

パネル：オブジェクト指向教育の現実

——はじめに問題認識ありき——

中所 武司
明治大学 理工学部 情報科学科
email : chusho@cs.meiji.ac.jp

オブジェクト指向はパラダイム

■技術的知識

- ・教育 vs. 学習
- ・講義 + 演習・実習
- ・技術移転

＜問題の定式化と解決方法＞



■パラダイム

- ・伝道 vs. 求道
- ・説教 + 修行(実践)
- ・問題認識の移転

＜従来のパラダイムで解決できない問題＞

オブジェクト指向は未定義言語

— 複数の独立した概念を含む —

- WHY : 何が問題？
- WHAT : 解決方法は？
- HOW : 実現方法は？



■基本概念

分散協調型モデル／抽象データ型
／クラスとインスタンス／クラス階層と継承機能

■問題がオブジェクト指向を定義する。

— 問題と解決方法(個々の概念)の関係 —

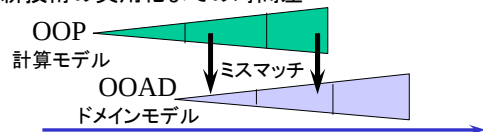
OOPとOOADは別もの

——問題認識の時間的ギャップ——

■研究開発の時間的流れ

WHY(問題)→WHAT(解決案)→HOW(実現方式)

■新技術の実用化までの時間差



- 解決すべき問題が変化すれば、対応する解決案も変化
- 時間的ミスマッチ：「未来学」や「考古学」

オブジェクト指向で何を解決？

■WHY > WHAT > HOW

- (1) 何が問題？何を解決したい？
- (2) どのOO概念が有効か？
- (3) その実現方法は？



■何を解決したいか？

開発期間短縮 / SEの生産性向上 / 再利用率向上 / 拡張性
／ 接続性 / 移行性 / EU主導で開発 / EUだけで保守 /

■システム開発技法

オブジェクト指向はモデリング技術

＜ドメインモデル / 計算モデル / データモデル＞

- 分析：業務モデルの構築
- 設計：ソフトウェア・アーキテクチャ / フレームワーク
- プログラミング：カプセル化・部品化・コンポーネント
／ クラスライブラリ / メッセージ / 並行処理

大学での教育(某教授の場合)

■＜ソフトウェア工学演習＞

- ・構造化分析 & テスト技法
- ← 現実のシステム開発の問題認識

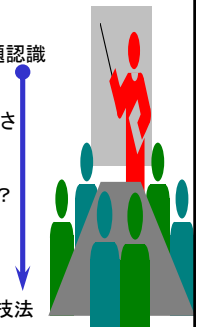
■＜ソフトウェア工学＞

- ・構造化技法(データ抽象化) ← わかりやすさ
- ・OO基本概念 ← モデリング
- ・OOP ← 分散協調型モデル
- ・OOAD ← ? マルチパラダイム?
- ・OO応用 ← オープンシステム

■＜ゼミナール・卒業研究＞

- ・昔C++, 今Java
- ・エンドユーザ指向のオブジェクト指向開発技法

■大学院＜OOAD＞



オブジェクト指向教育の難しさ

- 問題認識のない人に教える.
- 技術的バックグラウンドの異なる人に教える.
- (例) 言語屋さんとDB屋さん
- 最初に呼んだ本の影響が大きい.

パラダイムシフトの例

- アセンブラ言語 → 高級言語
- 手続き型高級言語 → 構造化技法