

OpenSim を用いた分散型メタバースサーバにおける レスポンス性能に関する評価

理学専攻 情報科学コース 松原 麻佑 (指導教員：小口 正人)

1 はじめに

メタバースサービスは、インターネット上に構築された現実世界に似た仮想 3 次元空間を指す。その空間に接続することで、よりリアルに、より容易にコミュニケーションを実現出来る場所を提供するものである。

近年、ブログや SNS が浸透し、ユーザが情報サービスに対して発信者であり創作者になっている潮流から、メタバースサービスは大いに注目されている。世界最大規模のメタバースサービスとしては、米 Linden Reserch 社が提供している Second Life が有名である [1]。一方で、クライアント端末、特にそのグラフィクスに要求されるスペックがある程度高いこと*1、レスポンスが遅いことから、期待されている程は普及していないという現実が挙げられる。そのため本研究では、クライアントを含めたメタバースシステムにおけるレスポンス短縮を目的とし、そのボトルネックの解析を行う。

2 評価環境

本研究ではメタバースが将来 Web のように普及することを前提として期待しており、そのためメタバースサーバを分散型で構築し、評価を行っている。

メタバースサーバのオープンソフトである、OpenSim0.6.8[2] を用いてメタバースサーバを分散型で構築すると、5 つのサーバモジュールに分かれ、役割を分散させることが出来る (表 1)。これらのサーバモジュールは、同一のマシン上で実行することも異なるマシン上で実行することも可能である。また実験で使用した各サーバとクライアントのスペックを表 2 に示す。

3 研究内容

本研究ではまず、広域分散環境における分散型メタバースサーバにおける評価を行った。しかしサーバモジュールが広域に分散したことによる性能への影響は少なかった。一方で、元々のログイン時間は、最速の場合でも約 54 秒かかった。ログインまでの時間としては、Web を基準に考えると明らかに長すぎる。そのため本研究では、メタバースサーバのログインレスポンスに関するボトルネックの解析を行う。分散型メタバースサーバは、複数台のマシンが協調動作を行うため、振舞が複雑で様々なボトルネックが存在し得る。そこで本研究ではまず、個々のサーバモジュールのログイン時の働きとその連携がどのように行われているか調べた。その結果、レスポンスの遅い理由として、以下のような可能性が考えられる。

1. ストレージアクセスが遅い
2. CPU 負荷が高い、メモリの利用が飽和、ネットワークの帯域幅が不足

*1 Second Life クライアントビューアの推奨動作環境は CPU が 1.5GHz 以上、メモリが 1GB 以上、グラフィックカードが Nvidia は 9600 以上、ATI は 4850 以上などとなっている

表 1: OpenSim サーバモジュールの役割

サーバモジュール	役割
Userサーバ	ユーザ (アバタ) の管理
ROBUSTサーバ	オブジェクトとアバタの持ち物とグリッドの位置などの管理
Messagingサーバ	メッセージ処理の管理
Regionサーバ	リージョン (土地、一般には256m x 256mの領域) の管理
Databaseサーバ	データベース管理

表 2: サーバとクライアントのスペック

	サーバ	クライアント
OS	Fedora Core12	Windows XP
CPU	Intel Xeon 3.6GHz、2.4GHz	Intel Pentium 4 CPU 3.00GHz
Memory	4GB、512MB	2.5GB
OpenSim	0.6.8	—
Graphic	—	Intel 82945G Express Chipset Family

3. ノード間のデータ通信とデータ処理の累積時間が長い

本研究では、これらの問題がボトルネックになっているか確かめるための実験を行う。

4 ボトルネックの評価実験

レスポンスが遅い理由として挙げた、3 つの可能性に対して、以下に述べる実験で確認を行った。

4.1 ストレージアクセス実験

ストレージアクセスが遅いという可能性に対し、SSD(Solid State Drive) を用いた実験で確認を行った。SSD とは、記憶媒体としてフラッシュメモリを用いるドライブ装置であり、高速アクセスが可能な、ハードディスクの代替として利用出来るものである。本研究では、インテル社の X25-M Mainstream SATA Solid-State Drive、容量は 80GB タイプの SSD を使用し、Database サーバと ROBUST サーバの HDD を SSD に替え、再度ログイン時間の測定を行った。

しかし、結果はどちらもログイン時間に全く変化はなかった。すなわち本実験環境の OpenSim システムによるログイン時間は、サーバ側のストレージアクセス時間はボトルネックになっていないと考えられる。

4.2 CPU、メモリ、ネットワーク使用率の測定

4.2.1 クラスタ管理ツールによる評価

CPU 負荷が高い、メモリの利用が飽和している、ネットワークの帯域幅が不足しているといった可能性に対し、実験で確認を行った。クラスタ管理モニタリングツール Ganglia を使用し、全メタバースサーバのログイン時における振舞を測定した。

結果は、どのサーバも CPU の使用率は低いが、メモリについては、ROBUST サーバと Messaging サーバが飽和している状態であり、ネットワークについては、User サーバと Region サーバの通信量が多かった。しかし、ネットワークの帯域幅は飽和している状態ではないため、1 つの可能性として、メモリ利用における飽和が、ボトルネックかもしれないと考えられる。

4.2.2 メモリ増設によるログイン時間の測定実験

メモリ利用における飽和が、ボトルネックである可能性について、メモリを増設し、再度ログイン時間の測定

を行うことで確認した。すなわちメモリ利用が飽和している、ROBUST サーバと Messaging サーバにおいて、メモリ量を 512MB から 1GB に増やし、ログイン時間の測定を行った。

結果は、ROBUST サーバと Messaging サーバにおいて、メモリ量を増やしても、ログイン時間の短縮には繋がらなかった。すなわち本実験環境の OpenSim システムによるログイン時間は、サーバ側のメモリ量はボトルネックになっていないと考えられる。

4.3 パケット解析

ノード間のデータ通信とデータ処理の累積時間が長いという可能性に対し、パケット解析により確認を行った。ログイン中にサーバ上に出てくるログを解析した結果において、ログイン処理の後半部分で、処理時間が長くかかっているものが多く見られた。そのため、通信パケットの後半部分に重点を置き、解析した。tcpdump コマンドを用いて、全メタバースサーバのユーザログイン時における通信パケットをキャプチャした。そして、IP アドレスを基に、どのサーバモジュールからどのサーバモジュールへパケットが送られているか確認した。

解析を進めると、処理時間が長いログインの後半のパケットは、全て Region サーバとクライアント間のパケットであることが判明した。つまり、両者のやりとりが、ネットワーク遅延等により長い時間がかかっている可能性がある。

5 マシンスペック変更時のログイン時間測定実験

パケット解析による結果を基に、Region サーバとクライアントのマシンスペックを上げ、ログイン時間の測定を行った。

5.1 高スペックな Region サーバとクライアント端末を用いたログイン時間測定実験

表 3 のように、Region サーバのスペックのみを上げ、ログイン時間の測定を行った。しかし、結果はログイン時間短縮には繋がらなかった。そこで、次はクライアント端末の性能のみを上げ、ログイン時間の測定を行った。使用したクライアント端末は表 4 に示す。

その結果、図 1 のように、ログイン時間は平均約 31 秒になった。また、再現性も保たれている。

5.2 考察

本実験において、クライアント端末のスペックを上げた場合に、ログインレスポンスの短縮に繋がることが明らかになった。本実験で使用したクライアント端末は、64bit CPU と 64bit 対応 OS を搭載しているため一度に処理出来るデータ量が多く、更にメタバースでは重要なグラフィックの処理が速くなったことが要因だと考えられる。この詳細を調べるため、次節の実験を行った。

6 各サーバモジュールにおけるパケット解析

ログイン時間が短縮した際、各サーバモジュールにおける振舞の違いを検証した。その手順は、クライアント端末を高スペックに変更する前後で、tcpdump コマンドによってパケットをキャプチャし、各サーバモジュールのパケット数と時間を比較した。結果を図 2 と図 3 に示す。Region サーバにおける処理パケット数とパケット通信時間が、他と比べて格段に大きいことがわかる。また、クライアント端末の変更後、Region サーバにおけるパケット数と通信時間が減少していることがわかる。

表 3: Region サーバのスペック変更

変更前	変更後
OS	Fedora Core12
CPU	Intel Xeon 3.6GHz
memory	4GB
	8GB

表 4: クライアントのスペック変更

変更前	変更後
OS	Windows XP
CPU	Intel Pentium 4 CPU 3.00GHz
memory	2.5GB
Graphic	Intel 82945G Express Chipset Family
	NVIDIA Quadro NVS 295

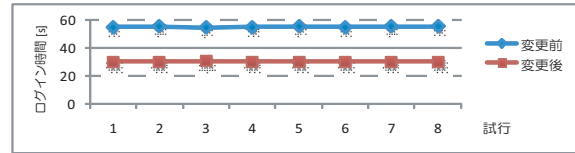


図 1: クライアント端末変更時のログイン時間

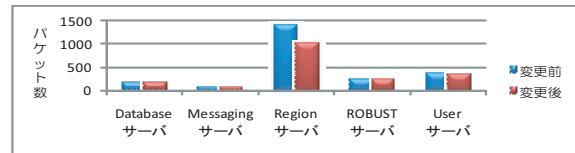


図 2: 各サーバモジュールにおけるパケット数

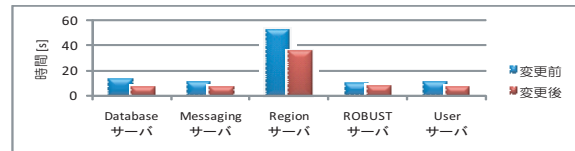


図 3: 各サーバモジュールにおけるパケット通信時間

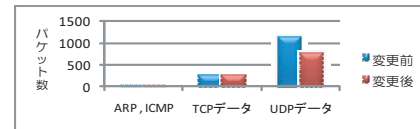


図 4: Region サーバにおけるパケット数

そこで、具体的にどのようなパケットが減少しているのか、解析を進めた。結果を図 4 に示す。

この結果により、UDP 通信におけるパケットの数が極端に減少していることがわかる。OpenSim システムにおいて、サーバモジュール間の通信には TPC(HTTP) が用いられており、UDP 通信は、クライアントと Region サーバの通信のみで使われている。これはメタバースの土地やアバタに関する精細なグラフィクス等の膨大なデータを両者で細かく処理しながらやりとりしている部分であり、クライアントの性能向上が、OpenSim システムにこのような形で確かに影響を与えたことが確認出来た。

7 まとめと今後の課題

本稿では、OpenSim システムにおけるログインレスポンスのボトルネックが、本実験環境ではクライアントのスペックであることを確認した。このような状況では、クライアントのマシン性能を上げることがログイン時間を短縮するために最も効果的であると言える。しかし将来的には、クライアント側で行われている処理の一部をサーバ側で行うことにより、クライアントとして高スペックな環境を用いなくても、ログイン時間を短縮出来るようになることが望ましいと考えられる。

参考文献

- [1] SecondLife, <http://jp.secondlife.com/>
- [2] OpenSimulator, <http://opensimulator.org/>