

流通基幹業務ダウンサイジングについて

Downsizing of Mission-Critical System in a Retailer

会 田 信 弘

要 約 勝ち組みに入る大手スーパーマーケットチェーン I 社は従来の汎用機からオープンシステムへのシステム再構築に踏み切った。開発期間は 15 か月、1200 人月という厳しい条件の開発を考察した。

第一に本システム構築のリスクとは何かを整理し対応について報告している。主たるリスクは、システムスコープ、ビジネスルール抽出、プラットフォーム、移行であった。

第二は、システムアプローチについてだが、アーキテクチャ中心設計とデータ中心設計アプローチを選択した経緯をまとめた。

第三は、プロジェクトデザインについて焦点をあてた。プロジェクトを構成するメンバーの経験度、技術的難易などの特性分析しプロジェクトマネジメントを行ったプロセスを紹介した。

プロジェクトの成功にはチームとしての成熟度と顧客との良い関係が重要であり、PM には実行力 (Performance) まとめる力 (Maintenance) が必要であった。

Abstract The regional super-market "I" Company is one of the "winners" in this industry in Japan. The "I" Company made decision to rebuild the company's computer system from mainframe system to open system. This paper introduces the essence of this system development that spent 15 months and 1200 man-months.

At first this paper assembles risks in constructing this system, and discusses the measures for them. The risks discussed here are categorized into the following area: system scope risk; accuracy of extraction in business rules; new technologies of Windows based platform and rule of transition from the legacy system to the modern system.

Second, the author discusses the system approach, presenting the background to the selection of the architecture-centric project approach and the data oriented approach because of characteristics of the mission critical system.

Third point focuses on the project design and analyze the experiences of project member and their technical difficulty, presenting the process of this project management including the risk hedge.

Project maturity and good customer relationship are the most important success factors in the system development. And the Project Manager is needed to have P (performance) and M (maintenance) ability.

1. は じ め に

デジタル家電の好調、自動車の世界的需要の拡大などから 2004 年の各経済指標も要漸く上向きになって来た。ネットバブルが崩壊したあと情報産業もここ数年思わしくなかったが、従来の汎用機のシステム寿命や 2007 年問題といわれる技術者の大量退職の時期を控え、流通業各社もここに来て再構築機運が盛り上がって来ている。

本稿は時流に先立ち 2001 年 3 月より汎用機の基幹業務を WINDOWS 系に全面的に切り替えを行った I 社の事例考察である。

2. I社のシステム再構築の背景

I社は広島に本社をおくグループ年商約4千億の大手スーパーマーケットチェーンである。新システムは2002年6月に稼働したが、それ以降も業績は順調に拡大し、いわゆる勝ち組みにはいる優良企業である(表1)。

表1 各SMの最近の業績(H16年2月換算)

	企業名	売上	売上伸び率
1	イオン	3兆5,462億円	11.5%
2	イトーヨーカ堂	3兆5,421億円	0.3%
3	ダイエー	1兆9,936億円	▲9.3%
4	ユニー	1兆1,676億円	▲1.0%
5	西友	1兆1,251億円	▲1.2%
6	ライフコーポ	3,762億円	▲0.1%
7	I社	4,079億円	6.3%
8	イズミヤ	3,689億円	7.3%

I社は20年ほど汎用機を利用し、情報活用も本部集中の仕組みであったが、業界での生き残りをかけ、コンパクトな本部機能と消費者に近い店舗での情報活用を目指してシステム再構築に踏み切った。1998年頃あるコンサルファームの診断を受け、再構築においてはWINDOWS系のプラットフォームを採用することを方針としたが、実際に汎用機なみの信頼性を確保できる確信がなく検討は中断していた。2000年に弊社の大型IAサーバであるES7000が登場するや、その性能の高さを評価し、弊社がハードの提供と開発を受託することとなった。

2001年3月より新システム開発に着手をしたが、それ以前の7年間はD社にアウトソーシングをしており、情報システム部門はわずか4名ほどで組織されていた。新たなシステムを一から作り上げるには明らかに要員不足の状況であった。開発のスタートにおいては弊社の上級APアーキテクトとI社のOBに参画してもらい、短期集中した要件定義を実施した。

システム再構築の一側面はTCOの削減であるが、ビジネス的にも新たな変化に対応できる土壌形成し、財務的にも効果がでるシステムとする顧客トップの意向があり、成功尺度としてBSC(バランスドスコアカード)^{*1}的な要素をもっていた(図1)。

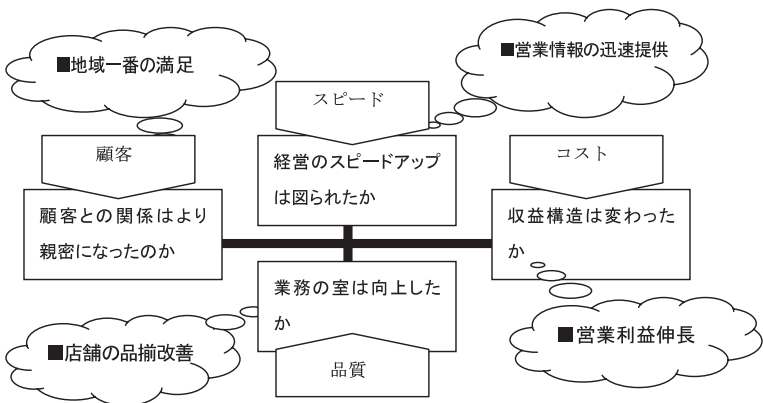


図1 新システムの評価基準

本システムはI社にとって全社プロジェクトであり、弊社にとっても WINODOWS による大規模ミッションクリティカルシステムの提供という大きな挑戦となった。

開発期間は 15 か月、開発工数 1200 人月、プロジェクトメンバは全員新規アサインであった。

3. システム構築のリスク

商品発注処理を中心とした外部連携が多い業務システム、20 年以上培ったシステムの歴史、I 社の情報部門の要員体制そして WINODOWS に統一した構築など開発当初より様々なリスクがあった。マネジメントとしては各リスクへの対策を決め計画に取り込み、プロジェクトの早い段階でのリスク解消に努めた。主なものを以下に紹介する。

3.1 システム開発スコープのリスク

I 社はシステムの再構築に当たり 2001 年 1 月より利用部門をメンバとした BPR プロジェクトを発足し、八つの分科会で業務上の課題とシステムに必要な支援機能の取りまとめを開始していた。総勢は 80 名でありその動向如何によっては事前調査をしていたシステムスコープと異なる可能性があった。本リスクに関しては従来のシステムのサービスレベルをクリアすることを最低レベルのハードルとして、明らかに追加となる機能に関しては、利用部門が費用対効果を算出し予算申請するという形をとった（図 2）。結果的に相当の絞込がなされ追加した機能は当初見積もりの約 8% であった。

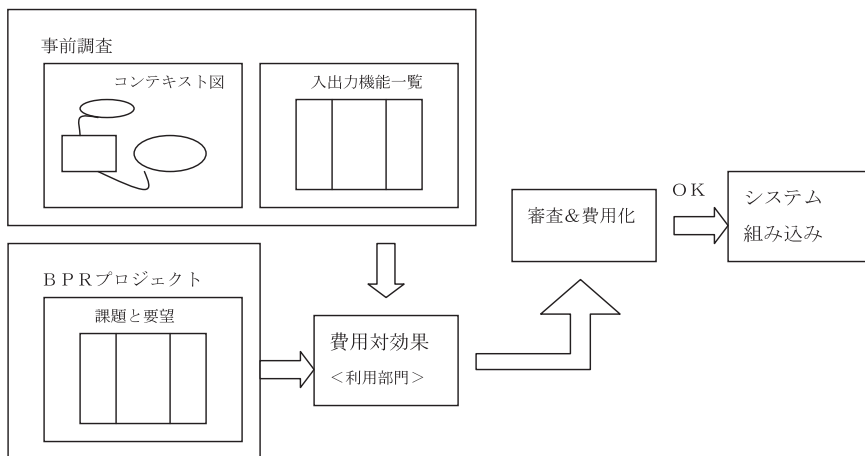


図 2 利用部門の要望取り込みプロセス

3.2 ビジネスルール抽出のリスク

本システムの開発期間 15 か月の内、要件定義は 3 か月であった。大規模開発の要件定義期間としてはあまりにも短い。その間の顧客情報部門および利用部門のヒヤリングでは業界の常識となっている要件要望ははず、特徴的なビジネスルールや現行の不具合改善に焦点が当たってしまう可能性があった。これではシステムの全体が見えてこない。その為プロジェクトを立上げるに当たっては流通基幹業務の開発経験者を組み入れることにした。

弊社には中規模企業向けのソリューションパッケージである OpenCentral^{*2}、IMPACT-SPA^{*3} などの販売しており、その開発経験者を本プロジェクトのチームリーダーとしてアサイン

した。

顧客との集中ヒヤリングではマーチャンダイジングサイクルに沿って業務ルールおよびシステム実現上の詳細ルールを解析していった。ヒヤリングでは分からない部分は、I 社 OB がプロジェクト要員として弊社側に参画し暗黙知の文書化を行った。それでも不明な部分は直接プログラムのコーディング解析を実施した。

分析は要件定義工程だけでは終わらず、論理設計段階、物理設計段階で詳細化を行った。

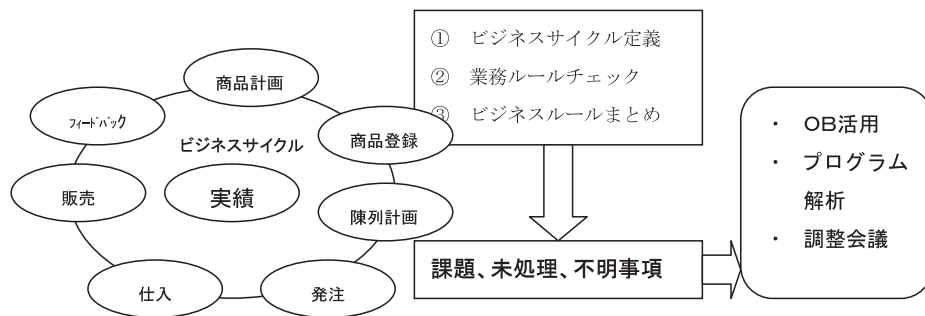


図3 ビジネスルール抽出プロセス

3.3 プラットフォームのリスク

本システムは3台のES 7000を中心とした、WINDOWSで統一したプラットフォームである（添付1）。提案時の構成で現時点（2004年秋）まで約2年間同一構成で稼働している。1日約100万件のトランザクションを処理しているが、当初は大型汎用機で稼働しているバッチ中心のシステムが移行できるか社内でも先行事例が殆どなく、基盤検討会議でもコミットするまでの結論には至らなかった。また、データベースはSQL-SERVERを使用した。当時とは異なる商用データベースが主流であり信頼性、習熟性の観点から意見の異なるメンバーもいた。

本リスクに関しては事前にパイロットシステムで検証することで目処を立てた。しかし本番システムと同等の試験は当然不可能であり、仮説—検証の実施とチューニングを繰り返して、当初の目標にたどり着いた。

システム構成上のボトルネックは、ネットワーク増強、より高速なCPUの採用で手は打てるが、アプリケーション的にはデータベースデザインが大きく影響するため、プラットフォームリスクを低減する上で、データベース選定とアプリケーションの開発標準化はきっても切れない問題である。詳述はしないが、各商用データベースのカーソル処理、ロックメカニズムについては十分理解して選定した。

ディスクに関しては、開発当初はリスク項目の一つであったが、採用したディスクは転送チャネル速度の高速化、十分なキャッシュメモリーなどから、本番稼働時問題になることはなかった。

オープンシステムの留意点として、各ベンダ製品の組み合わせを行ってシステム構築した場合のシステム稼働保証リスクをシステムインテグラーは法的な定義は別としても負わなければならない。

今回、OSからデータベースまでマイクロソフト社の製品で統一（図4）したが、障害があった場合に一貫した解析ができるという意味で正しい選択をした。

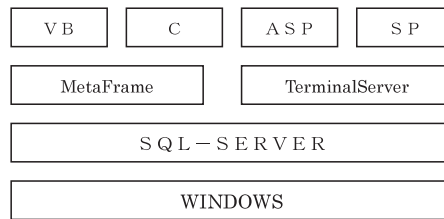


図4 システムのソフトウェア構成レイヤ

3.4 移行リスク

基幹業務再構築の大きなリスクの一つは移行である。システムは完成しても移行ができず本番遅延する事例が散見される。また切り替え時の移行負荷の大きさからその後の手順がスムーズに行かず切り替え後に支障がでるケースもある。

今回のシステム切り替えにおいてもI社と何回かのリリース判定会議を行い、慎重に切り替えを実施したが一部のテスト漏れが発生し何日かリカバリー作業を行った。

移行リスクを減らすにはできるだけ早い時期に移行プランを考えデータ移行と業務移行をシステム開発と同等レベルで計画することである。今回は移行データをテストの早い段階で利用することも計画し、三つある業務チームに移行担当を配置するとともに、全体の管理をする移行リーダーを設けた。移行リーダーは全体のプロジェクトの計画を熟知しており、複雑な切り替え時のトラブルは有ったものの全体のリスクは低減したと評価している（図5）。

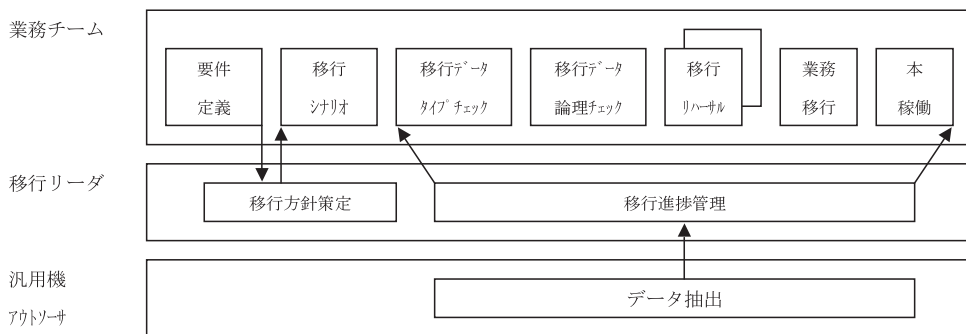


図5 移行概要

4. システムアプローチ

システムの実現において、OMG^{*4}が中心となり1990年代にCORBA仕様^{*5}を策定し、分散オブジェクトの概念をまとめたことはよく知られる所である。当時は他システムやレガシーシステムで作られたビジネスコンポーネントとあるインタフェースに合わせれば、自由に使えると期待が膨らんだ時代でもあった。しかし結局のところ実装技術では急送なWEBの発展により、HTTPプロトコルをベースにしたサービスが分散処理のベースとなった。GUI表記に関してはHTMLの延長としてXMLが急速に発展しSOAPが共通の伝達ルールとなってきている。またシステム間連携でも各社のEAIツールに優れたものが発表され疎結合モデルが現実的な選択となっている。

現在はオープンアーキテクチャおよびUMLによるモデル表記の普及もあり、ビジネスモデ

ルの実現アプローチとして、サービスオリエンテッドアーキテクチャ (SOA)、モデルドリブンアーキテクチャ (MDA)、アーキテクチャ中心アプローチなど様々な方法論が出てきている。各々の概要については技報 81 号「特集：アーキテクチャ」を参照されたい。

本システムの構築のベースは汎用機でのサービスレベルの達成が第一目標となっていた為、システムの構成を優先して決定するアーキテクチャ中心アプローチならびに、データの視点でシステムを分析するデータ中心設計に軸足を置いた。

4.1 アーキテクチャ中心設計

前述のようにハードウェアのプラットフォームは ES 7000、データベースは SQL-SERVER と決めていたが、そのベースライン上の品質目標として、汎用機並の信頼性、効率性、拡張性、可用性の各視点での評価基準を設けた。

実施にあたっては、インフラチームを設定しソフトウェアの選定、並びにパイロット検証も含めた各評価基準の内容を論理設計までに行なった。

主な方式を表 2 にまとめた。

表 2 品質指標とソフトウェアレイア

分類	信頼性	効率性	拡張性	可用性
システム全体	・ES7000 ・Windowsデータセンタ OS ・クラスタリング ・RAID構成	・各サーバの効率の確保検証 ・サーバ間の転送速度の確保	・システムのスケールアップとスケールアウトができる構成を組む	・クライアントからサーバまでWindowsに統一することにより、操作性、解析性を確保
C/Sアプリケーション	・事例が多く市場で洗練されている ・3層構造にするために、Metaframeを採用する販売ベンダとのアライアンスを確立している	・＜課題＞AP毎の独立性が高く、DBデザインに依存する	・Metaframe、TerminalServerのロードバランシング	・APの独立性は高いが共通機能に関して部品化を行い、再利用性を確保 ・GUI、印書などについては商用ツールを導入
情報系アプリケーション	・弊社で数十社実績のある、構築基盤を導入	・SQLSEVERでの処理効率を基盤として検証	・サーバのスケールアップに依存	・アプリケーション間でデータ連携、パラメータ連携をすることにより、ドリルダウン、情報連携が可能
パッチアプリケーション	・弊社のオープン系ソリューションを作成した時のパッチモデルを継承。障害時のリカバリ方法を統一。	・＜課題＞AP毎の独立性が高く、DBデザインに依存する。SQL文の記述により、効率が大幅に左右される		・ジョブスケジューラ(JP1)の導入

4.2 データ中心設計アプローチ

論理設計から物理設計への具体的なシステム実現をするシステムアプローチとして、大きくは、①プロセス中心アプローチ、②データ中心アプローチ、③オブジェクト指向アプローチ、がある。昨今は UML の普及により OOA^{*6} に注目が集まっているが、ビジネス系の基幹システム構築の上ではデータ中心設計に軸足を置きながらアプリケーション構成はオブジェクト指向を取り込むということが妥当な選択である。弊社の開発方法論の LUCINA^{*7} は同様な考え方をしている。

表 2 に表したが、画面系 AP、パッチ系 AP とも効率に関してはデータベース構造が大きく影響している。またデータ構造はオブジェクトモデルで表現した時の静的オブジェクトと呼ば

れるように、OOA でモデリングした時にも永続オブジェクトとして必ず表現されなければならないものである。

データ中心設計として独立するキーを抽出し各エンティティに分類することとそれに属する管理対象項目を整理していくことは、データベースモデラーに業務知識がないと困難であるが、今回業務開発経験が豊富なサブ PM にその役を担わせ、トップダウンによる設計を行った (図 6)。

データベースは工程に応じて、概念データベース、論理データベース、物理データベースと詳細化していくが、AP の手戻りを少なくする上でもデータベースの構造設計は非常に重要であった。因みに論理設計初期ではマスタを中心に 200 程度のスキーマであったが、物理設計終了段階では 1000 程度になっている。

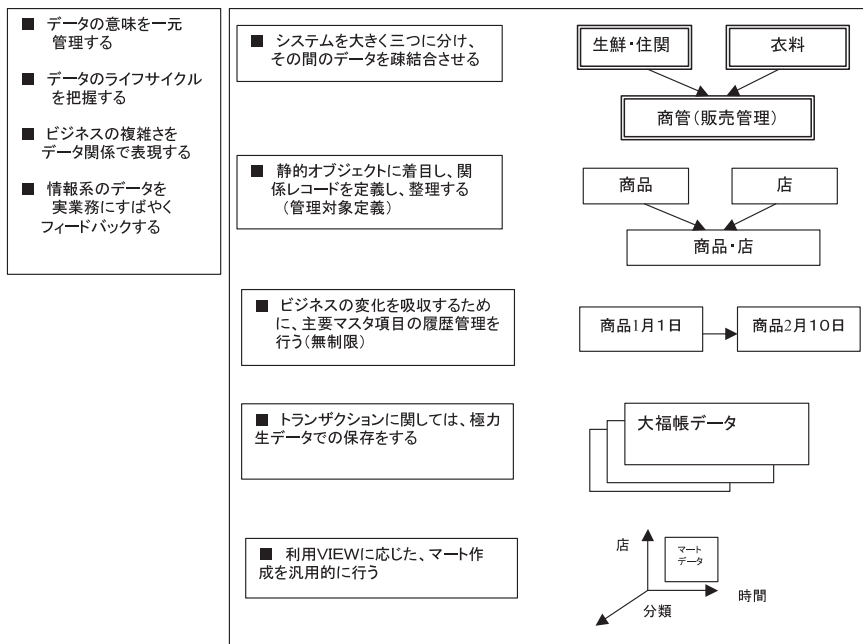


図 6 データ設計アプローチ

5. プロジェクトデザイン

システム開発の成功要因の一つは、プロジェクトチームの構成にかかっている。PM の責任感、実践力、統率力を発揮し、大洋に出帆した船の船長の様に目標を定め注意深くしかし大胆にプロジェクトを遂行しなければならない。ただし、船長一人では何もできない。優れたクルーを集めチームとしての力を結集することこそ、困難を乗り越える源泉なのである。

5.1 プロジェクトプロフィール

PM の初めの仕事はプロジェクトの特性を見極め、プロジェクト運営スタイルを決めることである。優秀な人材を当初より投入できるプロジェクトは少ないし、発注した顧客内でもすべてのメンバがシステム開発に合意しているとも限らない。まずはプロジェクトプロフィールの構成要素 (図 7) の見極めが肝心である。

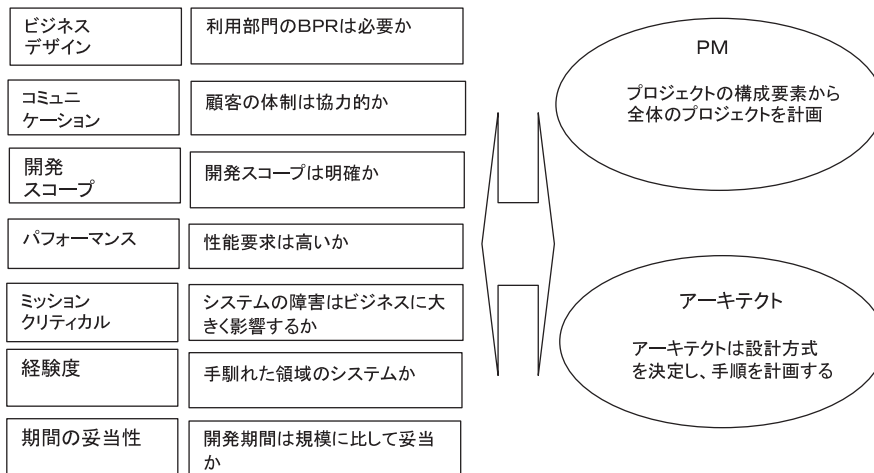


図7 プロジェクト構成要素

5.2 本プロジェクトの特性とプロジェクト計画

このプロジェクトのCSF[®]は以下の4点である。

- ① 既存汎用システムのシステム内容をカバーする
- ② コスト内、15か月で開発をする
- ③ 既存システム以上お効率と拡張性を確保する
- ④ IA サーバ ES 7000 と SQL-ERVER をプラットフォームにする

③と④は相互依存関係にあり、今回はアプリケーションアーキテクチャに先行して、システム化のスタートラインとして仮置きしている。汎用機であれば社内に形式知、暗黙知の蓄積が豊富であり、比較的容易に妥当性を導き出せる。オープンシステムの場合はパイロットシステムを走らせベンチマークテストを行うことが必須であり、そのWBS^{※9}を今回も組み込んでい

- る。
- ①は既存システムの分析と業務アーキテクトの投入ということになる。
 - ②は多分にもれず、開発着手前に顧客の予算や旧アウトソーサとの契約の関係から、詳細見積もり前に基本的な枠が決まっており、大きな制約であった。その為、開発体制は極力冗長度を避けプロジェクトの成熟度の急速な向上を目指した。また、営業も含めて顧客トップとのリレーション確保をして、両者のシステム運営委員会で問題の早期解決を図るようにした。

プロジェクトスタイルは図8、プロジェクトの運営会議は表3の通りである。

5.3 工程計画

工程計画は開発マイルストーンスケジュールを実現すべく、WBSの展開・詳細化と要員展開をしていくことである。計画着手段階で完全な計画を立てることは大規模開発の場合まず不可能であり、特に要員問題、検収問題から実際の工程期間がオーパフローすることが多い。本プロジェクトではオフィシャルな工程計画とともに、システムテストまでは遅延が1か月以内であれば本番稼働が可能となる財務指標計画（アーンドバリュ計画）をもってマネジメントした。一か月を超えた場合にはスケジュールのベースラインの変更が必要となるが、幸い最大遅

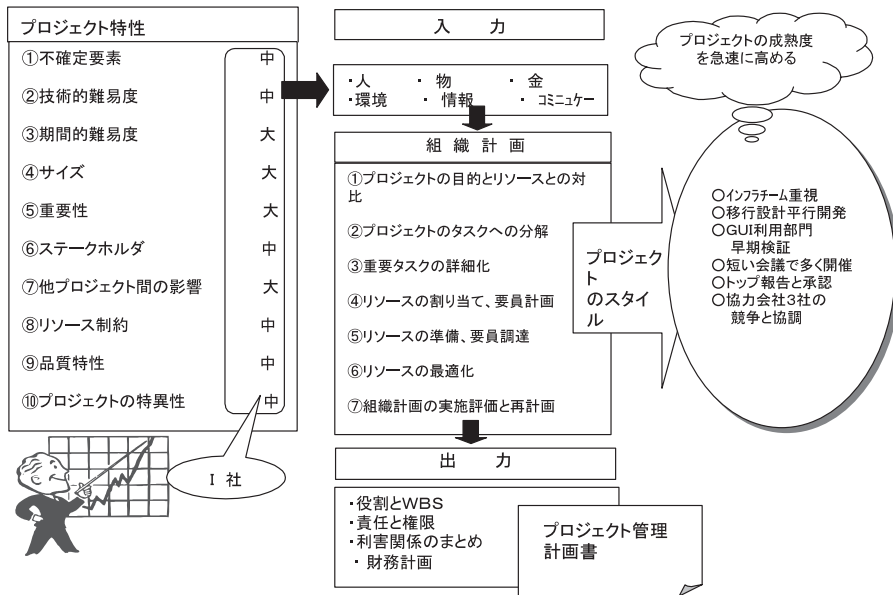


図8 プロジェクトスタイル

表3 プロジェクト会議体

- ① 必ず会議としての結論を出す
- ② 宿題の担当者と期日を設定する
- ③ PMまたはプロジェクトリーダーが会議を先導する
- ④ 短時間で終わらせる
- ⑤ 全員が何らかの発言をする

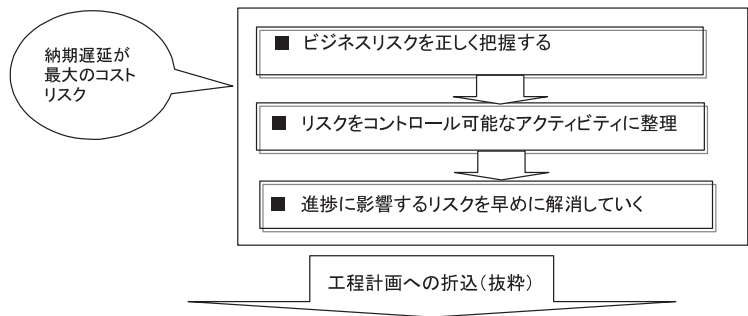
- 会議は2時間以内、通常1時間程度
- 技術はディスカスマネジメントはトップダウン

NO	会議名	目的	主な参加者	議長	サイクル	効果
1	プロジェクト報告	顧客トップとの課題調整	顧客オーナー、PM NUL PM、営業	顧客副社長	1／週	顧客の意思決定が迅速
2	リーダー会議	顧客の開発実務者とNUL間の進捗確認	顧客情報メンバ NUL PM、リーダー	PM	1／2週	プロジェクトの進捗の調整と役割分担
3	全体会議 (NUL内)	各自の状況報告	社内プロジェクトメンバ全員	PM	1／週	競争心によるモチベーションの向上
4	PMレビュー	各チームの進捗及び品質レビュー	PM、チームリーダー、協力会社リーダー	PM	1／2.5週	進捗遅延の防止 品質報告の遵守
5	チームレビュー	各チーム内の進捗、品質、技術レビュー	チームリーダー、協力会社リーダー、メンバ	チームリーダー	1以上／週	正しい進捗とメンバのやる気を把握
6	PMR/QAR	LOB内のレビューボードでの課題整理・対策	I&C部長 PM	部長	1／月	大局的な課題整理 ISBP遵守、CMM向上

延は3週間であった。

2章で記述したリスクはすべてスケジュールに影響するが、工程計画では事前に織り込むことでアプリケーション開発の手戻りを少なくすることができる。

工程計画におけるリスクコントロールは図9、実際の工程計画は表4の通りである。

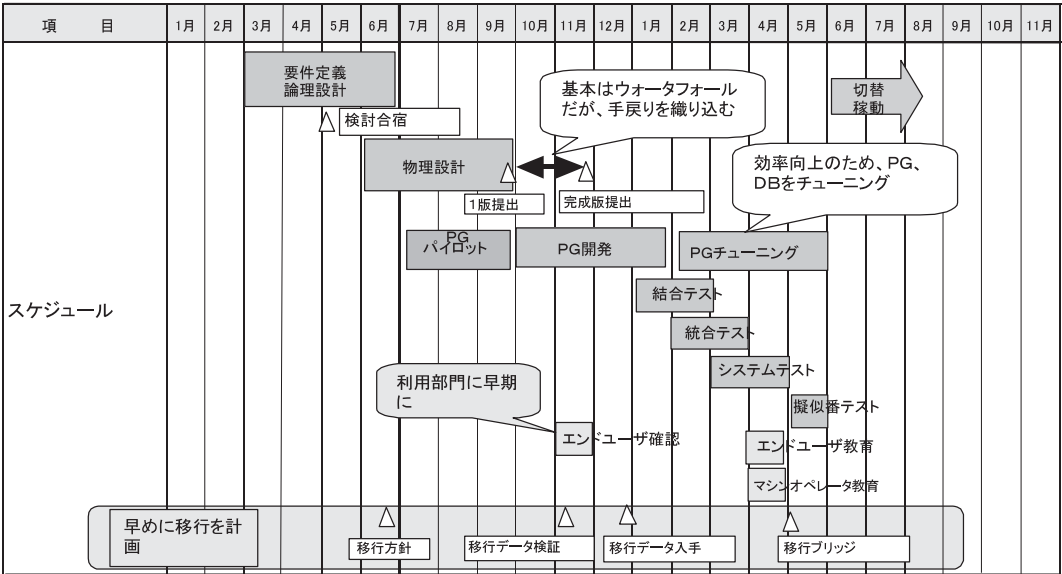


進捗影響度合	リスク名称	工程への組み込み
1	要件誤り	プロトタイプ作成
2	移行リスク	要件定義終了後に移行設計着手
3	協力企業プログラム品質	実データでのテスト計画
4	顧客検証体制リスク	擬似本番による、マッチングテスト

図 9 工程に織り込むリスク

表 4 工程計画

- 1 ウォータフォール型で開発を行うが、次工程にいても前工程変更があると予想して工程管理をする
- 2 店舗で利用する仕組みはテストと平行した教育工程を設ける
- 3 移行に関しては、論理設計時点から移行計画を立案する
- 4 チューニングを織り込む



6. PM マインド

今回紹介した事例は、本稼働時の品質に若干の問題は有ったものの、コスト・納期に関してはほぼ当初の目標通りであり、その後の顧客満足度調査でも高評価をいただいた。

当社の標準工数からするとずいぶん無理があったが、顧客、協力企業を初めとするステーク

ホルダの前向きな協力があり、プロジェクトのパフォーマンスを高めることができた。

PMという言葉であるが、プロジェクトマネージャという意味の他に、P=Performance, M=Maintenance という意味が含蓄されている。

Performance はもちろん、実力と実践力、責任感などやり遂げる自らの能力であるが、もう一つ忘れてならないのは Maintenance である。

プロジェクト作業の実際の受け手はプロジェクトメンバすなわち部下であり協力会社要員である。ある統計によればメンバのモチベーション次第で生産性は4倍異なる。そのモチベーションを高める為にメンタリングや公式、非公式のコミュニケーションが重要であることを忘れてはならない。図10はP2M^{*10}標準ガイドブックからの図を当プロジェクト用に当てはめてみたものである。

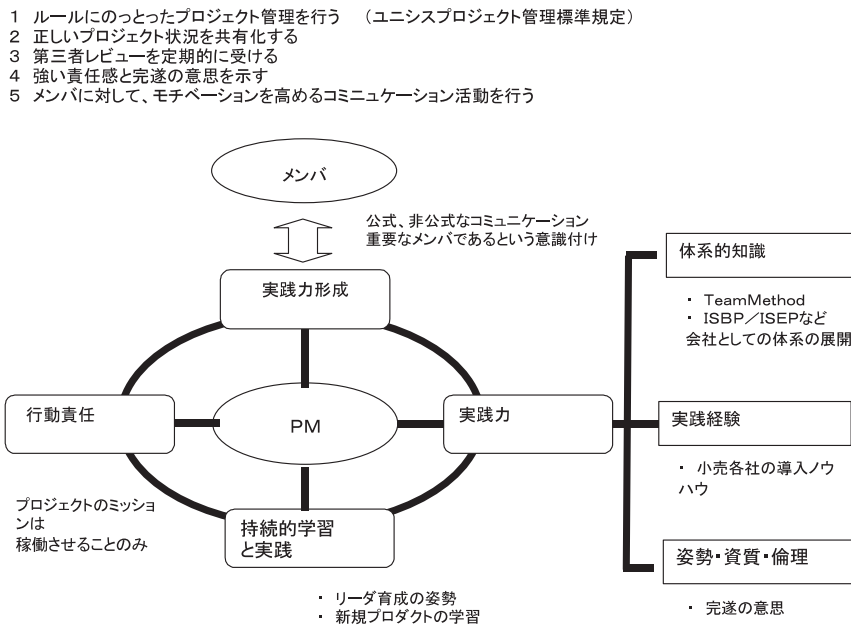


図10 PM マインド

7. お わ り に

ソフトウェア開発のプロジェクトマネジメントは数多くあるプロジェクトマネジメントの中で最も難しい部類に分類されている。大型プロジェクトの場合QCDを守ることができたのは全体の10%程度という統計もある。

米国で稼働する汎用機システムはいまや10%をきっており、日本でもレガシーシステムは終焉に近づいており、オープン化への流れが加速している。

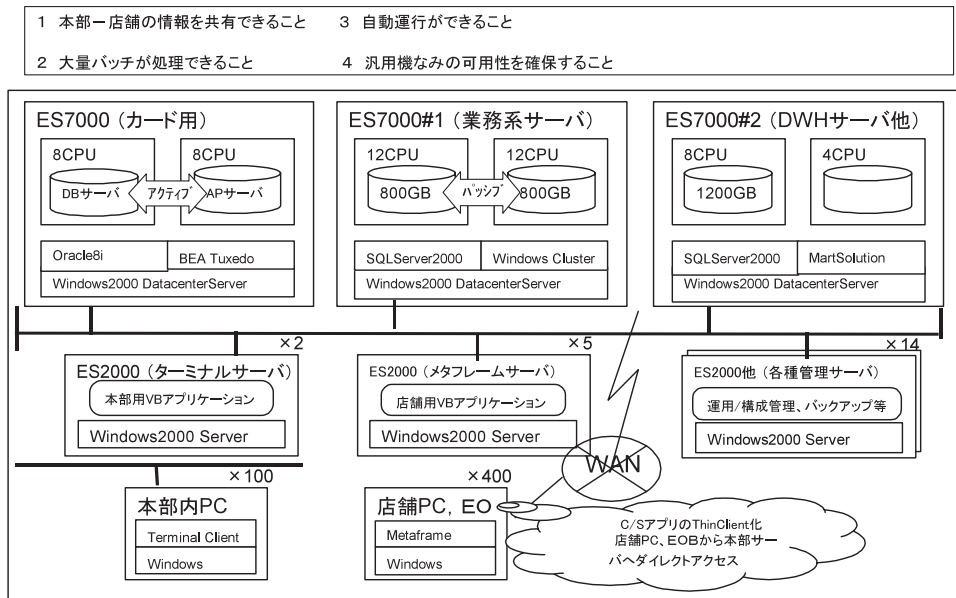
今回紹介した再構築事例はその先鞭に入る事例の一つである。オープン系システムはリスク回避が汎用機より難しいことを念頭にマネジメントすることが肝要であったが、振り返ると目新しい事をした訳ではなく、ひたすら可能性を信じメンバとともに一致団結して解決に当たったことが最大の成功要因だったと思う。

本番時の切り替えトラブルの中、顧客のトップマネジメントよりねぎらいの言葉をいただき、

プロジェクトを進めていく中でも、解決困難な問題が生じた時は常に精神的支えになっていた。

本システムは稼働して2年以上経つが、稼働の年プロジェクトメンバで山陰地方の旅行をした。今でもあの時の安堵感と達成感は思い出す。

当時のプロジェクトメンバに感謝の意を改めて表したい。



添付1 システム構成（基盤アーキテクチャ）

- * 1 BSC（バランススコアカード）：財務・顧客・社内プロセス・発展と学習の四つの観点に分類、それぞれの観点を評価する。トップマネジメントの支持が必要。
- * 2 OpenCentral：スーパーマーケットの本部業務サポートシステム。商品管理、発注管理、販売管理、仕入管理などを支援している。オープンシステム。
- * 3 IMPACT-SPA：主にアパレル業における本部サポートシステム。汎用機システム。
- * 4 OMG（Object Management Group）
CORBA（Common Object Request Broker Architecture）を策定している、分散オブジェクト技術の標準化団体。分散したオブジェクトを、ネットワーク上で共有し、統合するアーキテクチャの開発を目的としている。
- * 5 CORABA（Common Object Request Broker Architecture）
T.B.D ネットワーク上の分散環境でのオブジェクト間通信をサポートするためのミドルウェア基盤技術。
- * 6 OOA：Object Oriented Approach の略
- * 7 LUCINA：日本ユニシスが提供する開発方法の一連のプロセスとドキュメント。アプリケーションのタイプにより、雛型を用意している。
- * 8 CSF：クリティカルサクセスファクタの略。成功する上でのクリアすべく重要項目
- * 9 WBS（Work Breakdown Structure）
3階層程度に作業の詳細化を作業項目
- * 10 P2M（Project & Program Management）
プロジェクトマネジメントについて、日本の特性を加味して策定された体系。各プロジェクトの問題解決の為にプログラママネージャを設定する概念がある。

- 参考文献 [1] P2Mプロジェクト&プログラムマネジメント標準化ガイドブック（上巻）プログラムマネジメント編、2003 小原重信（編集）
- [2] UNISYS 技報 18号 特集：アーキテクチャ。

執筆者紹介 会 田 信 弘 (Nobuhiro Aita)

1976 年東北大学数学科卒業。同年日本ユニシス(株)入社。1987 年より流通業向けソリューション開発に従事。2001 年より大手 SM の基幹業務オープン化の PM として 2 年間従事。現在、日本ユニシス・ソリューション(株)製造流通サービス本部として、製造顧客、流通顧客を中心としたサービス部隊を率いている。情報処理技術者プロジェクトマネージャ、日本ユニシスプロジェクトマネジメント認定技術者。