

基幹系情報システム再構築プロジェクトにおける インド・オフショア開発の適用について



第3事業部
プロジェクトプロフェッショナル

岡本 祐二

Yuji Okamoto
yuji-okamoto@exa-corp.co.jp

オフショア開発の黎明（れいめい）期に、当社は主として低コスト開発を目的として部署ごとにオフショア開発を実践していた。そうした時期を経て、現在ではオフショア開発専門の支援部署を設置し、オフショア開発に全社で取り組んでいる。2004年からはインドIT企業の一つであるHCL社（HCL Technologies LTD.）をビジネス・パートナーとして迎えたため、特にインド・オフショア開発の手法をまとめ上げ、オフショア開発標準として体系化した。

本稿では、実際のシステム開発プロジェクトを事例に挙げ、準備・計画段階でのオフショア開発に特有の対策が最も重要であるとの認識の下、当社のオフショア開発標準をどのように適用し、インド・オフショア開発を実施したか、またその際に工夫した内容は何かということを中心に紹介する。また、開発遂行における様々な局面で得られたノウハウのうち代表的なものを紹介する。このようなオフショア開発のノウハウを蓄積し、次のプロジェクトにいかすことで、今後も継続的にオフショア開発を成功させていくことが可能となる。

1. はじめに

アプリケーション・システムの開発においては、お客様からのコスト低減要求がこの数年でますます強まってきている。また、コスト低減に取り組む際、納期と品質を犠牲にすることがあってはならない。つまり、短納期・高品質・低コストを同時に達成することが、お客様の要求にこたえシステム開発の業界で生き残り、サービスを提供し続けるための必要条件と言える。当社では、当初部署ごとに低コスト開発を目的としたオフショア開発を試行してきたが、現在ではオフショア開発専門の支援部署「オフショア推進センター」を設置し、オフショア開発に全社で取り組んでいる。オフショア推進センターが推進部署となり、短納期・高品質・低コストの実現のため、ISO9001、SEI CMM Level5、BS 7799の認証を取得しているインドIT企業に着目し、当社は2004年度よりインドHCL社（HCL Technologies LTD.）をビジネス・パートナーとして迎えている。

初年度の試行期間を経て当社オフショア推進センターではインド・オフショア開発の手法を体系化し、現在ではオフショア開発標準が確立している。オフショア開発を実施するシステム開発プロジェクトにおいては、準備・計画段階でのオフショア開発特有の対策が非常に重要である。本稿では、この認識の下、実際のシステム開発プロジェクトを事例として取り上げ、オフショア開発の問題点を提示し、その解決策を紹介する。その中では当社のオフショア開発標準をどのように適用し、どのように工夫したかということを中心に紹介する。また開発遂行における様々な局面で得られたノウハウのうち代表的なものを紹介する。このプロジェクトで実践したインド・オフショア開発の手法は、他のプロジェクトにも十分に適用でき、お客様の期待にこたえるための手段となりえるものである。

2. 事例プロジェクトの概要

本事例報告で紹介するプロジェクト（以下、当プロジェク

トと称する）は、製造業のお客様の販売・物流系基幹システムを再構築するものである。旧システムは、IBM社のメインフレームにてCOBOLで構築されていた。新システムでは、画面系の開発言語にマイクロソフト社のVisual C#、ASP.NET、バッチ系の開発言語にオラクル社のPL/SQLをそれぞれ採用している。

当プロジェクトにおいては、当社オフショア推進センターの協力を得て、HCL社へ開発を委託しインド・オフショア開発を行うことに決定した。決定した理由は次の3点である。

- ・一定範囲（詳細設計～結合テスト）をオフショア開発とすることで開発費を低減できる（トータルコストとしてお客様の要望にこたえる）
- ・欧米の先進的なシステム開発の経験が豊富であるためUML、.NET関連など最新技術に精通した技術者を大量に動員できる
- ・SEI CMM Level5達成に代表されるように安定した品質を実現する開発プロセスを持っている

HCL社に二つのサブシステムの開発を委託し、二つのサブシステムそれぞれに別の開発センターを指定した。先行したのは「販売系Xサブシステム」であり、正式着手前の準備段階において両社で入念に打ち合わせを実施した。もう一つは「物流系Yサブシステム」で、先行する販売系Xサブシステムでの取り決めや、日本側とインド側のメンバーの相互理解など、開始までの手順を踏襲し、そこより得た知見を活用した。プロジェクト完了後に測定した開発規模は、表1に示すとおりである。

開発プロセス標準は、RUP（Rational Unified Process）準拠¹⁾であり、当プロジェクトでオフショア開発を実施した範囲は、RUPの四つのフェーズにおけるフェーズ3（作成フェーズ）に相当する。フェーズ2（推敲フェーズ）、フェーズ4（移行フェーズ）の一部についても当プロジェクトの受注範囲であったが、オフショア開発は適用してい

表1 開発規模

	画面・バッチプログラム数			ステップ数		
	画面	バッチ	(合計)	画面	バッチ	(合計)
販売系Xサブシステム	46	47	93	134,513	34,609	169,122
物流系Yサブシステム	162	178	340	108,904	80,185	189,089

ない。

当プロジェクトにおけるオフショア開発の開始から終結までの開発作業のフローを図1に示す。まず、開発実施に先立ち、開始前の準備（正式着手前の打ち合わせ）、2回のKT（Knowledge Transferの略であり知識伝承を意味する）セッション、プロジェクト計画書作成の各タスクを実施した。次に、開発実施段階においては開発対象をロットに分割して、詳細設計から制作、単体テスト（UT）、ロット内結合テスト（ITa）までを一つのサイクルとして実施した。一つのサイクルが終わり次第、その結果をオフショア開発委託先と共同でレビューし、次のサイクルの開発手順へ反映させる方法を採用した。当プロジェクトではこのロット分割の単位を「ステージ」と称した。開始から終結までに関して、国内での開発の場合と比較すると次の特徴がある。

- ・実際の開発作業の開始前にKTセッションが必要であり、この部分では工期が数週間増加する
- ・委託先作成のプロジェクト計画書の確認が、オフショア開発では必須である
- ・適切な準備作業期間を設けることによって、複数のステージを並行して実施するか、または、一つのステージに含める開発規模を大きくするかのいずれかの方法を採用し、計画的に大量動員して工期を短縮できる
- ・結合テスト、受け入れテストについては、国内での開発の場合との相違はない
- ・開発が完了した段階では、次回開発に向けた総括のために関係者が終結会議を開催し教訓を確認し合う（オ

フショア開発の成功のためにはHCL社とのオフショア開発を継続して実施することが必要であり、互いの開発手順を確認し改善することによって、生産性向上、品質向上、納期短縮が期待できる）

3. オフショア開発の課題と対応ノウハウ

以下においては、当プロジェクトにおいて実施した、オフショア開発の課題への対応策をノウハウとして述べる。

3.1. 開始前の準備

先行した販売系Xサブシステムでは正式な着手前の準備として、まずオフショア開発委託先（HCL社開発センター）と当プロジェクトの間で開発プロセスのすり合わせを実施した。この開発プロセスすり合わせ会議は、当プロジェクトとオフショア開発委託先の初めての会合であることから関係者が直接会う対面形式（F2F、すなわちフェース・トゥー・フェース）で実施した。また、オフショア開発委託先の組織、技術などの事前評価を再確認する目的もあり、当プロジェクトとオフショア推進センターからそれぞれ1名ずつ、あわせて2名がインドの開発センターを訪問することにした。

このインド訪問の目的は、特に画面系における.NET開発経験と開発プロセスの確認、日本語への対応方法、および日本では避けられない仕様変更への対応の3点を確認す

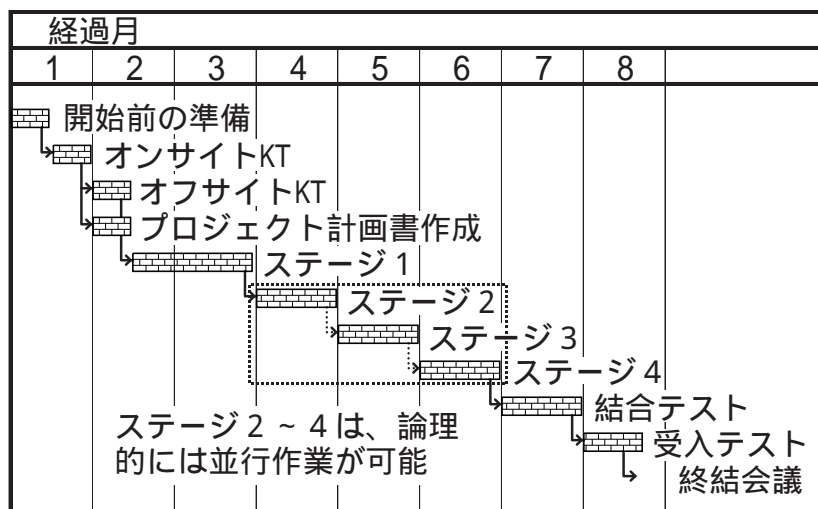


図1 今回のオフショア開発の開始から終結まで

ることであった。確認した結果は以下のとおりである。

- ・オフショア開発委託先(HCL社開発センター)の.NETに関する開発事例およびSEのスキルレベルと開発動員力の確認ができ、また、開発プロセスと開発成果物の種類は標準化されており、当プロジェクトとほとんど一致していることが判明した。(特に画面系の成果物はUMLベースであり、成果物の種類としては当プロジェクトとオフショア開発委託先の間で一致していた。)
- ・日本語への対応については、オフショア開発委託先でバイリンガルのSEをアサインすることにより、開発プロセス内で日本側とやりとりが必要になる部分に、翻訳処理ブロックを置いて、日本側に英語を原則意識させない方針であることを確認した。
- ・仕様変更については、日本国内での仕様変更と同様に仕様変更のルール、フローを事前に取り決め、それにしたがって運用していくということであった。仕様変更は随時取り込むのではなく、1ヶ月程度の蓄積の後に1度にまとめて取り込むということに合意した。

その後の会議では、インド訪問は頻繁には行わず、国際電話会議、テレビ会議を併用した。電話会議は、メールでのコミュニケーションに付きものの時間差の問題を解消し、また、複数サイトからの参加も可能とした。通常は電話会議を使用したが、図表を使用したドキュメントを用いて仕様の確認を行う場合には、テレビ会議を使用した。

3.2. 立ち上がり時のKTセッションの実施

当社のオフショア開発標準では、オフショア開発の開始時に、1) 仕様の概要レベルの理解、2) アーキテクチャの理解の2点をオフショア開発委託先と確認するKTセッションを設けることを推奨している。国内での開発の場合にはこのKTセッションは、通常は特別な期間を設けて実施することはせず、設計作業のQ&Aの中で逐次実施する。一方、オフショア開発の場合には、上記の目的に加えて日本側とオフショア側の相互理解のためにも、設計作業とは独立した作業項目とすべきである。

当プロジェクトにおいては、販売系Xサブシステムで開発委託を正式に決定した後、開発前の準備作業として、日本側オンサイトとオフショア側で2回のKTセッションを実施した。同サブシステムはお客様固有の作業負荷の高い

業務であり、システムとしての難易度も高かったため、オンサイト(当社開発チームのオフィス)でオフショア委託先のPM、PLの2名に対して2週間の計画でシステム説明を実施した。当社ではこれをオンサイトKTと呼んでいる。その後はオフショア委託先の現地にてサブリーダーも含めて、構築対象のシステム説明を2週間かけて実施した。こちらは、オフサイトKTと呼んでいる。また、オフサイトKTでは、画面系を中心に提供された多くの開発標準(画面デザインガイド、Web構成ガイド、アプリケーション基盤共通機能利用ガイド、コーディングルールなど)をオフショア側メンバーが理解するための作業も共同で実施した。

物流系Yサブシステムでも同様の考えに基づいて準備作業を進めることにした。販売系XサブシステムでオンサイトKT・オフサイトKTをそれぞれ2週間として作業を進めた際の経験を踏まえて、物流YサブシステムではオンサイトKT・オフサイトKTをそれぞれ3週間とした。これは、KTセッションに十分に時間をかければ、開発実施段階の効率が向上し、複数ステージを並行で実施すること、または、一つのステージに含める開発規模を大きくすることにより計画的な大量動員を行い、工期の短縮と全体での生産性向上が図れると判断したためである。

3.3. オフショア側のプロジェクト計画書

図1に示したとおり、オフショア開発委託先ではオンサイトKTセッションに続き、そこで得られた情報をインプットとしプロジェクト開発計画書を作成した。このプロジェクト計画書は当社側でもレビューをすることが、当社のオフショア開発標準で規定されている。

オフショア開発は言葉も文化も異なる遠隔地における開発であるため、開発プロセス・手順、役割分担、連絡体制などについて十分に協議し合意することが必要であり、その内容がプロジェクト計画書に反映されていることがオフショア開発の成功に向けた第一歩となる。

3.4. 開発プロセスへの翻訳作業の挿入

各ステージの中での開発作業の流れを図2に示す。原則として、日本における開発作業の流れと同じであるが、日本側とのやりとりが発生するプロセスでは前後に翻訳作業が入る。国内での開発と比較した特長は次のとおりである。

- ・日本語のインプット文書(当プロジェクトでは基本設

計画)をオフショア側で英語に翻訳する。開発チームは、基本設計書(英語版)を理解することから作業を開始し、単体テストまでを一連の作業範囲とする。したがって、インプット文書の英語への翻訳作業は、開発チームのクリティカルパスの外側に位置づけられる。(このままでは、この翻訳作業はプロジェクト全体のクリティカルパスに含まれるが、別の作業との並行作業とすることでクリティカルパスから外すことを考える。)

- ・ 詳細設計書を日本側でレビューする場合は、英語版の詳細設計書を対象とすれば、英語から日本語への翻訳によるクリティカルパスの延長や開発プロセスにおける翻訳作業の複雑化を招くことはない。これには日本側の要員にレビューのための英語力が必要となるが、詳細設計書に頻出する表現のパターンや英語版の設計書のレビューポイントを洗い出し、事前に習得しておくことで対処した。(実際には、英語版でレビューし日本語でレビューコメントを返すという方法をとった。日本人SEは詳細設計書レベルであれば英語であっても、事前に業務用語等の英和対照表を作成しておくことにより理解が可能であり、英語から日本語への翻訳作業を割愛できる。一方、レビュー結果のコメントを英語に翻訳することは苦手であり、インド側での翻訳作業で相互確認するのが確実であると判断した。)
- ・ 「単体テスト」を終えたチームは、同一ステージ内の次の開発対象について「英語版基本設計書の理解」に入る。同一ステージのすべての開発対象が「単体テスト」まで完了している場合は「ステージ内結合テスト」に入る。
- ・ 詳細設計書は最終的に日本語化するが、日本語化に際

しても翻訳レビューが必要である。通常、これは各ステージの終わりにまとめて実施する。

- ・ 上記の詳細設計書の日本語化作業が完了していなくても「ステージ内結合テスト」が完了していれば、開発チームは次のステージに入る。

3.5. 開発推進体制

オフショア開発の課題と対策として、日本側とオフショア側の体制に関する注意点が一般的にも認識²⁾されている。当プロジェクトにおいては、プロジェクト開始時点より当社オフショア推進センターのアドバイスに基づき体制構築に当たった。これが効果的に働いており、全体を通じて当プロジェクトでのインド・オフショア開発は成功することができた。以下ではこのノウハウについて述べる。

(1) 全般的推進体制

プロジェクトメンバーの多くはオフショア開発の実践経験がなかったため、当社オフショア推進センターからの次の支援が非常に効果的であった。

- ・ 契約管理
- ・ VPN等を用いた安全な共通開発環境の提供
- ・ 開発プロセス上の翻訳作業の考慮
- ・ KTセッション実施の支援
- ・ マイルストーンの設定
- ・ その他、アドバイスや実際の作業支援など

また、インド側での以下に示すオフショア開発の十分な実績やコンピテンシーがあったこと、および、プロジェクト開始時点より成功のための協力体制(以前からの開発を

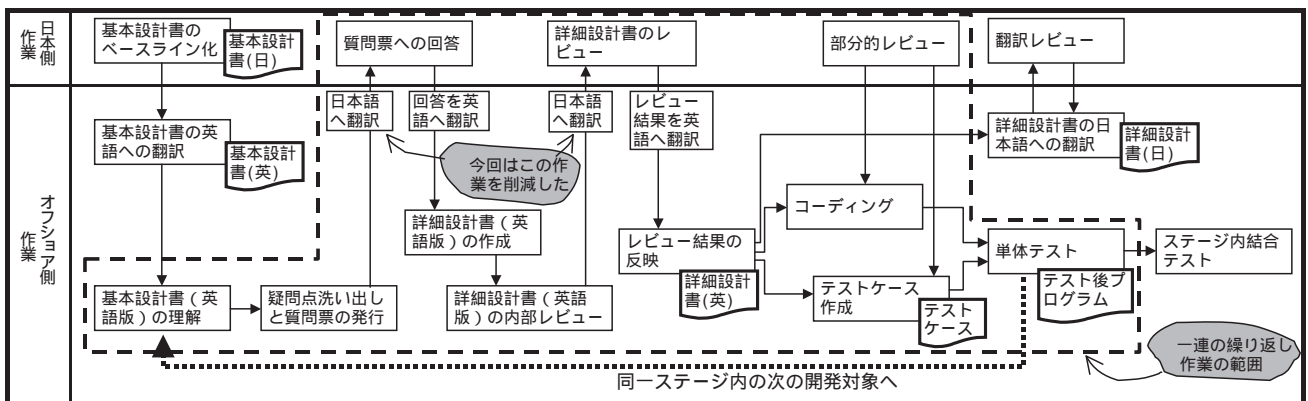


図2 各ステージでの開発作業フロー

通じて醸成した信頼関係に基づく)が確立していたことが、当プロジェクトの成功要因であった。

- ・ ISO9001、SEI CMM Level5に代表される品質管理の仕組み
- ・ 欧米のシステム開発の経験を有し、最新のシステム技術に精通したUML技術者や .NET技術者の大量動員力

(2) アプリケーションSEのオフショア開発適性

日本側では、業務仕様を伝達するSE(窓口)として、業務知見の深さ、仕様の理解力、システム開発の経験などが通常よりも高い水準のSEをアサインした。また、英語読解力(コスト増になるが、システム開発の経験が十分な要員と英語力のある要員をペアとすることで良い)、異文化交流への抵抗感がなく柔軟に対応できることなども考慮に入れた。

(3) 日本側アーキテクトのアサイン

当プロジェクトでは、画面系の開発基盤として .NETを採用したが、本プロジェクト開始当時は適用事例も少なかったため、技術面の体制強化も必要であった。すなわち、生産性の向上と品質の向上のためには、アーキテクチャの決定やデザインパターンの作成を十分な詳細さで早期に進める必要があった。このため、日本側にもアーキテクトをアサインしてオフショア側のアーキテクトと共同で技術的な問題解決に当たることとし、実践した。また、個々の開発者のソースコードについては、サンプリング・レビューを実施した。

3.6. 遂行時のQ&A実施方法

当社オフショア開発標準には、Q&Aの実施方法については、メールベースの方法とプロジェクト管理ツールのQ&A機能を利用する方法がある。当プロジェクトでは、利用者への導入教育の工数を削減するため、メールベースの方法を採用した。これは、特別のメール件名(Subject)を使用し、本文は定型化するルールである。オフショア開発に限らず、この方法は以前から国内の遠隔地開発で使用しており導入と適用が容易であるという特徴がある。また、特別のメール件名と本文の定型化は、Q&Aの進捗のトレース、内容の整理と分析などを容易にすることを目的としている。

先行した販売系Xサブシステムの例では、当初はメールベースでオフショア側のPMと通訳を経由してQ&Aを行った。ステージ1では画面系の開発の立ち上がりが順調ではなく、レビューでの指摘事項には仕様の理解が十分でないことに起因するものがあつた。これはQ&Aが十分に機能していないことが原因と考え、ステージ2の途中でQ&Aの実施方法を変更した。具体的には、それまでの「オフショア側のPMと通訳を経由する」方法から、「オフショア側ブロックリーダーから日本側のアプリケーション窓口(業務仕様を伝達するSE)が英語で質問を受け、日本側からは日本語で回答しインド側の通訳が英語に翻訳する」方法へ変更した。(図3に変更後のQ&Aの流れを図示した。)

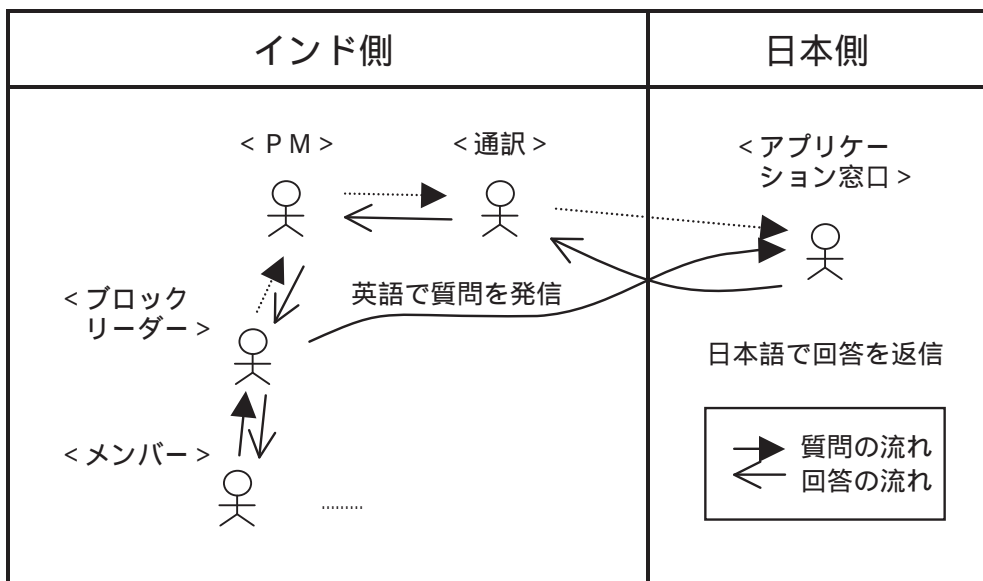


図3 Q&Aの基本的な流れ

この変更の結果として次のことが挙げられる。

- ・これを契機にインド側から質問がしやすくなりQ&Aが激増したが、仕様の確認を十分に行うことで誤解は確実に減少した（Q&A数の推移を図4に示す）
- ・質問の内容をオフショア側PMが精査する作業、および、質問を英語から日本語へ翻訳する作業がなくなり、インド側と日本側の担当者間でのQ&Aのレスポンスが改善した
- ・オフショア側PMが内容の精査をしないですべて日本側へ送信することで、一部にはささいな質問も含まれるようになった

3.7. 遂行時の仕様変更への対応

当プロジェクトとして最初のインド訪問の際に確認し合った取り決めの中で、「仕様変更の手順を取り決める、逐次対応ではなく一定期間蓄積し一度にまとめて変更を取り込む」という方針を立てた。

プロジェクト開始時点ではこの方法で合意していたが、プロジェクトの進行に伴い、優先順位の高い仕様変更、緊急対応しないと他への影響が大きく手戻りが予見される仕様変更などが発生した。このような仕様変更を蓄積することには日本側の担当者の心理的な抵抗感が非常に大きく、「仕様変更の蓄積、まとめて対応」の例外措置として緊急対応を何度か行ったが、必ずしも効果的ではなかった。つまり、手戻り工数の削減よりも、緊急に仕様変更の対応を行ったことによる開発現場の混乱や、仕様変更に対応した

後に少しの期間において同一箇所での再変更の発生など、例外措置の引き起こしたデメリットの方が大きかった。また、プロジェクト管理の面においても「仕様変更の蓄積、まとめて対応」方式は、仕様変更の対応によりスケジュール、コスト、生産性などの計画値が頻繁に変更になることを抑止し、進捗モニタリングが煩雑になることを防ぐ効果があることが分かった。

このように「仕様変更の蓄積、まとめて対応」の原則遵守は重要であり、例外的に仕様変更の逐次対応を行う場合も十分に優先度を検討し、実施の可否をマネジメントレベルで判断すべきである。また、当プロジェクトでは、「仕様変更の蓄積、まとめて対応」方式を採用することを、プロジェクトの開始時にお客様に説明し合意を得ていた。こうしたプロセスの提案については、プロジェクト立ち上がり段階で方針を定め、お客様の理解を取り付けることが必要である。

3.8. 終結時の終結会議開催

オフショア開発委託の完了に当たっては、当社のルールにのっとり終結会議をインドで開催した。その目的は以下のとおりである。

- ・開発委託の完了を正式に確認する
- ・完了の節目として、プロジェクトの実績データを収集し双方で確認する
- ・プロジェクト期間を通じて発生した問題点の共有、教訓としてのとりまとめ（これはオフショア開発では特

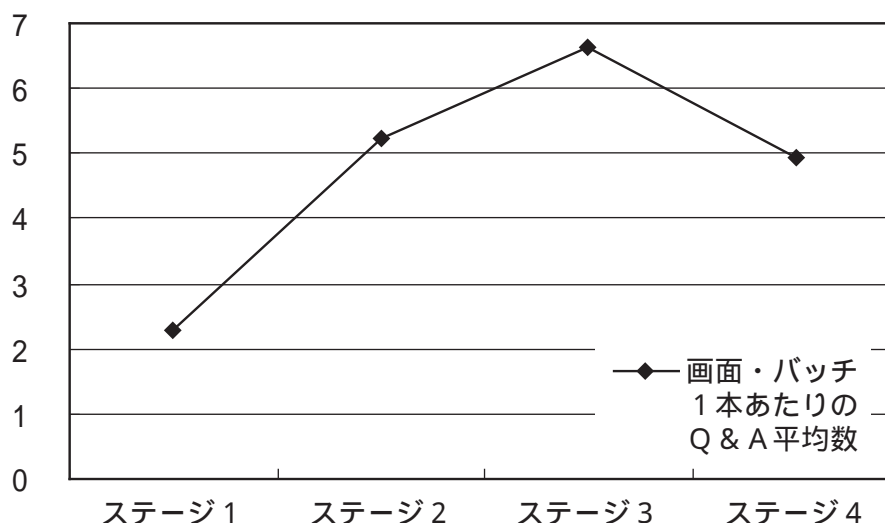


図4 販売系XサブシステムのQ&Aの推移

に重要)

- ・オフショア側メンバーへの感謝を表明(文化や言語が異なる遠隔地における開発のため、感謝の気持ちを伝えることが次回のプロジェクトの成功にもつながる)

4. オフショア開発のノウハウ項目まとめ

本事例で得た知見から効果があった対応策のポイントをノウハウ項目として次に挙げる。

(1) 事前準備:

- ・社内のオフショア推進部署の支援を取り付ける
- ・開始前の事前準備の重要性を認識し、現地での打ち合わせを計画する
- ・KTセッションには時間をかけ、開発実施段階の効率を高める
- ・オフショア側プロジェクト計画書レビューを通じて、開発プロセス・手順、役割分担、連絡体制などについての合意事項を最終的に確認する

(2) 変更管理:

- ・遂行時の仕様変更は一定の期間蓄積し、まとめて対応する

(3) 翻訳:

- ・翻訳作業は、オフショア側と日本側それぞれの相手側言語スキルに応じて、最大のパフォーマンスが発揮できるように開発プロセスに挿入する

(4) 終結会議:

- ・終結時には終結会議を共同で開催し、教訓を抽出し次回のオフショア開発につなげる

5. おわりに

本稿では、事例によりオフショア開発の問題点と対応策およびノウハウについて述べた。そこでは当社のオフショア開発標準を最小限にカスタマイズして適用し、インド・オフショア開発を実践した。このような工夫を更に進め、組織的な取り組みとして蓄積し再利用することで、より円滑にオフショア開発を実施することができる。今後も各種改善を加えながらインドIT企業との業務提携によるオフショア開発を継続的に実施し、より大きな成功に結びつけていきたい。

参考文献

- 1) フィリップ クルーシュテン, "ラショナル統一プロセス入門 第3版", アスキー, 2004
- 2) S openオフショア開発研究会, "オフショアリング完全ガイド ソフトウエア開発", 日経BP社, 2004

Visual C#, ASP.NET, .NETは、米国Microsoft Corporationの米国およびその他の国における商標または登録商標です。

PL/SQLは、米国Oracle Corporationの米国およびその他の国における商標または登録商標です。

RUPは、米国IBM Corporationの米国およびその他の国における商標または登録商標です。

CMMは、米国カーネギーメロン大学の米国における登録商標です。

UMLは、Object Management Groupの商標です。

この他、掲載されている会社名、製品名およびサービス名等は、各社の商標または登録商標です。