大規模企業情報システムの開発・管理 の実際

富士通(株)ソフト・サービス共通技術センター 橋本 惠二

hashimoto.keiji@jp.fujitsu.com

企業の情報システム開発は、従来になく大きなリスクを伴うものになってきた。それに対応するため、SI 企業が、マネジメント、エンジニアリングの双方から、どのような取組みを行っているかを述べる。

企業情報システムの変遷

企業情報システムの形態はここ十数年で大きく変化してきた. 80年代のメインフレーム中心のシステムは,90年代のクライアント/サーバ (C/S) 技術を経て,今では Web 技術をベースとするシステムが主流になっている.

システム技術の変遷とシステム化ニーズが合致して、 多くの企業情報システムは、複雑さを増大させながら 拡張されてきた。C/S技術は、それまでメインフレー ム上ですべてを開発してきたアプリケーションのうち、 マーケティングや人事総務などの情報活用業務や特定部 門業務の機能を切り出すかたちで新しいシステム形態を 作り出した。Web 技術は、企業外の一般消費者や他企 業との直接のチャネルを開き、e-Commerce や官庁の電 子申請システムなど、企業フロントの新しいシステム形 態を生み出した。他方、Web 技術は C/S 技術より運用 や変更コストが削減できるため、既存システムの Web ベースへの移行がここ数年急速に進んでいる。

この結果、企業情報システムは図-1に示すような構成が典型的になっている。企業外との窓口である Web ベースのフロントシステム、人事など部門業務で C/S システムからの流れをくむミドルシステム、そしてメインフレームを含む基幹業務のバックシステム、という形態である。それぞれのプラットフォーム上で適用され

る開発言語やミドルウェアもさまざまであり、この十数 年で、企業情報システムは相当に複雑なものとなって きた.

システム開発における混乱

このようなシステム環境の変遷は、システム開発にも大きな混乱をもたらした。かつてメインフレーム上では、1種類のプラットフォーム上で、OLTPや DBMS などのミドルウェアも固定されたものを利用し、基本的には1種類の言語(COBOL)で開発を行ってきた。開発標準、開発環境、開発管理ノウハウは、この安定な土台の上に積み上げられ、成果を上げていた。しかし、オープン化への急激な移行により、この積み上げが必ずしも継承されずに現在に至っている。

なかでも、アプリケーションの開発技術には、大き

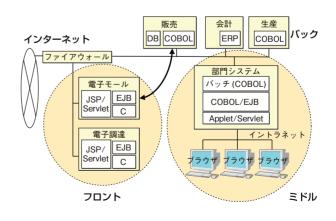


図-1 典型的な情報システムの形態

なギャップが存在する. その要因は2つある. 1つは, COBOL から Java などへの言語の変化である. COBOL 上に長年積み上げたノウハウを、Java ベースの環境に 読み替え、人材を再教育することが必ずしも成功してい るとはいえない、2つ目の要因は、アプリケーションの 実行環境の知識が、従来よりも要求されるようになった ことである. I2EE は. Web 環境での標準プラットフォー ムとして受け入れられてきた. しかし. ビジネスアプリ ケーションを構築するにはサービスが不十分といわざる を得ない. そのため、トランザクションログ取得、セッ ション管理、バッチ処理のリカバリなど、従来はメイン フレームのプラットフォーム機能として用意されていた ものを、アプリケーションで実現する必要が出てきた.

もう1つの重要な変化は、企業情報システムのビジネ ス環境にある. 企業合併や事業再編成といった環境の急 変、新サービスの開発スピード競争などの結果、情報シ ステムの開発, 再構築, 拡張に許される期間は大幅に短 縮している.一方で.90年代後半からの長期的な不況は. システム開発の予算を縮小した. その結果, 大規模シス テムの開発を、従来では考えられないような短期間と低 予算で実施せざるを得ない例も少なくない.

企業情報システムを取り巻く環境は、このように急激 に変化しており、システム開発の発注者、開発者の双方 にとって、大きなリスクを伴うものとなっている. 以下 では、このリスクをコントロールしていく方策を、富士 通の取組みを例に、マネジメント、開発技術の両面から 説明するとともに、残された課題に言及したい.

組織的なリスクマネジメントの事例

富士通の企業情報システム関連のサービスを担う技術

者, いわゆるシステムエンジニア (SE) は, 国内グルー プ企業を含めて42,000人を数える. サービスの中で、 情報システム開発を主とするシステムインテグレーショ ン (SI) ビジネスは, 一定規模以上でも, 国内で年間 1,000 プロジェクト近くを手がけている. 扱っている業種は, 金融(銀行,保険,証券,クレジット等),産業(製造, 流通, 運輸, 建築等), 公共(官公庁, 自治体, 学校等), 通信, 電力, ガス, 放送, 新聞, 医療, と多岐にわたる 1). SI プロジェクトには、100 万ステップを超えるソフト ウェア開発を伴う大規模プロジェクトが含まれる. これ らプロジェクトは、数は全体の数%に過ぎない、しかし、 金融機関や大企業の工場操業システムなど、納期、品質 の問題が顧客のビジネスの成否や、社会的影響に直結す るシステムを対象とするプロジェクトが大半である.「失

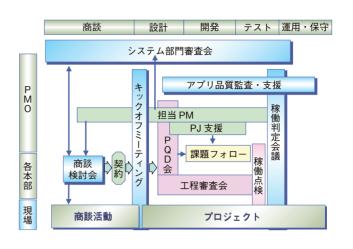


図-2 PMOによるプロジェクト監査の仕組み

敗が許されない」プロジェクトでリスクをいかに回避す るかは、富士通にとって、社会的責任でもある。

このような背景から、プロジェクトマネジメントオ フィス (PMO) 活動が 2001 年 4 月からスタートした. PMO 活動とは、個々のプロジェクトのリスクを組織的 にコントロールし、「長期的、短期的に組織内のプロジェ クトリスクを全体として極小化する」取組みである²⁾. 当初、10名程度のプロジェクトマネージャ(PM)で 個別プロジェクトの第三者モニタと改善指導から開始 した. 2002 年から、PM プロフェッショナル認定制度、 PMBOK に基づく PM ナレッジの体系化によって、組織 的にPMを育成する仕組みを整備してきた。2002年末 には、PMO を冠した組織に拡大し、あわせてシステム 開発技術を専門とする組織を併設することで、システ ム開発のマネジメントとエンジニアリングを両輪とする PMO が完成した³⁾.

現在, PMO の中心的な活動は, 図 -2 に示すようなプ ロジェクト監査の仕組みに要約できる.

第1段階の商談検討会では、顧客との契約以前の商談 段階で、適用技術、見積り、体制などの観点から初期の リスクを判定する.契約後の最初に行われるキックオフ ミーティングでは、プロジェクトの当事者が集まり、プ ロジェクトの特性、リスクの共通認識を持つことを主 眼とする. その後, PQD (Project Quality Design) 会 での初期設計の品質チェック, 各工程完了時での品質 チェックを行う.

これらの活動は、認定された約300名のPMがプロ ジェクトに専従することにより徹底される. さらに、後 述するプロセス改善活動の一貫として、アプリケーショ

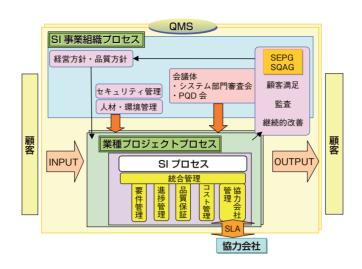


図-3 プロセス改善・品質保証活動のフレームワーク

ン品質の監査と改善活動が、PMO から派遣される専門家によって実施される。

このような仕組みによって、多くのプロジェクトに対してプロジェクト管理、IT、品質の専門家を有効に配置でき、個々のプロジェクトでは見えにくい客観性のある問題把握と、早期の対策を可能にしている.

プロセス改善と品質保証

上記の活動が、全社的に重要なプロジェクトに対するリスクマネジメントを主眼としたのに対して、各 SI 事業組織の日常的、継続的な仕事のプロセスを改善する活動も展開している。金融、産業、公共などの業種ごとの SI 事業組織では、常時複数のプロジェクトを運営し、それぞれの環境にあわせた SI プロセスを持っている。各プロジェクトの品質は、各事業組織が規定する SI プロセスの顕れであり、継続的なプロセスの改善なくしては、組織としての品質向上はあり得ない。

この考えに従い、富士通では、各 SI 組織に対して、QMS(Quality Management System)という活動を展開している。各 SI 組織が図 -3 のようなフレームワークに従った品質保証活動を実施する。現在、各業種 SI 組織に設置した SEPG(System Engineering Process Group)、SQAG(Software Quality Assurance Group)に、総数約 100 名の専門家を組織している。

この活動のなかで特に重要なのは、協力会社に対する品質保証の一貫した展開である. SI ビジネスにおいては、ソフトウェア実装を中心とする協力会社の貢献は必須である. これまでややもすると曖昧だった協力会

社との仕様提示条件と品質要求を、SLA(Service Level Agreement)として明確にし、健全化を図っている。

システム開発技術から見た問題

前述したように、ここ十数年のITとビジネス環境の変化は、情報システムの開発に混乱をもたらしてきた. 実際、Webベースの大規模システムが増加するなかで、開発技術に起因する問題が顕在化している。上記のマネジメントの取組みと並行して、システム開発の技術的問題に対する取組みは、PMOの重要な使命である。

問題の所在を分析すると、大きく次の3つの領域に分類できる.

(1) プラットフォーム構築上の問題

オープン環境での標準化の進展により、プラットフォーム、ミドルウェアの組合せの自由度が拡がり、最適な組合せでシステムを構築できるようになった。しかし、実際には、個々の製品のバージョンにより、組合せの不整合や性能問題が発生する。また、たとえば、J2EE標準により、市販のアプリケーションサーバ製品の機能に関しては差異がなく基本的に選択自由であるが、J2EE標準の上ではビジネスアプリケーション構築のサービスが不足しており、そのサービスをプロジェクトごとに作り込んできたのが実態である。ミドルウェア機能をアプリケーションでカバーしなければならないリスクは、標準化が進んでもなお残されている。

(2) ソフトウェア開発技術の問題

Java によるビジネスアプリケーション開発の技術も 人材も不足している. COBOL 環境で培った開発ノウハ

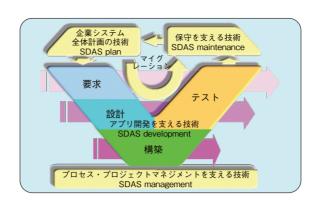


図-4 総合システム開発体系 SDAS の全体像

ウが Java 環境にうまく継承されていないのが大きな原 因の1つである. たとえば、ビジネスアプリケーション では常識的なメタデータの管理は、多くの Java 開発プ ロジェクトでは忘れ去られ、アプリケーション間のイン タフェースの不整合などのプリミティブな問題が発生し ている. また、アプリケーションの開発管理の基礎とな る単位(たとえばユースケース)がうまく定義できず、 品質,進捗,構成管理が機能しなかった例もある.

(3) 要求仕様の問題

この問題は、ビジネスアプリケーションの根深い問題 であり、システム環境が変わっても本質的には未解決の ままである. たとえば、業務仕様を利用者が厳密に定義 できず、開発者はある種の「常識」で不完全な仕様を補 うが、それがテスト段階で問題となり、仕様決定の責任 の所在をめぐって発注側と開発側で綱引きとなる. この ような要求仕様にまつわるリスクは、以前から何ら改善 できていない。むしろ、短期開発のプレッシャーのなか で、ますます増幅されているように思える.

開発技術の再構築

富士通では、これまで蓄積してきたシステム開発技術 を改めて再構築する取組みを進めている.

上記の問題(1)に対しては、2つのアプローチを進 めている. 1つは、プラットフォーム製品の組合せを パターン化し、検証されたテンプレートとして整備す る試みである. もう1つは, B².Sframework (Building Blocks System framework) という Web 環境におけるビ ジネスアプリケーション用のフレームワークの整備であ る. Web 環境には Struts のようなオープンソースのフ レームワークが存在するが、ビジネスアプリケーション

のプラットフォームとしての機能は不十分である. 高度 なセッション管理、ディレイドバッチ処理など、ミッショ ンクリティカルな機能を含めてフレームワーク化し、ミ ドルウェア製品として提供している⁴⁾.

上記(2)のアプリケーションの開発技術に関しては、 SDAS (System Development Architecture & Support facilities) という開発技術体系を再構築した (図-4).

SDASには、開発標準、フレームワーク、ツールなど の開発環境プロダクト, ビジネスアプリケーションの 設計, テスト技法などが含まれる. SDAS は 1987 年に データ中心型の方法論の技術体系として生まれた. 現在. Java を含めた環境を新たに融合させた方法論として公 開している⁵⁾

SDAS を通じて我々は、ビジネスアプリケーションの 原点の再確認を目指している. たとえば. アプリケーショ ンロジックは, 本質的に業務データの間に成立する検査 (validation)ルールや導出ルールの集合である. しかし, その業務データの項目が変更されることが多く、その影 響は業務データ、したがってルールにも波及する、その ため、業務データとそれに対するルールをクラスにラッ ピングするような単純なオブジェクト指向設計では、変 更に対して強い構造にならない、そのため、メタデータ を厳密に管理し、それを通じて変更の影響範囲を把握す ることは常套手段である. このような原点に立ち戻るこ とが必要で、この考えを基礎とした方法論を展開しよう としている.

要求仕様のリスク

問題(3)の要求仕様にまつわる問題は、SDASの中 でも依然として課題である。現在、業務仕様をシステマ

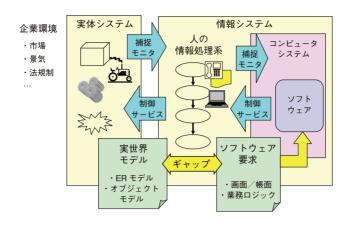


図-5 情報システムの要求構造

ティックに導出できるように、業務フロー分析や画面の プロトタイピングツールを開発しているが、要求仕様の リスクはこれらの技術の延長だけでは解決しない.

情報システムにおける要求仕様の難しさは、図-5に示すような構造からきている。物流や営業マンが活躍する企業の実体システムに対して、情報システムは支援系または制御系として存在する。ターゲットとなるコンピュータシステムは、情報システムの一部を担い、人の情報処理系が必ず並存する。うまく設計された情報システムは、人の情報処理系が、実世界で発生するさまざまな例外事象に対するバッファになる。たとえば、在庫に不足があっても実世界では代替品を出荷するようなことがあり得る。その場合、人の情報処理系は、コンピュータシステムに対して、あたかも最初から代替品の注文があったようにデータを入力し「つじつまをあわせる」ような役割をする。

コンピュータシステムへの要求仕様(ソフトウェア要求)を決定するには、実はこの人の情報処理系を含めた設計が重要なポイントとなる。人の情報処理系をうまく設計できないと、コンピュータシステムが実世界の例外を際限なく抱え込むことになるからである。残念ながら、多くの場合、この設計は暗黙のままに放置され、後工程にきて問題が発覚することになる。優秀なSEは、その暗黙知を利用者と共有する手だてを知っており、個人のノウハウに依存しているのが実態である。

この領域の問題は、ソフトウェア工学の延長では解決しない。画面上のプロトタイピングを何回行っても、 UMLをいくら高度にしても、背後にある人の情報処理 系の設計はできないからである。むしろ、経験ある SE の暗黙知を定式化する地道な努力が必要であろう ⁶⁾.

おわりに

IT の進展に開発技術の蓄積と継承が遅れてきたこと、 短期、低コスト開発が求められていることから、企業情報システムの開発はますますリスクを伴うものになっている。組織的マネジメントと開発技術の両面から、富士通のアプローチを例に、リスク回避の方策の一端を述べた。

環境が変わっても、ビジネスアプリケーション開発の 原点と本質的な難しさは変わらない。特に要求仕様にま つわる問題は、ビジネスアプリケーションの根底にある 問題で、それに伴うリスクは依然として未解決である。

少なくとも日本におけるビジネスアプリケーション 開発は、独自の経験則を培ってきた。そのプラクティス を表現し、公にする努力が我々 SI 企業に不足している ことを実感している。しかし一方で、ソフトウェア工学 などの基礎研究が、社会システムを支えるビジネスアプ リケーション特有の問題を直視していないようにも思え る。双方の歩み寄りと知恵の結集が必要であろう。

参考文献

- 1) http://segroup.fujitsu.com/
- 17 http://weightup.ntpstackin/
 20 Block, T. R. and Frame, J. D.: The Project Office, Crisp Pub. (1998), [仲村 薫(訳): プロジェクトマネジメントオフィス, 生産性出版 (2002)].
 3) 木野高史: プロジェクトマネジメントオフィスの進化と成熟度モデ
- 3) 木野高史:プロジェクトマネジメントオフィスの進化と成熟度モデルの考察,プロジェクトマネジメント学会誌,Vol.5, No.4, pp.12-17 (2003).
- 4) http://segroup.fujitsu.com/b2sframework/
- 5) http://segroup.fujitsu.com/sdas/
- 6) 板倉 稔,橋本惠二: スーパー SE がすすめる知のモデリング, 日科技 連出版社 (1996).

(平成 16 年 3 月 16 日受付)