



アプリケーション開発の最新状況を知る



2020年問題によるソース不足や就職の不人気、ビジネスモデル変更を迫るクラウドの台頭など、SIerやユーザーによるアプリケーション開発の現場は転換点を迎えている。NECの大場氏は「アプリケーション開発近代化への挑戦！～設計リポジトリの活用による設計情報の管理と品質向上～」と題し、そうした開発現場の状況と、今後どのような方向に進むべきかについて各種提案を行った。



最近のアプリケーション開発の現場

これまでのXupperユザ事例紹介セミナーで、幾度となくテクノロジーとプロジェクトマネジメントの最新動向を伝えてきたNECの大場氏。今回のセミナーでは、最近の開発現場を巡る状況を振り返りながら、労働集約型開発から知識集約型開発への道筋と具体的な対策を提示した。

大場氏はまず、最近のアプリケーション開発の傾向として、大きく3つの傾向があると指摘。1つめは開発期間の増加と開発コストの低下だ。目的や要求が複雑化して、要件定義や外部設計に多くの時間がかかるようになる一方、年率10%程度のコスト削減要求が求められているという。

2つめは、ライフサイクルや作り込みへの要求が強まったこと。

新規にシステムを開発するのではなく、システムを変更しながら継続的に使用する傾向が強くなっている。そのために、システムのライフサイクルを通じて「早く安く良く」開発すること、効率的に高品質を維持することが求められるようになった。

3つめは、開発リソースの確保が困難になってきたこと。SEは、新3Kと呼ばれるほど人気が低迷しており、特に優秀なリソースの確保が難しい。一方で熟練した高スキルのIT技術者は高齢化しているのが現状だ。

大場氏は「理系の就職人気ランキングを見ると、2010年以降、ITに関わる企業は上位10社に1社もランクされていない状況が続いている。人気だけでなく、実際に、優秀な人材が業界に入ってきていない」と人材不足の深刻さを語った。

労働集約型開発から知識集約型開発へ

そのような中、目指すべき開発手法として提案しているのが「知識集約型開発」だ。従来の開発手法は「労働集約型開発」で、中流工程（製造～評価）以降に人的リソースを規模に応じて投入する。開発要員のスキルにより生産性が変動するのが特徴だ。これに対し、知識集約型開発は、開発要員の労力を上流工程（設計）や評価試験の設計に注力させ、中下流工程はできるだけツールによる自動化を行う開発スタイルとなる。

■ AP開発テクノロジー		特徴	母体品質確保	開発期間短縮	追加・変更が容易な設計	影響調査が確実・容易	バグ混入防止
①アドバンスドDOA/SDE	レベル化	業務フローを徐々に詳細化（レベル化）し顧客との間で発生しがちな仕様齟齬・誤認識を防止。 上流の設計手法が統一され、下流での実現に依存しないため、要員のローテーションが可能。	○		○		
	設計ツールメタ連携	画面自動生成	○	○			○
		テストシナリオ自動生成	○	○		○	○
		クロスリファレンス	○		○	○	○
②業務モデル化/フレームワーク	純粋な業務処理に専念できるためAPの製造・試験の生産性が向上	○	○	○			
③テスト自動化	自動でテスト実行するため短時間で全てのテストができ、設計以降で作りこまれたバグを全て抽出可能。	○	○			○	
④イテレーション開発	机上で設計した画面の操作性や業務仕様の適性を早期に評価可能。 業務フローやUIの早期確立と開発要員の熟練度/スキル向上が可能。	○	○				

アドバンスドDOA/SDE		フレームワークベース開発		テスト自動化
要求分析	方式設計 (基本設計・実装設計)	詳細設計	製造・UT	評価 (AP&PF)
イテレーション開発 (AP開発&PF構築)				

図1：知識集約型ソリューションによる狙いと効果

特徴は大きく3つあり、(1) 上流工程の設計や評価試験の設計に時間と労力をかける、(2) ツールの自動化によって中下流工程を効率化させる、(3) 設計情報の共有および維持フェーズにおける効率化のために、設計情報を一元管理し、メンテナンス性の向上やデグレードの抑止を行う、となる。

「品質の作り込みを上流で行うが、仕様は必ずしも確定するわけではない。設計情報を一元管理することで、各フェーズで手戻りは容易。製造やテストは、自動化によって、作り込みバグのゼロ化を目指す」(大場氏)

つまり、短期開発やリソース不足、品質の作り込みへの要求、追加・変更開発によるバグ混入などの課題に対応できるアプローチというわけだ。そのうえで、大場氏は、知識集約型開発を推進するうえでのポイントは、アプリケーション開発のテクノロジーとプロジェクトマネジメントの融合だと主張した。

知識集約型 AP 開発のためのテクノロジーとツール

アプリケーション開発テクノロジーとしては(図1)のような技術を組み合わせて利用する。たとえば、「アドバンスド DOA (データ中心設計)」や「SDE (NECのSystemDirector Enterprise)」といったツールを使って、上流の設計情報からテストシナリオを自動生成するにすれば、「母体品質確保」「開発期間短縮」「影響調査が確実・容易」「バグ混入防止」といった効果が期待できる。

あわせて、プロジェクトマネジメントとしての取り組みを進めることで、効果を高める。たとえば「OODA (ウーダ)」と呼ばれる「観察 (Observe)」「予想 (Orient)」「決断 (Decide)」「実行 (Act)」をすばやく繰り返すことで迅速な意思決定を行うことで、プロジェクトが予定通り健全に遂行されているかを判断できるという。

大場氏は、こうした開発テクノロジーやツールでは、リポジトリが重要になると指摘した。

「リポジトリは、リポジトリ構造の中核であるビジネスフローを使って、ビジネスの世界とシステムの世界をつなぐ決め手になる“手続きの流れ”と“情報の流れ”を表現できる。また、リポジトリ情報を活用したインパクト分析が可能だ。階層化したレベルを上位からドリルダウンして、影響を受ける機能を特定できる。特定した機能の利用状況を相互参照にて分析し、変更に対する影響範囲

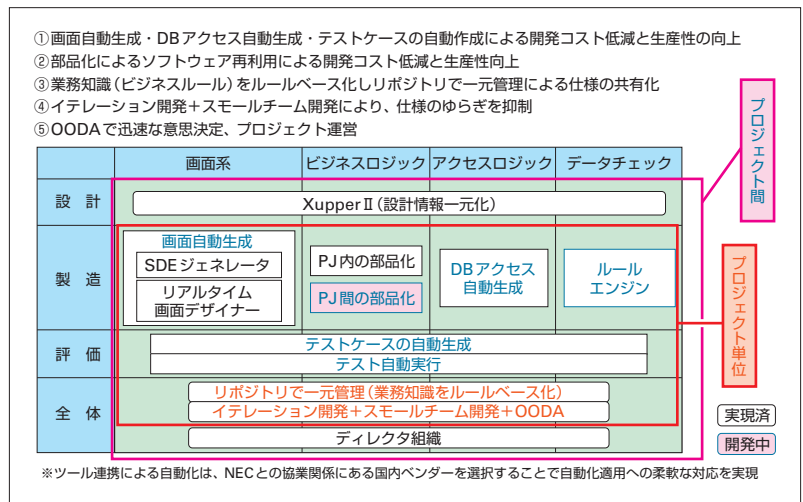


図2: 知識集約型 AP 開発ソリューションのツールマップ

を洗い出すことができる」

SDEを使えば、リポジトリからソースを自動生成することもできるという。また、テストについては、Xupper IIとテスト自動化ツールを組み合わせ、テストシナリオの自動作成からテストの自動実行を行うこともできるとした(図2)。

プロジェクトマネジメントで注目すべきポイント

プロジェクトマネジメントのポイントは、まずは、PDCAサイクルに加えて、先ほどのOODAサイクルを導入することだ。PDCAがQCDを確保(特に品質の確保)する取り組みだとすれば、OODAは、QCDの確保(特に品質の確保)がなされているかを確認することで、意思決定を迅速化する取り組みになるという。

また、組織体制としては、スペシャルSEによるスモールチームが重要だ。少数精鋭にすることによって知識の集約化を図り、多くの場数を踏ませることで優秀なSE(スペシャリスト)を育成していく。その際のチームビルディングでは、俊敏性を高めるために、「相互信頼」「直観的意思決定」「契約的リーダーシップ」「重点・焦点・方向づけ」という4つの属性を考慮する。これら4つは、OODAループを回す上でもポイントになるという。

開発自体は、スモールチームによるイテレーション開発のかたちとなる。イテレーション開発により、開発要員の習熟度・スキルの向上を実現し、開発リスクを低減することができるという。

最後に大場氏は「品質は上流工程で作られる。もし上流工程で作りがまなければ、後工程でそれを確保しようとしてもできない」と、品質確保における上流工程の重要性を重ねて強調した。