

CMMI レベル4 以下の EVM Applied EVM for under CMMI Level 4 IT Project

蓮尾 克彦 (Hasuo, Katsuhiko)
IT コーディネータ協会
hasuo@itc.or.jp

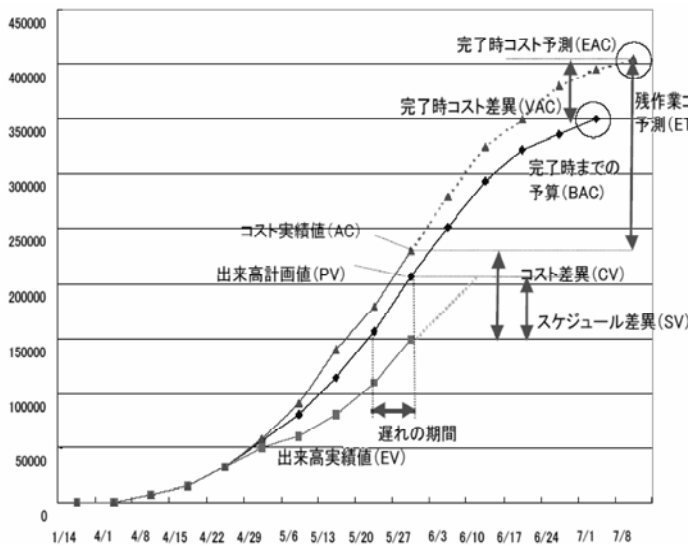
Abstract : In EVM that has succeeded in America, the engineering process of IT that becomes the object of managing is more than CMMI level 4. Classic project management techniques like WBS and PERT, etc. is effective in the serial engineering process more than CMMI level 4. However, these static techniques are not applicable by the system development with re-work. PERT of the next generation type was tried as a technique for managing under CMMI level 4 Project of Japan referring to the procurement model of U.S. Marines.

キーワード : PMBOK, CMMI, ITSS, SLCP, What-if, WBS, TOC, SD, NDP モデル

要旨 : Work Break Down Structure(WBS)等、古典的な手法のオペレーショナルな情報では進捗の遅れは判断できるが、マネジメントに必要な意思決定ができない。プロジェクト計画の変更や中止の意思決定に Earned Value Management (EVM)は有効な手法であるが、経済産業省の推奨[1]にも拘わらず普及していない(図1)。

Evaluation and Review Technique(PERT)やWBSなどフィードバック機能を持たないオペレーショナルな開発プロセスを対象とする古典的なプロジェクト管理技法の限界でもあり、アクティビティ内で生産性が変化したり、アクティビティ間で相互に生産性に影響のするエンジニアリング・プロセスを System Dynamics(SD)を使って表現できるモデルを試行する。単純なジェネリックモデルを複数組み合わせることにより、PERTと同様に、複雑な開発プロセスを簡単に表現できる次世代型のPERT手法を開発した。

このツールを活用し、発注者側はリスクを最小化でき、また請負側のソフト企業 Capability Maturity Model Integration(CMMI), IT Skill Standard(ITSS), Software Life Cycle Process(SLCP)などのソフト企業のベストプラクティスを組み合わせることにより、ベテランの暗黙知を形式知とし、ソフト会社の請負ビジネスのマネジメントプロセスの革新を図ることができる。



BAC(Budget At Completion) 完了までの予算、予定コスト。

PV(Planned Value) 出来高計画値 計画時に、各作業に割りあてられた価値。

EV(Earned Value) 出来高実績値 現時点までに完了した作業に対して、元々割り当てられていた出来高。

AC(Actual Cost) コスト実績値 作業を行うために実際に必要となったコスト。

SPI(Schedule Performance Index) スケジュール効率指数
 $SPI = EV/PV$ 各作業のスケジュール面から見た効率

CPI(Cost Performance Index) コスト効率指数
 $CPI = EV / AC$ 各作業のコスト面から見た効率。

図1 EVM 導入ガイドライン (情報処理振興事業協会)

日本の IT 調達の特異性にもよるが、Program

1 . PMBOK とエンジニアリング・プロセスの課題

1 . 1 IT 開発のプロセスと CMMI

PMBOK(Project Management Body of Knowledge)はこれまでのプロジェクト管理のノウハウが盛り込まれたプロジェクトマネジメントプロセスのベストプラクティスであり、プラント建設や国家的プロジェクトなどでの実績がある。IT 開発においても PMBOK や EVM は普及している。しかし、米国での IT 開発での成功事例ではプロジェクトマネジメントの対象となるエンジニアリング・プロセスが CMMI レベル4 以上のソフト企業であり、日本のソフト企業はレベル4 以下である。

CMMI はキャプラン等[2]が提言する戦略と業務の管理のダブルループと同じ考えで、Balanced Score

Card(BSC)が企業の組織能力(Capability)[3]の向上を狙いとしているのに対して、ソフト開発組織の能力向上を狙いとしたカーネギーメロン大学が中心となって開発した継続したプロセス改善(Software Process Improvement)活動の成熟度モデルである。

米国と同じく技術者の流動化の激しい日本でもベテランのノウハウをソフト企業に蓄積する手段として1995年頃から導入されている。しかし、中国やインドではCMMIレベル5のソフト企業が続出しているのに比べ、日本の多数のソフト企業はレベル4以下である。

1.2 CMMIレベル4のソフト企業的能力

CMMIでは管理すべき知識項目として24のProcess Areaが規定されているが、レベル4のソフト企業では2つの重要なプロセス能力が達成されている。

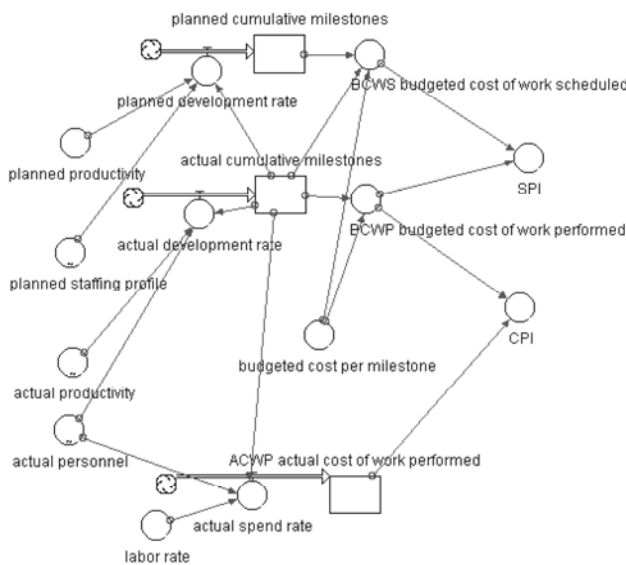


図2 南カリフォルニア大学のEVMのSDモデル
 1) 組織プロセス実績 (Organizational Process Performance) 過去の実績が組織的に蓄積できる。
 2) 定量的プロジェクト管理 (Quantitative Project Management) 定義されたプロセスを定量的に管理でき、品質目標およびプロセス実績目標を達成できる。

CMMIレベル4以下の多くの日本のソフト企業ではProject Manager(PM)が品質に関する責任を負う為、PMの能力に開発品質が依存する。レベル4以上のソフト企業ではCOCOMO(Construction Cost Model)[4]等を使った開発の見積もりが可能であり、南カリフォルニア大学(USC)ではSDを使ったEVMを提案している[5](図2)。しかし、これらのモデルやCOCOMO、ケーパージュンズ[6]等での見積もりはソースコード数やファンクションポイント数が確定した後の計算であり、手

戻りなど、フィードバックを持っていない為、手戻りの発生する日本のIT開発には適用できない。

1.3 米国のソフトウェア調達とSLCP

米国でのPMBOKやEVMの適用事例はソフトウェア要件が明確になった後工程のソフトウェア開発プロセスであり、顧客とコラボレーションしながら要件が明確になるビジネスアプリケーション開発例は少ない。

ビジネス要件をIT要件に変換するプロセスはヨードン等のソフトウェア工学の専門家は「社会学」の分野と位置付け、「ソフトウェア工学」の分野から除外している。ソフトウェア工学の分野ではソフトウェアの商取引のフレームとして世界標準のSLCPに対応してプロジェクト管理ができるが、手戻りによるフィードバックは表現できない。ソフトウェア工学の上流プロセスとして、顧客の要件を導出(Derive)するプロセスとして要求工学[7]が定形化されつつあるが、未だ未成熟である。

1.4 米国海軍の調達プロセス

顧客の要件が変化し、設計変更が発生する開発に軍艦の建造がある。米国海軍の資料によれば、要件が変化し、軍艦の完成時には基本設計段階での予算の2~3倍になるといわれている。これら課題に対応する為に、数年前から米国海軍のIntegrated Product and Process Development (IPPD)が制定され、その中ではTOCやWhat-if Theoryを取り入れたEVMとしてNaval Design Process Model (NDPM)[8]が使われている(図3)。このNDPMではデザインのスパイラルレート、レビューによるエラーのレート

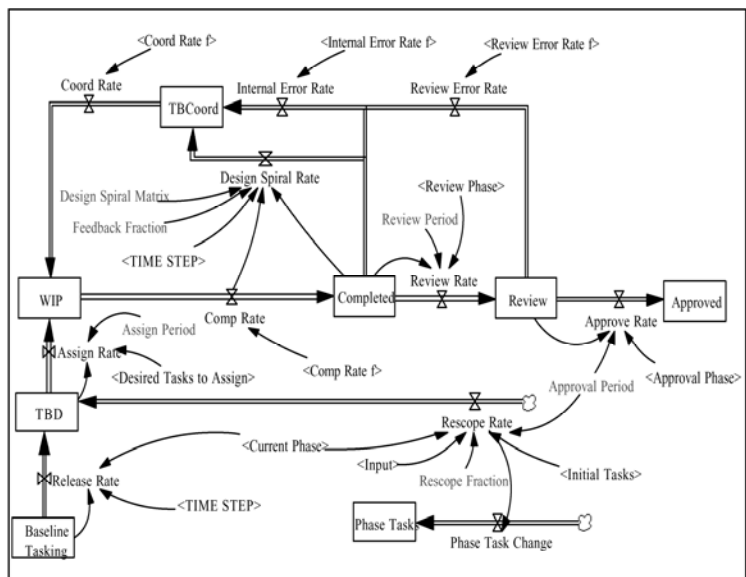


Figure 12 Naval Design Process Model Task Accomplishment Structure

図3 Naval Design Process Model

などを含むフィードバックモデルであり、日本のCMMIレベル4以下のシステム開発に類似している。しかし、

IT 開発に適用する為には、複雑な開発プロセスを忠実にモデル化する必要がある、モデルの作成には高度なスキルを必要とし、一般的な IT 開発の現場では使い難い。

2. 実績を反映した予測ができないEVMのツール

2.1 実績を反映した予測の困難さ

EVM が日本で普及しないのはエンジニアリング・プロセスの特殊性だけでなく、マネジメントの手法やツールに原因がある。EVM の目的は実績を把握し、予測することにあるが、スタティックな古典的な手法では実績を反映した予測の計算が複雑であり、並列プロセス等では計算が複雑になり、IT 開発現場ではほとんど適用できない。

2.2 プロジェクトの生産性の把握

プロジェクトの完了を予測する為には、現在進行中のプロジェクトの生産性を把握する必要があるが、この生産性の把握方法はスケジュール効率指数:SPI(Schedule Performance Index)とコスト効率指数:CPI(Schedule Cost Performance Index)の2種類ある。情報処理振興事業協会の「EVM 活用型プロジェクト・マネジメント導入ガイドライン」では、発注者側は現在の生産性の実績を $SPI = EV / PV$ (Planned Value) で把握する事を推奨している。

しかし、アクティビティごとに投入する人数と生産性は異なる。この為、SPI はそれぞれのアクティビティごとに異なり、これまでのEVをベースにアクティビティごとにSPIを計算し、完了時の予測(実績を反映したPV)を計算するのは困難である(図4)

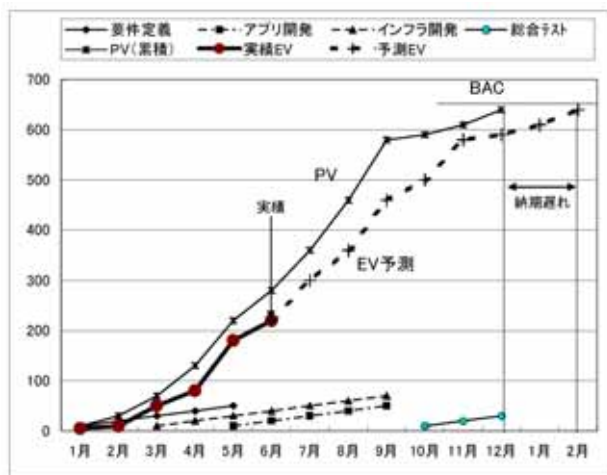


図4 実績を反映したEV

2.3 請負側の生産性の把握

請負契約の場合の発注者はPVとEVの実績を反映したSPIを管理すればよいので単純であるが、請負側はAC(Actual Cost)とEVを管理しなければならない。

EVを得るためのコスト効率指数CPIは $CPI = EV / AC$ で計測する。ACは単価と工数に分解でき、生産性が計算できる。開発プロジェクトで具体的な実績データとして把握できるのはACだけであり、EV/ACがプロジェクトの実効能力であり、真の生産性となる。この為、ACからどれだけのEVが生成されるかのマネージング情報を蓄積する組織としての管理がEVMには必須である。

2.4 ACのマネジメント情報

ACとEVが一致しない場合、PMは懸案事項として管理するが、この懸案事項には2種類考えられる。1つは見積もった作業が予定の工数で終わらない場合であり、プログラム設計の後工程のプログラム作成やテストの工程でみられる。これらオペレーショナルな情報の実績は統計学者のケーパージョーンズ等により、大小7,000のプロジェクトで収集されている。

もう1つは見積もりがされていない作業が発生する時であり、要件定義の作業中などで未知の事項(Unknown)が発見される場合である。これらの事象の発生頻度は前工程の品質に依存する。

Unknownに関してはケーパージョーンズ等のオペレーショナルな情報は参考にならない。ITプロジェクトの管理にはリスクとしてこれらUnknownに関するマネジメント情報が必要となる。

CMMIレベル3ではオペレーショナルな情報と、これらマネージングに必要な情報の組織的蓄積を要求しているが、日本のソフト企業でマネージング情報を蓄積している組織は皆無であり、ベテランのPMの経験と勘に依存しているのが実情である。

3. SDモデルの設計

3.1 ジェネリックモデル(サブモデル)の構成

10億円規模の開発ではWBSで展開される作業項目(アクティビティ)の数は1000項目程度になる。これらはグルーピングして1つのプロセス(工程)として管理できるようにWBSを作成する。これらの開発プロセスは複雑になり、相互作用も発生する。これを1次元のSDモデルで表現するのは困難である。

複雑な開発プロセスをモデル化する為には、期待値をEVに変換するプロセスと作業能力を生成する2つプロセスを持つ、NDPMに比べ単純なサブモデルを作成し、これをジェネリックモデルする。(図5)

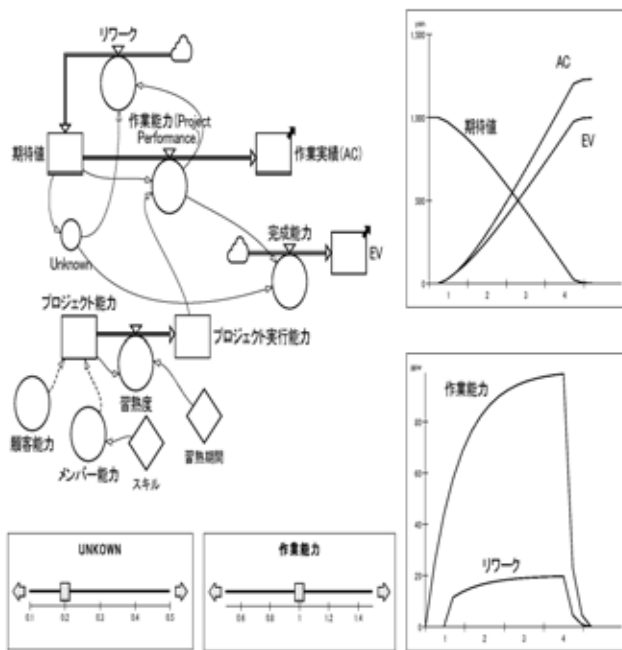
EVを生成するプロセスでは、Unknownに対するリワークが発生し、EVとACとは一致しない。また作業能力を生成するプロセスではプロジェクトがその能力を全て発揮するには習熟期間を必要とする。

ソフト開発のプロジェクトのサイズNが大きくなるとコミュニケーションロスが発生するが、Blocks[9]

によれば $N(N-1)$ である。RamiI 等の最近の研究では $0.03(N)^2$ が使われている [10]。日本では外注会社や派遣のメンバーが加わる為、さらに生産性は低下する。

USC では「納期」を生産性の要素にいれているモデルも研究されているが、今回はこれらの要素は反映していない。いずれにしろリワークの発生やコミュニケーション・ロスなどで、AC と EV は一致しない。

Unknown の発生率は前工程の品質に依存する。ベテランの PM はこの発生率を経験で体得しているが、CMMI レベル4以下のソフト企業ではこの発生率をPM個人の知識としてではなく、組織として蓄積することが必要である。また Unknown による生産性の低下は担当者のスキルの問題ではないことに留意した技術者の能力 (ITSS) を評価する必要がある。



- 1) 期待値 (Expectation EP) : 顧客が一定の費用で、期限内に獲得したい機能や品質 (単位: 円)
- 2) 作業能力 (Project Performance per week) : 一定期間内に価値を完成する能力 (単位: 円/週又は月)
- 3) 獲得価値 (EV) : 一定期間内に完成した EP に見合った作業価値 (単位: 円)
- 4) Unknown : 作業中に発見される未知の作業 (単位: 円) 価値を高める追加機能ではなく、当初想定した価値を維持する為には必要な機能。発生量は前工程の品質に依存し、前工程の欠陥 (洩れや間違い) が原因である。Unknown は 1/3 位作業に入ってから時点で発見される。

図5 リワークを考慮したジェネリックモデル

3.2 ITSSを使った期待値の計算

ITSS は IT 技術者の流動性を高め、日本の IT 技術者の能力向上を狙いとし、2003 年に初めて経済産業省が制定した、顧客の投資局面別の IT 技術者のレベルで

ある。顧客の期待するシステムの価値 (期待値) の算出にはこの ITSS を参照する。

$$\text{職種別人工} = \text{仕事量と難しさ} \div \text{作業期間}$$

$$\text{期待値 (金額)} = (\text{職種別人工} \times \text{職種別単価})$$

職種	スキル	達成度レベル
コンサルタント	経営戦略 ビジネス戦略策定	4~7
IT アーキテクト	課題整理 / 分析 ソリューション設計	4~7
アプリケーション スペシャリスト	コンポーネント設計 ソリューション構築	1~6
プロジェクト マネジメン	マネージング	3~7

図6 ITSS (顧客の投資局面からみた職種)

ITSS では 11 職種 (33 の専門分野) とそのスキルレベルを定義してあるが、顧客の投資局面からみた“職能”と同じ意味である (図6)。

3.3 ベースラインの作成

プロジェクトメンバーおよび顧客と合意した基準となるベースラインを策定する。

1) プロジェクトメンバーとの合意

作業開始の一定期間後に EV が得られる。さらにプロジェクトが所定の能力を発揮するまでには、知識獲得の時間が必要となる。この習熟期間を考慮したプロジェクトの実行能力を反映した EV を計画値 (PV) とし、PM はメンバーと合意する (図7)。

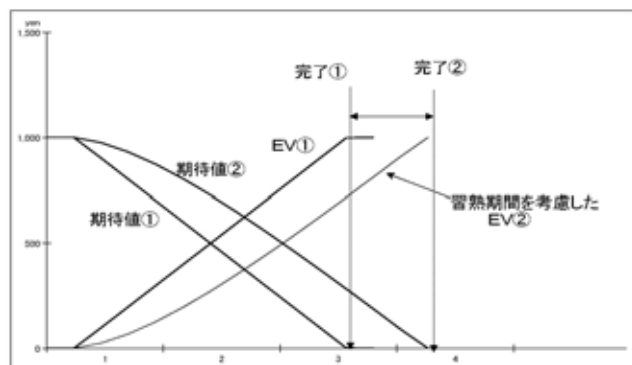


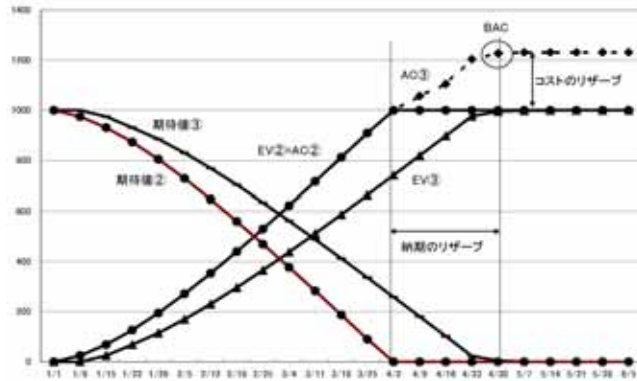
図7 メンバーと合意した PV

このプロセスは米国では Engagement Management とよばれているが、日本では「にぎり」と呼ばれている。この時の $PV=EV$ は技術者を評価するときの ITSS

の達成度指標の評価となるが、ITSS を企業内に定着させるには目標の達成の評価だけでなく、タスク特性（知識の共有 etc）を満足することを ITSS のガイドラインでは要求している。

2) 顧客との合意

メンバーと合意した PV をベースに発生が予想される Unknown を考慮したベースラインを設定する (図 8)。



リワークの無い時は EV は AC と一致する。

リワークがある時の AC は途中まで AC と同じである。

図 8 顧客と合意したベースラインと BAC

リワークのリスクを考慮したプロジェクトリザーブに関して PM は経営者(上司)と合意し、これをベースラインとし、予算金額 BAC(Budget At Completion)納期を決定する。上司と合意したベースラインは最終的には顧客との合意が必要である。この合意にもエンゲージマネジメント能力(ネゴシエーション能力)が必要であり、「貸し」「借り」が発生し、企業会計や税法と整合性をとる必要がある。

ITSS ではあいまいな契約条件や、あいまいな検収条件でも成功裡にプロジェクト完了させることを PM の能力として要求しているが、リスクの大きな仕事を請けるかどうかの判断は PM の権限ではなく、経営判断である (図 8)。

3.4 次世代型 PERT

それぞれのプロセスごとに期待値を算出し、ジェネリックモデルを使用し、次世代型 PERT を作成する。作成したジェネリックモデルを組み合わせ、それぞれのプロセスの開始条件を設定しベースラインとする (図 9)。

この時の開始条件は

- アプリケーション開発：要求定義の EV 値が 70~80%
- インフラ開発：予定開始月日
- 総合テスト：全アクティビティの EV 完了時である。

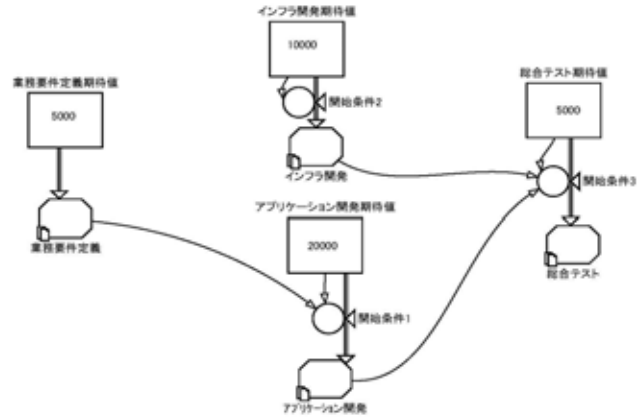


図 9 ジェネリックを組み合わせた次世代型 PERT

4. 実績の反映と予測

プロジェクトの遅れは種々雑多な要因で発生する。これらを分析し、対応するのは PM の役割であるが、プロジェクトの継続か中止を決定するのは経営マネジメントの役割であり、発注者側と請負側の経営者が理解しやすい報告を PM は求められる。

今後の進捗を鳥瞰する為には、EV の実績を記録し、現時点の EV の実績に合うようにそれぞれのアクティビティのプロジェクトの生産性(EV/AC)を調整し、現在の生産性を反映した予測をする。図 11 は実際のプロジェクトの WBS であるが、このプロジェクトのベースとなった計画を次世代型 PERT で作成する (図 10)。

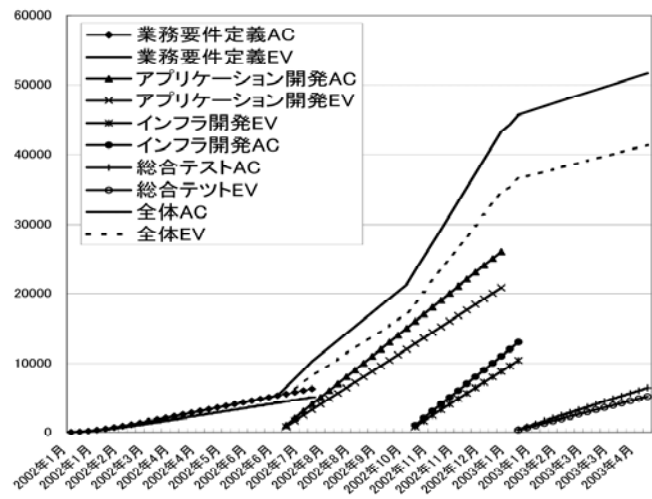


図 10 計画作成時の EV と AC

WBS でプロジェクトの遅れは把握できていたにもかかわらず、今後の予測が出来なかった為、経営の意思決定が4ヶ月遅れ、その結果ソフトウェアライセンス契約やハードウェアの購入等、結果的に15億円プロジェクトを中止した。回復不可能な遅れを発見したこの時点での AC は4億円であり、早期に意思決定できなかったことが悔やまれる。次世代型 PERT で作った図 10 に EV の実績値に合うようにシミュレートした予測が図 12 であり、WBS に比べ経営への説明がしやすいのは明

白である。納期が大幅に遅れたり、ACがBACを大幅に超える時、プロジェクト継続か中止の判断はPMの役割ではなく、発注者側の経営者の役割であり、説明責任はPMにある。予測したEVとBACの差が請負契約の違約金を超えるようであれば、契約を廃棄する判断はソフト企業の経営者の役割である。

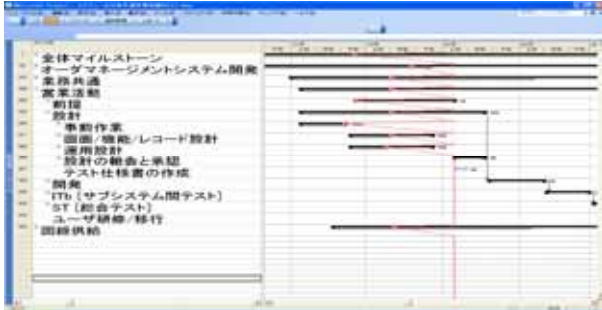


図 11 中断になったプロジェクトのWBS

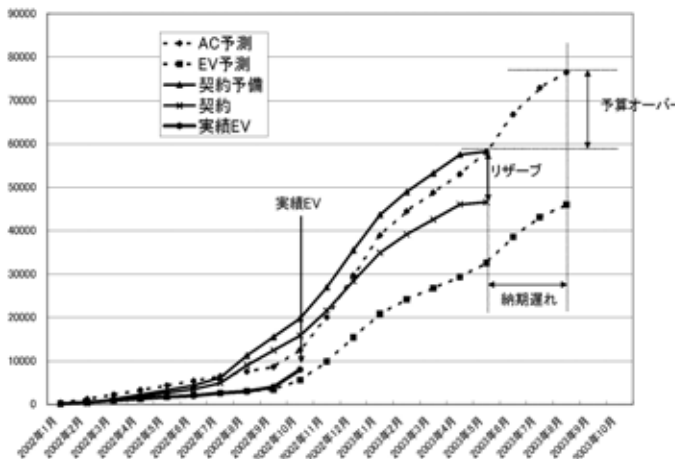


図 12 プロジェクト全体の鳥瞰図

5. 終わりに

不確定な要素が多いのが IT 開発(Development)であり、完成するまではそのリスクが観測できず、その結果、納期直前に問題が発覚し、発注者と請負側双方に重大な損失を蒙る例が多い。

開発のリスクは請負側のエンジニアリング・プロセスのレベルに依存する。請負側のエンジニアリング・プロセスが CMMI レベル 4 以上の信頼性のある組織であるときは、報告される EV を監視すればよい。しかし、請負側のエンジニアリング・プロセスが CMMI レベル 4 以下の場合、生産性に関する情報は組織内に蓄積されておらず、プロジェクト成功の鍵は PM の能力のみに依存する。

ビジネスシステムの開発は発注者と請負ベンダーのコラボレーションが必要であり、CMMI レベル 4 以下の IT ベンダーと協働する時には、発注者は EV と同時に AC を監視する必要がある。一般的にコスト効率指数 EV/AC は 0.8 以上は必要であり、それ以下

の場合は仕事量の見積りか、プロジェクト能力の見積りなど、スコープマネジメントに誤りがあったことになる。

発注側の担当者と請負側の PM が協働するのは当然であるが、プロジェクトの継続か中止の判断はプロジェクトの当事者ではなく、発注側と請負側の経営者の判断である。これら経営者に対して、進捗の遅れは判断できるが、今後のリスクに関しては判断できない WBS での報告は適切ではない。経営者の意思決定の情報として提供する EVM は有効な手法である。

プロジェクト成功を阻害する要因は多く、複雑に絡み合うが、単純なジェネリックモデルを複数組み合わせ複雑なプロジェクトを表現できる次世代型の PERT が有効である。常に将来を予測しつつ現状の EV を管理する手法は IT 開発以外のプラントプロジェクトにも適用できる。IT 開発においては CMMI, ITSS, SLCP などのベストプラクティスと組み合わせ、ソフト企業のマネジメントプロセスの革新に貢献できる (図 13)。

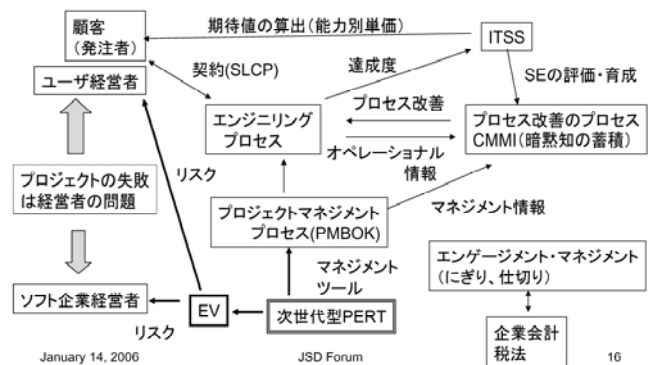


図 13 ソフト企業のマネジメントプロセスの革新

EVM を定着させるには管理プロセスと共に、簡単に使用できるツールが必須であり、オブジェクト技術を取り入れた新機能を取り入れて、今後さらに次世代型 PERT を改良する予定である。

参考文献

- [1] 情報処理振興事業協会: EVM 活用型プロジェクト・マネジメント導入ガイドライン、2003 年
- [2] ロバート・S・キャプラン、デビッド・P・ノートン: The Strategy-Focused Organization, 2001
- [3] 蓮尾克彦: SD による戦略的 BSC の解法、2005 年
- [4] Barry Boehm : COCOMO、1981年
- [5] Ray Madachy : CS599 Software Process Modeling(USC)1999
- [6] Capers Jones : ソフトウェア開発の定量化手法、2001 年
- [7] Barry Boehm: Expectations Management、1999
- [8] Thomas Laverghetta : Technological complexity and budgetary, Naval Engineers Journal, 1999年
- [9] Fred Brooks : The Mythical Month-Month, 1975年
- [10] Juan F Ramil : Introduction to System Dynamics, 2001年