

共有ゴーストオブジェクトによる 仮想世界同期法の提案とその評価

Evaluation of a Synchronization Method of the Virtual World using Shared Neighboring Ghost Objects in VWDB2: A Virtual World Database System

渡辺 知恵美[▼] 増永 良文[▲]
Chiemi WATANABE Yoshifumi MASUNAGA

我々が開発を行っている仮想世界データベースシステム第2版(VWDB2)では、仮想共同作業支援の視点からデータベース機能を備えたネットワークバーチャルリアリティシステムの実現を目指している。本論文では、仮想環境の同期性を保障するための仮想世界同期法を提案し、その有効性を検証する。VWDB2ではサーバによる集中管理方式を採用しているため、クライアント間の同期には限界があり、高い同期性を要求する場合の問題となる。そこで、我々は「共有ゴーストオブジェクト」という仮想世界同期法を新たに導入した。本方式ではサーバを介したクライアント間同期を行うと同時に、仮想世界で近隣にあるオブジェクトを操作しているクライアント同士でP2P通信を結び互いのオブジェクトの同期を行う。それによりサーバ-クライアント同期の役割と、クライアント間の同期の役割を分離することができ、その結果サーバ-クライアント間の同期間隔に関わらずクライアント間で一定に高い同期性を保証することができる。本同期法の有効性を実験によっても確認した。

VWDB2 is a network virtual reality (NVR) system with a database function that is realized by system-integrating a set of virtual reality systems and a single back-end database system. Since all transactions are issued to the back-end database system, an efficiency problem arises when high synchronism is required in the VWDB2 virtual work environment. In order to resolve this problem, the shared neighboring ghost objects are introduced that makes it possible for clients to do P2P communication each other. A novel synchronization mechanism is implemented on the VWDB2 based on this approach. By experiments, it is shown that the approach ensures both high reliability and high synchronism for VWDB2 clients, which are known as the essential features for realizing an efficient NVR-based shared work environment.

1. はじめに

我々は、VRシステムとDBシステムをシステム連携したVirtual World Database System (VWDB)の開発を進めてきた[1]が、最新のバージョンであるVWDB Version 2(VWDB2)では複数台のVRシステムと一台のデータベースシステムを仮想世界集中管理方式[2]で実現し、仮想共同作業環境を提供

▼ 学生会員 お茶の水女子大学大学院人間文化研究科博士
後期課程 chiemi@dbl-lab.is.ocha.ac.jp
▲ 正会員 お茶の水女子大学理学部情報科学科
masunaga@is.ocha.ac.jp

している(図1)。そのような構成のVWDB2は同じ世界に没入したユーザが共有化された仮想世界オブジェクトを共同で編集出来る仮想共同作業環境の信頼性(共有された世界の状態が矛盾しないこと)および同期性(各クライアントで表示される仮想世界が頻りに同期されること)を高く保障することが必要である。先行研究[1]では信頼性を保障するために仮想共同作業環境のためのトランザクションモデルを導入した。

本論文では、仮想共同作業環境の同期性も併せて保障するために「共有ゴーストオブジェクト」を導入した新しい仮想世界同期法とその有効性を示す。実験の結果、サーバとクライアント間の同期間隔の長さに関わらず、共有ゴーストオブジェクトは近隣クライアント間での高い同期性を実現できることが明らかとなった。以下、第2章で提案した共有ゴーストオブジェクトの概要を述べ、第3章でその有効性を検証する。

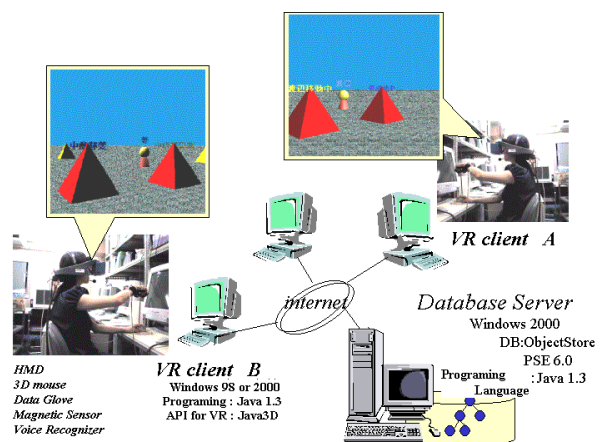


図1 VWDB2 のシステム構成

Fig.1 VWDB2 System Configuration

2. 仮想世界同期法

2.1 共有ゴーストオブジェクト

共有ゴーストオブジェクトはトランザクションの実行時、オブジェクトの状態を表示する半透明のオブジェクトである。移動トランザクションにおけるゴーストオブジェクトの例を図2に示す。なお移動トランザクションのように比較的長時間対話的に実行されるトランザクションは仮想世界における特徴的なトランザクションであり我々はこれを連続操作トランザクションと呼んでいる。連続操作トランザクションは操作の開始から終了までをひとつのトップトランザクションとし、それが複数のサブトランザクションで構成される2段階のトランザクションとなる[1]。オブジェクト O_A の移動トランザクション T_A を開始すると、ゴーストオブジェクト O_A' が生成される。ユーザは基本的にこのゴーストオブジェクト O_A' を移動させることになる。時刻 t_0 にサブトランザクション S_1 が発行された後、位置更新サブトランザクション S_2 が実行されるまでの間、ユーザはゴーストオブジェクト O_A' の位置更新操作を実行する。サブトランザクションが実行されると、その時点でゴーストオブジェクトがある位置にオリジナルオブジェクトを移動させるよう位置更新サブトランザクションが発行され、オリジナルオブジェクトの位置更新が行われる。

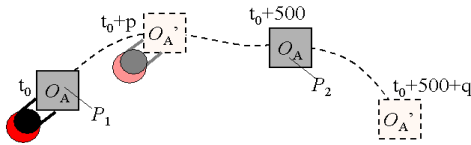


図2. ゴーストオブジェクト
Fig.2 Ghost Object

2.2 共有ゴーストオブジェクト

次にゴーストオブジェクトの共有による仮想世界同期法について述べる。基本的にはサーバを中心とした仮想世界集中管理方式を採用する。しかし、連続操作トランザクションで操作されている各オブジェクトに対し、その位置の近隣(後の実験では半径7m以内としている)にあるオブジェクトを検索し、近隣にあるオブジェクトを操作しているクライアント間で P2P による通信を行い、互いのゴーストオブジェクトを共有させる。例えば、クライアント C_A, C_B, C_C にてオブジェクト O_A, O_B, O_C の移動トランザクションを実行し、現在、データベース上の仮想世界が図 3(a)に示す状態であるとする。このとき、 O_A と O_B, O_A と O_C がそれぞれ近隣となり、 O_A の移動トランザクション T_A を発行しているクライアント C_A と、 O_B の移動トランザクション T_B を発行しているクライアント C_B との間に P2P のネットワーク通信路を確保して、 O_A のゴーストオブジェクト O_A' と O_B のゴーストオブジェクト O_B' を共有する(図 3(b))。なお、P2P 通信で共有するゴーストオブジェクトはあくまでオリジナルオブジェクトに対する仮の姿なので、オリジナルオブジェクトに比べて一貫性を厳密に確保する必要性は低いため、楽観的な一貫性管理とする。

また、クライアント C_A と C_B がゴーストオブジェクトを共有している場合に起こりうる障害には以下の 2 種類があげられる。

- (1) C_A または C_B で実行中のトランザクションがアボートする。
- (2) C_A と C_B の P2P 接続が途切れる。

(1) が起きた場合、オリジナルオブジェクトは VWDB トランザクションモデル[1]に従い、アボート直前にコミットされたサブトランザクションまでが有効となる。しかし、ゴーストオブジェクトはあくまで非永続的なオブジェクトであることからアボート直前に表示されていたゴーストオブジェクトの位置は有効にならず破棄される。また(2)が起きた場合は再接続する際にサーバ側に通知を行って C_A と C_B が操作しているオブジェクトが近隣にあるかどうかを再確認し、近隣にあればゴーストオブジェクトの共有を再開する。

2.3 共有ゴーストオブジェクト導入の意義

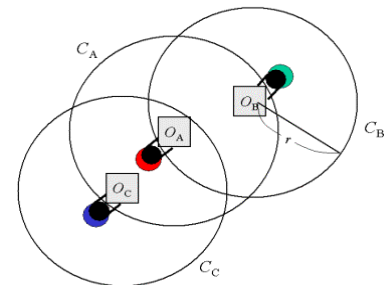
ここで、共有ゴーストオブジェクトを用いた仮想世界同期法の意義を明確にする。同期性とは仮想共同作業環境における同期の頻度であり、仮想共同作業環境における同期は以下の 2 種類に分けられる。

- (1) **クライアント間の同期**
複数の VR クライアント間での仮想世界の同期の頻度が高いほど、クライアント間の同期性が高くなる。仮想共同作業環境ではクライアント間の同期性が高いほど他のユーザの行っている操作や仮想共同作業環境の最新の状況をリアルタイムに確認することができ、共同作業をスムーズに行うことができる。
- (2) **クライアント-サーバ間の同期**
VR クライアント上に表示される仮想世界と、バックエンド DB 内に格納されている仮想世界との同期の頻度が高いほどクライアント-サーバ間の同期性は高

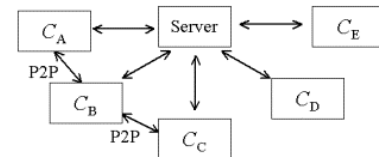
くなる。クライアントから発行されるトランザクションをサーバ側で処理しコミットすることによって同期が行われるため、クライアント-サーバ間の同期性が高いということは、クライアントで行われる操作が即座にデータベースでコミットされており、クライアント上に表示される仮想世界の信頼性が高い状態となる。

つまり、クライアント間の同期はユーザ同士の共同作業をスムーズにする役割を持ち、クライアント-サーバ間の同期はクライアント上の仮想世界の信頼性を高める。

共有ゴーストオブジェクトによる仮想世界同期法はこれら 2 種類の同期の役割を分離させ、また双方を表示することによって各々の同期の役割をユーザに意識させることに特徴がある。クライアント間の同期を共有ゴーストオブジェクトで表し、近隣のユーザ間での共同作業を行う際ユーザは共有ゴーストオブジェクトを視認する。またクライアント-サーバ間の同期をオリジナルオブジェクトで表し、ユーザは定期的にオリジナルオブジェクトが更新される様子を見ることによって、自らのおこなったトランザクションがコミットされたこと、現在のオリジナルオブジェクトのある位置が一貫性を持つ仮想世界オブジェクトの状態であること、またこの状態が他の全てのクライアントで表示されていることを確認することができる。



(a) 共有ゴーストオブジェクトを用いる例



(b) (a)の場合に行われる通信

図3. 近隣でのゴーストオブジェクトの共有
Fig.3 Sharing Ghost Objects in Neighbor

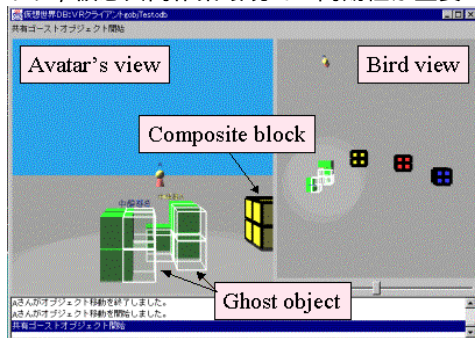
3. 共有ゴーストオブジェクトの有効性の評価

3.1 評価用アプリケーション

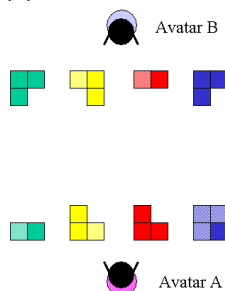
共有ゴーストオブジェクトによる仮想世界同期法の有効性を評価するために、これまで構築してきた VWDB2 プロトタイプシステム上に次のアプリケーションを実装した。

図 4(a)はアプリケーションのスクリーンイメージである。仮想世界は図 4(b)に示すように 4 タイプのブロックが 2 個ずつ計 8 個並べられている。この世界に二人のユーザが 2 列に並べられたブロックをはさむようにして向かい合い、同じタイプのブロックを合成していく。但し合成する際両方のブロックは誰かによって移動中でなければならぬ。このようにして最右の青いブロックから順にブロックを合成し 4 タイプ

全てのブロックを合成できたら作業完了となる。この評価用アプリケーションでは、仮想共同作業環境で2つ1組となるオブジェクトをピタリと合成させるという細かい作業を行う際に、お互いのオブジェクトのリアルタイムな状況を知る必要があり、仮想共同作業環境での同期性が重要となる。



(a)スクリーンイメージ



(b) 仮想世界を上から見た様子
図4. 評価用アプリケーション

Fig.4 Application for Evaluation

表 1. 主観的評価基準

Table 1 Subjective Evaluation Criteria

ランク	評価
5	とてもスムーズに全てのブロックを合成させることが出来た。
4	それほど苦勞せず全てのブロックを合成させることが出来た。
3	時々困難を生じるときがあったが、全てのブロックを合成させることが出来た。
2	ブロックを合成させるのが困難であった。
1	全てのブロックを合成させることが出来なかった

3.2 有効性の評価

前節で述べたアプリケーションを用いて今回提案した共有ゴーストオブジェクトの有効性の評価実験を行った。実験検証はゴーストオブジェクトを使用する場合としない場合、且つ4種類のクライアント-サーバ間の同期間隔(1000ミリ秒, 3000ミリ秒, 5000ミリ秒, 8000ミリ秒)の計8回行い、作業が完了するまでの時間を計測した。また各参加者に「他のブロックとスムーズに合成させることが出来たか」について表1にあらわす5段階評価でアンケートをとった。参加者は20名で、2人1組で10グループ(チームA-J)を組んでこれらの作業を試行した。なお、作業時間には慣れの問題が影響すると予想されたため、作業者20名ともにゴーストオブジェクトを使用する場合と使用しない場合とで1回ずつ練習をしたのち実験を行った。また同期間隔をユーザに告げず、計8回の試行をランダムに実行した。この実験では、データベースサーバにPC(Pentium III 850MHz, windows2000)を、クライアントにはPC(Pentium III 700MHz, windows2000 : 全て同スペック)を利用し、Ethernet(100BASE-T)で接続した。共有オブジェクト同士の通

信間隔は一定とし200ミリ秒とした。また、共有ゴーストオブジェクトを使用する場合は「近隣」とみなされる範囲を「オブジェクトの中心座標から半径7メートル以内」とし、範囲の広さを固定した。各試行におけるアンケートの結果を表2に、各試行にかかった時間を表3に示す。この結果を用いて、我々は以下の観点から共有ゴーストオブジェクトの有効性に関する考察を行った。

考察1: 共同作業における共有ゴーストオブジェクトの有効性

まず共同作業の際にユーザにとって共有ゴーストオブジェクトが有効に機能したかについて評価を行う。共有ゴーストオブジェクトの特徴はクライアント間の同期とクライアント-サーバの同期との役割分担を明確にすることにある。クライアント間の同期間隔は一定に200ミリ秒としていることから、オブジェクトの合成作業の際ユーザがゴーストオブジェクトの役割を理解し共有ゴーストオブジェクトを視認しながら作業をすれば、サーバ-クライアント間のP2Pによる同期間隔の長さにかかわらず作業時間及び評価は一定になるはずである。

表2を見ると、共有ゴーストオブジェクトを使用した場合の評価点は3.50から4.57であり、ゴーストオブジェクトを使用しない場合の評価点である2.71から4.43より全体的に評価が高いことが分かる。またゴーストオブジェクトを使用しない場合は、サーバ-クライアント間の同期間隔が長くなるほど評価が悪くなるのに対し、ゴーストオブジェクトを使用した場合は5000ミリ秒までは同期間隔が長くなるにつれて評価が良くなっている。また、表3をみると、ゲームにかかる時間に関しても、完了までにかかる時間は5000ミリ秒まで同期間隔が長くなるにつれ短縮されている。この原因としては、同期間隔が長くなるにつれてゴーストオブジェクトとオリジナルオブジェクトの間隔が適度に開くことによって合成作業がしやすかったことと、同期間隔が長くなるにつれてサーバへのアクセス集中によるボトルネックが解消されたため作業がしやすくなったことが挙げられる。

これらの結果から仮想共同作業環境における共同作業の際、ユーザにとって共有ゴーストオブジェクトが有効に機能していたことが分かった。

考察2: 適切なクライアント-サーバ間の同期間隔

考察1により、共有ゴーストオブジェクトを利用すれば、仮想共同作業環境での共同作業の際、ユーザは共有ゴーストオブジェクトを視認することでクライアント-サーバ間の同期間隔の長さ依存せずに作業を行うことができることが分かった。そこで、このアプリケーションにおけるクライアント-サーバの適切な同期間隔について考察する。

クライアント-サーバの同期間隔が長くなるにつれてシステム障害時のロールバックが問題となる。例えばクライアント-サーバ間の同期を5000ミリ秒とした場合、ゴーストオブジェクトは非永続的なものであることからトランザクションがアボートされる際アボート直前のゴーストオブジェクトの位置は保存されず、最悪の場合5000ミリ秒前の状態に戻るようになってしまう。そのためアボート時のゴーストオブジェクトの消失によってユーザの作業に困難が生じない程度の同期間隔を取る必要がある。この場合適切な同期間隔の長さはアプリケーションやユーザが頻繁に行う操作によって大きく異なると考えられ、データベース設計者または利用者が自由にできるように設定できるようにする必要がある。

ここでは評価用アプリケーションにおける適切な同期間隔を求める。本アプリケーションではロールバックが起きた場合でもユーザが違和感なく操作を再開できる距離を2.0メートル¹とし、サブランザクションが発行されてから次のサブランザクションが発行されるまでのオブジェクトの移動距離が2.0メートル以内である割合がゲームが終了するまでに発行されるサブランザクションの2/3以上であった場合その同期間隔は適切であるとする。表4はTeam I(ユーザqとユーザr)のゴーストオブジェクトを利用した場合の試行で採取した移動操作ランザクションのサブランザクション間の移動距離平均と、移動距離が2.0メートル以内である割合である。この場合5000ミリ秒の場合はユーザq,ユーザrともに移動距離が2.0メートル以内である割合が2/3を割っている。また平均でもユーザqは移動距離平均が2.0メートルを超えている。これらの結果より、本アプリケーションでの適切な同期間隔を3000ミリ秒とした。

なお考察1で表2と表3を見た結果ではゲーム終了までの作業時間ももっとも短く評価が良かったのは5000ミリ秒であったが、クライアント-サーバ間の発行間隔として最も考慮すべきことはシステム障害時のロールバックに対応できる同期間隔であると考え、3000ミリ秒を最適な同期間隔とした。

4. まとめと今後の課題

VWDB2上に実装した仮想共同作業環境に対して、高い信頼性および同期性を提供するために共有ゴーストオブジェクトを導入し、実験によってその有効性を示した。

今後の課題として、共有された仮想世界で複数人が同じオブジェクトを操作したい場合などの“協調”作業に対応した同時実行制御の定義を考えている。また、仮想共同作業環境における実用的なアプリケーションを実装する予定である。

[文献]

- [1]渡辺知恵美, 大杉あゆみ, 佐藤こず恵, 増永良文: “仮想世界データベースシステムにおける共有型作業環境のためのランザクション概念の導入”, 情報処理学会論文誌: データベース (TOD), Vol.43, No.SIG9, pp.55-67 (2002).
- [2] Meehan, M.: “Survey of Multi-User Distributed Virtual Environment”, in Course Notes: Developing Shared Virtual Environments, ACM Press (1999).

渡辺 知恵美 Chiemi WATANABE

お茶の水女子大学大学院人間文化研究科博士後期課程在学中。2000年お茶の水女子大学大学院人間文化研究科博士前期課程修了。仮想世界データベースシステムの研究・開発に従事。情報処理学会学生会員。日本データベース学会学生会員。

増永 良文 Yoshifumi MASUNAGA

お茶の水女子大学理学部情報科学科教授。1970東北大学大学院工学研究科電気及通信工学専攻博士課程修了,工学博士。情報処理学会データベースシステム研究会主査,ACM SIGMOD日本支部長,情報処理学会監事などを歴任。情報処理学会フェロー。電子情報通信学会フェロー。日本データベース学会副会長。IEEE-CS,ACM各会員。著書に「リレーショナルデータベースの基礎 データモデル編」(オーム社),「リレーショナルデータベース入門[新訂版]」(サイエンス社)など。

¹ Java3Dでは仮想空間での長さをメートル単位で表現している

表 2. アンケート結果

Table 2 Questionnaire Results

共有ゴーストオブジェクトの使用	使用する				使用しない			
	1000	3000	5000	8000	1000	3000	5000	8000
同期間隔(ms)	1000	3000	5000	8000	1000	3000	5000	8000
a(TeamA)	3	4	4	4	4	2	3	2
b(TeamA)	4	4	5	4	4	3	3	3
c(TeamB)	4	5	5	5	4	3	3	3
d(TeamB)	4	4	4	4	4	3	3	2
e(TeamC)	5	5	5	4	4	3	3	2
f(TeamC)	4	5	5	4	3	3	3	2
g(TeamD)	5	5	5	4	5	3	4	2
h(TeamD)	5	4	5	4	5	3	4	4
i(TeamE)	5	5	4	4	5	5	3	4
j(TeamE)	3	5	5	3	5	4	4	2
k(TeamF)	5	3	5	4	5	3	4	4
l(TeamF)	4	3	2	4	4	3	3	2
m(TeamG)	4	4	5	3	4	4	3	2
n(TeamG)	3	5	5	3	4	4	4	3
o(TeamH)	3	3	4	3	2	2	2	2
p(TeamH)	3	4	5	3	4	3	2	2
q(TeamI)	5	4	5	4	4	4	2	2
r(TeamI)	4	4	4	4	5	3	3	3
s(TeamJ)	4	4	5	3	5	3	3	2
t(TeamJ)	4	5	5	3	5	4	3	3
平均	4.07	4.14	4.57	3.50	4.43	3.43	3.14	2.71

表 3. ゲーム終了までにかかった時間

Table 3 Time to complete a game

共有ゴーストオブジェクトの使用	使用する				使用しない			
	1000	3000	5000	8000	1000	3000	5000	8000
同期間隔(ms)	1000	3000	5000	8000	1000	3000	5000	8000
Team A	126	106	120	120	166	140	180	180
Team B	150	147	120	115	157	177	165	165
Team C	160	156	161	220	154	155	140	168
Team D	119	115	103	140	94	116	120	140
Team E	110	113	98	152	108	110	160	189
Team F	133	122	100	128	98	110	116	120
Team G	113	102	82	110	106	120	126	150
Team H	129	110	106	190	110	128	209	189
Team I	112	100	94	120	101	120	160	160
Team J	148	132	125	189	113	213	204	220
平均	123.4	113.4	101.1	147.0	104.3	131.0	156.4	171.0

表 4. Team I の試行におけるサブランザクションのオブジェクト移動距離平均と2.0メートルを超えている割合

Table 4 Experiment Result for Team I

同期間隔 (ms)		1000	3000	5000	8000
user q	移動距離平均(m)	0.48	1.75	2.12	5.4
	2.m以内の割合(%)	96.9	73.7	49.9	39.6
user r	移動距離平均(m)	0.36	1.09	1.16	5.2
	2.m以内の割合(%)	100.0	79.4	64.4	36.8