の主体がしなければならないことを記述する。また、delegate によって他の主体にどのようなポリシーを委任できるかを規定できる。ポリシーは個体にではなく、管理者といった、組織における役職（ポジション）に対して規定される。もちろん、ポリシーは規則の集合であるので、ポリシーの衝突は起きる。ここで考えられているのは、authorization + vs authorization —、obligation vs authorization —、obligation vs refrain の３つの場合である．これらに対しては否定的なものやより詳細な規則を優先したりする。

（４）今回の会議の成果の概要

残念ながら、今回の会議の期間内（終了直後も含める）において、公表された FIPA 仕様文書はない。従来の文書は、Preliminary または、未公刊と位置付けられてしまった。しかし、各 TC で workplan とそれに対して責任を持つ会社がほぼ決められ、解散になった

TCD を除く全ての TC で、次の 4 月の会議までに preliminary レベルの仕様案を立ちあげ、4月会議前後で experimental まで持ち上げようということが報告された。

また、これに伴い workplan 作成用のガイドラインを公刊するなど、workplan の提案と実行に基づく新しい体制を立ち上げる方向では一歩前進したといえる。

D.2.4.2 抽象アーキテクチャについて

本節では、抽象アーキテクチャに関する議論の概要を報告する。

（1）抽象アーキテクチャの背景と狙い

これまでの FIPA における標準化作業では、

・インフラストラクチャに関する仕様化
　エージェント通信言語、エージェントサービス、オントロジ管理支援など
・エージェント応用に関する仕様化
　個人支援、旅行支援、ネットワーク管理などの応用分野を対象とした仕様

などに関する検討が行われ、その成果は、FIPA97、FIPA98、FIPA99 といった仕様ドキュメントとして公開されている。

その一方で、98～99 年における検討結果から徐々に明らかになってきたことがある。それは次の２点だという。

1）FIPA モデルの中核となるエージェント通信において、メッセージの伝送法（transfer）、メッセージの解釈法（strings/objects/XML）、メッセージに付随する選択的属性（認証、暗号化など）、などに関する多様性に対処することの有用性
2）実用的エージェントシステムを開発する上で既存技術、例えば、分散処理環境や分散プログラミング言語、メッセージ通信基盤（メッセージングプラットフォーム）、セキュリティサービス、ディレクトリサービス、相互接続技術などを咀嚼して活用することの重要性

これらに対する FIPA の選択肢としては、a)様々な特性追加に基づく仕様の漸増的改定、b) 総合的視点からの仕様策定、があった。そして、99 年１月時点で後者のアプローチが採られた。これが抽象アーキテクチャの検討につながっている。その狙いは、幅広く認知・利用されているメカニズム（メッセージ伝送、ディレクトリサービス、商用の開発プラットフォームなど）と同化できるアーキテクチャをつくろうとすることにある。

一方、抽象アーキテクチャの検討では、既存の仕様（FIPA97、FIPA98、FIPA99 の結果）との整合性を考慮することが必要となるが、それだけでは様々な実装形態に対応する上で十分満足できるものは得られないことは明らかである。すなわち、抽象アーキテクチャでは、特定の処理環境のもとで連続的に統合され、他の処理環境で稼動するエージェントシステムと相互運用するシステムが構築できるものを与えることが必要である。そこで、抽象アーキテクチャに関する検討では、次の３つのドキュメント、

・エージェントの構成要素とその相互関係を規定する仕様（抽象アーキテクチャ仕様）
・個別のソフトウェアや通信技術に基づくエージェントシステムの仕様化の指針（実装

ガイドライン）
・エージェント／エージェントシステムの相互運用性と仕様適合性を規定する仕様（相互運用ガイドライン）
を与えることを目指している.

（2）抽象アーキテクチャの必要性

エージェントシステムにおける相互運用性と再利用性を実現するためには、設計されるべきアーキテクチャ要素を明らかにする必要がある。特に、複数のシステムがその機能を実現する上で異なる技術を利用しているとすれば、こうした様々な方式に共通する特性を把握しておくことが必要となる。これがアーキテクチャの抽象化、すなわち、全ての妥当な実装に形式的に関わる抽象的な設計、の同定にむすびついてくる。

基本的な考え方は、種々のエージェントシステムを抽象レベルで記述することにより、それらの基本的構成要素相互間の関係を明らかにすることができるというものである。そして、これら要素間関係を記述することにより、相互運用可能となるエージェントシステムのつくり方が明らかとなる。例えば、二つのシステムがあり、一方のシステムがOMG IIOPプロトコルに基づくACLメッセージを使用し、他方がIBMのMQシリーズ企業メッセージングシステムを使用するとしたとき、送受信者によるメッセージ解釈やメッセージのエンコーディング／伝送方式などの解析を通して、アーキテクチャの抽象化が行われることになる。

一方、アーキテクチャ要素とその相互関係に基づく様々な具体的アーキテクチャを構成することができる。これらは共通的な設計（すなわち、抽象アーキテクチャ）を共有していることにより相互運用が可能となるだろう。もっとも、抽象アーキテクチャから多数の具体的アーキテクチャ（具体的実装）が生成されるものと考えるならば、実現システムを相互運用可能なものとするためのメカニズムも提供される必要がある。

（3）抽象アーキテクチャの守備範囲

抽象アーキテクチャの基本的な狙いは、メッセージ伝送法、エージェント通信言語、コンテンツ記述言語などが異なるエージェント相互間において、意味のあるメッセージ交換の実現を規定することにある。現時点では、抽象アーキテクチャの検討においては、

・メッセージ伝送における相互運用性
・様々な形式のエージェント通信言語の支援
・様々な形式のコンテンツ記述言語の支援
・複数のディレクトリサービスの支援

などに焦点を当てて検討を進めようとしている。勿論、個々の実装においては、ここで規定される特性が若干異なっていても相互運用可能となることが求められよう。

一方、こうした抽象アーキテクチャによる標準化の範囲外となる項目もある。例えば、

・抽象的に記述できない領域
・標準化の準備が整わないか、標準化に関する十分な合意が得られていない領域
・現時点では標準化を必要としない特殊な領域

などは、除外されるべきものと考えられる。例えば、現時点では除外すべきだと考えられ

ている項目としては、

・ライフサイクルと管理
・移動性
・ドメイン
・対話ポリシー
・エージェントの個性

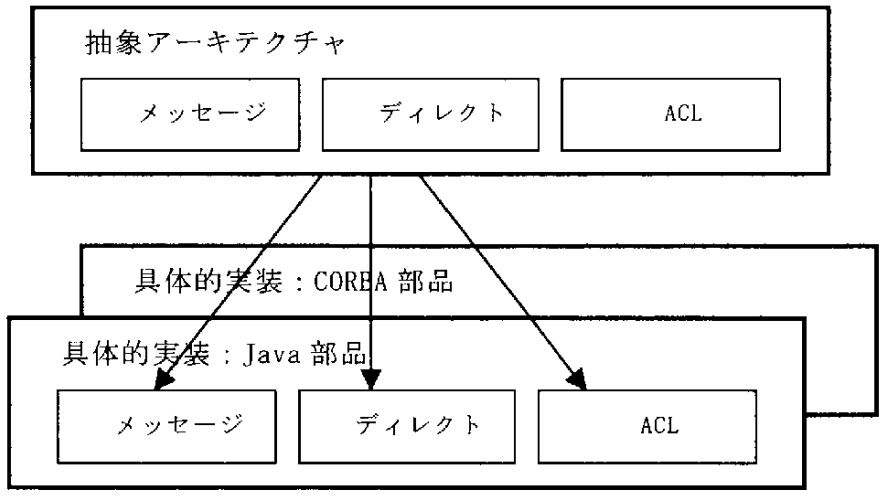
などが挙げられる。

（4）抽象アーキテクチャと具体的アーキテクチャ（実装）

抽象アーキテクチャは、それ自身が実現されるべきものではなく、実際に実装される具体的アーキテクチャの基盤となるべきものである。すなわち、具体的アーキテクチャでは、プログラミング言語やアプリケーションープログラム・インタフェースのような実際のソフトウェアに基づいて実現されるエージェントシステムの設計の詳細が記述される。こうした具体的アーキテクチャがFIPA準拠となるためには幾つかの条件が必要となる。まず、具体的アーキテクチャには、エージェント登録や検索、或いはエージェント間メッセージ伝送などのサービスを実現するメカニズムが存在するが、これらがFIPAの抽象アーキテクチャの対応する要素として明確に記述できることが必要がある。そこで、具体的アーキテクチャに含まれる抽象アーキテクチャ要素は「実装（realization）」と呼ばれる。そして、一般的には、具体的アーキテクチャは抽象アーキテクチャの一部もしくは全部を「実装している」と呼ばれることになる。

実装において実現される抽象アーキテクチャ要素の選択は、具体的アーキテクチャの設計者の自由裁量に委ねられている。同時に、実装は他の選択肢を含んでいても構わない。すなわち、抽象アーキテクチャは、優れたエージェントシステムを実現する上で有用な要素を導入することを妨げるものではなく、むしろ最小限必要とされる要素を規定していると考えるべきである。

上記の観点から、抽象アーキテクチャは選択的な要素を記述していると見なすこともできる。これは、抽象アーキテクチャでは選択的であって、実装においては必須の要素となる可能性があることを意味する。例えば、メッセージ伝送サービスのように、抽象アーキテクチャでは選択的な要素が実装で必要となり、その特性やインタフェースが詳細に仕様化されるような場合である。一方、抽象アーキテクチャの特定の要素だけが実現され、それが多数の実システムで利用されることもある。例えば、ディレクトリサービス要素のみが特定の実装環境に対応するように実現され、それが複数のエージェントシステムの実装で共通的に利用されるような場合である。



図表 D.2-1　抽象アーキテクチャと具体的アーキテクチャ（実装）

（5）今後の予定

抽象アーキテクチャの作成にあたっては、デザインパターンの概念を併用したUMLが利用されるようである．具体的なドキュメントの公開に向けて、本会議における抽象アーキテクチャの検討結果を反映したドキュメンテーションとそのレビューが予定されている。また、その中では、今回議論されていたゲートウェイ、すなわち、異種システム相互間での部分的／全面的な相互運用を支援する仕組みが追加されるようだ。また、抽象アーキテクチャ仕様と並行して、相互運用ガイドラインと実装ガイドラインを整備してゆくことも課題として残されている。

D.2.4.3　むすび

FIPA は、仕様化のプロセスの体制立て直しのため、1999 年度にいったんペースダウンした。workplan 方式に基づく新システムが、世界のニーズにあったスピードと品質を出せるかが今後の成功の鍵を握っている。

— 禁無断転載 —

次世代ネットワークと
産業フロンティア情報技術に関する
調査研究報告書
— ネットワークエージェント技術 —

発 行 平成12年3月
発行所 財団法人 日本情報処理開発協会
東京都港区芝公園3丁目5番8号
機械振興会館B1
電話 (03) 3432-9390

11－R001