

# モバイルエージェントに基づく分散型会議日程調整システムに関する研究 A Mobile Agent-based Distributed Meeting Arrangement System

影本 憲五<sup>†</sup>      本村 真一<sup>†</sup>      川村 尚生<sup>††</sup>      菅原 一孔<sup>††</sup>  
Kengo KAGEMOTO<sup>†</sup>    Shiniti MOTOMURA<sup>†</sup>    Takao KAWAMURA<sup>††</sup>    Kazunori SUGAHARA<sup>††</sup>  
<sup>†</sup> 鳥取大学 大学院 工学研究科    <sup>††</sup> 鳥取大学 工学部

## 1 はじめに

近年、コンピュータネットワークの発達にともない、グループでの共同作業を効率的に進めるためのグループウェアが注目されている。グループウェアの機能の1つとしてスケジュール調整が上げられる。既存のグループウェアでのスケジュール調整は、会議の参加者全員のスケジュールが共有され、あらかじめ正しく入力されていることが前提となっている。また、全員に都合のよい日時がなかった場合に、交渉を行う機能を有するグループウェアは存在していない。

そこで、我々はモバイルエージェントを用い、事前にスケジュールの入力を行う必要がなく、日程調整において必要であれば交渉を行う会議日程調整システムを開発している[1]。しかし、ユーザがシステム利用時にログインするための認証用コンピュータを特別に設け、常時起動しておく必要があった。また、ユーザが日程調整の途中でログアウトすることには対応できていなかった。

本稿ではシステム利用者のうち誰か一人は必ずコンピュータ(以後ノードと呼ぶ)を起動しログインしているという仮定のもとで、ユーザが日程調整の途中であってもログアウトすることが出来、かつ常時起動しているノードが固定されない分散型会議日程調整システムを提案する。

## 2 会議日程調整システム

### 2.1 会議日程調整システム概要

本システムは、我々の研究室で開発した Maglog[2]を用いて実装している。我々が対象としている日程調整は、会議の参加者と会議に要する時間及び会議開催の候補となる一定の期間が与えられ、その期間内でどの日時に会議を開催するのが適切かを決定するものである。

### 2.2 エージェントの動作

本システムでは、会議を招集しようとする時、調整エージェントと呼ばれるエージェントが招集者のノード上で生成される。生成された調整エージェントによって、収集エージェントと呼ばれるエージェントが参加者ごとに生成される。収集エージェントは担当する参加者から予定を収集し、調整エージェントに渡す。調整エージェントは、すべての参加者の予定を取得すると、参加者全員に都合のよい日時を探し会議の開催を通知する。全員に都合のよい日時がない場合、調整エージェントは、適切な日時において任意の参加者と交渉を行う。適切な日時は、交渉が必要な時間数、交渉が

必要な参加者数が少なくなるように選ぶ。招集者が開催を希望している時間も考慮する。

以上のエージェントに加えユーザ認証のためのユーザ管理エージェントが存在している。ユーザ管理エージェントは、ユーザのログイン時にログイン名とパスワードの認証を行う。加えてユーザがログインしたノードのIPアドレスを保持しており、他のエージェントがユーザの元へ移動する際にはこのIPアドレスを参照する。

### 2.3 実行例

実際にユーザが予定を入力している例を図1に示す。ユーザはGUIを用いて簡単に予定を入力することが出来る。preferable は会議の開催を希望する時間を表しており、招集者だけが入力できる。busy は会議に参加できない時間、tentative は交渉の余地のある時間、free は会議に参加できる時間を表している。busy と入力されている時間では会議は開催されず、必要であれば tentative と入力されている時間で交渉が行われる。



図 1: ユーザの予定入力例

## 3 分散型会議日程調整システム

### 3.1 分散ハッシュテーブル

分散型会議日程調整システムでは分散ハッシュテーブルの一種である Content-Addressable Network(CAN)[3]に基づく P2P ネットワークを用いる。CAN では (キー, 値) のペアが仮想座標空間に格納され、仮想座標空間はいくつかの領域に分割されて

いる。領域はノードによって所持される。ノードの参加、離脱によって領域は分割、統合され、領域を所持するノードは変化する。CAN に基づく P2P ネットワークを構築するためのエージェントを先に述べた 3 種類のエージェントに加えた。

そして、ユーザのログイン時には他のノードから領域を割り振り、ログアウト時には割り振られた領域を他のノードに返還するようにした。また、位置の特定が必要なエージェントを、エージェント固有の ID をキーとして分散ハッシュテーブルの 2 次元直交座標空間に配置した。つまり、各エージェントはそれぞれの固有の ID がキーとしてマッピングされている領域を管理しているノード上で動作する。

これにより、システムにログインしているユーザのノード上にはユーザ管理エージェント及び自分とは直接関係のない会議の調整エージェントも存在し得る。しかし、ログアウト時にはエージェントは他のノード上へ移動するためユーザは日程調整の途中でもログアウト出来、ユーザ認証専用のノードも必要ではなくなる。

### 3.2 動作例

図 2 にユーザ D がシステムにログインし、領域が割り振られている様子を示す。ノード A が所持していた領域の一部がノード D に割り振られている。

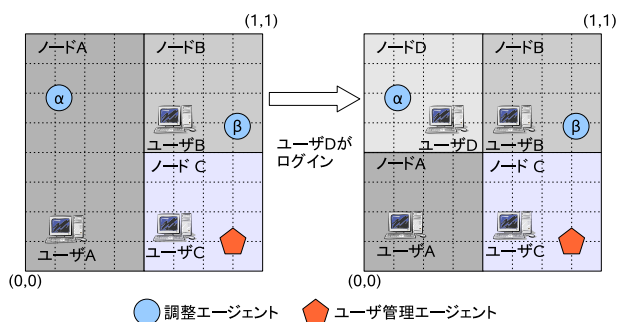


図 2: ユーザ D がログインする様子

図 3 にユーザ管理エージェントが存在しているノードが、領域の変化にともない動的に変わる様子を示す。ユーザ C がログアウトすることにより領域が変化し、ユーザ管理エージェントはノード C からノード B に移動している。すなわち、常時起動している必要のあるノードは固定されていない。

次に、図 4 にノード B において調整エージェントが日程調整の途中であり、ノード B を使用しているユーザ B がログアウトする例を示す。ユーザ B がログアウトすることにより、ノード B にいた調整エージェントがノード C に移動している。すなわち、日程調整途中でのログアウトが可能となっている。

## 4 おわりに

本稿では分散型会議日程調整システムを提案した。提案システムではユーザ管理エージェントは動的に移動するため、ユーザ認証専用の常時起動ノードは必要なくなる。また、ログアウト時に各エージェントを他

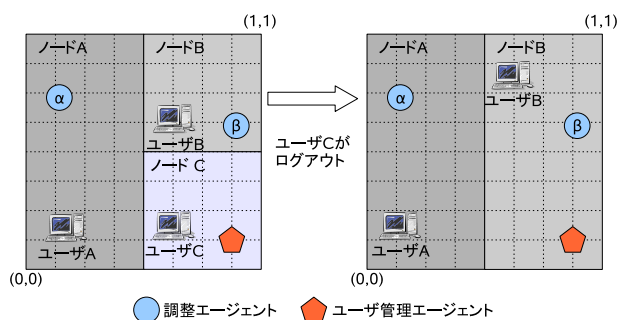


図 3: ユーザ管理エージェントの配置されているノードの変化

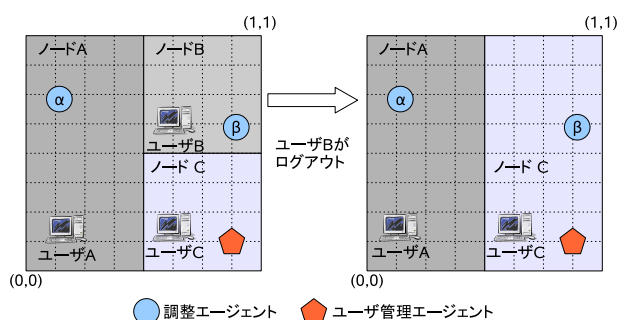


図 4: 日程調整の途中でログアウトする様子

のノード上に移動させることにより、誰かがシステムを利用しているという仮定のもとで、日程調整の途中であっても自由にログアウトすることが可能となる。

ただし、提案システムでは、領域を保持しているノードが最低 1 台は必要のため、ノードを起動しておくのはシステム管理者の仕事となり得る。しかし、領域を保持するノードを動的に変化させることにより、システムを止めることなく、ノードの管理、保守を行うことが可能となる。

今後の課題として、領域を保持しているノードが 1 台になった場合にもそのノードを終了でき、再起動時に終了前の状態から再開することも検討している。

## 参考文献

- [1] Kawamura, T., Kagemoto, K. et al.: Meeting Arrangement System Based On Mobile Agent Technology, *Proc. of the 2nd Intl. Conf. on Web Information Systems and Technologies*, pp. 117–120 (2006). Setubal, Portugal.
- [2] Motomura, S. et al.: Logic-Based Mobile Agent Framework with a Concept of “Field”, *IPSJ Journal*, Vol. 47, No. 4, pp. 1230–1238 (2006).
- [3] Ratnasamy, S. et al.: A Scalable Content-Addressable Network, *Proc. of ACM SIGCOMM*, pp. 161–172 (2001). San Diego, CA.