

Stakeholder分析手法の比較報告：CVCA vs. OnionModel

Stakeholders Analysis: CVCA vs. OnionModel

嶋津恵子^{†‡}

Keiko Shimazu^{†‡}

[†] 慶應義塾大学 デジタルメディア・コンテンツ統合研究機構

[‡] 慶應義塾大学 大学院 システムデザイン研究科

[†] Faculty of Commerce, Tokyo International Univ.

[‡] Faculty of Science and Technology, Keio Univ.

要旨

本書では、情報システム構築プロジェクトの失敗の原因が、Vee モデルで表現されるシステムエンジニアリングの左辺の最上位に位置する要求獲得にあると考えている。そこで、近年発表が相次いでいるステークホルダ分析手法を試用した。直感的に扱えるもの(CVCA)と、システムティックに網羅性高くステークホルダを抑えられるもの(OnionModel)を採用した。今回の試みでは、CVCA の導入は容易であり、直感的にモデリングをおこなうことが可能であったが、重要な(サービス仕様を決定するのに注目すべき)影響を与えるステークホルダを見落とした。一方、OnionModel を採用した場合、網羅的にステークホルダを把握し、またそれらの影響を確認することが可能であった。今後情報システムやソフトウェアの構築プロジェクトにおける十分な実証検証と、改良が必要だと思われる。

1. はじめに

1995年に発表された米国スタンディッシュ・グループ (Stahdish Group international, Inc) の ChaosReport は、IT 業界でプロジェクトに携わる者たちに大きな失望を与えた。これは、米国の企業や政府関連組織の、IT 関連プロジェクトの 1994 年の実態を調査・分析したものである。これによると、成功(succeeded)が 16% (大規模の場合 9%)、失敗 (failed) し中断もしくは廃棄されているものが 31%、残りの 51%は中断や廃棄はされていないものの課題が残されたまま (challenged) であると報告された。課題の内訳は、予算の超過 (対計画平均 189%)、納期の超過 (対計画平均 222%)、要求機能仕様の未達 (対計画平均 61%、大規模プロジェクトは 42%) である。これを受け、プロジェクトマネジメントに関する研究が進み、最適化と効率化の手法が導入され、2004 年度の報告では、成功したプロジェクトが 30%近くまで伸びた。一方、日本の状況も米国と大きな差はない。ChaosReport と同様の調査を日経コンピュータと JUSA(日本情報システム・ユーザ会 : Japan Users Association of Information systems)が実施している。前者は 2002 年の調査であり、これによると、成功したプロジェクトは 26.7%である[中村 03]。これに対し、3年後の JUSA の調査報告によると、成功したプロジェクトの割合は 30%まで伸びている[JUSA 06]。しかしながら、この成功率の大きさは満足できるものとはいえない。特に IT 産業は、その生産物が全産業の中で必要とされていることが他の産業と異なる点であり、最大の特徴になっている。つまり、IT 業界の停滞や衰退は、全産業界の競争力低下につながる可能性が高い。

今回、筆者は情報システムの構築に注目した。情報システムの構築は、IT 産業を構成するインターネット業界、ハードウェア業界、ソフトウェア業界、通信・プロバイダ業界のすべてにかかわるシステムエンジニアリングである。そこで、今回、情報システム構築プロジェクトの失敗の原因がどのフェーズで発生しているかを特定し、情報システム構築プロジェクトではほとんど導入が試みられていないモデリング手法を採用することを試み、その有用性を論じた。手法を利用した情報システム構築プロジェクトは、慶應義塾の学術コンテンツを学内横断で検索するサービスシステム[嶋津 06]である。

本書は次の構成を採る。2章に IT 系プロジェクトの成功率の低さの原因が、要求分析フェーズにあることを示す。3章に CVCA を利用した例、4章に OnionModel を利用した例を示す。5章に、2つのモデリング事例を考察し、6章にまとめを述べる

2. システムエンジニアリングとVモデル

21世紀に入ると日本では、情報システムの構築や更新に関する重大な問題が相次いだ。2000年問題への産業界あがての対応、2001年に相次いだ携帯電話機のソフトウェア不具合による回収騒動、2002年の第一勧銀、富士銀行、日本興行銀行が合併して出来た「みずほ銀行」のオンラインの大規模障害、最近では2005年ジェイコム株の誤発注事件があり、さらに2007年にはシステム障害により全日空の国内線のチェックイン端末が使用できなくなった。これらの騒動を通じ、日本の各産業界では、ソフトウェア品質の重要さが認識されるようになった。それにも関わらず、情報システム構築やソフトウェア開発のプロジェクトにおける具体的な解決策は提示されず、現在も依然として人海戦術で、情報システムをリリースする直前に可能な限りバグ（不具合箇所）をみつけ修復するという方法でしか、品質維持対策への対応が採られていない。しかもそれらの多くが、計画段階で見積もれなかった、“手戻り”(Incidental Development)作業だと言われている[渡部 98 他]。[経済産業省 05]によると、ソフトウェアの開発に際し平均280件の不具合が発見されているが、それらが発見（存在に気づく）された工程はテストと実装段階がほとんどである。一方、それら不具合がどの工程で発生したかを見ると、実装段階が47.3%、一方、要求分析から設計までの上流工程でも40.5%発生している。つまりテスト工程では、実装時の不具合だけでなく、上流工程で発生した不具合も解決することが求められている。仮に上流工程で作成された仕様が正確であれば、テスト工程では仕様どおりに修正する作業に集中できる。ところが仕様そのものに誤りがあった場合、これを精査しその上で情報システム全体を見直すことになる。最悪の場合は、ほぼすべての実装結果が無駄になることもある。できるだけ無駄にすることなく、既実装部分を活かそうとするとアドホック的なパッチファイルの量産になる。問題が複数みつかり、これらの作業の複雑さはさらに増す。これは、言わば上流工程で発生した問題をすべて下流工程で帳尻を合わそうとする“無理”であり、このような方法での要求仕様の正規化の精度は期待できるはずがない。前掲のChaosReportの2000年版では、何とか完成させた情報システムやサービスであっても、平均45%の機能がまったく利用されておらず常に利用する機能は7%にしか過ぎない。また、この“無理”は、保守・運用のフェーズにまで増員・コスト増加を引き起こしている[大山 04]注3。要求分析フェーズに情報システム構築プロジェクト上の問題があるという見方は、(社)日本情報システム・ユーザ会が毎年行っている企業IT動向調査の結果にも表れている。2002年以降の毎年の調査で、要求仕様の作成の不十分さを挙げる声が多い。この要求仕様の不十分さは、同報告に挙げられている別の問題点、“システム規模が大きくなるに従い予算、工期および品質に対する満足度の低さ”にも直結している。これらのことから、情報システムやサービス構築プロジェクトにおいて、成功の鍵は要求分析の正確さに大きく依存すると言える。

(情報システムだけでなく広く)システムエンジニアリングは、異なる分野の専門的アプローチを組み合わせ、全体最適化を実現することで完成度の高いシステムを構築することを目指す[incose 95]。

[incose 95]では、システムエンジニアリング上での

全体最適化を実現するためのフレームワークとしてVeeモデルによるマネジメントを推奨している(図1)。Veeモデル上の左辺上の(任意の)作業工程と、水平線上に位置する右辺上の作業工程間で最適化を計ることで、システム全体のそれを実現するものである。このフレームワークは、1990年代から導入されているが、特に最近では上流工程の最適化が求められ、顧客分析・特定(Stakeholder Analysis)の手法提案が注目されている[Alexander 06]。今回、直感的導入が可能なCVCAと大規模システムエンジニアリングで実績のあるOnionModelに注目し、それぞれの特徴を調べた。

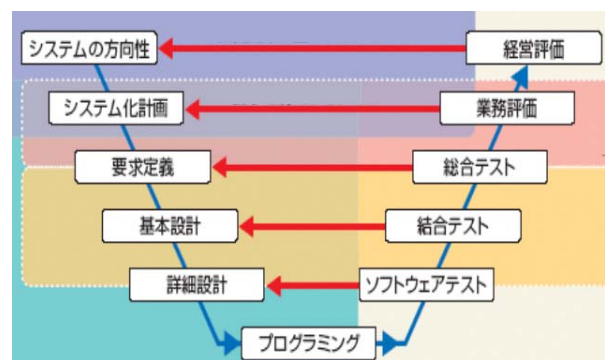


図1 システムエンジニアリングにおけるVeeモデル
IPAのホームページから転載

3. Stakeholder Analysis: CVCA

CVCA (Customer Value Chain Analysis)は、プロジェクトの目標である開発対象のシステム製品や、実現するサービスに対する真の顧客の特定するためのステークホルダ間の関係の表現方法である [Ishii 03]。一方 ER(Entity-Relationship)モデルは、情報システム構築において、バイブルとして用いられてきた [Chen 76]。これはデータベースにデータとして搭載する問題領域に対し、概念設計や業務分析するために用いられる。ER モデルが実体とそれらの関係だけで対象領域を表現しているのに対し、CVCA は、実体の中の特に”人”に注目して、ステークホルダを楕円やシンボルを用いて表現する [京屋]。さらに日と以外の実体は、流通するモノ (Object) としてとらえ、ステークホルダ間の関係を示す矢印線上に個別のアイコンで表記する。関係を持つステークホルダ間で製品、代金、クレーム情報が引き渡されるのが直感的に把握できる。CVCA は、ER の単純明快さを継承しつつ価値の連鎖 (Value Chain) を視覚化することに成功している。

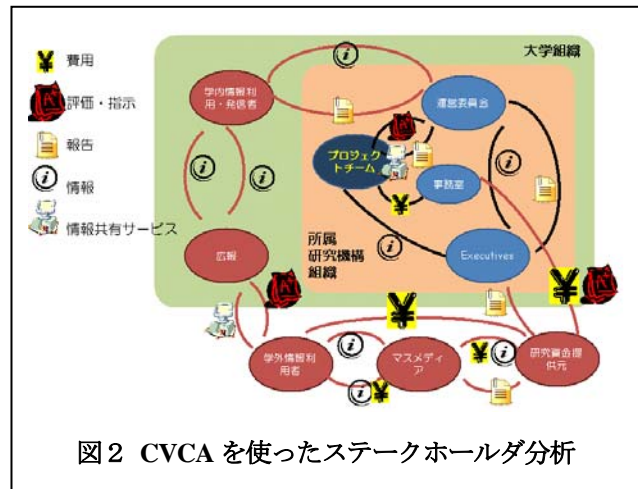


図2 CVCA を使ったステークホルダ分析

4. Stakeholder Analysis: Onion Model

[Alexander 05]は、システム開発における上流工程で、真のユーザを特定することの重要性を主張している。一般にシステム(何らかの立場で)に関わる人物を“ユーザ”と表現することが、誤った要求を仕様化することに繋がるとし、システムの企画、設計、開発、テスト、運用(maintenance および operation)のすべての工程にかかわる「人」を stakeholder とみなす。そしてそれらを [Alexander 05] が提案するモデル図上に展開することで、関係性と成果物(システム)の仕様に反映する最優先 stakeholder を特定することが可能になっている。CVCA を作成する際、システムに関係する人物や組織を並べ挙げるのに対し、OnionModel の作成では、雛型が用意され、それに該当する人物や組織を特定し埋めていく。配置用のマップは、同心円で描かれ、中心に開発するソフトウェア/ハードウェアプロダクトをおく。中心円から外側に向かい、構築するシステム全体、直接接続されている他のシステム、環境として影響を与えるもの(社会システム)を表現する。また各ステークホルダ雛型は、Slot と呼ばれステークホルダとしてのシステムに対する役割(Role)別に用意されている。それぞれのステークホルダ間の影響力(pressure)を矢印で示すことで、仕様に反映させる重要要求を発見する。

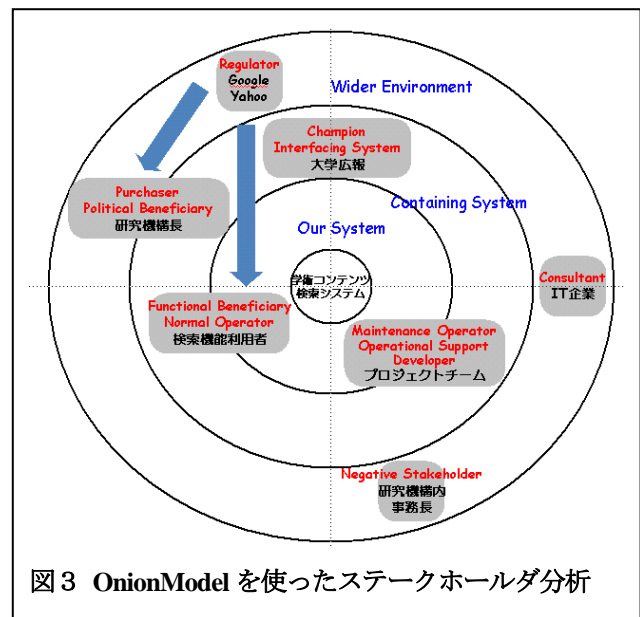


図3 OnionModel を使ったステークホルダ分析

5. 考察

今回の検証では、CVCA を用いたときには見落とされた Regulator が、OnionModel を用いたときに

は、明記された。さらにステークホルダ間の影響力を書き加えると、この Regulator が他のステークホルダに大きく影響を与えていると確認された。これにより CVCA による分析を行って設計した情報システムと、OnionModel による分析を行って設計した情報システムとで、実装仕様が異なった。Regulator の存在を認識した後、再度作成しなおした CVCA モデルを図4に示す。

また、モデルの作成に関し(両者とも専用ツールは使用せず)、要した手間にも差が発生した。CVCA は、作成ルールが単純であるため、特に学習の時間を必要とせず(プロジェクトメンバーとの打ち合わせを含めても)、2~3 時間で完成する。一方、Onion Model は、オブジェクト指向を基盤にしていることもあり、クラス概念を理解しておくことや、雛型として用意されているステークホルダの役割を学習していただくことが求められる。筆者の実験では、事前の学習に1日以上を要した。

両モデリングともシステム化対象領域に対するセマンティカルな解釈を前提としており、ER モデリングが実用性を高めたように、正規化の方針が必要である。

6. まとめ

本書では、情報システム構築プロジェクトの失敗の原因が、Vee モデルで表現されるシステムエンジニアリングの左辺の最上位に位置する要求獲得にあると考えている。そこで、近年発表が相次いでいるステークホルダ分析手法を試用した。直感的に扱えるもの(CVCA)と、システムティックに網羅性高くステークホルダを抑えられるもの(OnionModel)を採用した。今回の試みでは、CVCA の導入は容易であり、直感的にモデリングをおこなうことが可能であったが、重要な(サービス仕様を決定するのに注目すべき)影響を与えるステークホルダを見落とした。一方、OnionModel を採用した場合、網羅的にステークホルダを把握し、またそれらの影響を確認することが可能であった。今後情報システムやソフトウェアの構築プロジェクトにおける十分な実証検証と、改良が必要だと思われる。

参考文献

- [1] (社)日本情報システム・ユーザ協会, 企業 IT 動向調査報告書, 2006年4月5日, 2006
- [2] 中村 健助, 矢口 竜太郎, 2003年情報化実態調査, 日経コンピュータ, 2003年11月17日号, pp. 50-63, 2003
- [3] 嶋津恵子, “知の流通と再編を実現する攻めのコンテンツマネジメント”, IDG Japan 主催 Contents Management Forum 2006, 基調講演, 2006
- [4] 渡部亨靖, 「手戻りゼロ」で開発コスト半減に挑戦, 日経コンピュータ, 1998年09月28日号, pp. 118-134, 1998
- [5] 経済産業省, 2005年版組み込みソフトウェア産業実態調査報告書, 2005
- [6] Hanskins, C., Forsberg, K. and Kruger M., Systems Engineering Handbook, ver 3.1, August 2007, INCOSE-TP-2003-002-03.1, 2007
- [7] 大山繁樹, 経営者・IT責任者1000人に聞く「IT化の悩み」 人材不足と資金難が深刻, 日経アドバンテージ, 2004年4月号, pp. 43-61, 2004
- [8] Alexander, F. Ian., 10 small steps to Better Requirements, IEEE software, Mar/Apr 2006
- [9] Ishii, K. Course Materials, Design for Manufacturability (ME317) .Stanford University. USA, 2003.
- [10] Chen, P., The Entity-Relationship Model--Toward a Unified View of Data, ACM Transactions on Database Systems, Vol. 1, No. 1, March 1976, pp. 9-13, 1976
- [11] 京屋祐二, 野口国雄, 中野隆司, 東芝レビュー Vol.60 No.1, pp.1-36, 2005
- [12] Alexander, F. Ian., A taxonomy of Stakeholder, Human Roles in System Development, International Journal of Technology and Homan Interaction, Vol1, 1, pp.23-59, 2005

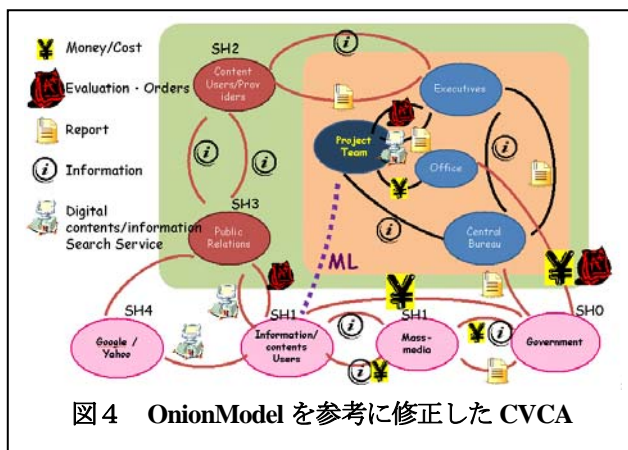


図4 OnionModel を参考に修正した CVCA