

Web アプリケーションのクラス設計最適化に関する研究

神奈川大学大学院 工学研究科 経営工学専攻

* 婁^{るう} 慶博^{けいはく}, 田中 裕介, 内田 智史^{さとし}

1 はじめに

近年、企業の基幹システムにおいては、J2EE(Java2 Enterprise Edition) で構築された Web アプリケーションが急速に増加し、その振舞いも複雑になってきている。Web アプリケーションの規模が大きくなると、データベースの数が増大し、データベース同士の構造も複雑になってくるため、Web アプリケーションの信頼性を保証する検証技術の重要性が益々高まってきた。このことからアプリケーションの改良・保守が開発者にとって非常に煩雑な作業となっている。ソフトウェアの開発の分析・設計段階で設計仕様をモデル化して検証すると、Web アプリケーションのパフォーマンスの向上と開発コストを削減することが可能となる。

本研究では、@ ODB (2 章参照) を利用して開発された既存の Web アプリケーションに対して、そのデータベース設計仕様をチューニングして、最適なデータベース構造を提案するシステムを構築する。近年、Web アプリケーション開発ではオブジェクト指向言語が導入されてきており、オブジェクト指向言語の Web アプリケーションの設計ではクラス図の作成が必要不可欠である。また、クラス図はプログラムを書く時に、直接使われるので実装に一番近い仕様である。@ ODB では、設計した 1 つのクラス図を元に 1 つのクラスをそのままデータベースのテーブルとして使用できる。

2 @ ODB

Object Based RDB は、神奈川大学工学部経営工学科内田研究室で開発された、RDB (Relational Database) に Java オブジェクトを暗号化して直接格納する技術である。具体的には、Java クラスより生成されたオブジェクトをシリアライズし、文字列に変換・暗号化し、文字列データとして RDB の中に格納する。RDB から取り出されたその文字列をオブジェクトに復元することにより、オブジェクトを RDB 中の 1 つのデータとして扱うことが可能になり、オブジェクト指向言語との相性が良くなる。しかし、この

手法を導入するにあたり、シリアライズや暗号化などプログラム作成の負担が増加する。そこで、Object Based RDB に関するプログラム及びデータベースアクセスプログラムなどを自動生成するツール@ ODB を開発し提供することにより、開発者への負担削減を行ってきた。

3 研究方法

本研究では、開発されたアプリケーションのデータベースの再設計方法を提案する。データベース検索に時間がかかるのは、複数のデータベースから検索を行っていることが大きいと考えらる。そこで、実際によく利用されているデータベース検索について分析し、データベースアクセス回数を最小化するデータベース構造にチューニングする。

まず、Java プログラムが保存されている任意のフォルダからファイルを取得し、そのクラス同士の関係・フィールド変数・メソッド名を抽出する。そこからクラス図やデータの一覧を作成して、クラス図と各クラスの情報からクラス全体の動的な振る舞いを表すモデルを作成する方法を提案する。

次に、データベースモデルでデータベース同士のパスを解析して、データベース同士の構造の最適化について検証する。本研究で提案するアルゴリズムの全体の流れは図 1 の通りである。

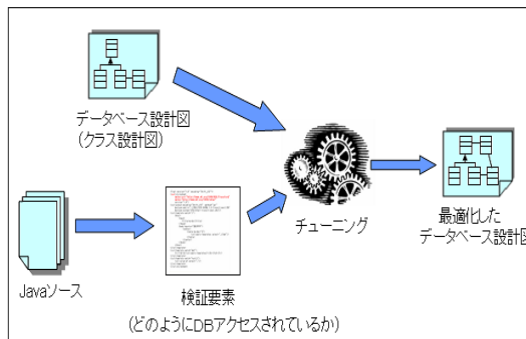


図 1: 本研究で提案するアルゴリズムのイメージ

3.1 テーブルモデルの作成方法

本研究では、Web アプリケーションの各ページで利用されるテーブル T_1, T_2, \dots, T_n について、次の手順で、テーブルモデルを作成する。テーブルモデルのイメージは図 2 の通りである。

1. 各テーブルに対応するテーブル間のパスを見つける
2. 各テーブルの検索キーと次のテーブルの検索キーを検出する
3. 条件分岐する場合は、各テーブルのアクセス優先度を決める
4. 各テーブル間のパスが組み合わせ可能ならば、テーブルを連結する
5. 組み合わせの中でデータベースアクセス数が最小のものを解とする

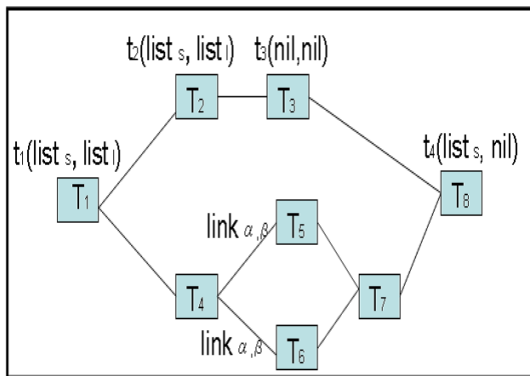


図 2: テーブルモデルの例

テーブルモデル例で記述するテーブル状態変数の意味は表 1 の通りである。

変数	意味
T_i	テーブル
t_i	検索キーのペア
$list_s$	そのテーブルの検索キー
$list_l$	次のテーブルの検索キー
$link_{\alpha, \beta}$	テーブル間のパス
nil	検索キーが存在しない

3.2 テーブルモデルのチューニング

ここで、テーブルモデルのチューニングとは、モデル化したシステムを用いてデータベース同士の構

造のパスを解析し、それがデータベースアクセスとして実行される時にアクセス経路が最短であるか検証して、場合によっては新しいデータベースを追加し、最適なデータベース構造を網羅的に検査する手法である。図 3 は検証によりクラス図を自動修正するときのイメージである。

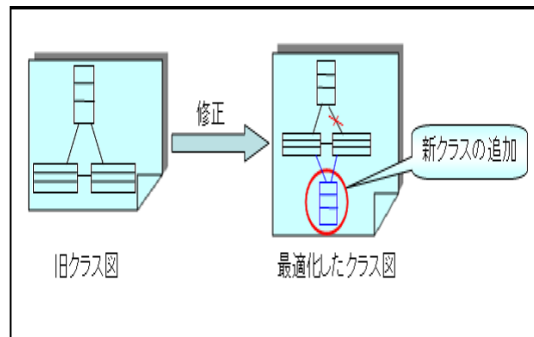


図 3: クラス図を自動修正するイメージ

4 まとめ

本研究では Web アプリケーションにおけるデータベースの再設計の際の効率について検討した。具体的には、複数のデータベース間のクラス構造を最適化するアルゴリズムを検討した。また、オブジェクト指向において重要なポイントはクラス設計であり、本研究で提案した手法により Web アプリケーションのパフォーマンスを向上できると考える。

5 今後の課題

- 提案した手法をモデル化という立場から詳細に分析し、問題点を洗い出すことにより、早急にシステムを完成させ、検証を行う。
- 提案手法の有効性を評価するため、実際に当研究室で稼働中の Web アプリケーションである WebLec を対象に適用実験を行う。

参考文献

- [1] 青木利晃, "オブジェクト指向分析モデルの検証支援環境", 第 19 回日本ソフトウェア科学会全国大会, 2002 年
- [2] 中西 啓之, 三浦 孝夫, "記述論理を用いた UML 整合性の検証システムの構成と実現", 電子情報通信学会データ工学ワークショップ, 2005 年