

オブジェクト指向とは何か？

オブジェクトモデリングスペシャリスト
土屋 正人

Masato Tsuchiya
m-tsuchi@sra.co.jp

◆オブジェクト指向とは何か？

「オブジェクト指向とは何か？」

「なぜオブジェクト指向なのか？」

これまで幾度も繰り返されてきた質問です。

オブジェクト指向は、よく現実世界のメタファを使って説明されますが、現実世界をそのまま実装することは出来ません。

オブジェクト指向はひとつの考え方であり、開発の救世主となるものでもありません。

◆オブジェクト指向で出来ること

オブジェクト指向で開発すると、保守性が向上する、再利用性が向上する等、様々な利点が宣伝されてきましたが、これは間違えです。

保守性や再利用性を向上するためには、昔から使われてきた品質指標である、モジュール凝集度とモジュール結合度を向上させる必要がありますが、**オブジェクト指向の考え方をきちんと理解した上で開発を行うこと**で、はじめて、これらを達成することができます。

◆モジュール凝集度と結合度

モジュール凝集度とは、モジュールが単一の責務のために作られているかどうかを示すもので、オブジェクト

指向言語であれば、クラスやパッケージ内の機能と情報間の関連性の強さを示す指標ということになります。例えば、関連し合う機能や情報が分散していると変更時の影響範囲が広がるため、これらは局所化されていることが望ましくなります。

従って、凝集度が強いほど良い設計・実装と言えるわけです。

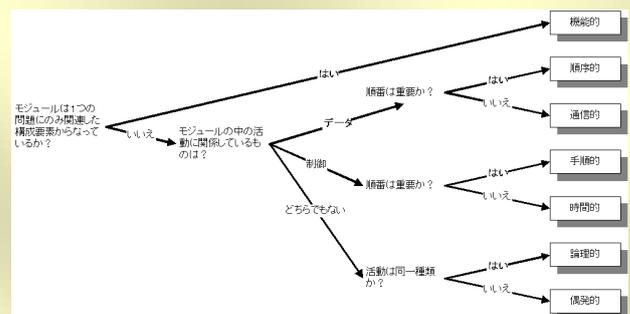
一方、モジュール結合度は、モジュールが他のモジュールからどれだけ独立しているかを示すもので、オブジェクト指向言語であれば、クラスやパッケージ間で呼び出しあうメソッドの依存の強さを示す指標です。例えば、クラスやパッケージ間の依存度が高いと、変更や再利用がやりにくくなります。

従って、結合度が弱いほど良い設計・実装と言えるわけです。

モジュール凝集度は、機能的、順序的、通信的、手順的、時間的、論理的、偶発的の7つのレベルに分類されます。機能的が最も良い(強い)凝集度で、偶発的が最も悪いものです。

オブジェクト指向の考え方を正しく理解してモジュールを設計・実装すると、モジュールは機能的強度になります。

作ったプログラムがどの凝集度レベルに相当するかを判断するためには、次の決定木が知られています。



モジュール結合度は、データ結合、スタンプ結合、制御結合、外部結合、共通結合、内部結合の6つのレベルに分類されます。データ結合が最も良い(弱い)結合度です。

ただし、他の考え方では良い凝集度と結合を達成できないというわけではありません。構造化手法であれ、データ中心手法であれ、きちんと設計することができれば達成できることです。オブジェクト指向の場合、他の考え方に比べて、カプセル化(データ抽象)や多相性などの独自のアイデアにより、これらを達成しやすくなるというだけです。問われているのは、**きちんと設計するスキル**です。

◆オブジェクト指向エクササイズ

オブジェクト指向の考え方が出来ているかどうかを判断するために、Jeff Bay の「オブジェクト指向エクササイズ」をやってみてください。

1000 行程度のコードを書いて、以下のルールに適合しているかをみてみます。

1. 1つのメソッドにつきインデントは1段落にすること
2. else 句を使用しないこと
3. すべてのプリミティブ型と文字列型をラップすること
4. 1行につきドットは1つまでにすること
5. 名前を省略しないこと
6. すべてのエンティティを小さくすること
7. 1つのクラスにつきインスタンス変数は2つまでにすること
8. ファーストクラスコレクションを使用すること
9. Getter, Setter, プロパティを使用しないこと

これらすべてに当てはまらなければオブジェクト指向になっていない設計・実装というわけではありません。しかしながら、これらを意識しながらプログラミングすることで、モジュール凝集度と結合度は**確実に良いものになる**でしょう。

◆◆◆◆ Yes? or No?

今日の開発に求められるもの ◆◆◆◆
産業第三事業部チーフコンサルタント 宮城豊

先日、仕事でインドの方とお話をする機会がありました。

通訳の方を通してですが、構築中のシステムの機能の説明をすると、担当の方が首を横に振ります。

通訳の方は、“分りました”と返します。

説明を続けている間、担当の方は首を横に振り、通訳の方は“分りました”、“大丈夫です”と肯定の返事を返します。

首を横に振ることで肯定の意を表す方々が居るということは知っていたのですが、いざ実際にそういう状況に遭遇すると、どうしても不安になり、つい、同じ事を繰り返して確認する、ということをしてしまいました。

調べてみると、インドの他に、バングラデシュ、スリランカ、ブルガリア等でも、同様の動作をするそうです。

システム開発の際に、作業の関係者全員が共通の認識を持つ為に用語の統一をする、ということは重要ですが、オフショア開発という開発スタイルが取り入れられる今日では、こういった風習の違いという点も相互に理解しておく必要が有るということを感じた瞬間でした。

SRA は、第 1 回国際カーエレクトロニクス技術展(2009/1/28-30:東京ビッグサイト)に出展します。見どころは、車車間通信、車載ネットワークエミュレータ、ソフトウェアプロセス改善コンサルティング。



「夢」を「希望」にするための架け橋としてのプロセス改善。ソフトウェアを作るのも、車を作るのも「人」。改善するのも「人」。マスコットキャラクターの“いのべー”とともに“やわらかいのべーしょん”をお届けします。

GSLetterNeo Vol. 6

2009年1月20日発行

発行者 ●株式会社 SRA 産業第三事業部

編集者 ●土屋正人、板沢智雄、柳田雅子、小嶋勉、野島勇

ご感想・お問い合わせはこちらへお願いします ●gsneo@sra.co.jp

株式会社SRA

〒171-8513 東京都豊島区南池袋 2-32-8