

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

賀来 健一[†] 那須 洋之^{*} 南 博康[†] 富井 尚志[‡]

[†]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 工学部電子情報学科

[‡]横浜国立大学 大学院環境情報研究院

〒240-8501 横浜市保土ヶ谷区常盤台 79-7

E-mail: † {d04hc008, d03hc049}@ynu.ac.jp, * b0144103@ynu.ac.jp, ‡ tommy@ynu.ac.jp

あらまし 我々はオントロジによって意味を共有し、意図を持った操作、検索が可能な仮想空間である「仮想オフィス環境」を提案してきた。この仮想空間では、(1)各ユーザが共有されている意味を用いた操作や検索（すべての論文を何もしまわれていない本棚にしまえなど）、(2)他のユーザが過去に行った操作の意図や履歴を用いた検索を行うことが可能である。本論文では、「仮想オフィス環境」に用いるブラウザの設計方法を提案する。さらにプロトタイプシステムを実装し、ユーザ数による負荷の変化などの評価を行うことにより、有効時間内で意味をキーとした検索や操作が行えることを示す。また、複数ユーザで「仮想オフィス環境」が共有できることを示し、このブラウザの有用性を示した。

キーワード マルチメディア DB, ユーザインタフェース, 時空間 DB, グループウェア

Proposal of Virtual Space Browser Enables Retrieval and Action with Semantics which is Shared by Plural Users

Kenichi KAKU[†] Hiroyuki NASU^{*} Hiroyasu MINAMI[†] Takashi TOMII[‡]

[†]Department of Information Media and Environment Sciences, Graduate School of Environment and Information Sciences, Yokohama National University

^{*}Division of Electrical and Computer Engineering, School of Engineering, Yokohama National University

[‡]Faculty of Environment and Information Sciences, Yokohama National University

79-7 Tokiwadai, Hodogaya-ku, Yokohama 240-8501 Japan

E-mail: † {d04hc008, d03hc049}@ynu.ac.jp, * b0144103@ynu.ac.jp, ‡ tommy@ynu.ac.jp

Abstract We proposed “Virtual Office Environment” which is a virtual space where semantics are shared by Ontology, and enable the retrieval and action with their intention. Each user can do two things in the virtual space. (1)Each user can do the retrieval and action with shared meaning (e.g. Put all papers into a bookshelf which is empty). (2)Each user can retrieve with the history and other’s action with its intention. This paper proposes the design methodology of the browser which is used for “Virtual Office Environment”. In addition, we have implemented prototype system. And this paper shows this browser is useful.

Keyword MultimediaDB, UserInterface, TimeSpaceDB, GroupWare

1. はじめに

近年の高度なグラフィックコンピュータの登場と広域情報ネットワークをもとに3次元グラフィックスやVRを用いたバーチャルミュージアムやバーチャルショッピングモールなどのサイバーコミュニティ空間が実現されている。さらにこれらの情報技術の進歩

により、オフィスの形態にも変化が生じてきている。例えば、自宅で仕事をこなすSOHO、本社とネットワークで結ばれたサテライトオフィス、自分の席を持たずに仕事を行うフリーアドレスオフィスなどがある。これらのオフィスでは、人々が所々の場所に分散しており、コミュニケーションはネットワークを通じて行われることが多い。ネットワークを介したコミュニケ

ーションの手段は、メール、チャット、掲示板などがある。これらだけでなく、空間を共有しアバタなどを通じて情報交換をし合う PAW[1]に代表される仮想空間もコミュニケーションの手段として挙げる事ができるであろう。

仮想空間を DBMS で管理し、DB でオブジェクトの形状データなどを管理すると、検索や更新などの多様な操作が可能になると考えられる。このように、仮想世界、あるいは現実世界の 3 次元情報と DB を融合させる研究事例がいくつか見られる。例えば、増永らの VWDB[2,3]では、共有時空間のためのデータモデルの提案を行っている。田島らは、DB に蓄積されたオブジェクトに対するアクセス制御に関する提案[4]を行った。また、現実世界の 3 次元情報をモデル化しデータベースで管理するシステムとして RWDB[5,6]がある。

しかし、仮想空間内で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、この仮想空間に参加しているユーザの空間中での行動やオブジェクトへの操作に込められた意図は分からない。例えば、仮想空間の引き出しの中に書類を保管する場合、座標を指定してオブジェクト(ここでは書類)を移動させるよりも、「引き出しの中にこの書類を保管しておけ」と命令する方が、命令者の「引き出しの中に書類を保管しておきたい」という意図が明示化されており、空間の他のユーザに伝わりやすい。

そこで、このような空間に参加しているユーザの意図を表した操作の実現のために、仮想空間の持つ意味を明確化する。例えば、机の持つ意味に注目すると「机」という名称を持ち、「引き出しに筆記用具をしまえる」や「書類を上板の上で書ける」のような機能や役割がある。このような空間の持つ意味を抽出し、明示的かつ体系的に管理して空間中のオブジェクトと結びつければ、空間の利用者の意図を表した操作を実現できる。

その手段として、本プロジェクトでは、空間の意味を明確かつ体系的に管理する手段として、対象世界の意味の捉え方を明確化するオントロジ[7,8,9]を利用する。このオントロジと形状データとを対応付け、空間の利用者間の意味の固定化と共有化を図る。このように、空間の意味を表現することにより、空間の機能を検索するという研究も行われている[10]。

我々はこれらを用いることによって、仮想空間で意味を利用した検索「この本がしまっている棚はどこ？」や、利用者の意図を持った操作「書類をしまえ」といった利用者の意図が明示化された操作が可能である仮想空間「仮想オフィス環境」[11]の実現をプロジェクトとして取り組んでいる。この仮想オフィス環境では、意味と共に利用者の操作の意図を共有し、検索や操作の結果を 3 次元のイメージによって提供するため、オフィスの対物活動における情報伝達やルール習得が迅速に行える。

本論文では、この仮想オフィス環境のブラウジングに必要な仮想オフィス環境ブラウザの設計方法を提案し、実装したプロトタイプシステムについて述べる。本手法の評価のために、プロトタイプシステムを用い

て、仮想オフィス環境に複数の利用者が参加した状態で、利用者数や利用者が行った操作によって更新されるオブジェクト数を変化させる実験を行った。そして、後にあげる 2 つの差分取得方法のどちらがこの仮想オフィス環境に適しているかを決定する。これによって、有効時間内で意味をキーとした検索や操作が行えることを示すことで、本論文で提案したブラウザの有用性を示す。

2. 仮想オフィス環境

本章では、仮想オフィス環境ではどのようにして空間の意味と形状データの共有を行い、空間の利用者が意図を持った検索や操作が可能になるかを説明する。

2.1. 意味と意図

従来の PAW などのような仮想空間では視覚的な情報や位置情報のみが共有され、その空間に存在する暗黙的な意味情報は個々の利用者の認識に任されていた。そこで、空間の情報共有を実現するために、本プロジェクトでは意味情報を三次元形状データと関連付けて DB に蓄積するモデル化手法を提案してきた[11]。それにより、視覚的な情報のみならず、意味情報をも共有することが可能になり、三次元仮想空間内を利用者間で統一された価値基準で、検索や操作が可能になると考えた。ここでモデル化の対象を明確にするために、以下の二つの概念を定義する。

定義 1. 意味

仮想空間に存在する形を持ったオブジェクトや特定の領域の概念で、機能や役割を併せ持ち、一般に名詞で表現する。

例 1: 本棚(しまう、取り出すなどの機能を持つ)

例 2: 休憩スペース(喫煙する、飲食するなどの機能を持つ)

定義 2. 意図

仮想空間でのオブジェクトの移動や形状変化などに対して明示的あるいは暗黙的に含まれる空間の利用者の考えや目的。

例: 椅子の整頓(椅子のオブジェクトの座標移動が利用者の椅子の整頓という意図を持つ)

2.2. 三層構造モデル

空間の意味の共有化を図るためには、空間内に存在するオブジェクトの形状データとの対応が重要となる。既存の Web3D などのファイル形式では、空間の意味は形状データの属性値として表現されており、構造を持った意味の表現が難しい。また、検索を行うときにも文字列マッチングの全検索になってしまうためデータの再利用が非常に困難である。そこで本プロジェクトでは、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[11](図 1)を提案してきた。

意味層では、空間の意味の明確化、体系化をおこなうために、オントロジを用いて意味から必要な概念を抽出し、それを意味層のエンティティとしている。ここで用いるオントロジとして 3 つのオントロジを定義した。

定義 3. 形状オントロジ(Shape Ontology)

形状オントロジとは空間のオブジェクトまたはあ

る特定の領域を指すエンティティであり、以下の属性を持つ。

- 概念名を表すラベル
- 形状の大きさや値段などの属性の型と値域の定義
- 概念定義：自然言語などで定義される概念の詳しい説明

なお、形状オントロジによって定義されるエンティティ間の関係は、Is-a と Part-of がある。

定義 4. 機能オントロジ(Active Ontology)

機能オントロジは空間のオブジェクトや特定の領域を指すエンティティであり、以下の属性を持つ。

- 概念名を表すラベル
- 概念定義：自然言語などで定義される概念の詳しい説明

尚、機能オントロジによって定義されるエンティティ間の関係は、Is-a がある。

さらに形状オントロジと機能オントロジとの間に成り立つ関係を明確にする役割を持つオントロジも定義する。

定義 5. 関係オントロジ(Relation Ontology)

形状オントロジと機能オントロジの間に成り立つ関係を明確にするエンティティであり、EDR 概念辞書 [12]で使われる 8 つの関係子を用いる。関係オントロジは、以下の属性を持つ。

- 概念名を表すラベル
- 概念定義：自然言語などで定義される概念の詳しい説明

以上の 3 つのオントロジによって、必要な意味の要素を明確かつ体系的に定義でき、その組み合わせによって様々な意味表現が可能となる。例えば、「引き出しにしまえる」という意味表現は「形状オントロジ：引き出し」「関係オントロジ：in」「機能オントロジ：しまう」間の関係付けによって表現可能である。更に目的語を伴うような「引き出しに参考書をしまうことができる」という意味表現は、上述の 3 つと「関係オントロジ：objective」「形状オントロジ：参考書」間の関係付けによって可能となる。

形状データ層は、形状データの頂点データ、面ループ、色や形状データ間の相対座標などの階層構造を持つ。これらの形状データは意味を一切持たない。したがって、意味層と形状データ層が分離され独立に管理できる。

存在エンティティ(EE)層に蓄積される存在エンティティとは、オントロジの具体物で、対応するオントロジの概念と Instance-of 関係を持ち、空間の存在を表す。ここでオブジェクトや特定の空間の存在を表すエンティティを定義する。

定義 6. 形状存在エンティティ(Shape Existing Entity)

形状オントロジで定義した概念の具体物であり、その具体物が空間に存在することを示すエンティティである。形状存在エンティティは、所属する形状オントロジの属性値や簡易ポリゴンを持ちうる。

仮想オフィス環境では、利用者が意図を持った操作を行うことによって、その操作が行われていない状態から操作が実行中である状態へ移行する。したがって、

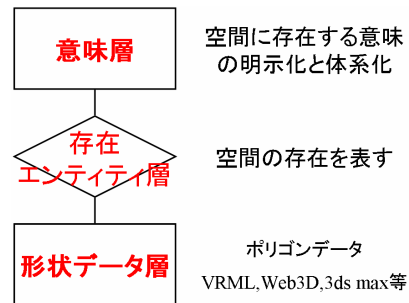


図 1 三層構造モデルの概念図

操作が実行中である状態を表す情報が必要となる。そこで、その情報を機能オントロジと Instance-of 関係となる機能存在エンティティによって表現する。次に機能存在エンティティを定義する。

定義 7. 機能存在エンティティ(Active Existing Entity)

機能オントロジで定義した概念の具体物であり、その存在により空間中である操作が実行中である状態を表すエンティティである。

形状存在エンティティと機能存在エンティティを定義したが、これらの間には所属する形状オントロジと機能オントロジとの間に成り立つ関係と同様の関係を定義できる。これを関係存在エンティティとして次に定義する。

定義 8. 関係存在エンティティ(Relation Existing Entity)

関係オントロジで定義した概念の具体物であり、形状存在エンティティと機能存在エンティティの関係を明確にするエンティティである。

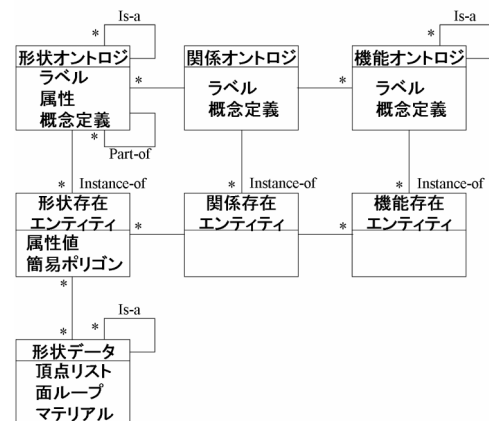


図 2 概念データモデル

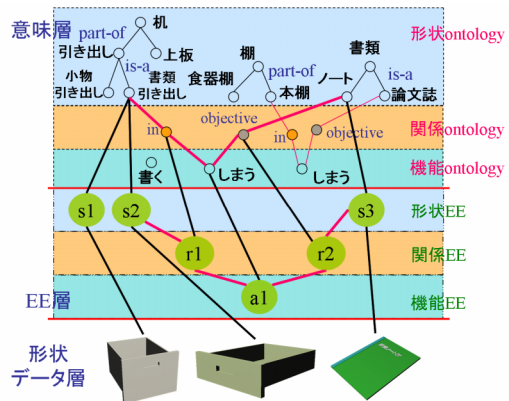


図 3 三層構造モデルの記述例

これまで示した設計手法の概念モデルを図2に示す。図3では、(書類引き出し,in,しまう,objective,ノート)によって「書類引き出しにノートをしまえる」、(本棚,in,しまう,objective,論文誌)によって「本棚に論文誌をしまえる」という空間での意味を表現している。また、(s2,r1,a1,r2,s3)の関係によって「s3がs2にa1の状態(しまわれている)である」という空間で操作が実行中である状態を表現している。

2.3. 意図を持った検索・操作

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[11]。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。本研究ではこのような操作には以下のような二種類があると考え、定義した[13]。

定義 9. 明示的操作

利用者が「この部屋にあるすべての書類をこの書籍棚にしまえ」のように操作の意図を明示的に宣言する操作。仮想オフィス環境では、コミュニティ内で行われる操作の意図は意味情報層で明示的かつ体系的に定義したため、利用者が希望した操作に対し、あらかじめ操作の意図を体系的に表現した意味情報層を用いることで、DB およびコミュニティにその意図に見合った形で反映可能になる。

定義 10. 暗示的操作

利用者がオブジェクトに対して行った行動(例えば、机の上のゴミをゴミ箱の中に移動させる)についてプログラムがその操作の裏にある意図(ゴミを捨てた)を読み取る操作。明示的操作と同様に、暗示的操作でも意味情報層を用いる事でその操作の「意図」をDBに反映可能である。

本論文では、この2種類の意図を持った操作や検索が可能なブラウザを設計・実装を行う。

2.4. 仮想オフィス環境の共有

利用者が仮想オフィス環境を共有する概念図を図4に示す。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」として各利用者が取得する。意味情報付きシーングラフを以下のように定義する。

定義 11. 意味情報付きシーングラフ

意味情報付きシーングラフとは、形状データとオブ

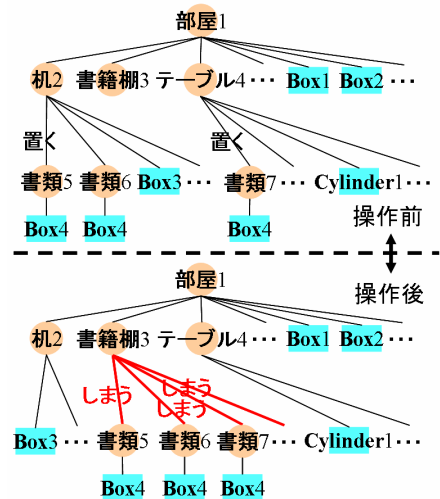


図5 操作実行前後(意味情報付きシーングラフ)

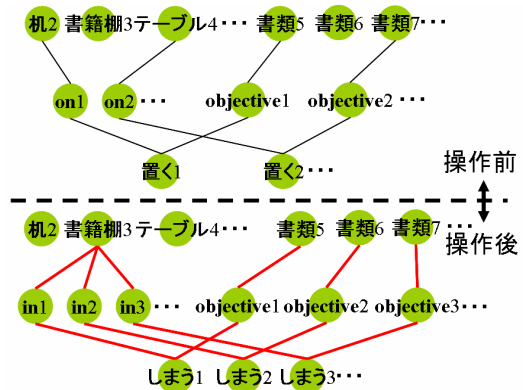


図6 操作実行前後(EE層)

ジェクトに付加する意味情報が記述されたシーングラフであり、図5のような構造を持つ。この図では、丸ノードが形状EE、四角ノードがその形状EEの形状データを表し、枝が形状EE間の相対座標、回転や状態を保持している。さらに各丸ノードはその形状EEの意味にあたる形状オントロジーのラベルと概念定義、そして可能な操作の知識(例えば、形状EEの意味が論文集の場合、論文集は書籍棚にしまうことができるなど)を保持している。このシーングラフをブラウザとDBの中間データとすることは、DBに蓄積されている仮想空間の形状データをCGで扱いやすく、利用者が空間の構造・状態を理解しやすい階層構造に直すために必要である。また、EEのIDを参照することによりDBとシーングラフが一意に対応しているため、DBのビューとしての役割も担っている。これによって、同じ空間でも利用者ごとに見え方を変えるなどの、利用者のパーソライズの実現を可能にしている。

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(図5)をDBに反映し(図6)、変更部分(図5太線の枝と、上下のノードのID)を他の利用者の意味情報付きシーングラフに渡すことで、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。

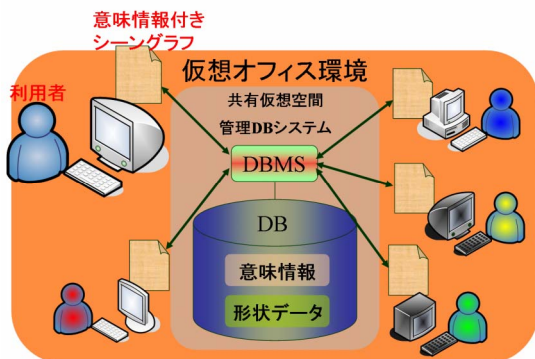


図4 空間の共有

3. 仮想オフィス環境ブラウザの設計

仮想オフィス環境をブラウザに表示し、そのブラウザ上で利用者が意図を持った操作や検索を行うために、本研究ではブラウザには以下のような5つの機能[14]が必要になると考えた。

- 機能1. DB に蓄積した仮想空間を以下のようにして表示する
 - i データを意味情報付きシーングラフに変換する
 - ii そのシーングラフに書かれた形状データを読み込む。
- 機能2. 1.で読み込んだ仮想空間の中をウォークスルーでき、さらにマウスなどでオブジェクトの指定・移動ができる
- 機能3. 利用者の意図を明示的に宣言する操作（明示的操作）や検索を仮想空間に対して（ブラウザ上で）行うことができる
- 機能4. 利用者がオブジェクトに対して行った操作についてプログラムがその裏にある意図を読み取る（暗示的操作）ことができる
- 機能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）

本章では、これらの機能を付加したブラウザを設計するために作成した以下のクラス図(図7)、空間を表

示するまでのシーケンス図(図8)、明示的操作のシーケンス図(図9)を用いて説明する。

3.1. 空間表示の設計

本節では、図8のシーングラフの取得、ブラウザの表示について説明する。2.4.節で述べた理由により、利用者側のブラウザとサーバ側のDBとのやり取りを意味情報付きシーングラフによって行う。まず利用者はDBConnectionクラスのMakeSceneGraphメソッドを呼び出す(図8(i))ことでDBにシーングラフ作成クエリを発行する。その結果として、利用者は意味情報付きシーングラフを取得する。

そして、この意味情報付きシーングラフに書かれている仮想空間をSceneGraphReaderクラスのReadメソッドを用いて(図8(ii))、EEクラスのインスタンスとして取得し、仮想オフィス環境ブラウザのMainBrowserに仮想空間を表示する。EEクラスとは、意味情報付きシーングラフと同様の構造(図3)を持っているクラスで、形状EE(図5丸ノード)が形状データを保持している場合のみ、ObjectDataクラスのインスタンス(図5四角ノード)を保持する。ObjectDataクラスは、そのEEの形状を構成している基本形状(Boxなどの)頂点データ、面ループ、基本形状間の相対座標、回転、マテリアルなどの値を保持している。このEEクラスにより、階層構造を考慮したデータの更新(オブジェクトの移動など)やその更新を意味情報付きシーングラフに反映することが容易

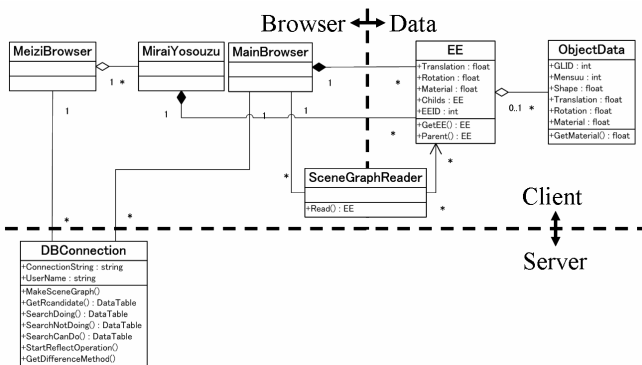


図7 クラス図

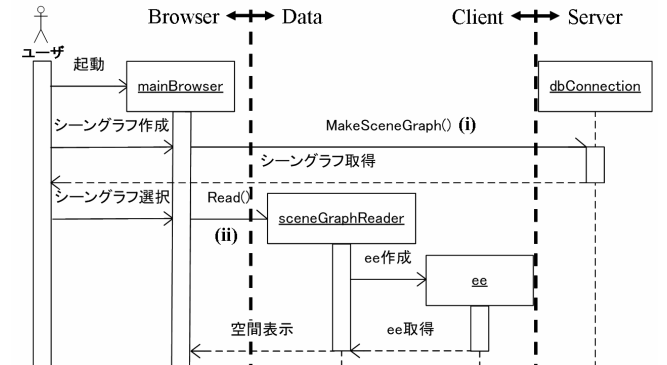


図8 シーケンス図(空間表示)

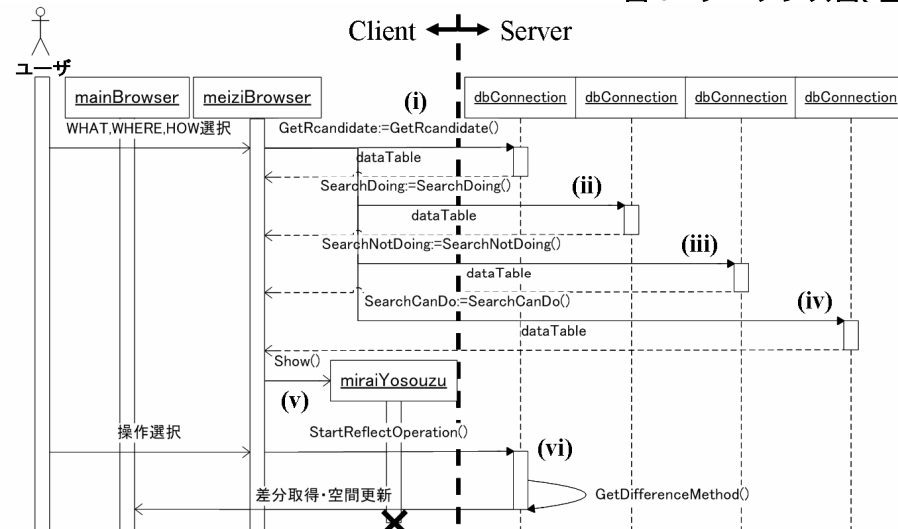


図9 シーケンス図(明示的操作)

になる。

3.2. 明示的操作の設計

本論文では機能 3. にあたる明示的操作 (図 9) について説明する。

明示的操作は以下のようなデータの流れによって可能になると考えた。

- (1) 利用者は, MeiziBrowser で操作 (図 9(i)) か, 3 種類の検索 (現在行われている操作検索 (同(ii)), 行われていない操作検索 (同(iii)), 可能な操作検索 (同(iv))) を選択し, その後に操作の対象 (WHAT), 操作先 (WHERE), 行いたい操作 (HOW) を選択する。もしくは, 仮想オフィス環境ブラウザ上からオブジェクトをピックアップすることで WHAT を選択する。
- (2) それら (WHAT, WHERE, HOW) を用いてそれぞれのクエリに対して DB で検索される
- (3) DB がそのクエリの検索結果 (WHAT の EEID と操作の場合, 操作後の translation, rotation と WHERE の EEID) を返す
- (4) (3) で取得した値を用いて, 操作後の空間や検索結果を視覚化したものである MiraiYosouzu (「未来予想図」) に表示する (図 9(v))
- (5) 操作の場合, MiraiYosouzu の操作の候補の中から操作を行いたい WHAT を選択 (ピック) する
- (6) (5) で選択された値を用いて MainBrowser とシーングラフ, DB が更新される (図 9(vi))

明示的操作の例として, 「机の上のすべてのモノをその机の引き出しにしまえ」などが挙げられる。DB への更新操作を簡便にするために, ブラウザとのデータのやり取りは 2.4 節で述べたような変更部分 (操作の対象 (What) と操作先 (Where) の EE の ID とそれらの間の translation, rotation) のみとした。

3.3. 差分取得方法の設計

2.4 節で述べたようにある利用者が行った DB の更新を伴う操作を受けて, 更新されたデータ (操作の対象 (What) と操作先 (Where) の EE の ID とそれらの間の translation, rotation) を差分として他の利用者に渡すことで複数利用者による仮想空間の共有 (マルチクライアント) を実現する。

4. プロトタイプシステムの実装

4.1. DB の実装

2.2 節で述べた図 3 のような概念データモデルに基づき, かつ機能 1. ~ 機能 5. の機能が実現可能なように設計した論理データモデルを図 10 に示す。この論理データモデルをスキーマとして, RDBMS 上に実装した。実験用 DB サーバの実装環境を以下に示す。

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

4.2. 仮想オフィス環境ブラウザの実装

以上に述べた方針に基づいて, 明示的操作と暗示的操作を実現できる仮想空間表示器である「仮想オフィス環境ブラウザ」を実装した。

実装の手段として, 意味情報付きシーングラフの記述には表現能力が非常に高く, API が充実している

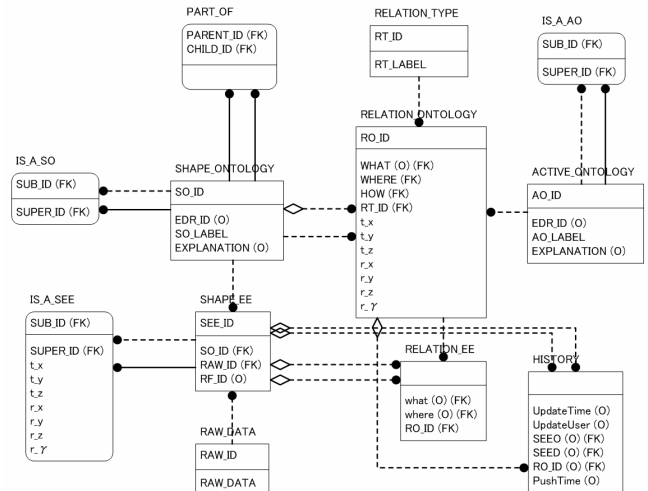


図 10 論理データモデル

```
<?xml version="1.0" ?>
- <world>
- <subject>
- <object>
  <id level="1">部屋1</id>
  <EE_ID>1</EE_ID>
- <edge>
  <translation>0 0 0</translation>
  <rotation>0 0 0</rotation>
  <state>0</state>
</edge>
- <ontology>
- <shape_ontology>
  <label>部屋</label>
  <explanation>主として研究活動を行う場所</explanation>
</shape_ontology>
- <relation_ontology />
</ontology>
- <subject>
- <object>
  <id level="2">論文集178</id>
  <EE_ID>178</EE_ID>
- <edge>
  <translation>-367.1 -129 -336</translation>
  <rotation>0 -1 0 -1.197</rotation>
  <state>0</state>
</edge>
- <ontology>
- <shape_ontology>
  <label>論文集</label>
  <explanation>論文一覧が掲載されている書籍</explanation>
</shape_ontology>
- <relation_ontology>
  <WHAT_WHERE_HOW>論文集 書籍欄にしまう</WHAT_WHERE_HOW>
  <WHAT_WHERE_HOW>論文集 白机の上で読む</WHAT_WHERE_HOW>
  <WHAT_WHERE_HOW>論文集 白机の上に置く</WHAT_WHERE_HOW>
</relation_ontology>
</ontology>
- <subject>
- <object>
  <id>Box52</id>
- <shape>
  <vertex_list>-10 0 1.5,-10 0 -1.5,10 0 -1.5</vertex_list>
  <vertex_list>10 0 -1.5,10 0 1.5,-10 0 1.5</vertex_list>
  :
  <vertex_list>-10 24 1.5,-10 24 -1.5,-10 0 -1.5</vertex_list>
</shape>
- <transform>
  <translation>0 0 0</translation>
  <rotation>0 0 0</rotation>
</transform>
<material>0.8863 0.898 0.898</material>
</object>
  :
```

図 11 意味情報付きシーングラフ

XML を用いた。また, シーングラフから仮想空間を描画するためのグラフィック API として, OpenGL を用いた。これは, OpenGL がプラットフォームに依存せず, かつグラフィックスハードウェアにアクセスするプログラムを書くことができ, さらに機能 2. にあたるブラウザ上でのオブジェクトの指定 (ピッキング) に適した機能も持っているためである。仮想オフィス環境ブラウザの作成には C#を用いた。また, シーングラフの記述には XML を用いているため, XML の読み取りには XML の階層構造も考慮できる DOM を用いて形状データを取得した。実際の意味情報付きシーングラフを図 11 に示す。そして, このブラウザの動

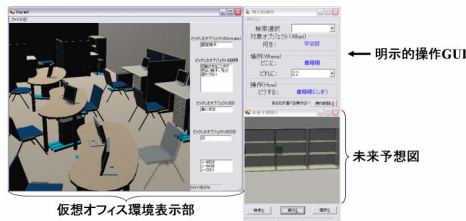


図 12 プロトタイプシステム

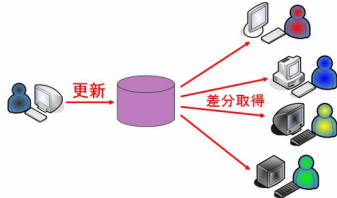


図 13 評価方法

作画面を図 12 に示す。

4.3. 差分取得方法の実装

他の利用者のブラウザに更新を反映するに当たり、次のような方針で評価を行う。

- 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は短い方が良い
- 更新反映の確実性が高いほうが良い
- DB への負荷は少ない方が良い

以上の考えから以下の2つの差分取得方法を実装した。これらを比較し、どちらが仮想オフィス環境ブラウザに適しているかを評価する。

- A) Pull 型差分取得：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
 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
- B) Push 型差分転送：UDP を用いた Broadcast（ネットワーク内で不特定多数の相手に向けてデータを送信すること）による差分送信
 メリット：他の利用者への反映時間が短く、DB への負荷が少ない
 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（データ不整合）の可能性あり

5. プロトタイプシステムの評価

本章では、実装したプロトタイプシステム（図 11）を用いて複数の利用者で一つの空間を共有したときの評価を行う。

利用者は DB の更新が起きた場合、現在表示されている空間と 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、3.3 節で挙げた 2 種類の差分取得方法を用いて、1 人の利用者が更新を行い、他の使用者がその差分を受け取る（図 13）とき、一回の更新で取得する差分タプル数（1 タプル：図 5 太線の枝（float 型の変数 7 つ）と、上下のノードの ID（int 型の変数 2 つ））を変化させることで、どちらの差分取得方法が仮想オフィス環境ブラウザに適しているかを評価した。利用者数は 10 人、測定対象は 5.1 節で用いたオフィスの仮想空間で、全 EE 数が 1000 個、一回の操作で影響を受ける最大の EE 数が 250 個とした。

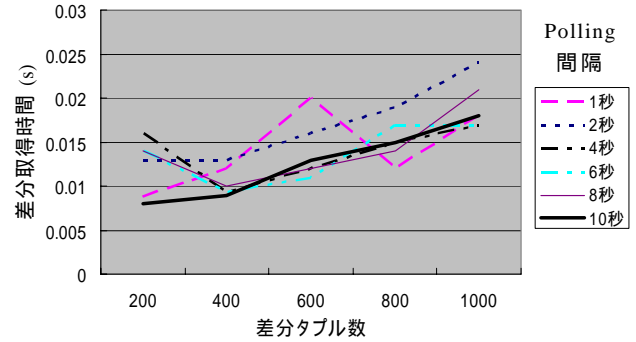


図 14 Polling 実験

- (ア) Polling にて差分を取得する場合、Polling 間隔を固定する必要がある。そこでまず適切な Polling 間隔を求める。Polling 間隔を 1～10 秒で変化させたときの差分タプル数-差分更新時間グラフは図 14 のようになる。また、最大差分タプル数は空間の全 EE 数と等しくなる。これは、Polling 間隔の間に同じ EE に対して複数の操作が行われた場合、最新の値のみを取得するためである。この図より利用者が 10 人程度では、Polling 間隔を変化させてもあまり違いは見られない。本来なら Polling 間隔は短いほうが望ましいが、Polling はブラウザ側が動作を管理するため、あまりに間隔が短すぎると DB だけでなく、ブラウザにも負荷がかかりすぎてしまい、動作が重くなるなどの支障が出てくる。よって、Polling 間隔はブラウザの動作に支障が出ない 10 秒（図 14 太線）が最適であると考えられる。また、利用者数 10 人のとき差分タプル数をさらに変更させることで、このシステムの限界も評価した。Polling 間隔 10 秒のとき、Polling の限界実験での差分タプル数-差分更新時間グラフは図 16 のようになる。この図より、差分タプル数が 40000 までではほぼ比例して差分取得時間が増えていくが、40000 を超えると差分取得時間が急に増大し始める。これは利用者 1 人あたりの差分取得にかかる時間が大きくなりすぎて、他の利用者のトランザクションをロックする回数が増えるためであると考えられる。よって、Polling による差分取得での差分タプル数の限界は 40000 タプルであると考えられる。
- (イ) Broadcast による差分送信の方法をとった場合、最大取得タプル数は 250 になる。これは、この差分取得方法では DB の更新が起こるたびに差分を取得するため、一回の操作で影響を受ける最大の EE 数が最大差分タプル数と等しくなるからである。この方法の場合、パケットを損失する場合があるため、伝送路が混んでいる場合（320kbps のストリーミング放送を流した）でも実験を行ったが、差分タプル数が 250 程度ではパケット損失は起きなかった。また、UDP 伝播時間も差分タプル数 250 のとき、図 15 より 0.32 秒程度のため問題はないと考えられる。次に、Broadcast による差分取得での差分タプル数の限界は図 15 より、1819 であることがわかる。これは、一回の Broadcast で送ることが可能なデータの最大量がヘッダを含めて 65536Byte で固定されているため、

1 タブル 36Byte である今回の実験では差分タプル数 1819 が限界であると考えられる。

今回の実験結果より、より実際の利用に近い環境で、オブジェクト数が 1000、利用者が 10 人程度の規模が小さい空間の場合、ブラウザや DB に負荷がかからず、よりリアルタイムで空間を共有できる(イ)の Broadcast による差分送信が最も適していると考えられる。しかし、もっと大規模な仮想空間を大人数の利用者で共有する場合、(イ)の方法と、確実に差分を取得できる(ア)の Polling を組み合わせた方法が最も適していると考えられる。

本論文では、図 13 のように更新を起こす利用者を 1 人に固定して実験を行ったが、複数の利用者が同時に更新を起こそうとした場合、以下のような問題が生じると考えられる。現在は DBMS の同時実行制御を用いて制御しているので、一瞬でも早くクエリを発行した利用者のトランザクションを先に処理し、その間はその他の利用者のトランザクションにはロックがかけられる。現状のシステム設計では、利用者が操作を行う際には、まず操作の候補を視覚化したものである「未来予想図」を表示する。この候補の中から利用者は結果として反映させたい操作を手動で選択する。このため、ある利用者が未来予想図を表示している間に他の利用者によって操作を行いたいオブジェクトに対して先に操作が行われてしまった場合、思い通りに操作が更新されない場合が生じる。この問題の解決方法として、未来予想図を表示している間からロックをかけてしまう方法が考えられる。しかし、この方法だと選択プロセスが手動であるため、ロックがかけられる時間が非常に長くなり、他の利用者が操作を行うときの待ち時間が長くなってしまいうという欠点が出てきてしまう。これらの方法は一長一短なので、これらの評価は今後の課題である。

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、意図を持った操作、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、本論文では意味をキーに用いた検索や、利用者の意図を表す操作が可能な仮想オフィス環境ブラウザの設計・実装方法を示した。また、このブラウザを利用した複数利用者での仮想空間共有の評価を行い、ブラウザの有効性を示した。複数の利用者が同時に更新を行ったときのロックの方法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 14780197)の支援による。

参考文献

- [1] 松田晃一, 上野比呂至, 三宅貴浩, “パーソナルエージェント指向の仮想社会「PAW」の評価”, 電子情報通信学会論文誌 D-II, Vol.J82-D- , No.10, pp.1675-1683, 1999 年 10 月
- [2] Masunaga Y. and Watanabe C., “Design and Implementation of a Multi-modal User Interface of the Virtual World Database

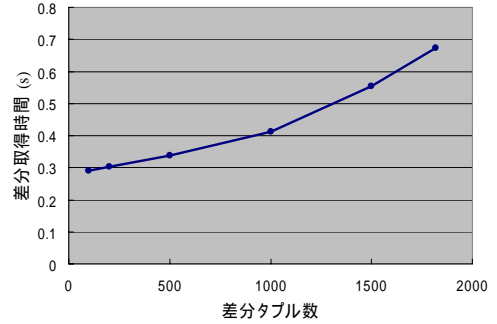


図 15 Broadcast 実験

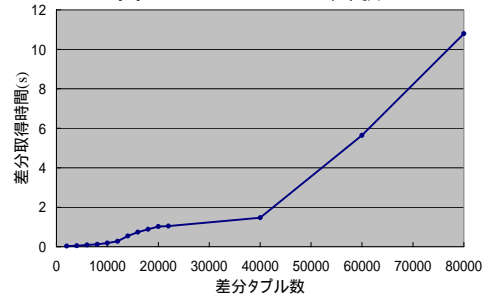


図 16 Polling 限界実験

System(VWDB)”, International Conference on Database Systems for Advanced Application(DASFAA)(2001)

- [3] 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. 44, No. SIG 8(TOD 18), 2003.
- [4] 田島敬史, 依田和也, 田中克己, “複数詳細度を持つ CG データのためのアクセス制御とアクセス権管理”, 信学論, J-82-D-1, pp.193-200, 1999 年 1 月
- [5] H. Arisawa, T. Sato, “Human-Body Motion Simulation Using Bone-Based Human Model and Posture Database”, 4th International Scientific Conference on Prevention of Work-Related Musculoskeletal Disorders (PREMUS 2001), pp.128, Amsterdam, the Netherlands, Oct. 2001.
- [6] T. Tomii, K. Salev, S. Imai, H. Arisawa, “HumanModelling and Design of Spatio-Temporal Queries on 3D Video Database”, Visual Database Systems 4(VDB4), Y. Ioannidis & W. Klas (Eds.), IFIP, Published by Chapman & Hall, pp.317-336, 1998.
- [7] 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, No6, Nov. 1999
- [8] C. Chen, L. Thomas, J. Cole, C. Chennawasin, “Representing the Semantics of Virtual Spaces”, IEEE Multimedia, April-June 1999, pp. 54-63.
- [9] J. F. Sowa, “Knowledge Representation: Logical, Philosophical, and Computational Foundations”, Brooks/Cole, 2000.
- [10] 松尾 豊, 高木 朗, 平塚 誠良, 橋田 浩一, 中島 秀之, “空間の意味表現と空間機能検索”, セマンティックウェブとオントロジー研究会資料, SIG-SWO-A303-01, 2004.
- [11] 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.
- [12] 通信総合研究所, EDR 概念辞書, <http://www.crl.go.jp/overview/index-J.html>
- [13] 高木良成, 岡田直也, 竹島広人, 谷岡洵, 富井尚志, “DB で管理された 3 次元仮想コミュニティにおける意図付操作モデルの導入”, 信学技報, Vol.103, No.356, DE2003-111, pp.1-6, Oct. 2003.
- [14] 賀来健一, 富井尚志, “高度コミュニティ空間のためのインタラクティブなブラウザの実装”, 情報科学技術レターズ(FIT2004), vol.3, pp.63-66, 2004.