

新人研修報告書

平成 20 年 6 月 30 日

和才 友昭

1.目的

アプリケーション開発が行えるスキルを身に付けることを目的とする。

2.方法

IBM が配布している Java ゲームプログラムの「robocode」を使用して、様々なロボットを作成することで、オブジェクト指向を理解する。また、Struts を用いて Web アプリケーションの開発を行うことによって、Web アプリケーションの開発方法を身に付ける。

3.結果

robocode を使用したロボットを作成していく中で、デザインパターン及びオブジェクト指向を理解することができた。また、Web アプリケーションの開発を行うことで、SQL、Web アプリケーションでのデータベース接続方法、Struts での Web アプリケーションの構築方法を理解することができた。

4.結論

オブジェクト指向、Web アプリケーションの開発方法を理解することができた。しかし、作成したプログラムを見ると変数名・メソッド名のネーミングが解りにくかった。このようなプログラムでは、他者がプログラムを修正する場合に煩わしいことになる。実務ではプログラムのメンテナンス性も重要である為、メンテナンス性の向上が今後の課題である。

5.コメント

製品である学校給食会システムを Web アプリとして再開発を行うことで、実務と同等の緊張感を持つことができた。技術面では Java の基礎部分しか学んでいないが、この基礎を活かして、今後も Web アプリケーションの開発方法への理解を深めて行く。

目次

	頁
[1] 目的.....	1
[2] 対象システム.....	1
[3] 方法.....	2
3.1 オブジェクト指向の習得.....	2
3.2 デザインパターンの習得.....	3
3.3 Dao パターンの習得.....	3
3.4 Struts を用いた学校給食会システムの作成.....	4
[4] 結果.....	5
4.1 オブジェクト指向を理解する.....	5
4.2 デザインパターン.....	6
4.3 Dao パターンについて理解する.....	8
4.4 Struts を用いた学校給食会システムの作成.....	9
[5] 結論.....	12
[6] 参考文献.....	12
6.1 参考書.....	12
6.2 参考 Web サイト.....	12

[1] 目的

アプリケーション開発を行える技術と知識を身に付けることを目的とする

[2] 対象システム

新人研修で使用する開発環境等を以下に示す。

PC

- ・ Compaq dx2000 St
- ・ メモリ 512MB

Java

- ・ Java SE1.5.0_14

Eclipse

- ・ eclipse-SDK-3.3.1.1

EclipsePlugin

- ・ Tomcat Sysdeo プラグイン
- ・ Amateras プラグイン

MySQL

- ・ mysql-5.0.45
- ・ mysql-connector-java-5.0.8

Tomcat

- ・ Tomcat6

Strtus

- ・ Struts 1.3.8

[3] 方法

3.1 オブジェクト指向の習得

Java 言語の特徴であるオブジェクト指向を理解する為に、IBM が配布している「robocode」を使用して以下のロボットを作成する。

- ・ 三角形、四角形の動きを行うロボットの作成
- ・ 三角形、四角形の動きを交互に行うロボットの作成
- ・ 被弾するたびに動きを切り替えるロボットの作成
- ・ ランダムにスピンを行うロボットの作成
- ・ 画面の中央に移動後、敵を 360 索敵し弾を発射するロボットの作成

上記のロボットを継承、オーバーロード、インターフェースを用いて作成し、オブジェクト指向を理解する。

以下に継承、オーバーロード、インターフェース、抽象クラスの概要を記す。

継承

- ・ スーパークラスの機能をサブクラスで使用できるようになる
- ・ 多重継承を行うとはできない

オーバーライド

- ・ スーパークラスで定義しているメソッドにサブクラス独自の処理で実装し直すこと

インターフェース

- ・ 抽象メソッド(処理を実装しないメソッド)を定義する
- ・ インターフェースは複数実装することができる
- ・ オブジェクトを作成することはできない

抽象クラス

- ・ 抽象メソッドと処理を実装しているメソッドの両方を定義できる。
- ・ abstract 修飾子を class の前に付けることで抽象クラスとなる。
- ・ オブジェクトを作成することは出来ない。

3.2 デザインパターンの習得

以下に示すデザインパターンを使用し、オリジナルロボットを作成する。

Strategy パターン

- ・ 状況に応じてアルゴリズムを切り替える

Factory Method パターン

- ・ インスタンスの生成を行う

Singleton パターン

- ・ インスタンスが 1 つしか生成されないことを保障する

上記 3 つのデザインパターンを利用することを考えてクラス図を作成し、オリジナルのロボットを作成することで、Strategy パターン、Factory Method パターン、Singleton パターンを理解する。

3.3 Dao パターンの習得

Java 言語から、データベースへアクセスするクラスを、Dao(データアクセスオブジェクト)パターンを用いて作成する。

Dao パターン

- ・ メインロジックからデータベースへのアクセスを切り離す

3.4 Struts を用いた学校給食会システムの作成

アプリケーション事業部にて開発した学校給食会システム¹を Web アプリへ移植を行うことで、Web アプリケーションのシステム開発の技術、知識を身に付ける。以下に学校給食会システム開発の期間、担当プログラムを示す。

期間 : 2008/4/30 ~ 2008/6/12

担当プログラム : 地区マスタ保守画面、地区マスタリスト、製品分類マスタリスト、
入庫調整入力画面、入庫調整入力チェックリスト、米飯重量合計一覧表、
学校別重量別数量一覧表(米飯)、学校別重量別数量一覧表(パン)、
コネクション取得クラス、プロパティ取得クラス

担当プログラムを作成し、以下のスキルを習得、理解する。

- ・ Struts を使用した Web アプリケーションの作成方法の習得
- ・ アプリケーションサーバでデータベースへのアクセスを管理する方法の習得
- ・ SQL でデータベースのデータを操作する方法の習得

Struts を使用した Web アプリケーションの開発方法の習得

Struts を用いて学校給食会システムの開発を行うことで、Struts での Web アプリケーションの開発方法を習得する。

アプリケーションサーバでデータベースへのアクセスを管理する方法の習得

学校給食会システムでは、データベースへのアクセスをアプリケーションサーバ (Tomcat) で管理させる。アプリケーションサーバでデータベースへのアクセスを管理する方法を習得する。

SQL 文についての知識の習得

学校給食会システムでは、Web 画面から入力されたデータを基にデータベースへデータの検索、追加、更新、削除を行う為、SQL でデータベースのデータを操作する方法を習得する。

¹ 学校給食のパン製造及び米飯を納品しているお客様を対象としたシステム

[4] 結果

4.1 オブジェクト指向を理解する

様々な動きを行うロボットを作成することで、継承、オーバーライド、インターフェース、抽象クラスを理解することができ、オブジェクト指向とは操作側(ロボット)は内部の詳細な動きを意識しないでコーディングを行う考え方である、ということを理解した。

- ・ 継承

robocode で用意されている Robot クラスを継承させることで、ロボットに関する処理を気にすることなくコーディングを行うことができた。ロボットに関する処理とは、「ロボットが被弾した時」「ロボットを動かす」「弾を発射する」などの処理である。Robot クラスで「ロボットが被弾した時」「ロボットを動かす」といったメソッドが用意されているので、ロボットを作成する時にはそれらをオーバーライドして使うだけでよかった。継承を行い、スーパークラスの機能を使うことで、サブクラスでロジックを組む必要がなくなることを理解した。

- ・ オーバーライド

robocode で用意されている Robot クラスを継承し、ロボットの動作を行うメソッド(run)を再定義することで、独自の動作を行うことが可能であることを理解した。このメソッドを再定義することをオーバーライドと呼び、再定義したメソッドから元々あったメソッドを呼ぶことも可能であることを理解した。

- ・ インターフェース

「三角形、四角形の動きを交互に行うロボット」を作成する時に、インターフェースを用いて作成した。インターフェースでロボットの「動き」を定義し、三角形の動きと四角形の動きを行うクラスをそれぞれ作成した。それぞれのクラスにインターフェースを実装させ、操作側(ロボット)では、インターフェースを用いてロボットの動きを制御させた。

操作側(ロボット)では、「インターフェースが動きを行うものである」ということを知っているだけでよく、インターフェースが保持しているインスタンスを気にすることなくコーディングを行えた。今回の課題では、複数のインターフェースを実装させることはなかったが、インターフェースは複数実装することができるメリットもある。

- ・ 抽象クラス

抽象クラスは、抽象メソッドと処理を実装したメソッドの両方を定義することができる。処理を実装したメソッドは定義しなくてもよい。抽象クラスを使用する時には、継承してサブクラスを宣言し、インターフェースと同じく抽象メソッドをオーバーライドすればよい。抽象クラスは、インターフェースのメリット(複数実装することを除く)に加えて、抽象クラス内の処理を実装させたメソッドもサブクラスで使うことができるメリットがある。

4.2 デザインパターン

オリジナルロボットを作成することで、Strategy パターン、Factory Method パターン、Singleton パターンの3つのデザインパターンを理解した。

Strategy パターンは、メインロジックに依存せずにアルゴリズムを切り替えることができるパターンであることを理解した。図 4-1 に、今回作成したオリジナルロボットの Strategy パターン部分のクラス図を示す。

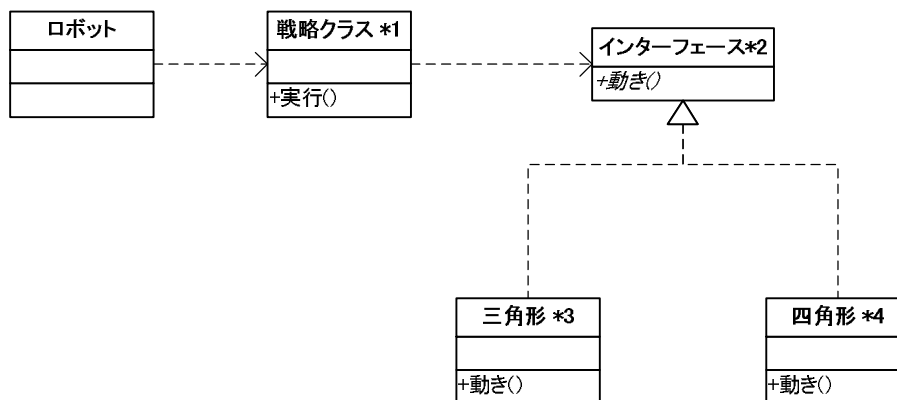


図 4-1 Strategy パターン

戦略クラス*1 で、被弾するたびに保持しているインスタンス(インターフェース*2)を三角形*3 もしくは四角形*4 に切り替える。そうすることにより、ロボットを実装する時に、アルゴリズム(動き)を意識する必要がなくなる。

Factory Method パターンは、スーパークラスでインスタンスの作り方を決め、サブクラスで生成するインスタンスを決定する。インスタンス生成と、どのインスタンスを生成するかを分けて作るということである。今回のオリジナルロボットでは、戦略を作る抽象クラスを定義し、戦略を作るクラスで被弾するたびに動作を切り替える戦略を生成している。図 4-2 に、今回作成したオリジナルロボットの Factory Method パターン部分のクラス図を示す。

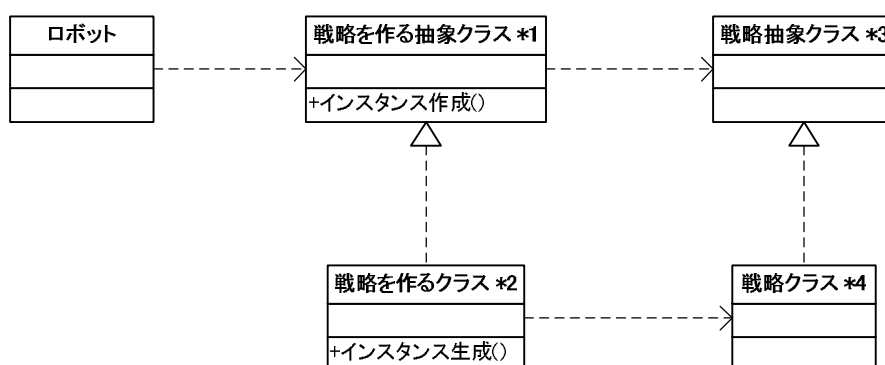


図 4-2 Factory Method パターン

戦略を作る抽象クラス*1で、インスタンス生成を定義し、戦略を作るクラス*2で戦略抽象クラス*3を実装させた戦略クラス*4を生成する。もし戦略が増えても、戦略を作るクラス*2を増やすだけで新しい戦略が作られるので、メインロジックに依存することはない。

Singleton パターンは、インスタンスが 1 つしか生成されないように保障する場合に適用するパターンである。今回のオリジナルロボットでは、動作確認を行うためにログを取るクラスを Singleton パターンで作成した。図 4-3 に、今回作成したオリジナルロボットの Singleton パターン部分のクラス図を示す。

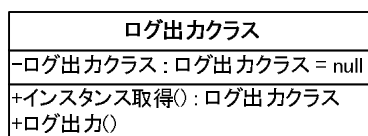


図 4-3 Singleton パターン

ログ出カクラス内でインスタンス取得メソッドを用意し、このメソッドを介さないとインスタンスを取得できない作りとした。クラスフィールドで、static にしたログ出カクラスのインスタンスを用意し、一意のものとする。インスタンス取得メソッドで、インスタンスが一度も生成されていない場合はインスタンスを生成し、インスタンスが生成されている場合は、オブジェクトをそのまま取得する。

4.3 Dao パターンについて理解する

Dao パターンとは、メインロジックからデータベースへアクセスする処理を切り離すことによって、データベースへの保存先やアクセス先が変更になっても、メインロジックには影響がでないようにする方法であることを理解した。図 4-4 に今回作成した Dao パターンと Factory Method パターンを組合わせたクラス図を示す。

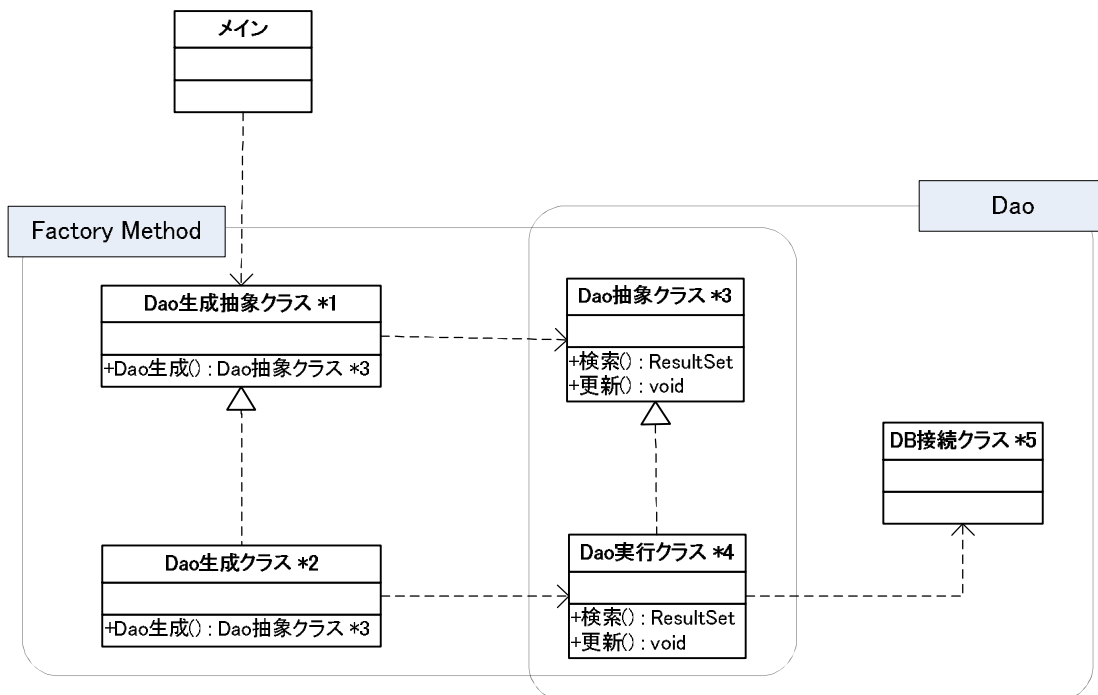


図 4-4 Dao パターン、Factory Method パターン使用図

Dao パターンと Factory Method パターンを組合わせることにより Dao 実行クラス*4 が複数作成された場合でも、メインクラスに依存しない作りとした。メインクラスでは Dao 生成抽象クラス*1 で用意している Dao 生成メソッドを使うだけでよい。Dao 実行クラス*4、DB 接続クラス*5 をメインクラスから切り離して作り、実際に SQL 文やデータベースへの接続先を変更しても、保守が行い易くメインクラスではロジックの修正が必要ないことを確認し、Dao パターンでの実装方法を理解した。

4.4 Struts を用いた学校給食会システムの作成

学校給食会システムで担当したプログラムを作成していく中で、以下の知識を習得、理解することができた。

- Struts を使用した Web アプリケーションの開発方法
- アプリケーションサーバでデータベースへのアクセスを管理する方法
- SQL 文についての知識の習得

Struts を使用した Web アプリケーションの開発方法

学校給食会システムを Struts アプリケーションで開発することにより、Struts での Web アプリケーションの開発方法を身に付けることができた。図 4-5 に Struts での Web アプリケーションの流れを示す。

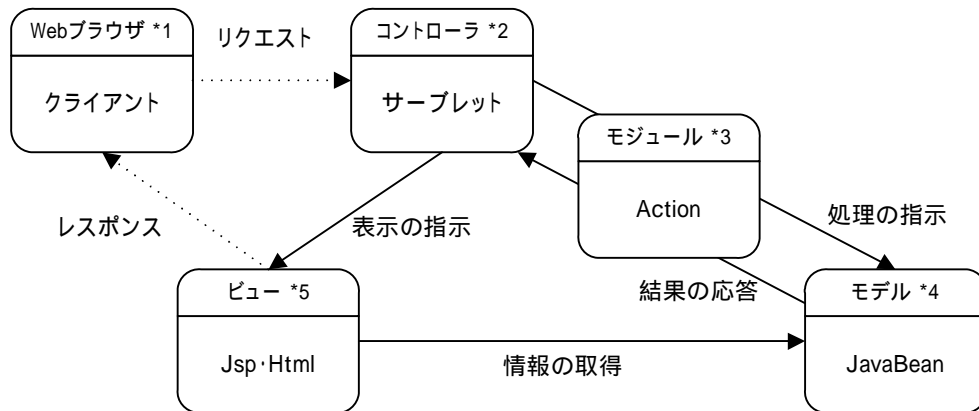


図 4-5 Struts アプリケーションの流れ

図 4-5 に示すように、Web ブラウザ*1 からのリクエストはコントローラ*2 で処理される。コントローラ*2 は、送られてきたリクエストに対しての呼出しモジュール*3 を決定する。モジュール*3 では、ビジネスロジック(モデル*4)を実行し遷移先をコントローラ*2 に戻す。コントローラ*2 は戻された遷移先(ビュー*5)を表示する。遷移先の Jsp ページは、スコープに登録されたデータを表示する。クライアントは、最終的に Jsp によって生成された結果(html)を、レスポンスとして取得する。

アプリケーションサーバでデータベースへのアクセスを管理する方法

アプリケーションサーバ(tomcat)にデータベースへのアクセス情報を設定し、コネクション取得クラスから JNDI 経由でコネクションを取得することにより、アプリケーションサーバでデータベースへのアクセスを管理する(図 4-6)方法を理解することができた。

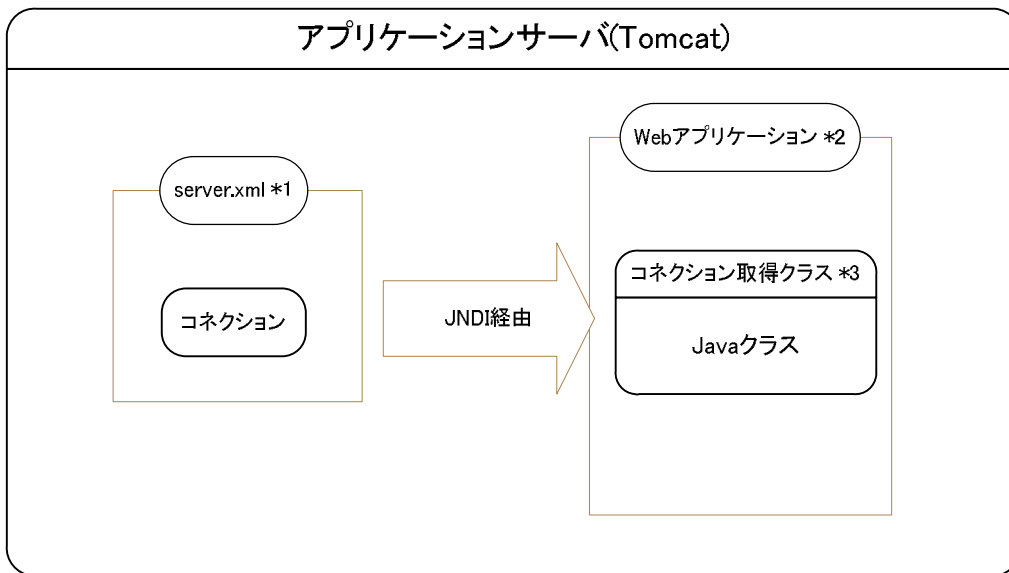


図 4-6 アプリケーションサーバでのデータベースアクセス管理

図 4-6 に示すように、アプリケーションサーバ内の server.xml*1 にデータベースアクセス情報を設定し、コネクション取得クラス*3 から JNDI 経由でコネクションを取得する。JNDI 経由でコネクションを取得することによって、データベースへのアクセス先が変更になっても、server.xml*2 のデータベースアクセス情報を修正するだけで済み、java ファイルや jsp ファイルのロジックの修正、コンパイルをする必要がなくなる。

SQL についての知識の習得

Web 画面から入力されたデータを基に、データベースのデータの抽出、検索、追加、削除を行うプログラムを作成することで、SQL でデータベースのデータを操作する方法を理解することができた。以下に、学校給食会システム開発で身に付けた SQL の基本的な構文を示す。

- データの抽出

```
SELECT カラム名 FROM テーブル
```

- データの検索

```
SELECT カラム名 FROM テーブル [WHERE カラム名 = 対象データ]23  
[GROUP BY カラム名(グループ化するカラム名)]  
[ORDER BY カラム名 [ASC|DESC]]
```

- データの追加

```
INSERT INTO テーブル [(カラム名)] VALUES(対象データ)
```

- データの更新

```
UPDATE テーブル SET カラム名 = 更新データ [WHERE カラム名 = 対象データ]
```

- データの削除

```
DELETE FROM テーブル [WHERE カラム名 = 対象データ]
```

WHERE 句とは

- 指定されたカラム名が対象データのものを指定する

GROUP BY 句とは

- 指定されたカラム名で、同じ値を持つデータごとにまとめる

ORDER BY 句とは

- ASC を指定した場合は昇順、DESC を指定した場合は降順、どちらも指定しなかった場合は昇順で指定されたカラム名のソートを行う。

² 文字列の場合は、シングルクォーテーションで文字列を囲む必要がある

³ []内の SQL 文は省略可

[5] 結論

Struts を用いた Web アプリケーションの開発と構築方法については理解できた。しかし、変数名や、メソッド名のネーミングが解り辛いといった問題点が残った。実務では、プログラムを複数の人が修正する可能性がある。その時に、誤解を招くようなネーミングをつけていると変数やメソッドの再確認を行うのに時間がかかってしまう。今後の課題として、変数名、メソッド名のネーミングセンスの向上がある。

[6] 参考文献

6.1 参考書

- ・ Java 言語で学ぶデザインパターン入門
- ・ 基礎から始める Jsp&Servlet
- ・ ApacheStruts 入門 アプリケーション開発
- ・ Apache Struts ハンドブック 改訂版

6.2 参考 Web サイト

- ・ @IT - アットマーク・アイティ [<http://www.atmarkit.co.jp/>]
- ・ スタイルシート入門 [<http://www.tohoho-web.com/css/index.htm>]
- ・ SAK Streets – SQL 開発言語資料
[http://homepage2.nifty.com/sak/w_sak3/doc/sysbrd/sak3sql.htm]
- ・ 概要 (Java 2 Platform SE 5.0)
[<http://sdc.sun.co.jp/java/docs/j2se/1.5.0/ja/docs/ja/api/index.html>]
- ・ TECHSCORE
[<http://www.techscore.com/tech/DesignPattern/>]
- ・ Welcome to Programmer's Room
[<http://park.geocities.jp/gdfsm000/poi/>]
- ・ ITpro 聞く力と話す力を磨く
[<http://itpro.nikkeibp.co.jp/article/COLUMN/20070221/262695/?ST=bizskill>]