大規模システム開発におけるプロジェクト・マネジメントの実際

The Project Management in Large-scale Development Actually

葛　谷　幸　司

要　約　プロジェクト・マネジメントについては，IT 業界のみならず，PMI/PMBOK をはじめとした教育や資格取得の推進，更には参考文献・書籍も多数発表されており，それらを知識として修得する事は比較的容易である．しかしながら，その知識をもってしても実践する事は容易ではないという現実や実態もある．本稿は，特に難易度が高い大規模開発のプロジェクト・マネジメントについて，実際に日本ユニシスグループと(株)百五銀行が共同で実践した次世代オープン基幹系システムの開発プロジェクトである「S-BITS プロジェクト」を通し，プロジェクト・マネジメントとして特に工夫したポイントである「透明性あるプロジェクト運営」について，重要成功要因などを中心に述べている．

また本稿は，大規模開発におけるプロジェクト・マネジメントの実際を理解し，その上に改良・工夫を加えて，より失敗のないプロジェクト・マネジメントを実践する一助にしていただく事を目的としている．

Abstract　Learning project management is not considered to be difficult based on various kinds of training programs prepared by IT vendors and public authorities like PMI/PMBOK, and a lot of books are published for reference. On the other hand, it is well known fact that managing projects in real situations are very tough missions even with well educated knowledge's. This paper reports about the project management schemes applied in “S-BITS project”, a very large scale project with highest difficulties jointly done by Nihon Unisys Group and The Hyakugo Bank, Ltd. , focusing its “transparent project management methodology” with its critical success factors. This paper may be useful to understand the real features of large scale projects and to find out original improvements and refinements to general schemes aiming successful projects management.

1. は　じ　め　に

情報・通信技術を含めいわゆる情報システムは，日進月歩で進化し続けている．この背景には，社会や企業における情報システムとしてのニーズがあり，ひいてはその企業が提供するサービスを利用する顧客のニーズがあるのも事実と言える．月並みな表現ではあるが，現代における情報システムはもはや社会基盤の一部またはそのものとして機能しており，それらを利用する顧客，提供する企業共に極めて依存性が高く，また重要な位置付けになってきている．

一方，これら情報システムを構築・提供する側としては，様々な技術革新へのキャッチアップにはじまり，ビジネス環境の急速な変化やそれに伴う新たな機能要求と対応，更には高品質で安定した情報システムをより早くより安く，計画期限と計画コスト内で提供するという使命を負う事になる．そうした情報システムの構築は，新規開発または追加・変更などの形態で，大小規模は様々であるもののプロジェクトを組成し実行されるのが通例となってきている．

そのプロジェクトを立上げ～計画～実行～監視コントロール～終結へと，遂行する実行責任者がプロジェクト・マネージャ（PM）であり，この一連のプロセスがプロジェクト・マネジメントである．

また，こうした重責を担うPMをいかに短期間で育成していくかという事も大きな課題となってきており，PMI/PMBOKをはじめとした知識体系の普及活動をはじめ，日本ユニシスグループも当然の事ながら，国内外，IT業界を問わずプロジェクト・マネジメント知識の修得やPMの育成に力を注いでいる状況である．

しかしながら，こうした取組の一方で実際のプロジェクトにおいて計画した品質・コスト・納期通りに進まず，中断，延期やコストオーバを余儀なくされてしまう失敗プロジェクトが後を絶たないのも事実であり，日本ユニシスグループでは，これを撲滅するためのケーススタディを通した研修をはじめ，ISBP/TeamMethodという社内標準（PMI/PMBOK準拠）の開発プロセスやプロジェクト運営について様々な規定や規約を早くから定め運用してきている．また，同様に失敗プロジェクトを起こさないための参考文献や書籍も数多く発刊されており，今や社会または企業経営における共通の重要課題であると言えよう．

本稿では，特に難易度が高いとされる，大規模開発のプロジェクト・マネジメントについて，実際に日本ユニシスグループと(株)百五銀行が共同開発した次世代オープン基幹系システムの開発プロジェクトである「S-BITS[*1]プロジェクト」を例に取り，プロジェクト・マネジメントとして特に工夫した次の「重要成功要因」を中心に述べている．

1） プロジェクトの全体統制
2） 会議体運営
3） 課題/リスク管理
4） 間接業務

なお，プロジェクト・マネジメントとして重要な三大要素となる，品質（Quality），コスト（COST），納期（Delivery）の管理については，開発規模を問わずやるべき事や手法などが体系化され，HowToに関する書籍や情報も数多く存在すると考えるので，先のポイントで間接的に触れる程度であえて深く言及はしていない．

プロジェクト・マネジメントについては，実務や教育を通して知識を修得/向上し経験を積んでいくが，その知識や経験をもってしても実践する事はなかなか容易ではないという現実や実態もある．本稿は，大規模開発におけるプロジェクト・マネジメントの実際を理解し，その上に改良・工夫を加えて，失敗のない大規模開発プロジェクトを実践するための一助になる事を目的としている．

2. プロジェクト・プロフィール

開発プロジェクトの特徴は，顧客である(株)百五銀行とパートナ企業であるマイクロソフト(株)他，及び日本ユニシスグループ（協力パートナ企業含む）の大きく3社の共同プロジェクト体制である．

2003年12月に，前述の3社にてWindows®ベースの次世代オープン基幹系システムの開発着手を対外発表し，3ヶ月間のプロジェクト立上げ・準備期間を経て2004年4月に開発プロジェクトを発足し要件定義工程を開始した．以降，設計/製造工程，統合テスト工程，システムテスト工程とその間の稼働判定会議（計3回）の評価を経て，当初計画通りの2007年5月

6日に本番稼働を開始し，同年6月末をもって本プロジェクトは終結/解散となった．

2.1　プロジェクト体制

プロジェクト体制は，基幹系システム（国内勘定系，国際勘定系，対外チャネル系，融資稟議支援系等）の全てをオープンプラットフォームで刷新するという特性があったため，各機能・役割単位にプロジェクトを6分割した複数プロジェクト体制とした．更には，各プロジェクト配下には詳細機能別にグループやチームを構成し，より機動力を発揮できる組織体制を確保した．また，最大の特徴である共同開発プロジェクトは，日本ユニシスグループと百五銀行

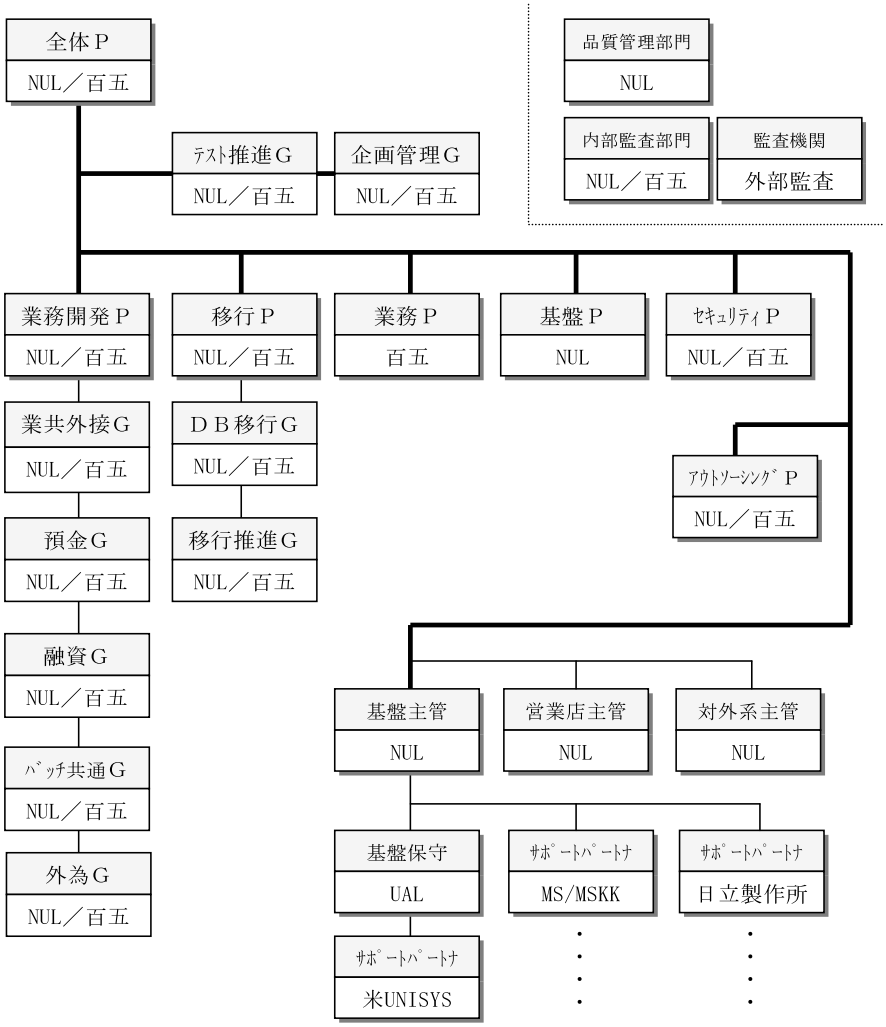

図1　プロジェクト全体体制図（2007年5月本番稼働時点）

の要員で構成し，PM も二人一組で役割責任を相互牽制し補完しうるマネジメント体制を敷いた（図 1）.

なお，プロジェクト体制については要件定義から終結・解散までの各工程や状況を鑑み，適宜最適化を図るべく変更を実施し，各フェーズでより機動力が発揮できるよう柔軟な組織運営を実施した.

全体プロジェクトは，ピーク時の構成要員数が約 1,000 名という大規模な体制であり，各プロジェクトで一番大所帯となるのがオープン勘定系システム「BankVision」を中心として開発責任を担った「業務開発 P」でありピーク時約 500 名の体制であった.

各プロジェクトの役割については，図 1 の役割・機能名称で概ね理解可能なので詳細は割愛するが，規模は異なるもののその一つ一つはそれ相応のプロジェクトであり，それぞれに PM や GL（グループリーダ）を配置し，通常開発プロジェクトに必要とされる機能（QCD 管理など）も各々で保持し，独立組織として運営した.

大規模プロジェクトの定義として，まず第一に構成要員数や開発規模が大きい事が挙げられるが，役割・機能別に組織をグルーピングするという構成が通例で，それらを統括する機能が必要となる．本プロジェクトでは全体 P がすなわち統括プロジェクトであり，全体統括 PM がその責務を担った .

2. 2　開発計画

開発計画は，プロジェクト発足から終結・解散までの 33 ヶ月間であるが，全体のスケジュールは中核となる BankVision の開発計画を主軸として立案した．課題としては移行をはじめ各種サブシステムの並行開発であり，これらをいかに同期させ各プロジェクトそれぞれに支障

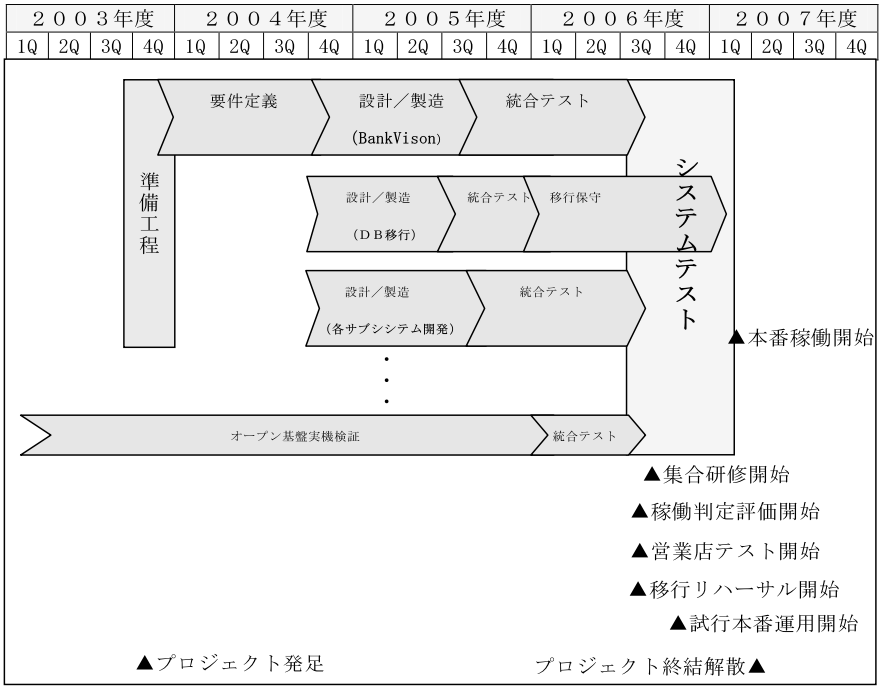

図 2　開発スケジュール（概要）

をきたさず統合させていくかであった.

開発の各工程期間はプロジェクト毎に若干異なるものの，統合テスト工程内でシステム連携を中心としたテストへ，更にシステムテストでは本番稼働環境を用いた運用テストを中心に計画・実行した．また，オープン基盤の実機検証はこれら開発を先行する形で，プロトタイプの業務プログラムを用いて各種検証作業，最終チューニング作業へと進めた（図 2).

2. 3　開発規模と役割

開発全体規模は約 20,000 人月強，プロジェクト要員数は平均 600 名（ピーク時 1,000 名）の大規模プロジェクトであり，共同開発での役割分担は概ね次の通りとなる.

1)　要件定義・・・共同
2)　設計/製造・・・日本ユニシスグループ主体
3)　統合テスト・・・日本ユニシスグループ主体（最終検証は百五銀行）
4)　システムテスト・・・共同
（但し，営業店テストや移行リハーサル等のイベントテストは百五銀行主体）
5)　システム運用・・・アウトソーシングサービスとして日本ユニシスグループ

なお，開発ツールや言語などはサブシステム毎に若干異なる点やシステム数が多い事，加えて本論文の主題とは直接的な関係が薄いため割愛する.

3.　プロジェクトの重要成功要因

本章では，主題である大規模開発プロジェクトの重要成功要因について，実際の全体プロジェクト運営を踏まえ考察し述べていく.

まず，この重要成功要因を一言で表すならば,「透明性あるプロジェクト運営」に尽きる．例えば 10 人体制のプロジェクト（PM）であれば各メンバの顔やスキルをはじめ，人柄や性格までをも把握できる．更に，開発成果物に関する進捗管理や品質管理もその規模感から比較的容易に把握でき，結果として開発全体のプロジェクト状況が必然的且つ正確に把握できると言える．しかしながら，この規模が 100 人，200 人，500 人･･･と数が増して大規模になると，小規模であれば自身の目で見えていたものが見えなくなってしまうという状況になる．では，この見えない部分をいかに見えるようにするのか．これを実現するのが「プロジェクトの透明性」の追求だが，その実践がなかなか上手くできないのも事実であろう.

では，この課題をどう克服していったのか，特に影響の大きい要素と考える次の重要成功要因について，それぞれの実際とポイントを掘り下げる.

1)　プロジェクトの全体統制
2)　会議体運営
3)　課題/リスク管理
4)　間接業務

3. 1　プロジェクトの全体統制

プロジェクト構成要員（ステークホルダ）が常時 100 人，200 人を超えるような大規模プロジェクトの場合，一つのプロジェクト（PM）配下に全メンバを横ならびで構成するのではなく，システムの機能や分野等でプロジェクトを分割したり，更にグループやチームを編制し多

段階的な組織編成にするのが通例である．これそのものはより機動力，組織力を発揮させるためであって否定するものでないが，現実はこのステークホルダの人数に比例するような形でプロジェクト運営の難易度は高くなると考えられる．その代表的な理由の一つは，プロジェクトという組織を分割したり多段階化する事によって，強い統制力が必要となる事である．

統制力を十分に発揮できるようにするためのいくつかの要素の中からポイントを挙げるならば，基本的なコミュニケーションと権限委譲（役割定義）であると考える．

3.1.1　コミュニケーション

一般的に，平等に与えられた時間内でPMが1日に会話できるメンバの数はおおよそ10名程度である．いかに大規模になろうがメンバとの実務上のコミュニケーションはやはり10名程度になろう．この10名とのコミュニケーションを日々密に取る事は言うまでもない．

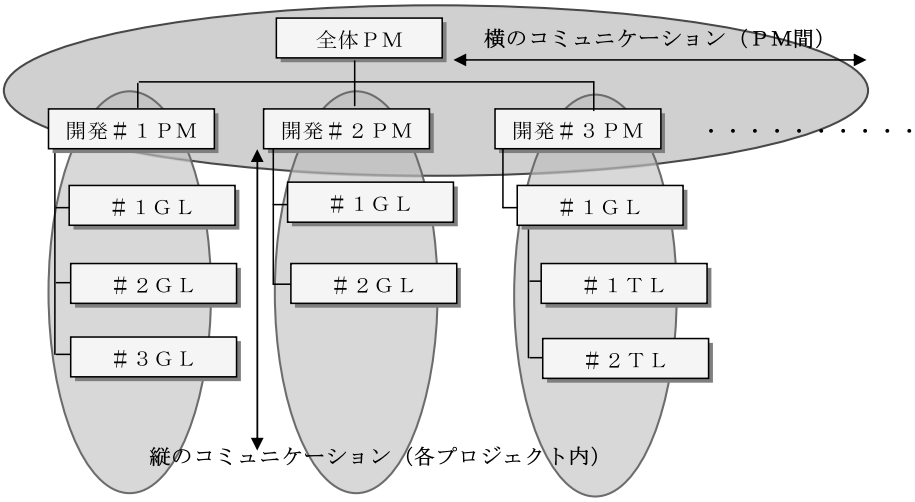

図3　横と縦のコミュニケーション

プロジェクト全体の状況把握のベースとなるのは，全体PMと各PM間の「横のコミュニケーション」であり，この横のコミュニケーションは各プロジェクト内で行われる「縦のコミュニケーション」がスムースに実施されてこそ意味のあるものになる（図3）．

また，コミュニケーションとは単なる情報伝達ではなく，上下左右の情報交換に始まり各プロジェクトの状況や課題，更にはプロジェクト全体の指揮命令も含まれる．これらは一見単純で当然の事であるが，現実は会議などに忙殺され，それらが実施できていないという状況に陥っているPMも少なくないであろう．

更に，次に述べる権限委譲を含め，このコミュニケーションの継続こそがプロジェクト統制における重要な要素となる．

3.1.2　権限委譲

コミュニケーションの内容を大きく左右するのが各々の役割定義（責任範囲）であり，各PMやGL，TL（チームリーダ）に課せられる役割を定義した上で，各種リソース（要員，予

算，各種ファシリティ等）をどのように使い当初計画通りのQCDを実現するのかについて，役割責任と権限を委譲した上で効率性と機動力を発揮する必要がある．当然ながらそれらを遂行できる要員のアサインメントは重要な要素となるが，アサインメントできたとしてもこれらが正しくなされないケースでは，常に様々な判断をより上位層へ求める形態となる．更には，判断を促すための情報提供をはじめ各種説明会やレビュ等を頻繁に実施する事となり，結果としてより多くの時間やコストを費やすという状態に陥るケースが現実には少なくない．

大規模プロジェクト運営では，こうした権限委譲を正しく行った上で，かつそれぞれの立場や役割責任の上でコミュニケーションを実施する事で，はじめてプロジェクト全体の統制が図られると考える．

特に，大規模PMの立場であればより正確な情報を得る事でより正しい判断ができ，正しい指示が出せるようになる．言い換えるならば，大規模ほど不正確な情報が断片的に入ってきて正しい判断をするに至らない．または判断するのに更なる情報収集が必要となり，より多くの時間をロスし，結果としてプロジェクト運営が立ちいかなくなる等の状況に陥ってしまう．

このような事から本プロジェクトでは，体制図に合わせて各役割定義を行い，プロジェクト全体の計画内運営に関わる権限は全体統括PMに，それに準じて各PMやGL/TLへと各役割責任と権限委譲を実施した．

3.2　会議体運営

会議による業務時間の圧迫・占領，不本意な出席により時間を無駄にしている焦燥感，会議体の意思決定機能不全など，PMのみならず開発作業で忙殺されるエンジニアも持つ不満である．大規模プロジェクトになるほど，多くのステークホルダに影響を与えるので，会議体の運営には気を配る必要がある．

3.1節で述べた「プロジェクトの全体統制」をより確実に実施するためには，プロジェクトの会議体運営が重要な位置付けになる．それについて実際のプロジェクト運営を通し，より効果的な会議体運営を考察する．

3.2.1　会議体の定義

通常，「プロジェクト管理計画書」で明文化するプロジェクト運営管理の一つとして会議体運営要領を定める事になる．特に，大規模且つ企業経営にもその成否のインパクトが大きいプロジェクトに関しては，単なる報告の場ではなく，正しい経営判断をする場としての位置付けも必要となる．これらを踏まえると概ね以下の会議体の定義が必要である．

1)　経営トップ向けのプロジェクト全体状況報告（判断）
2)　開発責任者（経営層またはプロジェクトオーナ）への報告・調整・判断
3)　プロジェクト全体の報告・調整・判断
4)　各プロジェクト内の報告・調整・判断
　（同様に，プロジェクト配下のグループ/チーム内の報告・調整・判断）

その他として，イベントベースで開催する各工程の終了判定会や外部監査・内部監査，情報セキュリティ委員会等が構成されるであろう．

表１　会議体運営要領（概要）

NO	会議体名称	開催間隔	レビューア	被レビューア	エスカレーションパス
1	役員報告会	４半期	経営トップ	開発責任者	
2	開発管理委員会	月次	開発責任者	全体PM	↑
3	全体プロジェクト会議	隔週	全体PM	各PM	↑
4	プロジェクト会議	週次	PM	GL/TL	↑

本プロジェクトでは，主要な会議体を表１のように定め，プロジェクトの発足から解散まで継続実施した．エスカレーションパスが，役割権限を越えた判断や承認を得る際のルートとなる．なお，各プロジェクト内の各種会議体は任意に定義され運営されているがそれらは割愛している．

3.2.2　会議体の重要ポイント

全体 PM の管理として重要な会議体は，「開発管理委員会」と「全体プロジェクト会議」（表1. No.2 と No.3）であり，それぞれのポイントは次の通り整理できる．

1）　全体プロジェクト会議

プロジェクト全体の横串会議であり，計画をベースとした各プロジェクトの進捗状況や品質状況，加えて重要な課題と対応状況をレビュする．この場の調整や判断がプロジェクト遂行上の要となるので，各プロジェクトから正しい情報や状況の報告が必要となる．加えて，この会議が全体 PM としての実質的なデシジョンボードになる．

2）　開発管理委員会

本会議は，全体 PM がプロジェクト全体の計画と進捗状況を開発責任者（経営層）へ説明し，必要に応じて調整・判断を求める場であり，例えばプロジェクト全体で定めた当初計画（体制，進捗，開発計画，予算，リスク管理状況など）に変更が必要となる場合等の最終的なデシジョンボードとなる．

以上二つの会議体が，大規模プロジェクトを効率的に進める上での重要な場であり，プロジェクトの意思決定と全体統制を図る機能をも包含している．

また，各会議体を効率的に継続実施する事で，プロジェクト状況の変化がより把握し易くなる．加えて会議に費やす時間を最小限にすべく各会議体のアジェンダ（報告，課題調整，上申の区分）を定義・宣言し，各会議のはじめにゴール（目的）を明確化する事も重要な要素と言えよう．

こうした演出も含めて，プロジェクト全体の状況や重要事項等についてタイムリな情報共有化の推進がより一体感を醸成し，各 PM も含めたメンバ自身をゴールへとモチベートするきっかけにもなると考える．本プロジェクトは，これら会議体の資料や議事録を全プロジェクトメンバが閲覧できる環境の作成，フィードバックする報告会の開催など，情報のシェア（共有化）も行った．更なるポイントとしては，大規模プロジェクトが陥るプロジェクト間の課題の調整，プロジェクト全体のリスク管理が挙げられるが，これらについては次節で述べる．

3.3　課題/リスク管理

課題やリスク管理の必要性は言うまでもないが，特に大規模になると各プロジェクトやグループ間に潜む課題や問題そのものが浮き彫りにされず，結果として放置され，気づいた時には手遅れになっている事も少なくない．これは，先に述べた役割定義や権限委譲を実施しても，場合によっては防げない．なぜならば，文字ベースで定義・表現できる内容には限界があるからである．また，プロジェクト状況は工程などによって変化していくため，そもそも法律を定義するよう精緻に役割定義や権限委譲を実施しても確実なる費用対効果が求められない事も原因である．本節では，これらについてどう対応したのか実際の運営を踏まえ考察する．

3.3.1　課題管理

繰り返しになるが，大規模プロジェクトにおいて課題の抽出や対応は重要であり，これを怠ると命取りとなりかねない．特に問題解決者が曖昧な場合それらを見過ごさず確実に抽出していく事が重要である．

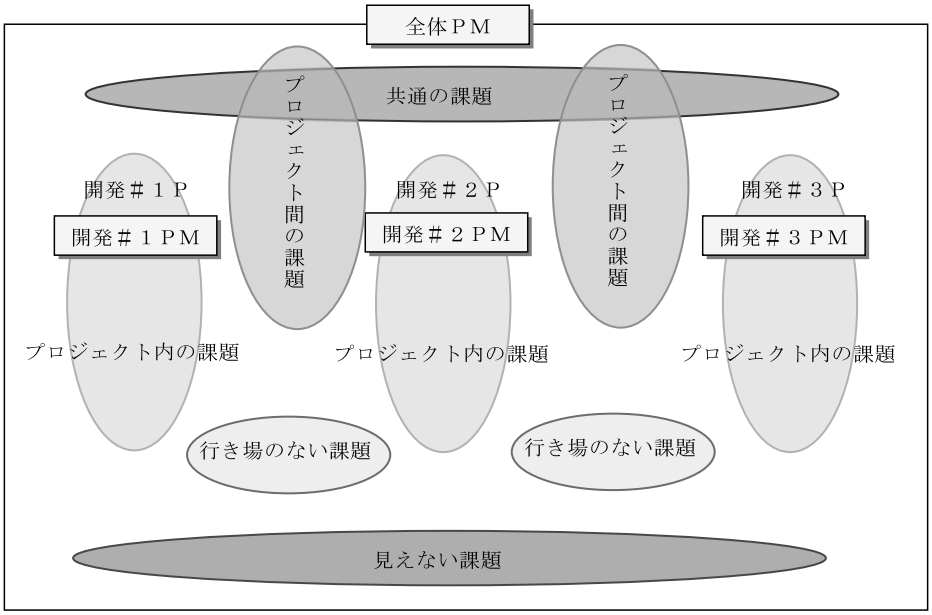

図４　全体プロジェクト内に存在する５種類の課題

一般に，全体プロジェクト内に存在する課題の種類は概ね図４に示す５種類に種別される．ここで問題とするのは，各々が認識している課題（各プロジェクト内の課題）を除いたもの全てと言え，それらをどう整理し解決へと推進するべきかについてポイントを述べる．

1)　共通の課題

この課題は，プロジェクト全体の共通課題である．例えば，各システムでプリント機能の運用の統一化を図りたいとする．各システムはプリント機能へのニーズが異なる事が予想されるため，誰かが要求仕様を取りまとめ実現に向けた対策を講じるといった対応が必要になってくる．このようなケースの場合，短期間でそれらを検討し解決へと導くためのタスクフォース（本件を解決するチーム）を組成し対応に当たらせるのが有効な手段と考えられる．

影響度が大きく期間的に余裕がないと判断されるような課題については，リーダの任命またはPM自身がリーダとなり，タスクフォースを組成し対応に当たる事が課題解決の近道と言えよう．

なお，実際のプロジェクトにおいてこのような共通の課題については，多くのタスクフォースを作り対応完了をもって解散するという事を実践し多くの成果があげられた．

2）　プロジェクト間の課題

この課題は，開発＃1Pの対応を待って，開発＃2Pが更なる対応を行うケースが該当する．例えば，システム基盤系の不具合修正作業と基本動作テストが終了したら，業務系の正常確認作業を行うというように連携が必要なケースが挙げられる．この例の様に単純明快であれば問題はないが，より多くのステークホルダが関わり，期間も２～３ヶ月程度かかるような場合は，これらを確実にトレースし影響範囲を見極めていく必要があると言える．

3）　行き場のない課題

この課題は，各種定例会でレビュを実施している時などに検出される課題であり，課題は認識できるが役割として明確な解決者が定められていないもの，または，全てではないものの，複数のプロジェクト間の共通課題と認識できるものなどが該当する．

このようなケースの時，全体PMとしては実際に次の様な対応を実践していった．

①課題解決に一番適任と考えられるプロジェクトやグループのアクションとする．

②共通性が高く，影響度も大きいと判断した場合はプロジェクト間共通の課題としてタスクフォースを組成する．

③前述のいずれも該当しない．または，影響度は大きいが適切な課題対応者が判断できない場合は，全体PMのアクションとして課題を消滅させないようにした．なお，このケースは全体PM自身がその初動の対応（方針確定など）に当たり，それを受けて実際のアクションへと繋げて行くケースが多かった．

4）　見えない課題

存在しているのに認識できていない課題である．実際は，外部監査（プロジェクト外部の社内第三者も含む）を実施した場合に，指摘される勧告・改善項目が該当し，監査のヒアリングやレビュを受けている時に認識できるケースも少なくはない．この「第三者の目」は時に非常に有益であり，実際に外部監査や監督官庁（日本銀行）のシステムリスク監査等を受ける上で，プロジェクト運営やQCDについての各種資料や証跡がわかりづらい等，外部からプロジェクトを見た際に，内部からでは見えていなかった課題が数多く検出されたのも事実である．

5）実際の課題管理と対応

５種類に種別される課題の存在とポイントを述べてきたが，実際の課題管理と対応について次に述べる．

①課題管理一覧（アクション・アイテム一覧）の項目

・　管理No（プロジェクト単位のシーケンス番号）

- 重要度（高/中/低）
- 発生日（年/月/日）
- カテゴリ（自プロジェクト固有 or プロジェクト共通）
- 解決（期限/完了日）
- 状況（未着→検討→着手→完了）
- 課題内容
- 対応状況
- 課題提起者（担当プロジェクト or グループ名）
- 課題解決者（担当プロジェクト or グループ名）
- リスク番号（別紙，リスク管理一覧との関連性）

②実際には，以上の項目を網羅した一覧表を作成し，チーム/グループ/各プロジェクト/全体プロジェクトで共通に使用し，プロジェクト発足から解散まであらゆる会議体で利用・管理した．また，大規模になると扱う量も多くなるため，全体プロジェクト会議開催時には，カテゴリが「プロジェクト共通」とされる項目のみを抜粋して進捗や状況管理を実施した．ポイントは，全プロジェクト内に山積する課題（アクション・アイテム）に優先度を付け見過ごさないで確実にトレースしていく事であり，更に，これらをプロジェクトメンバへ公開する事で各プロジェクトの状況を含め情報の共有化，ひいてはプロジェクトの透明性も高める効果があると言えよう．

3.3.2　リスク管理

プロジェクトのリスク管理については，日本ユニシスグループ内でもそのプロセスや手法が策定され，実施が義務付けられている．また，これらについて関連書籍も多く発刊されておりその重要性や手法は比較的容易に理解できる．しかしながら実践となると，現実は日々の直接業務に追われ疎かにしてしまう傾向があるのも否めない．その理由は概ね次の通りに整理できる．

- リスクを定量化しても，直接プロジェクトの資金化（予算化）には繋がらない．
- ステークホルダ（プロジェクト関係者）は，時間や労力を無駄にしたくない．
- ステークホルダ（プロジェクト関係者）は，リスクが現実化するとは思わない．
- 潜在要因として，環境や社会学的トレンドとしてポジティブになる事が強調される傾向にあるため，リスクについて考える事を避けがちになる．

しかし，現実の大規模プロジェクトでこれらを無視する事は命取りであり，その重要性を分かり易く整理し理解する事で，より負荷の少ない継続したリスク管理が実践でき，その有効性も高くなると考えられる．

本項では実際に工夫した事項を中心にポイントを述べる．

1)　全体プロジェクトのリスク相関図の策定

色々と考えられるリスク項目を整備し，最終的に本プロジェクトにおけるリスク大項目（11 項目）として相関関係も含め図 5 のように定義した．

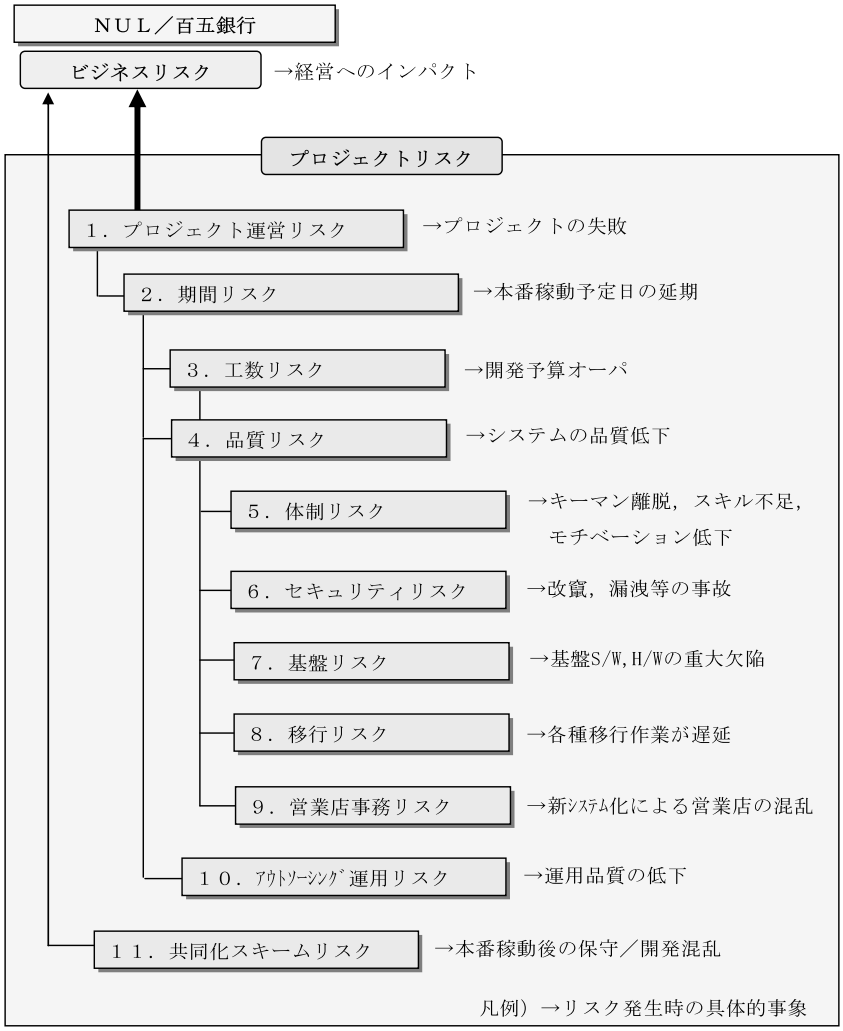

図 5　S-BITS プロジェクト全体のリスク相関図

ここで整理したポイントは，プロジェクトのリスク範囲とその大項目の定義であり，加えて，当該リスクが発生した場合にどのような事象に陥るのかを分かり易い表記とした点である．簡潔に述べると最大のリスクは，「プロジェクトの失敗」であり，それは「本番稼働予定日の延期」であると定義した．また，本番稼働予定日が延期となる要素は，「開発予算のオーバ」「システムの品質低下」によってもたらされ，更にそれらを発生させる要素は，「体制面でキーマンが離脱する」「基盤系 S/W，H/W の重大欠陥が発生する」などと定義していった．

2)　リスク管理の運用

結果的にこうして 11 大項目のリスク項目を定義し，実際には一覧表「リスク管理一覧・対応状況」として各種会議体でのレビュや報告を実施した．なお，このリスク管理手順の実際は，各リスク（小）項目の発生・認識時に必ず 11 大項目に紐付けを行い，更には課題管理一覧上（前述，3.3.1 項　課題管理 5）実際の課題管理と対応）の各項目がどのリスクに

結びつくのかの紐付けも適時実施していった．また，合わせて当該リスクの影響度（大/中/小）や状況（回避/発生/消滅），対応コスト等を管理し，状況に応じて必要とあらばリスク対応費用の使用可否判断を開発管理委員会（プロジェクトの最終意志決定機関）へ諮る事で予算措置を講じた．このようなリスク管理一覧表を用い継続的なコントロールを実施した事で，より分かり易く効率的なリスク管理運営が実践できた．

3)　考察

いずれのリスク項目も，根本的な要因を突き詰めていくと最終的に「人の問題」に繋がるケースが少なくない．すなわちシステム構築（開発）とは，知的労働集約産業と言われる様に，システムの品質や作業の進捗，プロジェクトの管理・運営も人に依存するところが大きい．特に，「体制リスク」で定義した，キーマンの戦線離脱/要員のスキル不足/メンバのモチベーション低下は，大規模になるほど留意し，早めに回避策を講じていく事が重要である．

3.4　間接業務

プロジェクトでの開発に直接関係する業務を直接業務，その反対に直接関係しない業務を間接業務と区分けできる．実際，PM やそのスタッフは間接業務にも多くの時間を割く事になり，大規模開発プロジェクトであるほど，些細な事象にまで注意を払う必要が生じてくる．

本節では，実際に大規模ゆえに発生し対応に苦慮した代表的な作業項目を列挙し，そのポイントを解説する．

1)　各種開発ファシリティの手当て

①開発場所，それらにかかわる備品（部屋，机やイスなど）

②開発機器（H/W，S/W，N/W など）

2)　分散開発における人・物の移動

①客先の開発現場へ長期出張や転勤となる場合など住環境の確保

②これら人の移動に伴い発生する物の移動（PC や荷物）

③分散化に伴う，物理的なメール便の確保（紙・媒体ベースの定期的な搬送）

3)　効率化（経費削減含む）施策

① TV 会議の導入

②電話や Web 会議の導入

4)　メンバのメンタル・ヘルスケア

①定時退社日（ノー残業デー）の設定と励行

②外部カウンセラーによる面談（社内規定や運用に準拠）

③スポーツジムなどの厚生施設の手配（社内規定や運用に準拠）

以上が，代表として挙げられる作業である．人や物に関係した項目が多くなるが，大規模プロジェクトであるほど「多数・大量」という数による負荷が増大する．

これらに実際に対応するにあたり，専属スタッフを配備する事はもちろん，社内の関係スタッフ（人事総務部門）や社外の専門業者（不動産，運送会社など）の協力を得て効率的に対応していく必要がある．また，「外部カウンセラーによる面談」など中にはプロジェクト内では対応しきれない課題もあるが，これらは社内規定上の仕組みを最大限活用する，その仕組みが存在しなければ必要性を提言し，重要性を理解させ実行に繋げられるよう会社を誘導していく

こともケースによっては必要であろう．

このように大規模プロジェクトであるがゆえに考慮し実施すべき間接作業は決して無視できず，プロジェクト運営において与える影響（作業効率やコスト）も大きいと言えるので，開発計画を踏まえ，どの時期に何が必要となるのかを早め早めに整理し手をうっていく事が肝要であろう．

4. お わ り に

大規模プロジェクトを通し，プロジェクト・マネジメントとして工夫したポイントに絞り述べてきたが，特に，大規模PMとしてプロジェクトに取り組む姿勢や目線について言うならば，常に先を見越し（2～3ヶ月先または次工程），その状態へ引き寄せていく様なアプローチやマネジメントが必要かつ有効であると考える．言い換えるならば，今日や明日を考え実行するメンバ，一週間以上先までを見越して行動するチームリーダ/グループリーダ，更に一ヶ月以上先を見越すPM，更には次工程の事を考え，あるべき状態へ誘導する全体統括PMという役割分担で，これらを日々のルーチンワークとして意識し実践する事，上手く実践できている事が大規模プロジェクトを成功させる基本的な要素であると考える．

また，こうした大規模PMの役割を遂行するには，当然ながら資質は重要な要素であるが，まずは小中規模のPMを担当する事で開発全工程の立上げから終結までを実践し経験とキャリアを積んでいく事も重要な要素であろう．

まとめとして，いささか情緒的，精神論的な表現となるが大規模PM実践の勘所・心得を記す．

1. プロジェクトの開始準備・立上げは重要．
2. 開発計画や開発方針の定義は重要且つバイブルに．
3. 勘と経験と度胸も時には重要，されど定量的な裏づけは無視できない．
4. 大きな船は簡単に旋回できない．（止まれない，動けない）
5. 一日のロスは数十人月規模のロス．（またその逆もしかり）
6. 今日明日も重要であるが，3ヶ月先，半年先に目線を見据える事．
7. 間接業務系のスタッフも重要なプロジェクトのキーマン．
8. 悪い事ほどオープンに．
9. 統制されたプロジェクトはルーチンワーク．
10. 情報共有の徹底で，プロジェクトのエンパワーメントを最大限化へ．
11. プロジェクトは一人のスーパーマンだけでは決して成功しない．

最後になるが，本稿の題材とした「S-BITSプロジェクト」は，大規模開発且つ前人未踏のオープン基盤全面採用，更には実現困難とされる共同開発という大きなハードルをクリアできた．これは共同開発パートナである(株)百五銀行殿との強固なリレーションシップがあって実現できたものであり，また，プロジェクトにかかわったメンバ一人一人の絶大なる協力と実行力をもって成し得た事であり，この場を借りてプロジェクト関係各位に深く感謝したい．

＊1　S-BITS（= Succeeding Banking Information Technology for Success consortium，エスビッツ）新たなバンキングシステムを研究し，その実現のため，対応策の策定と実証モデル（ひな型）開発を行い，参加行に提供することを目的とした，次期バンキングシステム検討・検証コンソーシアム．日本ユニシスと日本ユニシス地方銀行勘定系ユーザ7行で構成されている．2000年11月15日に設立．また，コンソーシアムで策定したグランドデザインを実現するために発足した次期オープン基幹系システムの共同開発プロジェクトをいう．

参考文献
［1］ Paul S.Royer，峯本展夫訳，「プロジェクト・リスクマネジメント」，生産性出版，2003年10月
［2］ Robert L.Glass，山浦恒夫訳，「ソフトウェア開発－55の真実と10のウソ」，日経BP社，2004年4月
［3］ プロジェクトマネジメント協会（PMI），「プロジェクトマネジメント・プリンシプル」，（株）アイテック，2006年9月
［4］ 吉野良成，「システム構築プロジェクトの現状と課題」，ユニシス技報，日本ユニシス，Vol.27 No.2 通巻93号，2007年8月，P3～15

執筆者紹介　葛　谷　幸　司（Koji Katsuya）
1985年日本ユニバック（株）入社．証券システムのSEサービス，1992年より地銀勘定系パッケージ「TRITON」の基盤開発と適用サービスに従事．その後いくつかの金融プロジェクト（PM）や新事業の企画部門を経て，2004年より日本ユニシスグループと（株）百五銀行で共同開発した次世代オープン基幹系システム『S-BITSプロジェクト』の全体統括PMを担当．現在，S-BITS適用統括プロジェクト長．