

ユビキタス環境におけるエージェント指向ビデオ会議 システムのフレームワークの設計

高橋 秀幸[†], 菅沼 拓夫[†], 白鳥 則郎[†]

ユビキタスコンピューティング環境において、家電製品などの様々な機器を組み合わせてサービスを提供するシステムが登場しつつある。本研究では、これまで主に PC とその周辺入力装置によって構成・提供されてきたビデオ会議サービスに着目し、ユビキタス環境下でビデオ会議サービスを実現するユビキタスビデオ会議システム (UVCS: Ubiquitous Videoconference System) を提案する。UVCS とは、具体的には PC や専用機器に加えて、室内の AV 家電製品等も活用することでビデオ会議サービスを動的に構成するシステムである。本稿では、UVCS の効果的な構成を実現するために、エージェント化したサービスコンポーネントの動的な連携に基づく UVCS の組織構成手法を提案する。

Design of A Framework of Agent-based Videoconference System in Ubiquitous Environment

Hideyuki Takahashi[†], Takuo Suganuma[†] and Norio Shiratori[†]

In the ubiquitous computing environment, systems that provide services by combining various devices including the home electric appliances, are emerging. In this work, we focus on the videoconference service that has ever been composed and afforded by PC and its peripheral devices. We propose Ubiquitous Videoconference System (UVCS) that provide videoconference service in the ubiquitous computing environment. UVCS is a videoconference system that is constructed by dynamic coordination with not only PC and dedicated peripheral equipments, but also with various devices including audio-visual home electric appliances and all that in the room. In this paper, We propose an construction method for UVCS by using dynamic cooperation among agent-based service components.

1 はじめに

ユビキタスコンピューティング技術の進展に伴い、室内の家電を含む身の周りの様々な機器がネットワークに接続し、室内外を問わず、いつでもどこでもネットワークサービスが利用可能になりつつある。また、マルチメディア通信を利用したコミュニケーション環境の発達によって、家電製品を含めた様々な機器を組み合わせてサービスを提供するシステムが登場しつつある [1][2][3]。

一方、ユビキタスコンピューティング環境を背景に、テレワークに関する社会的関心が高まりつつある [4]。テレワークにおいては、SOHO (Small Office・Home Office) 等を対象としたテレワーク支援システムの一つとして、ビデオ会議システムを利用した遠隔コラボレーション支援技術が重要である。しかし、従来の PC ベースのビデオ会議システムは、PC や専用機器を利用するため、ネットワークやマルチメディア通信アプリケーション等のための専門的なスキルを持った利用者を対象としたシステムであった。そのため、ネットワークを用いたマルチメディアコミュニケーションに不慣れた利用者が、安心にかつ簡単に利用することが困難であった。ま

[†] 東北大学電気通信研究所/情報科学研究科 Research Institute of Electrical Communication/Graduate School of Information Sciences, Tohoku University

た、本来の作業とコミュニケーションのための作業を同時に行うことによるパフォーマンスの低下や操作性の劣化等の問題もある。

そこで本研究では、これまで PC とその周辺入力装置群のみによって構成・提供されてきたビデオ会議サービスを拡張し、ユビキタスコンピューティング環境においてビデオ会議サービスを提供する、ユビキタスビデオ会議システム (UVCS: Ubiquitous VideoConference System) を提案する。UVCS とは、室内のユビキタスコンピューティング環境の中で、PC や専用機器に加えて、AV 家電製品を含めた身の周りの多種多様な機器を効果的に利用することによって、様々な利用者が、より簡単にビデオ会議サービスの利用を可能にするシステムである。

本稿では、UVCS の技術的課題を明らかにし、その解決法を与えるエージェント指向フレームワークの設計について述べる。

2 ユビキタスビデオ会議システムとその技術的課題

2.1 ユビキタスビデオ会議システム

ユビキタスビデオ会議システム (UVCS) とは、ネットワークに接続した身の周りの AV 家電製品を含む多種多様な機器の動的な連携によって、PC や専用機器を利用せず構成されるビデオ会議システムであり、利用者の要求や QoS に適応することによって、計算機やネットワークの専門知識を持たない非専門家でも簡単に利用可能な利用者指向のビデオ会議サービスを提供する。

図 1 に、2 人の利用者 User A と User B が UVCS に基づくビデオ会議サービスを利用する例を示す。この例では、User A は画像出力装置として TV、画像入力装置として Digital video camera、そして、音声入力・出力装置として Headset を用い、一方、User B は画像入力・出力装置として Desktop PC に接続している Display と CCD Camera、音声入力・出力装置として Minicomponent stereo に接続している Mic、Speaker を用いることによって、ネットワークを介したビデオ会議システムが構成されている。

このような構成になったシナリオとしては以下の通りである。すなわち、User A に関しては、ビデオ会議サービスを開始する直前に、Note PC 内の音楽ファイルを Headset を用いて再生し、聞いていた。その後、User B とのビデオ会議サービスの要求を行ったので、そのままビデオ会議の音声入

力と音声出力に Headset を利用することになった。また、User A はビデオ会議を行う際には、通常大きな画面を好むので、画像出力サービスとして最寄りの大きな画面の TV が採用された。また画像入力機能としては、TV の横の Digital Video camera と、Desktop PC に接続している CCD Camera があったが、より鮮明に User A の画像をキャプチャすることが可能であるという判断から Digital video Camera が利用された。

User B に関しては、TV で好きな番組を観ている時に User A からビデオ会議開始要求が発生したため、User B やその家族が普段利用している Desktop PC の Display、CCD Camera を画像出力、画像入力サービスとして利用した。また User B がビデオ会議を行うことを知った TV は、出力ボリュームを下げながらもサービスを続け、さらに、最寄りのミニコンポに接続している Mic と Speaker が音声入力・出力サービスを提供することによって、TV を見ながら User A とビデオ会議を行う環境が構成された。

以上のように UVCS は、ユビキタスコンピューティング環境における利用者の機器利用コンテキストや嗜好、ビデオ会議に対する基本要件、機器の接続状況や動作状況等に応じて、利用者が望む QoS を可能な限り満たすビデオ会議サービスを動的に構成する能力を有する。

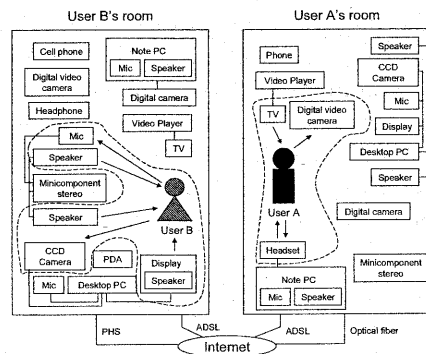


図 1: UVCS の概要

2.2 技術的課題

UVCS を実現するためには、以下のような技術的課題を解決する必要がある。

(P1) 利用者の各種情報やサービス要求の獲得が困難

UVCS では、利用者の機器に関するスキル、嗜好、状況に応じたサービス提供を行うために、利用者の情報・要求等を獲得する必要がある。利用者

の情報・要求としては、プロフィール、プリファレンス、位置情報、サービス要求等があるが、UVCSではPCの利用を前提としていないため、利用者の情報・要求等を獲得・管理することが困難である。(P2) 利用要求や QoS を考慮した、多種多様なビデオ会議システムを構成する要素間の連携が困難

様々な特性、性能を持った構成要素が偏在するユビキタス環境において、利用者要求、資源状況に応じて適切な QoS でビデオ会議サービスを構成する必要があるが、そのためのハードウェア、ソフトウェア、利用者、ネットワーク等の異種な構成要素間の要求・機能調整、QoS 調整等の連携が困難である。

2.3 解決方法

以下に、前章で挙げた技術的課題に対する解決法を挙げる。

(S1) 利用者情報・要求の獲得機能

まず、利用者のスキルや状況、要求等を的確に表すための利用者に関する知識表現を与える。次に、多種多様な構成要素を動的に組み合わせ、要求獲得機能を構成する方法を与える。以上により P1 を解決する。

(S2) 異種構成要素間のリレーションに基づく組織化

各構成要素間の関連性(リレーション)を活用し、利用者個人に適応した QoS 要求をできる限り満たすようなビデオ会議システムを構成するためのプロトコルを提案する。さらに、効果的な組織化のために、ビデオ会議サービスを提供した組織に対する利用者からの評価を反映させる仕組みを導入する。以上によって、P2 を解決する。

3 ユビキタスビデオ会議システムの設計

3.1 UVCS のフレームワーク

図2に共生コンピューティングに基づく UVCS のフレームワークを示す。

共生コンピューティングとは、ハードウェア、ソフトウェアなどが高度に連携することに加え、利用者の社会的活動や人間関係、要求等に基づきシステムが構成される、人間・機械共生空間を実現するためのコンピューティングの概念である [5]。

本研究では、UVCS を構成するコンポーネント(機能要素)をエンティティと呼ぶ。エンティティはその特性によっていつくかに分類される。すなわち、AV 機器や PC などといったデバイスを表現するハードウェアエンティティ、主に PC 上で動作

するソフトウェアを表現するソフトウェアエンティティ、ネットワーク機能を表現するネットワークエンティティ、UVCS の利用者を表現するユーザエンティティなどである。

これらの個々のエンティティの状況を監視したり、制御したりするために、エンティティに関する様々な知識をエンティティに付加したものをエージェントと呼ぶ。またエンティティがエージェントとして振舞うことができるようにするプロセスをエージェント化 (Agentification) と呼ぶ。

エージェントは図2に示すように共生コンピューティングプラットフォーム (Symbiotic Computing Platform) 上に存在し、そこで他のエージェントと協調的な動作を行う。本フレームワークの特徴は以下のとおりである。

- ホームサーバ機能等のサーバを特に置かず、分散型でエンティティが管理される
- ハードウェア、ソフトウェア、利用者、ネットワークなどの、特性の異なったエンティティ間で密な協調が行えるため、様々な制約が存在するユビキタスコンピューティング環境でのサービス構成が効果的に行える。
- 利用者を系に含んでいるため、UVCS の構成に対して利用者の積極的な関与が可能である。

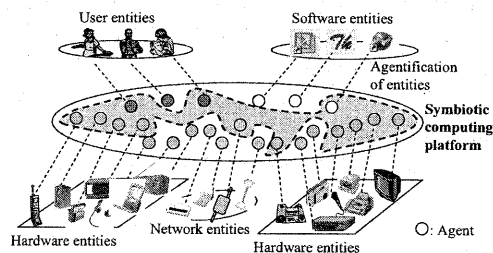


図2: 共生コンピューティングに基づく UVCS のフレームワーク

3.2 エージェントアーキテクチャ

本システムで用いるエージェントのアーキテクチャを図3に示す。Entity は、前節で述べた、エージェントの保持するエンティティである。エンティティに対して、図3で示すようなアーキテクチャを加え、エージェントとして動作可能とすることがすなわちエージェント化である。Cooperation Mechanism (CM) は、他のエージェントとのメッセージ交換を行うためのメカニズムである。エージェント間では、エージェント間通信言語によって規定された形式に

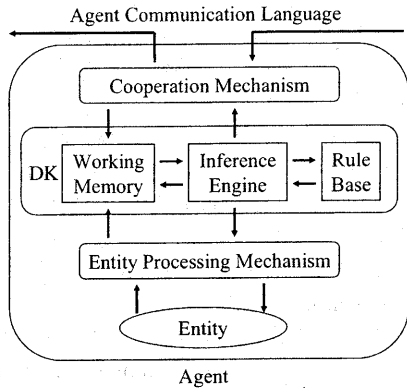


図 3: エージェントアーキテクチャ

より、メッセージ交換を行う。Domain Knowledge (DK) は、そのエージェントの持つ Entity に関する様々な領域知識を保持し、その知識に基づき Entity の監視・制御を行ったり、他のエージェントに対して働きかけを行うための知識ベースシステムである。Entity Processing Mechanism (EPM) は、Entity を直接的に制御するためのインタフェースである。例外通知などの Entity からのイベントを DK に通知し、また DK の指示により、Entity の動作を制御する。

DK はさらに、Working Memory, Inference Engine, Rule Base によって構成される。Working Memory 内には Fact の集合が、また Rule Base には Rule の集合がそれぞれ保持され、Inference Engine がそれらを参照することによって、プロダクションシステムとしての動作を行う。これにより、CM を介した他のエージェントとのインタラクションや、Entity に対する制御・監視などのアクションを実行する。

3.3 利用者情報・要求の獲得機能

利用者のスキルや状況に適応した要求獲得方法として、(1) ユーザ・エンティティの情報・要求の知識表現と (2) ユーザ・エンティティと多種多様なエンティティ間の状況に応じた要求獲得組織の構成について述べる。

(1) ユーザ・エンティティの情報・要求の知識表現

利用者の各種情報、サービス要求等を表現するための知識表現モデルをユーザ・エンティティに与える。具体的には、静的なパーソナルプロフィール (Personal Profile)、ビデオ会議に対する嗜好 (Videoconference Preference)、および動的なビデオ会議要求 (Videoconference Requirement) の3つ

のモデルを構成し、それらをユーザ・エンティティをエージェント化したユーザ・エージェントの DK 内に記述する (図 4)。

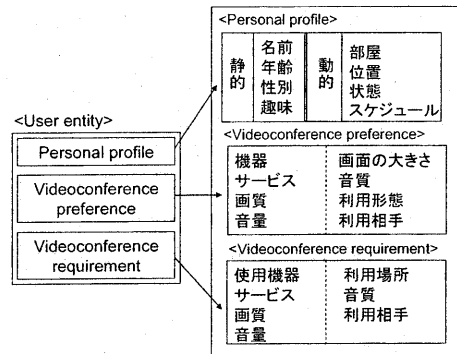


図 4: ユーザ・エンティティの知識表現モデル

(2) ユーザ・エンティティと多種多様なエンティティ間の状況に応じた要求獲得組織の構成

利用者の情報・要求獲得に関して、契約ネットワークプロトコル [6] に基づき、要求獲得機能を実現するエンティティの動的構成手法を与える。

要求獲得は、ビデオ会議に対するサービス開始要求時、サービス調整要求時、サービス終了要求時に必要となるが、利用可能な機器群を動的に組み合わせることによって、要求獲得機能を構成する。例えば、PC を使えるスキルを持つ利用者に対しては、PC を用いて要求入力を求めることも可能だが、PC を使えるスキルがない利用者からは、利用可能なマイクと音声認識ソフトを組み合わせる要求獲得組織が構成される。この機能の詳細については本稿の対象外とする。

3.4 エージェント間のリレーションに基づく組織化

ユビキタスコンピューティング環境下においては、刻一刻と状態を更新する各エンティティが、サービス要求発生と同時にその要求を満たすことが可能かどうかを判断し、エンティティの組織を構成してサービスを提供する必要がある。そこで、エンティティ間にさまざまな関係性 (リレーション) を与え、環境の利用コンテキストに応じてそのリレーションを更新する仕組みを導入する。このリレーションはエージェントによって管理される。すなわちエンティティ間のリレーションをエージェント間リレーション (Inter-Agent Relationship: IAR) として定式化し、エージェントの知識として保持・管理する。エージェントは IAR に基づき、従来の契約ネットワークプロトコルを共生コンピューティング向けに拡張し

た、拡張型契約ネットプロトコルを用いて、サービスを提供するエージェント組織を構成する。

図5に、IARを中心としたUVCSの動作概要を示す。

まず、ハードウェア、ソフトウェア等の各エンティティに対して、分類ごとに適したドメイン依存の知識表現モデルに基づきエージェント化を施す。次に、各エージェントが、ユビキタスコンピューティング環境の利用コンテキストに応じ、IARの更新を行う。UVCSの利用要求が発生した時、(S1)で述べた要求獲得組織構成機能により、利用者要求を獲得、分析する。その要求に基づき、IARを反映したUVCSの組織が形成され、サービスが提供される。サービス提供が終了した際、提供されたサービスのユーザ評価を獲得し、IARへ反映させる。

以下、各フェーズの詳細について述べる。

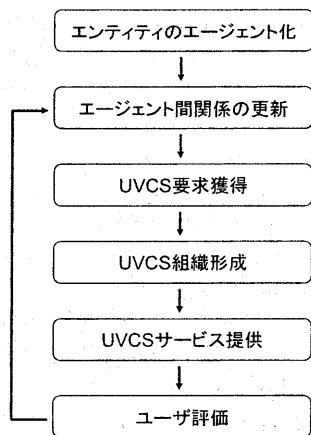


図5: エージェント間リレーションに基づくUVCSの動作概要

(1) エンティティのエージェント化

異種エンティティ間のIARによる組織化を行うために、各エンティティに対して、ドメイン依存の知識表現モデルに基づいたエージェント化を施す。図6にソフトウェアエンティティとハードウェアエンティティの知識表現モデルの例を示す。各エンティティは、それぞれハードウェア、ソフトウェア、ネットワークといった分類型のドメインに適応した知識表現モデルに基づいて記述される。その知識表現モデルでは、ユビキタス環境で想定される各エンティティの動的な状況の変化などを考慮し、静的情報、動的情報に基づいた知識を記述している。これにより、エンティティ間の連携の際に発生した場合の状況変化の通知や状態の把握を可能にする。また、この静的情報や動的状況の変化が生じた際には、エンティティを監視しているエージェントが、

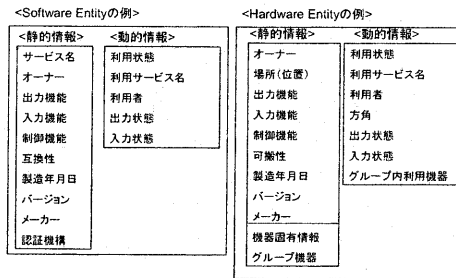


図6: 各エンティティに対する分類型知識表現モデル
常に情報の更新を把握する。

(2) IARのリレーション属性と更新法

効果的な組織形成と利用者に適応したサービス提供のために各エージェント間のIARに対して、a) Broad-Relationship(BR), b) Competing-Relationship(CR), c) Narrow-Relationship(NR)といった3種類の分類を行う(図7)。

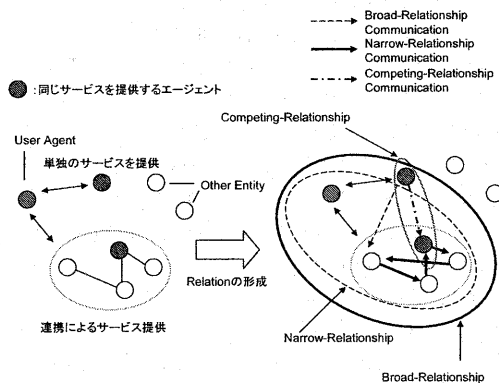


図7: リレーションの分類

a) Broad-Relationship(BR)

エージェントの単体で、利用者に何らかのサービスを提供したエージェントが属する範囲である。例えば、利用者がテレビを見た場合、テレビのエンティティはBRに属する。そして、BR範囲内に属するエージェントは、優先的にtaskが通知される。また、BRに属したときに、現在BRに属しているエージェント全てに対して、自分自身の存在を広告する。

b) Competing-Relationship(CR)

BR範囲内に属したエージェントから広告されたサービスが競合する場合、すなわちサービス属性が等しい場合、互いに利用者とのIARの更新を通知しあうことができる。これにより、taskを受け取った際、他のCR内のエージェントを考慮して

bid を返すかどうかを判断することが可能である。

なお、各エージェントはサービス提供毎にリレーション値を持つが、現時点では、エージェントが単体であるいは連携によって、利用者にサービスを提供した回数をリレーション値とする。各エージェントが持つ他のエージェントに対する IAR は、定期的リレーション値などを考慮して更新することが出来るため、利用者の嗜好の変化にも対応することが可能である。

c) Narrow-Relationship(NR)

複数の連携によって、利用者に何らかのサービスを提供したエージェントが属する範囲である。この関係を形成しているエージェントは、関係を形成しているエージェントが bid を返す可能性があると判断した場合に、Counter Proposal を行うことが可能である。ここで、Counter Proposal とは、task において、ある条件が満たせないために bid を返すことができない場合に、その条件を満たすことが可能であれば Counter Proposal メッセージを返すこととする。なお、この関係が形成されているエージェント間では、常に状況の変化を同じ NR 内のすべてのエージェントに通知する。

(3)IAR を用いた UVCS の組織形成

IAR を用いた UVCS の組織形成は、例えば、利用者のビデオ会議サービス開始要求が発生した際、ビデオ会議構成マネージャが、利用者に対する BR 範囲内の全てのエージェントに対して task の通知を行う。その際、task が通知されたエージェントには、a) CR が形成されている場合と b) NR が形成されている場合が考えられる。そこで、以下に a) CR と b) NR が形成されている場合の組織形成についてそれぞれ述べる。

a) CR が形成されている場合

CR が形成されている場合は、同じサービスが提供可能なエージェントの状況や IAR 情報などを常に通知し合っているため、task が通知された際、各エージェントが bid を返すことが可能な場合、CR 内の相手の情報を参照する。そこで、マネージャに対して、bid を返す前に、相手が bid を返し、award を受け取ると判断した場合は、task に対して条件は満たすが、bid を返さないという判断が可能である (図 8)。

b) NR が形成されている場合

NR が形成されている場合は、まず task を受け取った際に、自分がその条件をある理由によって満たすことが出来ないと判明した場合、NR に属するエージェントが bid を返す可能性があるかどうかを参照する。そこで、NR 内のエージェントが誰

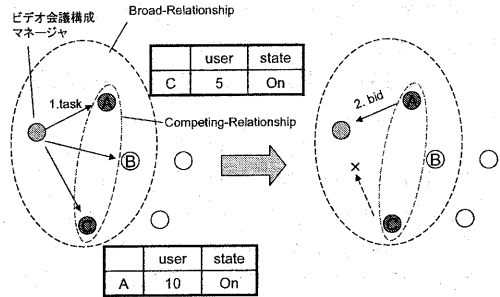


図 8: Competing-Relationship が形成されている場合の組織形成

も bid を返さないと判断した際には、そのままその task を無視するが、NR 内のエージェントが bid を返すと判断した際には、Counter Proposal メッセージをマネージャに対して返す (図 9)。task を通知したマネージャは各エージェントからの bid を受け取り、その中に Counter Proposal メッセージが含まれていることを判断すると、bid を含めて通知してきたエージェントに対して、refusal メッセージを返し、すべての bid を無効にする。そして、Counter Proposal を受け入れるかどうかを利用者に対して問い合わせる。そこで、その結果をもとにもう一度 task の通知を行う。その際、ユーザが Counter Proposal メッセージを受け入れる場合は、次の Counter Proposal メッセージを bid と同様にして扱い、利用者が Counter Proposal を受け入れない場合は、次の Counter Proposal メッセージは破棄し組織構成を行う (図 10)。

加えて、Counter Proposal によるサービス提供が行われる場合、Counter Proposal メッセージを通知したエージェントが award を受け取り、連携組織が確定した際に、サービスを構成するエージェント間で NR が生成される。そこで、お互いの情報を通知し合うことによって、Counter Proposal による award を受け取ったエージェントの現状ではサービスが提供できないことを判明する。そして、

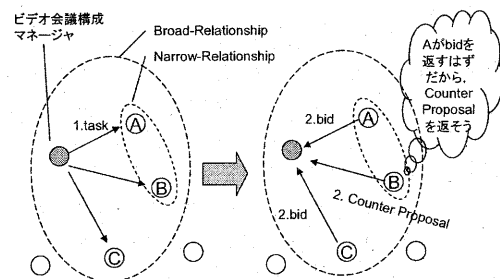


図 9: Counter Proposal メッセージの通知

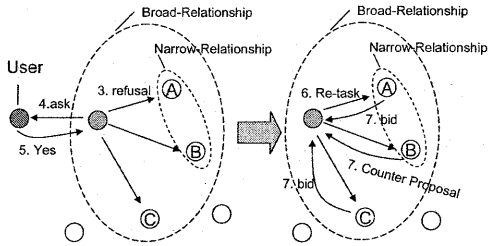


図 10: ユーザが Counter Proposal を許可する場合

現状ではサービス提供が不可能なエージェントは、NR 内の何らかの出力サービスが可能なエージェントに対して、不可能な状態を利用者に可能な状態に変更してもらうためのメッセージの通知を依頼する。例えば、メッセージを受け取ったディスプレイ・エージェントはディスプレイに依頼されたメッセージをテキスト表示することなどが挙げられる(図 11)

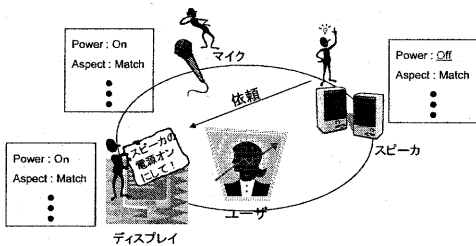


図 11: Counter Proposal によるサービス提供

(4) ユーザ評価

ビデオ会議サービス終了後に、受けたサービスに対して、利用者が評価を行う。評価は、現時点では、良い、普通、悪いなどの簡単な評価を設定し、ビデオ会議サービス終了後に、利用者の最寄のデバイス等が評価を尋ねることとする。また、評価においても利用者の機器に対するスキルや状況を考慮することによって、直接入力を行う場合や音声入力などを用いた入力方法などを利用者の身の回りのエージェント間の連携によって構成し、評価を獲得する(図 12)。その際、利用者からの評価に対して、良い場合は 1、普通の場合は 0、悪い場合は -1 をポイントとして扱い、そのポイントをビデオ会議サービスを提供したエージェントに対して与える。そして、利用者にビデオ会議サービスの提供を行ったエージェント間のリレーション値を変化させることによって、NR 内の IAR も変化することになる。また、サービス提供中に、利用者が現状のサービス構成に対して不満があり、サービス構成機

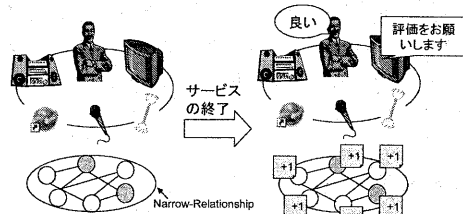


図 12: ユーザ評価の例

器の変更の要求を行った場合に発生するリレーション値の変動モデルも検討中である。

4 実装

本フレームワークの実装においては、ADIPS フレームワーク [7] の DASH1.7.9h[8] を用いる。DASH を用いる理由は、3.2 節で述べたエージェントアーキテクチャを実現するのに適しているためなどである。

共生コンピューティングプラットフォームは、DASH のエージェント動作環境 (Workplace) として実装する。共生コンピューティングプラットフォームは可能な限り各ハードウェアエンティティ上に存在すべきであるが、現状ではコンピューティング環境を持たない機器等も数多く存在するので、将来の拡張性を考慮しつつ、より多くの機器を本フレームワークに取り込む必要がある。具体的には、コンピューティング環境を持たない機器では、近接の共生コンピューティングプラットフォームを持つ機器上にエージェントの知識処理部 (DK) を置き、EPM に拡張を施すことで、コンピューティング環境を持たない機器の監視・制御を行う仕組みを導入する。また、将来、それまでコンピューティング環境を保持していなかった機器に環境が導入された際には、可能な限りエージェントの知識記述レベルでは互換性を維持し、EPM の置き換えにより対応可能とする。

5 まとめ

本稿では、ユビキタスコンピューティング環境において、利用者に適応した各種機器の効果的な連携によって、ビデオ会議サービスを提供する UVCS について述べた。共生コンピューティングの概念に基づく本システムの構成により、利用者とハードウェア、ソフトウェア、ネットワークの密な連携による、柔軟で高度なビデオ会議サービスの提供が可能となることが期待される。今後は、UVCS のプロトタイプによる実験・評価によって、本手法の

有効性を検討したい。また、本手法における IAR とサービス提供後のユーザ評価の効果的な反映のメカニズムの検討とリレーション値の詳細化などが課題である。

参考文献

- [1] Minar, N., Gray, M., Roup, O., Krikorian, R. and Maes, P.: Hive: Distribute Agents for Networking Things, proc. ASA/MA'99, the 1st International Symposium on Agent Systems and Applications and 3rd International Symposium on Mobile Agents, 1999
- [2] T. Iwamoto and H. Tokuda: PMAA: Media Access Architecture for Ubiquitous Computing", IPSJ, Journal, Vol.44, No.3, March, 2003.
- [3] T. Itao, T. Nakamura, M. Matsuo, S. Ttanaka, T. Suda and T. Aoyama: Relationship Mechanism for Dynamic and User Preference-Aware Service Creation, IPSJ, Journal, Vol.44, No.3, March, 2003.
- [4] 榊原 憲: 「e-office」のメッセージ通信ログからみた分散勤務者のコミュニケーションニーズ, 第3回日本テレワーク学会研究発表大会論文集, pp.77-82, June, 2001.
- [5] 菅原 研次, 白鳥 則郎, 木下 哲男: ポスト・ユビキタスコンピューティングに向けて-共生コンピューティング-, 信学技報, AI2003-96, August, 2003.
- [6] R. G. Smith: The Contract net protocol: High-level communication and control in a distributed Problem solver, IEEE Trans, Comput., Vol.29, No.12, pp.1104-1113, 1980.
- [7] S. Fujita, H. Hara, K. Sugawara, T. Kinoshita and N. Shiratori: Agent-based Design Model of Adaptive Distributed Systems, The International Journal of Artificial Intelligence, Neural Networks and Complex Problem-Solving Technologies, Vol.9, No.1, pp.57-70, 1998.
- [8] Distributed Agent System based on Hybrid architecture(DASH).
<http://www.agent-town.com/>