

D-024

## 仮想オフィス環境におけるマルチクライアント機能の実装と更新反映伝播時間の評価 Implementation of Multi-Client Function and Evaluation of Reflection and Distribution Time of Update in Virtual Office Environment

那須 洋之†      賀来 健一†      富井 尚志‡  
Hiroyuki Nasu      Kenichi Kaku      Takashi Tomii

### 1. はじめに

近年、コンピュータ技術の発達とネットワークの広帯域化により、オフィスの形態が変化してきている。例えば、自宅や郊外の小さな事務所まで仕事をこなす SOHO などがある。このようなオフィスでは人々がそれぞれ異なる場所で仕事をしている場合が多く、3次元の仮想空間でのコミュニケーションを使用することもできるであろう。

3次元の仮想空間と DB を連携させ、3次元空間の情報を管理する研究事例はいくつか見られる。例えば、共有空間のためのデータモデルの提案を行っている VWDB[1] がある。しかし、これらのように空間のオブジェクトの位置情報や動きを DB で管理するだけでは、空間内の意味情報はわからない。例えば、オフィスのような共同体意識を持って生活を行う空間においては、その中に存在する物体の「意味」やその中でのみ有効な知識や約束事、共通の「意図」が存在する。そこで、空間の持つ意味を抽出し、空間中のオブジェクトと結びつけられれば、空間の利用者の意図を表した操作を実現でき、空間内の人々の意思疎通の助けになると考えられる。このような仮想空間を本プロジェクトでは仮想オフィス環境と名付けた。そして既に仮想空間を表示し、意図を表した操作を行う GUI として仮想オフィス環境のためのインタラクティブなブラウザを実装している[2]。

本論文では、この仮想オフィス環境ブラウザで行った意図を持った操作を他の利用者のブラウザに反映させるマルチクライアント機能の設計・実装をし、プロトタイプ環境において評価を行った。具体的には、I) ある利用者が操作を行ってから他の利用者のブラウザに反映されるまでの時間は短いほうがよい、II) データの受け渡しは確実性が高いほうがよい、III) DB サーバに対する負荷は少なくする、という設計方針の下に3つの方法のマルチクライアント機能を設計し、仮想オフィス環境ブラウザにマルチクライアント機能を実装した。3つの方法とは、i) 一定時間ごとに DB に問合せを行う Polling, ii) ある利用者が操作を行ったときにそれを知らせる文字列を Broadcast する更新通知, iii) ある利用者が操作を行ったときにその操作により変更された空間内のオブジェクトの情報を Broadcast する差分送信である。そしてそれら3つの方法のどれが適しているかを評価するために、応答時間から DB サーバの負荷を評価する差分取得時間、ある利用者が操作を行ってから他の利用者のブラウザへ反映されるまでの時間を評価する更新反映伝播時間という2つの評価基準を定義し、仮想オフィス環境におけるマルチクライアント機能の評価を行った。

†横浜国立大学大学院環境情報学府  
情報メディア環境学専攻

‡横浜国立大学大学院環境情報研究院

### 2. 仮想オフィス環境

本章では、仮想オフィス環境においての意味や意図とは何か、そして仮想オフィス環境における検索、操作について述べる。

#### 2.1 意味と意図

従来の仮想空間では、空間内のオブジェクトの形状や位置情報、動きといった視覚的な情報のみが利用者に共有され、その空間に存在する暗黙的な意味情報は利用者個々の認識に任されていた。そこで本プロジェクトでは予めそのような意味情報を抽出し、それを3次元形状データと関連付けて DB に蓄積することにより、視覚的な情報だけでなく意味情報も共有できると考えた。意味情報の共有が可能になれば、仮想空間内を統一された価値基準で検索、操作ができるようになるであろう。そこで本プロジェクトでは、モデル化する意味情報を明確にするために意味と意図を定義した[2]。このように意味と意図を体系的かつ明示的に定義しておく手法としてオントロジ[3]を用いた。

空間の意味の共有を実現するためには、空間の形状データとの対応が必要になる。本プロジェクトでは図1のように空間に存在する意味を形状データと分けて、その間を「仲介者：存在エンティティ」で結ぶ三層構造モデル化手法[4]を提案してきた。この手法により、意味と形状データを分離して管理することができ、構造を持った意味の表現、データの再利用が可能になる。

#### 2.2 意図付き操作モデル

前節で述べたようなモデル化手法をもとに本プロジェクトでは、空間の利用者の検索の定式化と意図を持った操作の形式化を行った[4]。この形式化によって、意味層、存在エンティティ層、形状データ層への意図を持った検索やそれらをまたぐ検索と空間の利用者が指定したオブジェクト群に対して意図を持った操作を実行することが可能となる。本プロジェクトではこのような操作には、利用者が操作の意図を明示的に宣言する操作(明示的操作)と利用者がオブジェクトに対して行った行動についてプログラムがその操作の裏にある意図を読み取る操作(暗示的操作)があると考え、これらを定義した[2]。

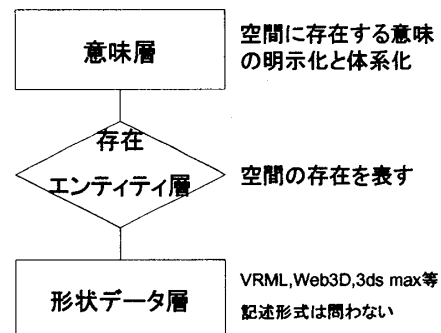


図1 三層構造モデル

## 2.3 仮想オフィス環境の共有

前節で述べた操作を行うことによって、変化したオブジェクトの状態が表す意味は、DB上でexplicitなentityとして表現される。一方、分散して世界の一部だけを利用する本システムの利用者は、必要に応じてクエリによりビューとして局所空間を共有し、利用できるれば、利用者サイドの計算機資源も少なくすむので、メリットが大きい。そこで、DBのビューとしてオブジェクトのIDや状態の意味を表すentity IDを利用者側に局所的に持たせる構造とした。

すなわち、DBMSによって管理された仮想空間は、任意の利用者固有のクエリにより作成される「意味情報付きシーングラフ」[2]として各利用者が取得する。このシーングラフを用いることで、仮想空間をブラウザに表示する。そして、このシーングラフに対する操作をDBに反映し、他の利用者の意味情報付きシーングラフを更新することで、結果的にDBと意味情報付きシーングラフの整合性を保ち、複数の利用者が同一の仮想空間を共有することが可能になる。

## 2.4 仮想オフィス環境ブラウザ

本プロジェクトでは、図2に示すように仮想オフィス環境を表示し、利用者が検索や意図を持った操作を行える仮想オフィス環境ブラウザを実装している[2]。このブラウザに必要な機能として、1) DBに蓄積された仮想空間のデータをXML形式の意味情報付きシーングラフとして取得し、そのシーングラフに書かれたデータを読み込むことにより、ブラウザに仮想空間を表示する、2) その空間の中をウォークスルーできる、3) 空間の中のオブジェクトを指定・移動させる、4) 明示的操作と呼ばれるユーザの意図を明示的に宣言した操作を行う、5) 複数の利用者がそれぞれのブラウザからひとつの仮想空間を共有することができる(マルチクライアント機能)、がある。

1)～4)については既に設計・実装について詳しく述べている[2]。そこで本稿では5)にあるマルチクライアント機能の設計・実装について述べる。特に、オブジェクトの座標だけでなく、DB上での意味情報と各利用者の保持する意味情報を整合させなければならない。本システムは、5)の有効な実装方法が重要な位置を占める。

## 3. マルチクライアント機能の設計

ある利用者が行った操作を他の利用者のブラウザに反映させるには、各利用者が所持している意味情報付きシーングラフを、操作による変更が反映されたDBと整合性が取れた状態にしなければならない。その方法として以下の2つが考えられる。

- ① DBに意味情報付きシーングラフ作成クエリを発行し最新のシーングラフを得る
- ② 操作により変更が起こったオブジェクトのデータのみを取得し、そのデータを用いてブラウザ側で意味情報付きシーングラフを更新する

①の方法では、局所空間すべてのデータが入った意味情報付きシーングラフを新たにDBから得る。したがってDBとの整合性は保たれる。それに対して②の方法では、操作によって変更されたオブジェクトのデータをブラウザがDBより取得し、そのデータを基にそれぞれの利用者側で意味情報付きシーングラフを書き換える。したがって、DBの意味情報との整合性を保つためには、書き換えに必

要となるのに十分なデータを取得する必要がある。この2つの方法を比較すると、①の方法では操作が起こるたびに数メガバイトのシーングラフを作成し、全員に送信しなければならない。したがってDBやネットワークにかかる負荷が大きく大規模な空間では難しい。そこで、②の方法を用いてマルチクライアント機能の設計を行う。

②の方法を行うにあたり、DBと意味情報付きシーングラフの整合性を保つ更新に必要な、オブジェクトのデータ(差分データ)を考える。意図を表した操作は、操作の対象となるオブジェクトの意味を変更することはない。そこで、1オブジェクト当たりの差分データは以下のデータとなると考えられる。

- I. 操作されたオブジェクトを識別するためのID
- II. 操作の対象となる場所のオブジェクトを識別するためのID
- III. 操作されたオブジェクトの位置情報

これらの差分データをすべての利用者に渡すことができれば、DBと整合性を保ったまま意味情報付きシーングラフを更新でき、ブラウザ上の仮想空間を更新できる。

そこで、更新時にこれらの差分データを全利用者に渡す方法を、以下のような設計方針の下に設計を行った。

- 利用者が操作を行ってから他の利用者にデータが届くまでの時間は短いほうが良い
- データの受け渡しは確実性が高いほうが良い
- DBへの負荷は小さいほうが良い

以下設計した3つの方法について詳しく述べる。

### 3.1 Pull型差分取得

データの受け渡しの確実性が高いと考えられる方法としてTCPを用いるPull型差分取得を設計した。この方法ではすべての利用者が一定間隔ごとにDBに更新があったかどうかを問い合わせにいくPollingを行う(図3)。メリット・デメリットとしては以下のようなことが挙げられる。

メリット：少数利用者のとき確実性が高い

デメリット：Polling間隔と整合性がトレードオフ、利用者数の増加によりDBの負荷が増加

### 3.2 Push型差分転送

DBの負荷が低く、通信速度が速いと考えられる方法として、UDPを用いて一対多の通信を行うBroadcastを使ったPush型差分転送を設計した。操作が行われるとDBサーバが差分データを取得し、すべての利用者にBroadcastする(図4)。このようなUDPを使った通信はネットワークゲーム等のネットワークアプリケーションでデータ送信の際に通信速度を減らす方法としても研究がなされている[5]。メリット・デメリットとしては以下のようなことが挙げられる。

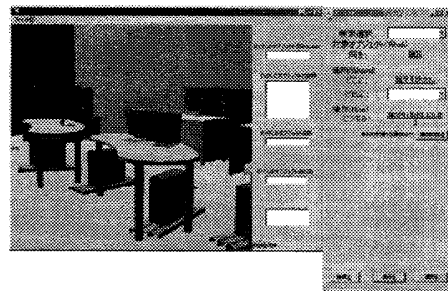


図2 仮想オフィス環境ブラウザ

メリット：DB への負荷が少ない，データ転送速度が TCP を用いるより速い

デメリット：クライアントの伝送路状況に依存しパケット損失（データの不整合）の可能性あり

### 3.3 更新通知

Polling より DB の負荷は小さいが，確実性が低くなると考えられ，また Broadcast より DB の負荷は高いが，UDP を使い送信するデータ量が少ないので確実性が高くなると考えられる方法として，更新通知を設計した。UDP を用いた Broadcast で文字列を送信することにより操作が行われたことを通知し，それを受け取った利用者が DB に問い合わせにいきデータを取得する(図5)。

メリット：更新時のみ DB に問い合わせる

デメリット：操作が行われた時にアクセスが集中する，クライアントの伝送路状況に依存しパケット損失（通知の失敗）の可能性あり

## 4. マルチクライアント機能の実装

本章では，前章で設計した3つの方法を仮想オフィス環境ブラウザに実装を行う。実装を行うにあたり，以下のよ

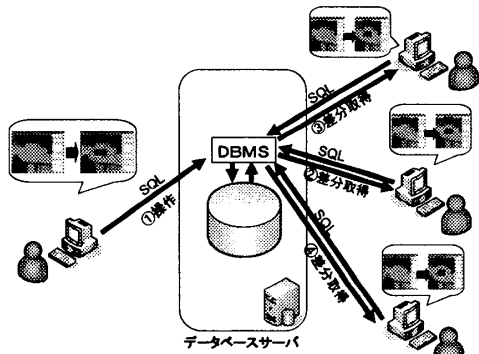


図3 Polling

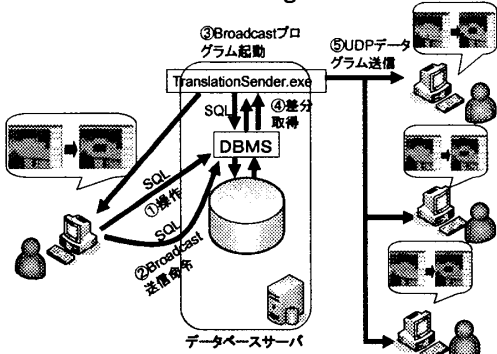


図4 Broadcast

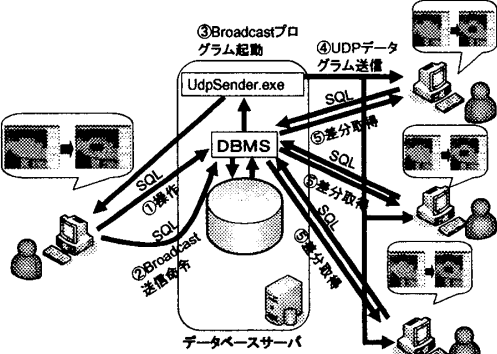


図5 更新通知

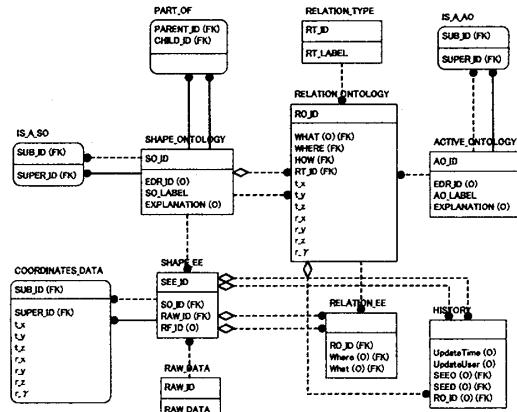


図6 論理データモデル

うな DB サーバ上に図6の論理スキーマを実装した。

ハードウェア：Dell Precision 530，CPU：Intel Xeon 2.4GHz (DUAL 構成)，Memory：1GByte，OS：Microsoft Windows Server 2003，DBMS：Microsoft SQL Server 2000

差分取得は History テーブルと RELATION\_ONTOLOGY テーブルを(図6)結合することにより，操作されたオブジェクトを識別するためのIDである SEEO，操作の対象となる場所のオブジェクトを識別するためのIDである SEED，操作されたオブジェクトの位置情報である t\_x~t\_z と r\_x~r\_y を取得するようにした。DBMS が Broadcast プログラムを呼び出す方法としては，Microsoft SQL Server 2000 のシステムスタッドプロシージャである xp\_cmdshell を使用した。また Broadcast において1オブジェクトの差分データが 36Byte であり，1回の Broadcast で送ることが可能なデータ量がヘッダを含めて 65536Byte で固定されているため，1回の Broadcast で送信できる差分データは 1819 オブジェクトが限界であると考えられる。そこで 1819 以上のオブジェクトが操作されたときには 1819 オブジェクトごとに分けて送信するようにした。

## 5. マルチクライアント機能の評価

本章では，前章で実装したマルチクライアント機能を用いた実験を行い評価する。

実験における DB の規模は，我々が現在使っている研究室(使用人数 15 人)をモデルに構築した空間に存在する全オブジェクト数が 1000 なので，それより十分に大きな規模としてその 10 倍である 10000 オブジェクトの規模まで実験を行った。

利用者は DB の更新が起きた場合，現在所持している意味情報付きシーングラフと DB との差分のみを取得することで空間を共有する。そこで，利用者数 15 人のうち 1 人の利用者が更新を行い，前章で実装した3種類の差分取得方法を用いて，他の利用者がその差分を受け取る時，一回の更新で取得する変更オブジェクト数(差分オブジェクト数)を変化させることにより，どの差分取得方法が優れているかを評価した。同じ評価基準で評価するために，差分取得時間と更新反映伝播時間という評価基準を以下に定義する。

- 差分取得時間：各利用者が差分取得クエリを発行してから差分データを取得するまでの時間
- 更新反映伝播時間：ある利用者が操作を行ってから他の利用者が差分データを取得するまでの時間

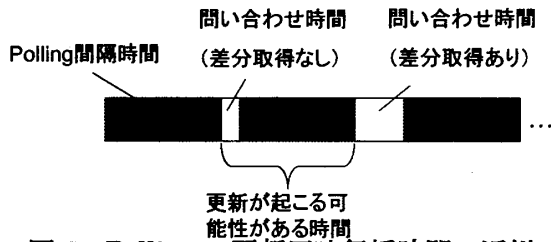


図7 Pollingの更新反映伝播時間の近似

Pollingの更新反映伝播時間は、Polling間隔時間の中で更新が起きたかによって変わってしまう。そこでPollingにかかる時間の詳細(図7)を見ると、差分取得なしの問い合わせはPolling間隔時間に比べて十分に小さく、また利用者全員の平均を考えれば、Pollingの更新反映伝播時間はPolling間隔時間の1/2と差分取得時間の和であると近似できると考えられる。したがって、今回の実験ではそのように近似し評価を行った。

### 5.1 差分取得時間の評価

Pollingと更新通知で差分を取得する場合、利用者が自らDBにクエリを発行し差分を取得する。そこでDBへの負荷を評価するために差分オブジェクト数を変化させたときの差分取得時間を測定した。結果は図8に示す。この結果より、Polling間隔が短いとDBへの負荷が大きくなるので、DBの負荷が大きくなり差分取得時間が長くなったと考えられる。また差分オブジェクト数が増えるとPollingのDBへの負荷の増え方が更新通知より大きいことがわかる。これは、Pollingは差分オブジェクト数が増えると1利用者あたりの差分取得時間が長くなる影響で、ある利用者が差分を取得中に既に差分を取得した利用者のPollingが来る回数が増え、それだけDBの負荷が大きくなるからだと考えられる。この結果より、差分オブジェクト数が増えるとPollingより更新通知が有効であることがわかる。また、各利用者が差分取得クエリを出さないことからBroadcastの差分取得時間を測定していないが、LocalのDBに自らの1回のみアクセスで済むBroadcastが一番DBへの負荷が小さいと考えられる。したがってDBの負荷は、Polling > 更新通知 > Broadcastであると考えられる。

### 5.2 更新反映伝播時間の評価

操作が行われてから最も速く更新が反映できる方法を求めるために、差分オブジェクト数を変化させたときの更新反映伝播時間を測定した。結果は図9に示す。この結果より、操作が行われてから利用者が更新を受け取るまでの時間が一番短いのはBroadcastであることがわかる。しかし、Broadcastは差分オブジェクト数6000以上では、データの損失が起こりすべての差分を受け取ることができなかつた。これは、UDPを用いているのでデータの損失が起きたと考えられるが、その原因については、ネットワークが混雑している環境やサーバを強化して実験を行うなどして今後確かめていきたい。

今回は、操作を行う利用者は1人という限定の下で実験を行ったが、本来はそれぞれの利用者が自由に操作を行っているはずである。また、大きなオフィスになると利用者数や仮想空間内のオブジェクト数も増加すると考えられる。したがって、DBの負荷が大きくなる方法では操作に支障をきたすと考えられる。今回の実験ではBroadcastが一番よい

結果であったが、差分オブジェクト数が増えるとデータを損失することがわかった。そこで、今後、1オブジェクト当たりの差分データのデータ量の削減方法の提案や利用者数を増やした環境やネットワークが混雑している環境での実験を進めていきたいと考えている。

## 6. まとめ

本プロジェクトでは、オントロジによる空間内の意味の統一により、利用者間で意味を共有し、意味を持った操作、検索が可能な仮想空間である「仮想オフィス環境」の提案と実装を行っている。そこで、本論文では複数の利用者での仮想オフィス環境ブラウザを使った仮想空間共有の機能の設計・実装方法を示し、プロトタイプシステムにより評価を行った。実用に近い環境やさらに大規模な仮想空間での評価が今後の課題である。

## 謝辞

本研究の一部は文部科学省科学研究費補助金(課題番号17700097)の支援による。

## 参考文献

- [1]渡辺知恵美,増永良文,“仮想世界データベースシステムにおける仮想世界同期法”,TOD,Vol.43,No.59(TOD18),2003
- [2]賀来健一,那須洋之,南博康,富井尚志,“複数利用者間で意味を共有した検索・操作が可能な仮想空間ブラウザの提案”,データ工学ワークショップ論文集(DEWS2005),Mar,2005
- [3]溝口理一郎,“オントロジー研究の基礎と応用”,人工知能学会誌Vol.14,Nov,1999
- [4]南博康,賀来健一,富井尚志,“意図共有を実現するオフィス仮想環境の設計”,信学技法,Vol.104,No.345,DE2004-114,pp.1-6,Oct,2004
- [5]井芹威,堀真人,藤川和利,下条真司,宮原秀夫,“多人数参加型ネットワークアプリケーションの広域ネットワーク環境における利用実験”,信学技法,IN2000-121,pp.21-28,2000

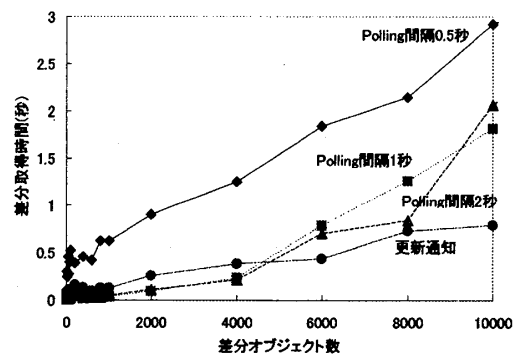


図8 差分取得時間の評価

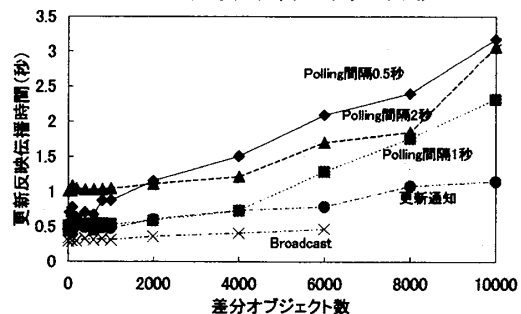


図9 更新反映伝播時間の評価