

2K-03

Thin クライアントモデルの枠組みへの Thick クライアント導入に関する一検討*

小口正人[†] 齊藤忠夫^{†,‡} 野原光夫[‡]
[†]中央大学 [‡](株)トヨタ IT 開発センター

1. はじめに

分散システムのクライアント・サーバモデルは、機能をサーバとクライアントでどのように分担するかという観点から、処理をサーバで集中して行くクライアントの機能を最小とする Thin クライアントモデルと、逆に処理をクライアントで分散して行く Thick クライアントモデルに分類できる[1]。常時接続が普及しつつある有線ネットワーク環境とは異なり、オフラインでの使用が多いテレマティクスのようなモバイル環境においては、Thick クライアントモデルをベースにした情報・サービスの提供システムの実現が望ましいと考えられる。本稿では、現状において広く用いられている Thin クライアントを対象とした Web ベースのシステムの枠組みに Thick クライアントを導入していく手法の提案と検討を行う。Thick クライアントをサーバと接続した実験プラットフォームを構築し、データベース上での処理を行うアプリケーションを Thin クライアントと Thick クライアントそれぞれの形で実行して比較する。

2. Thin クライアントモデルと Thick クライアントモデル

Thin クライアントモデルと Thick クライアントモデルを比較すると、一般に Thin クライアントモデルが注目されることが多く、従ってサーバサイド・プログラミングが盛んに研究されている。

しかしテレマティクスなどのモバイル環境においては、クライアントにも大量のデータを蓄積・処理する能力のある端末を用いることが可能になってきており、またネットワークに関しては今後無線 LAN や DSRC など広いバンド幅の通信が行える場所、いわゆるホットスポットが部分的に存在する環境が増えてくるものと考えられる[2]。そのような状況においては、データを使用するたびにサーバにアクセスしなければならない Thin クライアントモデルより、ホットスポットで必要なデータを持ってきておき、その後はオフラインや細い回線のみで接続された状態になっても要求を満たすサービスを続けられる Thick クライアントモデルの方が好ましいケースも多い。またパーソナライゼーションなどの高度なサービスを行う場合にも、Thick クライアントが有利である。

しかし現状では Thin クライアントモデルをベースにしたシステムが一般的であり、今後も発展が予想されることから、Thick クライアントモデルのシステム開発を Thin クライアントモデルと共通の枠組みで考えるべきである。さらには上記のように通信環境が多様化していった場合、コストパフォーマンスは端末価格だけではなく、ネットワークも含めたトータルなもので評価するべきである。

そこで本研究では両モデルの共存環境を目指し、HTTP プロトコルで通信を行う現状の Thin クライアントモデルの枠組みに Thick クライアントを導入する構成を検討した。これにより、アプリケーションプログラマが両者の違いを出来るだけ意識せずにシステムを構築できるシームレスな環境の実現を目標とした。

* A study of thick client model adaptation to a thin client model-based web service framework.

Masato Oguchi[†], Tadao Saito^{†,‡}, and Mitsuo Nohara[‡]

[†]Chuo University

[‡]TOYOTA INFOTECHNOLOGY CENTER CO., LTD.

3. 提案方式と実験プラットフォーム

Thin と Thick が共存し得るシステム構成案を得、これを実験プラットフォーム上に構築した。

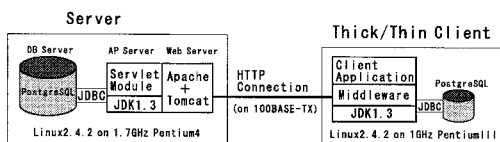


図 1. 実験プラットフォームの概観

図 1 に示す本システムは以下のように動作する。まず Thin クライアントモデルの形で実行（実験では Thick クライアントマシンを用いてシミュレート）する場合、クライアントはサーバ側の Web サーバと HTTP コネクションを張り、これを通して AP サーバのサーブレットモジュールが呼び出され実行される。サーブレットモジュールは JDBC を通じて DB サーバにアクセスし、必要な SQL 処理を実行して結果がサーブレットに返され、これが Web サーバと HTTP コネクションを通してクライアントに返って来て表示される。サーバサイドは現状で一般的になりつつある 3 ティア構造のサーバ・アーキテクチャそのものの動作である。

一方 Thick グライアントモデルの場合、処理の前半部分は Thin クライアントモデルの場合と同様である。Thick クライアントモデルの場合には、サーバ側の DB にアクセスした時点で、そのまま SQL が実行されるのではなく、まず DB の内容をクライアント側へ移送する処理を始める。クライアントは図 1 のようにアプリケーションとそれをサポートするミドルウェア及び JDK からなり、アプリケーションはミドルウェアを通してサーバやローカル DB にアクセスする。DB サーバで読み出されたテーブルは HTTP を介してクライアント側に送られ、受け取ったクライアントはローカル DB に JDBC でアクセスしてデータの書き込みを行う。これが完了した後、アプリケーションからミドルウェアを介しローカル DB に SQL 文が送られ、Thin クライアントモデルの場合にサーバ側で実行されたものと同じ処理が行われて結果が表示される。本研究では以上のようなシステムを実現すべく、クライアント側のミドルウェアおよびサーバ側のサーブレットモジュール等のソフトウェア環境を開発した。

4. DB アクセスとデータ転送の実験

構築した実験プラットフォームを用いて、DB アクセスとデータ転送の基礎的な実験を行った。アプリケーションとして、クライアントからのリクエストに基づきサーバの DB 上にある 1 つのテーブルに検索文でアクセスして中のタプルをすべて読み出すという処理を、Thin クライアント型と Thick クライアント型の両方で実行させた。DB としては、属性値に数値とキャラクターを計 5 列持つテーブルを 1 つ作成しサーバ側に格納した。タプル数は 10 万件であり、データサイズは約 12MB である。これを Thin 型と Thick 型で 2 回ずつ実行させた。表 1 に示す実験結果において、処理時間 1 はサーバにおけるタプルの作成及び DB への挿入時間、処理時間 2 はサーバにおけるタプル読み出し時間（Thin 型ではサーバにおける DB 処理時間）、処理時間 3 はデータ転送時間、処理時間 4 はクライアントにおける DB へのタプル挿入時間、処理時間 5 はクライアントにおける DB 処理時間である。

表 1. 実験結果（機能毎処理時間[sec]）

実行モード \ 処理時間	1	2	3	4	5
実行前	232				
Thin 1 回目		6	9	—	—
Thin 2 回目		6	9	—	—
Thick 1 回目		6	9	133	8
Thick 2 回目		—	—	—	8

5. まとめ

基礎的な実験結果から、たとえば DB に繰り返しアクセスするアプリケーションにおける Thick クライアントの優位性がわかった。今後は Thin クライアントと Thick クライアントの使い分けをさらに明確化し、より現実的な評価実験を行う予定である。

参考文献

- [1] I. Ahmad, "Network computers: The changing face of computing", IEEE Concurrency, vol. 8, no. 4, 2000, pp. 9 - 11.
- [2] J. Ala-Laurila *et al.*, "Wireless LAN Access Network Architecture for Mobile Operators", IEEE Communications Magazine, vol. 39, no. 11, November 2001, pp. 82 - 89.