

DBFlute 導入ポイント

久保 雅彦(jflute)

DBFluteとは？

○ 現場指向のO/Rマップ

- 自動生成ありき
- 現場指向な全体最適機能
- ConditionBeanと外だしSQL(OutsideSql)

資料作成時点：

DBFlute-0.9.5

DBFlute.NET-0.8.9

とりあえず読むページは？

- DBFluteの紹介
- アーキテクトのためのチュートリアル
- デベロッパーのためのチュートリアル

実際のページを
みながら説明します。

サポートされる言語は？

- Java 5.0/6.0
- C# 3.0

別の言語が増える予定なし
新バージョン対応はする予定
C#版はDBFlute.NETとして独立

サポートされるDBは？

○ MySQL

○ PostgreSQL

○ Oracle

Java : 全てのDBでExampleあり
C# : MySQLのみExampleあり

○ DB2

○ SQLServer

Firebirdは確認環境(Example)がないが、
JDBCドライバ・各種クラスは用意されている

○ H2 Database (Javaのみ)

○ Apache Derby (Javaのみ)

Exampleプロジェクトを
みながら説明します

サポートされるDIコンテナは？

- Seasar (S2Container) 2.4
- Spring Framework 2.5
- Google Guice 1.0
- Lucy
- Slim3

C#版はSeasar(Quill)のみ

Exampleプロジェクトを
みながら説明します

DB設計のマナー (on DBFlute)

- PK制約を付けよう
- 複合PKは「代理キーPK + ビジネスキーUQ」
- FK制約を付けよう
- UQ制約を付けよう
- 名前に特殊記号を使うのをやめよう
- 名前に予約語を使うのをやめよう

次ページよりExampleDBを
みながら説明します

PK制約を付けよう

- ユニークに識別できる値がないと非常に不便
- PKを付けるコストは低い

【DBFluteでの制限】

PK制約がないテーブルは更新系処理は全て外だしSQLでやる必要がある。(Insertだけはオプションでできるようにはなる)

複合PKは 「代理キーPK + ビジネスキーUQ」

- 複合PKは、SQL上でもアプリケーション上でも扱いづらいため、一つの値で識別できるようにした方がいい。
- 代理キーPKだけだとビジネス的な制約が無くなるので、必ずビジネスキーにUQ制約を付与すること。

【DBFluteでの制限】

複合PKだとサブクエリ系の機能が利用できない。
(LoadReferrerやExistsSubQueryなど)

FK制約を付けよう

- 不整合なデータが発生して泣くのはリリース後
- 不整合なテストデータでテストの質が悪い
- FK制約付けることによるテストデータ登録の煩雑さは工夫次第でどうにでもなる。(例えば、DBFluteのReplaceSchemaを活用するとか)

【DBFluteでの制限】

テーブル間の関連をDBFluteの設定ファイルで指定する必要がある。(additionalForeignKeyMap.dfprop)

UQ制約を付けよう

- 不整合なデータが発生して泣くのはリリース後
- 不整合なテストデータでテストの質が悪い
- UQ制約がないとone-to-oneをERDから読み取れない

【DBFluteでの制限】

one-to-oneの関連をDBFluteの設定ファイルで指定する必要がある。(additionalForeignKeyMap.dfprop)

名前に特殊記号を使うのを やめよう

- なにかにつけて不便になる
- アプリで回避できても、周辺ツールや将来のシステム移行後にどんな災いがあるかわからない

【DBFluteでの制限】

そのテーブルは自動生成できないので、Viewを使うなどして回避。(該当テーブルは自動生成対象外に設定すること)

名前に予約語を使うのをやめよう

- なにかにつけて不便になる
- アプリで回避できても、周辺ツールや将来のシステム移行後にどんな災いがあるかわからない

【DBFluteでの制限】

DBの予約語とプログラム言語の予約語と二種類あるので注意

そのテーブルは自動生成できないので、Viewを使うなどして回避。(該当テーブルは自動生成対象外に設定すること)

制約系に関して補足

チェック制約やトリガを使った制約など、突き詰めるとキリがなく完璧な制約にするのは至難ではあるが、対費用効果を考えて「ちょっとした頑張りで大きな効果が得られる制約」は積極的に付けていくのが良い。(PK/FK/UQなど)

全体最適化をしよう！

- 共通カラムの設定を自動化！
- 区分値をタイプセーフに！
- 業務的one-to-oneを関連付ける！
- デバッグしやすいログを出力！
- Buriと連携！

代表的なものを挙げています。
ExampleDBやデモを交えて
(時間のある限り)説明します。

Maven DBFlute Plugin

とうとうMavenでDBFluteを
操作できるようになりました

別途資料で説明します