

履歴再生・レイヤー管理機能を持つ CMS の実装

岩田明典[†] 高丸尚教

The implementation of CMS with action-rollback and multi-layer manager

AKINORI IWATA,[†] and HISANORI TAKAMARU

1. 序 言

コンピュータとネットワークを用いた応用ソフトウェアとして CMS や LMS と呼ばれる学習履歴管理・資料提示のシステムは欧米の大学でその有用性を認められ導入が進んでいる。日本国内においても欧米に追随する形で普及が進んでおり、実際に学生がいつでも講義資料にアクセスし、予習・復習に役立てる機会も多くなってきた。

しかし、従来の CMS 等は資料の一次配布や情報提供側が決定した情報階層や構造を提示するということが最適化されている面は否めない。

我々はコミュニケーション構造を扱うための CMS を構築するモデルとして発言者主導モデル¹⁾ という設計モデルを提唱している。提案モデルは発言を透明なレイヤを用いて層状に時間軸記録し、レイヤを情報コンテナとすることで、柔軟なアクセス制御と情報階層・構造の自由な組み換えを可能とした。発言者主導モデルの発言領域という概念は、蓄積・交換される発言を利用者が「公」と「私」のどちらに分類するかという行動を重視している。インタフェースとして、利用者の注目している箇所を推定し、コンテンツ間の関連をマウスカーソルの軌跡に注目した直接的方法による文書アノテーション手法を提案²⁾ している。

本論文では、データのレンダリングやファイル送受信をバックエンドで一括処理する (図 1) ことにより、(1) フロントエンド開発 TCO を削減すること、(2) 時間の掛る処理からのフロントエンド開放、(3) バックエンドに複数拠点のデータ統合機能を備え、データ分

散配置下でもユーザに煩雑な設定を排除可能という 3 つの利点を有することを示す。

2. フロントエンド開発 TCO の削減

インタフェースとなるフロントエンド開発の多様性を確保するためには、フロントエンド開発特に、バックエンドとの通信に関する規約を簡潔にして処理の多くをサーバ側に依拠するというデザインが有効である。

第一は、CMS 上で意味解析をするためには時系列の履歴管理・レイヤー挿抜イベント管理が必要不可欠である。このような処理をフロントエンドに個々に実装していくことは作業の重複、データ複製等冗長な開発作業をすることに繋がる。TCO 削減のためバックエンドでデータの一貫性を保ち、フロントエンドとのインタフェースを単純なリクエストと情報の粒度に差を持たせた (画像 > バウンディングボックス > 各オブジェクトのプロパティ) レスポンスによって構成する。

第二は、ユーザ或いはユーザ間の共同作業領域を画面よりも広いキャンバスを考え図 2 に示すようにユーザはビューポートを通してキャンバスの一部分を見る。キャンバスとビューポートの概念を用いることで、フロントエンド側の座標管理を軽減する。また、ビューポートで任意の領域をページとして登録することで、スライド表示や重要箇所の表示が可能となる。

3. フロントエンドの応答時間削減

CMS や LMS のような教育のための応用システムではファイルアップロードを活用することで作品レビューや共同創作へ継げることができる。そのためフロントエンドを UI をブロックする処理から解放し応答時間を削減したい。そのため以下に示す 2 点を考慮した。

[†] 中部大学大学院工学研究科情報工学専攻
Graduate school of Computer Science, Chubu University

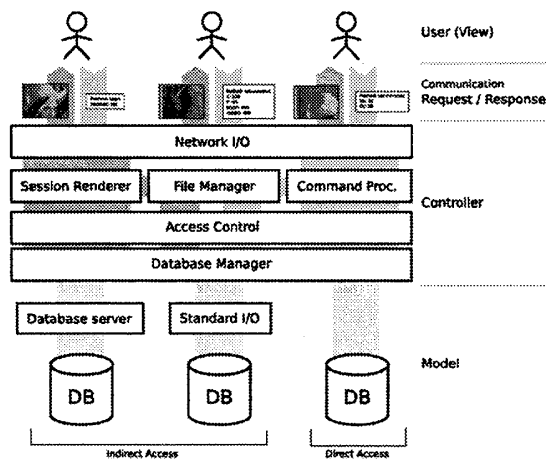


図 1 システム概観

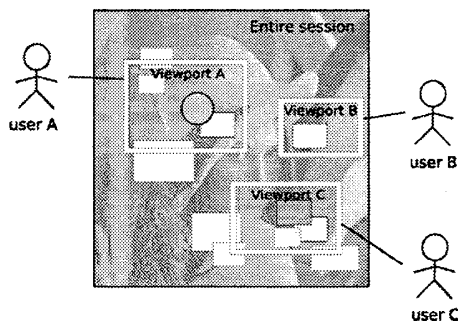


図 2 セッション全体とビューポート

第一に、ウェブアプリケーションの場合は Atwood³⁾ が指摘しているように、ブラウザではアップロード中に別のページに移動することも、途中でウィンドウを閉じることも一般にできない。この問題を解決するため、バックエンドで動作するサーバにクエリを発行し、フロントエンドからアップロード処理を代行する手法を取った。具体的には対象のファイル名とアップロード先の情報をリクエストとして発行すると、バックエンドでアップロード処理を継続し、フロントエンドが正常終了しなかった場合でも処理が正常に行われる。

第二は、ビューポートの導入によるレンダリング時間の短縮である。図 2 に示したようにキャンバスには大量の共有データが配置される可能性がある。これらのデータを全て取得しレンダリングするのではなく、視界に入っている領域だけを処理対象とすることによってデータの絞り込みを行い、レンダリングするオブジェクト数を削減することで、画像生成の処理時間を短縮する。

4. バックエンドの情報拠点統合機能

グループワークのような数台のコンピュータで構成す

る小規模の LAN における CMS 運用と活用のため、大規模環境から小規模環境まで対応できる柔軟なデータベース管理が必要となる。

小規模環境への対応のためサーバ型データベースを導入したり、導入作業が煩雑となる事態を避け、メモリ使用量が少く、導入作業の省力化を計ることができる組み込みデータベースを用いる。本バックエンドでは SQL による操作が可能 SQLite⁴⁾ を利用している。SQLite データベースはネットワークを介した接続には一切対応しないが、小規模な LAN を用いる作業に対応するため、Python によるデータベースサーバとアクセス用のモジュールを作成し、ネットワーク経由での SQL 発行に対応した。接続するデータベースのファイル名、或いはホスト名とポート番号の組をローカルサーバのシステム DB に登録することによって、あたかも、単一のデータベースに接続しているかのように、クエリを発行することが可能である。ユーザからは、複数のホストあるいはファイルにデータベースが分割されていることを意識せず、クライアントアプリケーションは設定なしで複数のデータベースを同時利用することができる。

5. 結 論

本文で挙げた 3 つの利点を念頭に試作アプリケーションを開発した。バックエンドに処理を分散することにより、データの一貫性の維持を可能とした。バックエンドへの処理の分担はフロントエンド動作の効率化を計ると同時に、異なる規模のネットワーク環境にも適応することができる。データベースに部分的に採用した生産者/消費者モデルは様々な情報源を統一的に扱う方法である。この手法をさらに推進し多次元データが存在するデータベースにおいても同手法による有効性について検討したい。

参 考 文 献

- 1) 岩田明典, 高丸尚教: 発言者主導モデルを採用したリアルタイム次世代 CMS の開発, 情報処理学会研究報告, Vol. 2007, No.6, 情報処理学会 (2007).
- 2) 岩田明典, 高丸尚教: 仮想スポンジを用いた文書アノテーション手法の提案, 平成 19 年度 電気関係学会東海支部連合大会 講演論文集, O-83 (2007).
- 3) Atwood, J.: Why Are Web Uploads So Painful? (2007). <http://www.codinghorror.com/blog/archives/000964.html>.
- 4) Hipp, D.R. and Kennedy, D.: SQLite, <http://sqlite.org/> (2000).