




ソフトウェア開発プロセスの研究動向について
 ~SPW(Software Process Workshop)2005の内容をもとに~


 奈良先端科学技術大学院大学
 情報科学研究科/情報科学センター
 飯田元
 tida@itc.naist.jp


 2005年11月25日
 38A関西プロセス分科会

本日の話題


- 最近の動向(国際会議の内容を元に)
 - ◆ SPW2005@北京(中国)


 2005年11月25日
 38A関西プロセス分科会

SPW2005
 (Software Process Workshop)


Unifying the Software Process Spectrum
 May 25-27, 2005, Friendship Hotel, Beijing, China

※河野善彌先生(EASE研究員)のご報告も参考にさせていただきました。


 2005年11月25日
 38A関西プロセス分科会


Software Process Workshop 2005

- 主催: 中国科学院軟件研究所
- 協賛:
 - ◆ USC Center for Software Engineering (米)
 - ◆ UMASS Laboratory for Advanced Software Engineering Research (米)
- 日時: 2005年5月25~27日
- 会場: 北京友誼賓館, 北京, 中國
- 位置付け: ICSE2006@上海のプレミーティング
- 構成:
 - ◆ 基調講演: Boehm 他10人
 - ◆ 論文発表: 30編(日本から2編)
 - ◆ ポスター: 19編(日本から2編)
 - ◆ デモ: 6件(日本から2件)~うち一件はNAISTから(PReP/SGT)
 - ◆ 参加者: 約150人(日本は約15名)


 2005年11月25日
 38A関西プロセス分科会

ワークショップの内容

- いわゆるワークショップではなくむしろシンポジウム。
- プロセス関連のビッグネームを一同に集めての基調講演が目玉
 - ◆ EASEの海外アドバイザー4名(Basili, Boehm, Jeffery, Rombach)を含む
 - ◆ 三日間, 午前中はすべて基調講演(一日3~4名)
- 一般の発表は, ほとんどが中国および共催団体(USC/UMASS)のもの
 - ◆ 質のばらつきが大きい
 - ◆ 個人的には興味のある発表も
 - Aspect-oriented Process Engineering
 - Process Patterns in COTS Process
 - Process Tailoring Framework
 - etc.


 2005年11月25日
 38A関西プロセス分科会

**KN1:
The Future of Software Processes**

Barry Boehm (Professor, University of Southern California)

- ◆ COCOMO, スパイラルプロセスモデルなどの提案者。
- ◆ 近年「Agileと規律」を出版, EASE海外アドバイザー。

ソフトウェアプロセス・ソフトウェア工学分野を俯瞰し, 将来の展望について述べた。

- 技術的進歩, 社会情勢・市場の変化への追従が必須である
 - ◆ 様々なカルチャー, 手法の融合
- 大規模システムではAgileとPlan-drivenプロセスの組み合わせが必要
 - ◆ Agile vs. Plan-drivenの指針: レーダーチャート
- インクリメンタルなプロセス, 手法, ツール, スキル
 - ◆ プロアクティブな技術, 市場モニタリング
 - ◆ 例) インターネットスパイラルプロセス


 2005年11月25日
 38A関西プロセス分科会

KN1: (つづき)
The Future of Software Processes

■分散化・グローバル化:エンタープライズアーキテクチャ

- ◆ SISOS (Software-intensive System of Systems) :ソフトウェア中心のシステム集合体
- ◆ システムをでたために結合していると破綻する.
- ◆ SISOSのためのプロセス管理が必要
 - ただ購入するのではなく、適応と制御の活動に近い
 - ブランドリブンな拡張やアジャイルな変更管理, バリューベースで優先順位を再編
 - 新しい外部調達の実践とスキルが必要
 - アウトソーシングのためのスパイラルプロセス(OODA):
 Observer, Orient, Decide, Act: 観察/方向付け/決断/行動 のスパイラル

CASO 2005年11月25日
SEA関西プロセス分科会

KN1: (つづき)
The Future of Software Processes

■これからはCOTSである

- ◆ 研究, 人材育成におけるCOTSの優先度を上げよ
 - プログラミングとは別の概念なので, 新しいプロセスが必要
- ◆ レガシーシステムは存続する
 ~新しいシステムを追加する前にレガシーシステムの撤去を考えよ

■先端技術への対応

- ◆ 新しいプラットフォーム, 強力な自己モニタリングソフト, より高度な抽象化, より強力なツールなど
- ◆ ワイルドカード: ロボット, バイオコンピューティング

■人材育成

- ◆ 応用レベルではプログラミングスキルの重要性が低下
- ◆ 他のドメインでのスキル, 予防・適応能力, COTS統合技術, 調達能力が, より重要
- ◆ 最も重要: 学習能力

CASO 2005年11月25日
SEA関西プロセス分科会

KN1: (つづき)
The Future of Software Processes

■ソフトウェアプロセス研究の今後

- ◆ 実証しつつ成長する(Empirically-evolved)プロセス技術
 - 言語, 方法論, マトリクス, モデル, ツール
 - インクリメンタルに曖昧さを排除
 - 不完全で非形式で部分的な仕様への対処
 - 形式度の度合いやライフサイクル, カルチャー間のギャップの橋渡し
 - 実証環境に基づいた成熟・移行の推進
- ◆ シミュレーション
- ◆ 教育用ゲーム
- ◆ 信頼性と敏捷性のバランスをとるためのバリュー・ベースのプロセス
 - SOSのためのスケーラビリティと拡張性
 - 他変数のトレードオフ分析のための一般的技術
- ◆ 技術プロセスと調達プロセスの融合
- ◆ ハイパーコンピューティングの応用
- ◆ ワイルドカード技術の応用

CASO 2005年11月25日
SEA関西プロセス分科会

KN2:
Software: A Paradigm for the Future

Watts S. Humphrey (Fellow of SEI, CMU)

- ◆ SEIの創立メンバー, CMMの主要開発者, PSP/TSPの発案者

チームワークサポートの重要性を説いた

■集団化: 生き残るため⇒国家の形成と征服のため⇒工業製品の生産のため

■次のチャレンジ:
 知的生産物の大規模生産に対する科学的手法の適用
 (Intellectualization)

- ◆ ハードウェアで達成したような生産性向上をソフトウェアで達成できないのか?
- ◆ これまでの経過は思わしくなかったが見通しは明るい

CASO 2005年11月25日
SEA関西プロセス分科会

KN2: (つづき)
Software: A Paradigm for the Future

■コストとスケジュールの管理

- ◆ ハードウェアの手法に倣いCMM, ISO-9000などが登場(1986)し, それなりに受け入れられ成功している

■品質の管理

- ◆ ハードウェアの手法は適用可能(Cleanroom, Praxis, PSP/TSPなどによる実績)
- ◆ しかし: 個人による品質や故障の管理方法はハードウェアの場合と大きく異なる(マネジメントの優先順位やスタイル)

■人の管理

- ◆ 知的作業のために集中管理を超える方法を考案できるか
- ◆ Demingらはハードウェア分野においてひとつの方法を示した(自己管理が鍵である)
- ◆ TSPはソフトウェアチームが管理者とうまく交渉できることを実証しつつある
- ◆ 継続的なチームのコーチングとマネジメント支援が必要

CASO 2005年11月25日
SEA関西プロセス分科会

KN2: (つづき)
Software: A Paradigm for the Future

■なぜソフトウェアはやっかいなのか

- ◆ ソフトウェア自体の性質: 複雑で様々な解釈がありうまく定義できない
- ◆ 様々な規模のプロジェクト(一人~数千人)
- ◆ 現在では多くのビジネスにとってクリティカル

■セキュリティの問題: Defective software is not secure

- ◆ 被害の90%がソフトウェアの欠陥につけこんだもの
- ◆ 欠陥の上位10件が脆弱性全体の75%の原因に(CERT/CC)
 - 典型的なアプリケーションは1KSLOCあたり1~7個の欠陥
 - もしそれらの欠陥のたった5%でも脆弱性につながるなら, MSLOCあたり50~350の脆弱性があることに..
 - 大規模ユーザは典型的には\$250,000以上を修正のインストールに費やしている
- ◆ テストではすべての欠陥を除去できない
 - どれくらい欠陥が多いとダメなのかははっきりとは決められない

CASO 2005年11月25日
SEA関西プロセス分科会

KN2: (つづき)
Software: A Paradigm for the Future

- 品質管理の戦略 (PSP/TSPベース)
 - ◆ できるだけ優れたデザインメソッドを用いる (現実レベルでフォーマルなもの)
 - ◆ 要求・設計・コードをすべてレビューし、欠陥をトラッキングする
 - ◆ 見つかった欠陥を分析する (修正、類似欠陥の発見、プロセスの改善)
 - ◆ PQI (Process Quality Index) を用いたプロセス品質の評価
 - 設計/コーディング時間, デザインレビュー時間, コードレビュー時間, コンパイル品質, 単体テスト品質に基づく指標
 - TSPはCMM5よりはるかに優秀
- この戦略を有効に用いるため必要なこと:
 - ◆ PSP/TSPベースの規律をチーム・個人で身につける
 - ◆ セキュリティの重要性を認識し、脆弱性を防ぐ方法を知る
- 教育が重要
 - ◆ やってみるまでは誰も信用しなかった
 - ◆ PSPコース: 10個のプログラム開発. 徐々にPSPの実践を導入.

CASC 2005年11月25日 32人関西プロセス分科会

KN3:
Integrated Software Process & Product Lines

H. Dieter Rombach (Prof. Univ. Kaiserslautern / Fraunhofer IESE所長)

- ◆ 産学連携のビジネスモデルで成功 (IPA/SECやJAXA, リコーなどもクライアント)
- ◆ EASE海外アドバイザー

ソフトウェア・プロセス&プロダクト・ラインのコンセプト紹介をおこなった

- プロダクトとプロセスの両方においてPro-active (事前) の再利用を行うことが今後の鍵
 - ◆ Reactive: 保存しておいて役立つよう祈る
 - ◆ Proactive: 再利用を見込んで予め計画
- 再利用のためには対象 (プロセス/プロダクト) のテイラリングは必須
 - ◆ 事前にパラメータ化, テンプレート化などを用いて準備し, 対象プロジェクトのコンテキストに基づいて変更
 - ◆ 共通部分 (コモンリティ) と可変部分 (バリエビリティ) の切り分けを行う
 - プロダクトにおいては成功: プロダクトライン手法の紹介
 - プロセスの再利用はまだ未成熟
 - プロダクトライン手法をプロセス自身に適用することを提案

CASC 2005年11月25日 32人関西プロセス分科会

KN3: (つづき)
Integrated Software Process & Product Lines

- ソフトウェアプロセスの問題点
 - ◆ 事前に規定されたプロセスと現実との乖離⇒ ガイダンスが得られない, 測定ができない, 知見のフィードバックができない
 - ◆ 組織標準プロセスと分野やプロジェクトに特化したプロセスとの関係が定義されていない
 - 現場から上がってこずにトップダウンで規定された標準プロセス
 - 総括的で巨大な標準プロセス (共通点を抽出するよりもむしろすべての特徴を併合して作成されるため)
 - テーラリングのためのガイダンスがない

※新しいVモデルXT (ドイツ公共プロジェクト用標準プロセス定義) では テーラリングガイドラインが含まれている

CASC 2005年11月25日 32人関西プロセス分科会

KN3: (つづき)
Integrated Software Process & Product Lines

- ソフトウェアプロセス・ライン
 - ◆ 例
 - 自動車サプライ企業のための組込ソフト開発ライフサイクル・モデル
 - セーフティ・クリティカル・デバイスのためのライフサイクル・モデル
 - コンフォートデバイスのためのライフサイクルモデルなど
 - ◆ プロセスラインの形成
 - トップダウン: 標準規格+テーラリングガイド
 - ボトムアップ: プロジェクト実行プロファイルからプロジェクト間での共通部分を抽出
- プロセスラインとプロダクトラインの統合 (まだ提案段階)
 - ◆ 両方の統合リポジトリを形成する
 - テーラリングが容易になる
 - プロジェクトをまたがった学習が容易になる (共通部分と可変部分が明確に見えるようになるから)

CASC 2005年11月25日 32人関西プロセス分科会

KN4:
Unifying Microprocess and Macroprocess Research

Leon J. Osterweil (Prof., University of Mass.)

- ◆ プロセスプログラミングの提唱者, ICSE2006@上海のGeneral Chair

プロセスに対するミクロな視点, マクロな視点と両者の関係を紹介し, それらの橋渡しとしてのプロセス言語の位置づけを語った

- ◆ マクロな視点: プロセスが何をするか, どう振舞うか
- ◆ ミクロな視点: 振る舞いに何がどう影響するか

- プロセスはファーストクラスオブジェクト
 - ◆ 入力=アーティファクト, リソース (要員, 予算, ツール, 時間)
 - ◆ 出力=アーティファクト/効果, 他の振る舞い (コスト, 時間, エラー)

CASC 2005年11月25日 32人関西プロセス分科会

KN4: (つづき)
Unifying Microprocess and Macroprocess Research

- マクロ視点での研究 (オブジェクトの外部仕様)
 - ◆ オブジェクトの振る舞い
 - ◆ 入力関係, 入力の変化に対応した反応 (コスト, スピード, 品質)
- ミクロ視点での研究 (オブジェクトの内部メカニズム)
 - ◆ オブジェクトがどう機能するか (入力がどう利用されて出力に結びつくか, 振る舞いにどう影響するか)
 - ◆ リソースをどのように消費するか, 新たなリソースを投入したらどう使われるか
 - ◆ 現在と異なる新たな振る舞いをするか
- マイクロとミクロとの橋渡し
 - ◆ どのようにして振る舞いが形成されるのかを説明する
 - ◆ 変更の示唆と効果の予測
 - ◆ 変更に対する検証

CASC 2005年11月25日 32人関西プロセス分科会

KN4:(つづき)
Unifying Microprocess and Macroprocess Research

■例) フェーズ指向開発プロセス

- ◆ 初期開発:
 要求,アーキテクチャ,設計,実装,テスト・検証, ライフサイクルの分析確認
- ◆ 継続的発展:
 分析とテストにより誘導
- ◆ マクロプロセスの力点=
 振舞: 要求, アーキテクチャ, テスト・検証をどのように進めるか
- ◆ マイクロプロセスの力点=
 機能するか: コーディング, 分析, テスト・検証で何が行われるか
- ◆ 継続的発展には両アプローチの統合が必要

CASP 2005年11月25日 38人参加プロセス分科会 19

KN4:(つづき)
Unifying Microprocess and Macroprocess Research

■マイクロプロセスのためのツール

- ◆ 記述言語と実行セマンティクス(デザイン検証のサポートのため)
- ◆ アナライザ(正しく振舞うか・検証, 発展的変更による影響予測)
- ◆ テストツール(パフォーマンスの観察・計測, 要求との比較, 変更決定の手がかり)

■プロセス記述言語に対する要件

- ◆ 正確性, 完全性, スコープの広さ, 明確さ, 適度な抽象化
- ◆ ダイアグラムでは不十分(マイクロ)
- ◆ リワークの扱い
 - リワークは元の状態に戻ることでない
 - 新しいコンテキストで以前のプロセスを実行すること
- ◆ Little-JILダイアグラム/セマンティクスの紹介

CASP 2005年11月25日 38人参加プロセス分科会 20

KN4:(つづき)
Unifying Microprocess and Macroprocess Research

■プロセスアナリシス

- ◆ 静的解析: 欠陥の特定, 変更の影響予測, 最悪時パフォーマンスの予測
- ◆ 実績:
 - オークションプロセスの欠陥指摘
 - 労働管理プロセスにおけるプライバシー侵害リスクの発見
 - 医療プロセスにおける欠陥, 誤りの発見
 - 投票プロセスにおける問題と改善点の指摘
- ◆ 動的解析: プロセス実行のモニタ(支援, 誤りの指摘), シミュレーションによるパフォーマンステスト, リソース配置の最適化
- ◆ 実績:
 - オークションプロセスのシミュレーションで「共謀」を再現
 - 医療プロセスのシミュレーションも計画中(要員配置, オーバーロード検知)

CASP 2005年11月25日 38人参加プロセス分科会 21

KN5:
Achieving Software Development Performance Improvement Through Process Change

Ross Jeffery, (Prof., Univ. New South Wales, Australia / NICTA)

- ◆ 産学連携プロジェクト組織NICTAをマネジメント

プロセスイノベーションの概念と実践事例の紹介をおこなった

■NICTAの説明

■動機

- ◆ CMM/CMMI 1v.5の認証を受けても実際には生産性はあがっていなかったり, ばらばらで, 改善に役立っていない
- ◆ 現状のプロセスを書いても改善にはあまり役立たない
- ◆ 中小企業はCMMIAセメントを改善手段として却下
- ◆ もっと広い視野での改善をすべき

■Process Improvement からProcess Innovationへ

CASP 2005年11月25日 38人参加プロセス分科会 22

KN5:
Achieving Software Development Performance Improvement Through Process Change

■背景

- ◆ プロセス改善のコンテキスト(Morton MIT90's Framework)
- ◆ プロセス研究のフレームワーク(Sambamurthy & Kirsch, 2000)
 - ファクタ指向
 - 説明としてのプロセス
 - コンセプト分類としてのプロセス
 - イベント系列としてのプロセス

■プロセスイノベーション(Davenportのモデルに基づくもの)

- ◆ 改善とは違う
- ◆ 明示的に実施されなければならない
- ◆ テクノロジに依存する
- ◆ 人と組織を大規模に変更する
- ◆ 戦略にリンクされたビジョンによってガイドされるべきである
- ◆ すべてのタイプの企業やプロセスで実施可能である

■Allete Systemにおける成功例

- ◆ いくつかの支援ツール
 - プロセスガイド(EPG), コスト予測ツールなど

CASP 2005年11月25日 38人参加プロセス分科会 23

KN5:(つづき)
Achieving Software Development Performance Improvement Through Process Change

	プロセス改善	動的プロセス改善	プロセス革新
変更の度合い	インクリメンタル	インクリメンタル	ラディカル
開始ポイント	既存プロセス	既存プロセス	何もない状態から
変更頻度	1回/継続的	継続的	1回
必要な時間	短い	長い	長い
参加形態	ボトムアップ	トップダウン/ ボトムアップ	トップダウン
典型的なスコープ	機能に限定	機能間にまたがる	機能間にまたがる
リスク	限定	限定	高い
プライマリ・イネーブ	統計的制御	テクノロジー	テクノロジー
変更のタイプ	文化的	個人の役割とスキル/ テクノロジー/マネジメント	文化的/構造的

CASP 2005年11月25日 38人参加プロセス分科会 24

KN6: What Beyond CMMI is Needed to Help Assure Program Success?

Arthur Pyster (Senior Vice President, Science Applications International Incorporated)

- ◆ システムのコンサルタント等の後1997-2004の間米連邦航空局(FAA)に勤務

FAAでの調達の実験などを通じ、CMMIに不満な点、今後の展望などを紹介

- FAAの紹介:
 - ◆ CMM/CMMIやISO9000の使用経験長い
 - CMMI : 多くがレベル3か4
 - ISO9001:2000取得部署も
- 米政府(国防省)の見解:
 - ◆ CMMIで高い成熟度を達成した組織では、当然後続のプロジェクトが高い成熟度をもって実施(=コスト、スケジュール、パフォーマンスが計画通りに達成)されるものと期待している。
 - 現実にはそうならない...

CASO 2005年11月25日 25 3rd SEA関西プロセス分科会

KN6: What Beyond CMMI is Needed to Help Assure Program Success?

- CMMI成熟度レベルを解釈する際に調達者が犯す4つの間違い
 - 1) アプライザは小さなサンプルを見るだけですべてのプロジェクトを見るわけではない
 - 大規模な組織なら、成熟度の高いプロセスを組織的に使っていないと高いレーティングを受けることが可能
 - 多様なビジネスを行っている組織では、代表するサンプルの選択が困難
 - 選択の仕方によっては不当に高すぎるまたは低すぎるレーティングを受ける
 - 2) プログラムを評価なしに判定はできない
 - 契約時にはチームが未完成
 - 実際には高成熟プロセスを使わない可能性
 - 調達側の時間や金の制約で評価を行わないことも
 - プロセスの経時劣化の可能性
 - 分野違いプロセス適用の可能性
 - テーラリングに失敗する可能性

CASO 2005年11月25日 26 3rd SEA関西プロセス分科会

KN6: What Beyond CMMI is Needed to Help Assure Program Success?

- CMMI成熟度レベルを解釈する際に調達者が犯す4つの間違い
 - 3) 低い成熟度のアクワイアは高い成熟度のサプライヤと衝突する
 - 安定した要求を維持できなかったり、契約変更を管理できない
 - 新しい大規模調達に慣れていない
 - システムがすぐに必要でスパイラルなプロセスより「ビッグバン」方式をとる
 - 供給者側の高成熟度に無理解で品質管理やテストに金を払わない
 - 4) 優れたプロセスだけでは不十分である
 - プロセス、人、環境、技術の4つが不可欠
 - CMMIには大規模システムによるエンタープライズのフォーカスが欠如: FAAの新通信システムサービスは2年の設計(CMMIのフォーカスはこの後、4-5年掛けて配置され、さらに10年運用される

CASO 2005年11月25日 27 3rd SEA関西プロセス分科会

KN6: What Beyond CMMI is Needed to Help Assure Program Success?

- 今後の課題
 - ◆ CMMIに基づくプロセス改善のコミュニティは、CMMI設計に深く関わり、スポンサーとなった米政府の人間と緊密に仕事をを行うべき。
 - ◆ 米政府のスポンサーを継続するためにも、上に上げた問題点の解決を行い、調達の品質を上げることが急務である
 - ◆ 質疑でのコメント:
 - CMMIの一番の問題点は委員会が編纂されたこと。あまりに複雑で膨大。
 - むしろSW-CMMのほうがハンフリーらごく少数のスタッフで作成されたのでコンパクトでよくまとまっている

CASO 2005年11月25日 28 3rd SEA関西プロセス分科会

KN7: Active Models: A Possible Approach to the Integration of Objective and Subjective Process Models

Bryan Warboys (Prof. University of Manchester)

プロセスとプロダクトを統合したモデルによる進化的開発アプローチの提案

(コンセプトの話ばかりで実践が不明のため本日は省略)

CASO 2005年11月25日 29 3rd SEA関西プロセス分科会

KN8: Expanding the Horizons of Software Development Processes: A 3-D Integrated Methodology

Mingshu Li (中国科学院軟件研究所長)

- ◆ SPW2005ホスト

ソフトウェア工学およびソフトウェアプロセスのトピック・分野を総括してコメント

- Technology-Human-Processの3視点を統合した開発モデル(TRISO-M)の提案とそれに基づくソフトウェア工学環境(TRISO-DEVELOPER)の構築
 - ◆ Human → Cost
 - ◆ Process → Management
 - ◆ Technology → Product
- 大組織で包括的に活動中(研究、プラットフォーム開発、ツール開発)
 - ◆ Human(コスト)視点ではコスト予測にCOCOMO-IIを適用

CASO 2005年11月25日 30 3rd SEA関西プロセス分科会

KN9: Empirically Evolving Software Engineering Techniques*

*プログラムのタイトルは"Evolving Defect 'Folklore': A Cross-Study Analysis of Software Defect Behavior"

Victor R. Basil (Prof. Univ. of Maryland / Fraunhofer MD)

- ◆ NASA SEL創設者, GQMパラダイム, Experience Factoryの提唱者
- ◆ EASE海外アドバイザー

Empirical Software Engineering パラダイムと関連する活動の説明

- 基本コンセプトの紹介
 - ◆ QIP (Quality Improvement Paradigm)
 - ◆ Experience Factory
 - NASA/SELでの実績紹介
- 関連プロジェクト
 - ◆ CeBASE (Center for Empirically Based Software Engineering)
 - ◆ NASA High Dependability computing Program
 - など
- 最後にもとのタイトルの内容を紹介
 - ◆ SELのデータをもとに実証的にソフトウェア欠陥の分析や分類を行った事例

CASP 2005年11月25日 31 第1回関西プロセス分科会

KN10: Rigorous Software Process For Development Of Embedded Systems

Wilhelm Scheafer (Prof. University of Paderborn, Germany)

組込みソフトウェアのモデル駆動型開発 (MDD: Model Driven Development)を行った事例をもとに, そのプロセスを解説

- ◆ プラント制御
- ◆ 次世代自律運行型ミニトレインシステム

(プロセス自体の話はあまりなかったので省略)

CASP 2005年11月25日 32 第1回関西プロセス分科会

KN11: Software Are Processes, Too

Jack Estublier (Prof., Genoble Univ. France)

MDA(Model-Driven Architecture)に基づいてプロセスやメタプロセスの話をした.

- プロセスのモデル階層は, ソフトウェアそれを扱う人間のモデル階層より一段抽象度が上にあり, メタプロセスのモデル階層はさらに一段上にある, といった話

(概念整理以上の意味はあまりなさそうなので詳細は省略)

CASP 2005年11月25日 33 第1回関西プロセス分科会

一般講演:

- Aspect Oriented & Software Process
 - ◆ Stanley Sutton Jr. (IBMワトソン研究所 研究員, Arcadiaプロジェクト等でのプロセス言語設計の中心)
 - ◆ AOP / SOP (Subject-oriented)
 - アスペクト: モジュール間にまたがって存在するフィーチャー
 - AOIは product, process, process languageのいずれにも有効
 - プロセスプログラミング必須ではない(OOSDとOOPの関係と同じ)
 - ◆ プロセス言語の観点からは:
 - 言語自体をいじる・いじらない二つのアプローチ
 - アスペクトごとの言語セットを用意
 - マクロプロセスとマイクロプロセス
 - マクロ: プロセスアーキテクチャ, ふるまい, ...
 - マイクロ: 厳密, 簡潔, 正確, 検証
 - メソプロセス (Mesoprocess) 二つをつなぐ中間層
 - プロセスエンジニアリング
 - 言語選択, 環境設定
- Process Patterns for COTS-Based Development
 - ◆ Ye Yang
 - ◆ COTSベースでの開発プロセスのパターン
 - ◆ 三つのパターン層を提案
 - ライフサイクル・パターン: プロダクトとデザインの連鎖
 - アクティビティ・パターン:
 - アセスメント (A)
 - テイラリング (T)
 - グルー (G)
 - カスタム (C)
 - ワークフロー・パターン: A.T.G.の内部のワークフロー (それぞれをパターンと呼んでいる)

CASP 2005年11月25日 34 第1回関西プロセス分科会

総括:

- ちょっとした同窓会?
 - ◆ ICSP1~4, IWSP1-9以来, ICSE以外でこれだけそろるのは珍しい
 - ◆ 中国の若い研究者には良い勉強の機会? (河野先生の印象)
- 中国はこれからソフトウェア・プロセスにも力を入れるのか?
 - ◆ YESだが, 3Dアプローチなどの有効性や本気度は正直言ってわからなかった.
 - ◆ SPWで査読した限りでは中国からの論文は質のばらつきが大きかったが:
 - ソフトウェア工学/プロセス系の論文投稿数は年々増加
 - プロセス中心型統合環境の開発を根気強くやっている研究者もいる
 - ◆ USC, UMASSと軟件研は活発に交流している模様
- 日本では?
 - ◆ 大学における理論系の研究は80-90年代で一段落 (東大, 京大, 東工大, 阪大など), 実践系の活動は企業が主体に (JASPIC, SEA-SPINなど)
 - ◆ 企業や政府からの根強い需要はあるので, 大学側であらためて活動を盛り上げるには良い契機かもしれない

CASP 2005年11月25日 35 第1回関西プロセス分科会