



2006 年度下期未踏ソフトウェア創造事業 採択案件評価書

1. 担当PM

並木 美太郎 PM (東京農工大学大学院 共生科学技術研究部 助教授)

2. 採択者氏名

開発代表者:村上 直 (高エネルギー加速器研究機構 計算科学センター 助手)
共同開発者:なし

3. プロジェクト管理組織

株式会社メルコホールディングス

4. 委託金支払額

2,927,824 円

5. テーマ名

DBPowder: RDBMS 活用フレームワーク

6. 関連Webサイト

なし

7. テーマ概要

RDBMS は、データを管理・利用するための強力なコンポーネントであり、様々なアプリケーションのバックエンドとして広く利用されている。

しかし、RDBMS の活用は容易ではない。容易でない背景として、以下の4つがあげられる。

ア) 適切なリレーショナルスキーマの構築は難しい

イ) スキーマ内のテーブル連携を手軽に実現できるアプリケーションに乏しい

ウ) RDBMS プログラミングは煩雑である

エ) 開発後のスキーマの再変更は非常に困難である

これらの問題点は相互に絡み合っており、そのことが問題点をより複雑にしている。これらの一部を解決するフレームワークは存在するが、全てを解決するフレームワークは存在していない。

我々は、これらの4つの問題点を解きほぐし RDBMS の有効活用を支援するためのフレームワーク、DBPowder を提案し、実装する。

DBPowder は、リレーショナルスキーマを簡潔に記述するための言語 DBPowder-mdl と、RDBMS 活用ソースコードを生成するコードジェネレータ DBPowder-factory を提供する。DBPowder-factory は、DBPowder-mdl が定義したスキーマを活用するウェブアプリケーションのソースコードを生成するとともに、そのスキーマを RDBMS 上に作成する。このウェブアプリケーションは、データの操作・閲覧機能を持ち、複数人で同時に利用できる。また、生成されたソースコードは、プログラマが拡張しやすいように構成されているので、本格的な RDBMS アプリケーションを構築するための土台として利用することができる。

RDBMS が抱える構造的な問題として、開発後のスキーマ再変更が非常に困難な点が挙げられるが、DBPowder-factory は、スキーマ再変更に伴うソースコードの変更箇所を可能な限り自動修正し、自動修正が危険な箇所についてはプログラマに指し示す。これによって、スキーマの再変更を容易にする。

8. 採択理由

Web アプリでは、DBMS との連携は重要であり、PHP では SQL 文を、Java ではクラスライブラリにより、RDBMS と連携処理を記述する。しかし、スキーマ定義などはコード中に分散し、決して開発効率や保守性が良いものとは言えない。

本申請は、スキーマ記述言語を定義し、この言語で記述されたプログラムから Web アプリを生成するアプローチを提案しており、Web アプリ開発・保守のスキーマの再変更を簡便化しようという試みは興味深い。DBMS と一貫した体系を提供できれば、実用性も高く、未踏性もあることから採択とする。

9. 開発目標

本プロジェクトでは、

- ・適切なリレーショナルスキーマの構築を助ける
- ・スキーマ内のテーブル連携を手軽に実現できるアプリケーションとして作用する
- ・RDBMS プログラミングの煩雑さを軽減する
- ・開発後のスキーマの再変更を可能にする

の四つの問題点を解消するソフトウェアとして、DBPowder を開発する。本プロジェクトでは、利用頻度の高い、RDBMS に Web を組み合わせた開発を主なターゲットとし、RDBMS を用いたウェブアプリケーションの構築を助けるフレームワークとして DBPowder を提供する。特に、RDBMS を中心とし、MVC ごとに Web アプリを分割し効率よく、また保守性の高い Web アプリの開発環境を実現する(図4)。

10. 進捗概要

本プロジェクトでは主に次の四つの機能を持つソフトウェアを開発した。

- (1) データベース開発・利用支援機能
- (2) スキーママッチング機能
- (3) ユーザインタフェース機能
- (4) DBPowder 統合利用機能

Web アプリ開発者が作成した図4の各 DBPowder-schema、DBPowder-navigation、DBPowder-html を、それぞれ結合し、HTML、SQL、サーブレットに変換する Java プログラムとなっている。

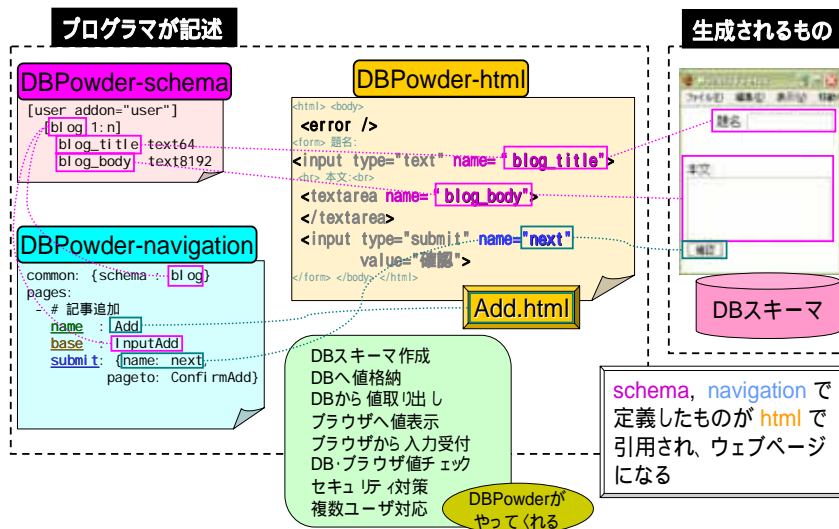


図4 DBPowder による Web アプリ開発

11. 成果

DBPowder では、ウェブアプリケーションの Model-View-Controller アーキテクチャに対応した記述言語としてそれぞれ、DBPowder-schema, DBPowder-html, DBPowder-navigation を提供している。DBPowder-schema のみでウェブアプリケーションを構築できる一方、DBPowder-navigation, DBPowder-html と連携させることで、ページ遷移やレイアウトを様々に実現できる。これにより、開発の簡略さと柔軟さの両方を獲得することに成功した。

DBPowder-schema は、簡潔で直感的なツリー型の表記でありながら、逆方向リンクの記述を可能にすることで、リレーショナルスキーマの豊富な表現力を維持している。また DBPowder-schema は、リレーショナルスキーマ構築で煩雑になりがちな、主キー・外部キーを自動的に補うことで、簡潔な表記を可能にしている。DBPowder-navigation, DBPowder-html のウェブページ構成機能は、DBPowder-schema のスキーマ記述力をベースに成り立っている。

DBPowder では、スキーマに関する記述を DBPowder-schema に集約している。これにより、煩雑になりがちなスキーマの構築や修正の工数を下げることができた。また、生成コードの拡張ポイントを明らかにすることで、DBPowder-API を用いてカスタムコードの記述が可能になっている。これは、プログラマによる独自拡張を実現する上で必要な機能であり、DBPowder でもサポートされている。

以上の内容により、RDBMS を中心とする Web アプリの開発環境を提供できた。

12. プロジェクト評価

Web アプリでは、DBMS との連携は重要であり、PHP では SQL 文を、Java ではクラスライブラリにより、RDBMS と連携処理を記述する。しかし、スキーマ定義などはコード中に分散し、決して開発効率や保守性が良いものとは言えない。

本プロジェクトは、スキーマ記述言語を定義し、この言語で記述されたプログラムから Web アプリを生成するアプローチを提案しており、Web アプリ開発・保守のスキーマの再変更を簡便化しようという試みは興味深い。DBMS と一貫した体系を提供できれば、実用性も高く、未踏性もあることから採択としたプロジェクトである。

当初は明確でなかった Web アプリ全体における各要素の記述もほぼ固まり、DB だけでなく、Web アプリ全体を見渡した設計になった点はよかった。開発全体として、より完成度を向上し、実用的な Web アプリを複数開発し、その有効性を主張できるようにしてほしい。

13. 今後の課題

公開するための完成度の向上、文書作成などが今後の課題である。