

システム運用知識抽出法の提案

山本 修一郎

名古屋大学 情報連携統括本部 情報戦略室

愛知県名古屋市千種区不老町

A Proposal on System Operation Knowledge Elicitation Method

Shuichiro YAMAMOTO

Strategy Office,
Information and Communication Headquarters
Nagoya University

Furo-cho, Chikusa-ku, Nagoya Aichi Japan

概要

情報システムの運用は日常的に実施されているにもかかわらず、情報システム開発に較べると、運用知識の実体が必ずしも明確ではなかった。本稿では、口頭で伝達されることが多く、明示的に文書化されにくいシステム運用知識を活動理論に基づいてカード形式で抽出するモデルと方法を提案し、具体例に基づいて提案した知識抽出モデルの妥当性について考察する。

Abstract

Although information system operations are now executing on every day, operational knowledge on information systems is rather unclear than those of system development. In this paper, an elicitation method for operational knowledge is described based on the inter actor communication model. Especially, the knowledge is described with the cards those items are defined by Activity Theory and actor situation model. The validity of the proposed method is also discussed by applying to extract operation procedures of an actual system.

1 はじめに

SECのプロセス共有化WGでは、「運用に入る前の要求品質の確保」について議論しており、「開発側は保守／運用する立場でも考えるべきだ」「運用／保守の見える化が弱い」と指摘している[1]。また、このようなテーマは「奥が深く、まだまだ検討を要する」としている。

また、筆者が運用サービス企業担当者に実施したヒヤリングの結果、現場の運用業務を整理する上で、「運用手順と生産物との関係をどのように対応付ければいいのかよく分からない」という問題や、「経費を投じて作成した運用手順が実際の運用とずれてしまってどうしたらいいか困っている」という問題が明らかとなった。この結果を整理すると、次のようになる。

理すると、次のようになる。

【問題1】 運用手順を記述する方法が明確でない

【問題2】 運用手順と生産物とを対応付けて記述できない

【問題3】 運用手順の変更管理が困難である

これらの問題の原因は、運用活動の主体と対象には状況に応じた相互作用関係があるにもかかわらず、この相互作用関係が明確に定義されていないことと、そのために運用上の変更箇所を特定できず、変更管理を適切に実施できないことにある。

このため、本稿では、運用活動の主体と運用対象をそれらの状況とともに明確化することにより、運用手順の可視化と変更管理を容易化する運用手順の記述方法を提案する。また、本手法の初期評

価結果についても報告する。最後に、以上の結果に基づいてシステム運用知識を抽出する上での課題と今後の展望について述べる。

2 関連研究

2.1 アクタ関係モデル

アクタ関係モデル[2]では、2次元行列を用いてアクタ間の関係を分析する。アクタ関係行列は、代表的なゴール指向要求工学手法である i*フレームワークの SD モデル[10]に相互変換できる。

2.2 アクタ相互作用モデル

アクタ相互作用モデルでは、ソフトウェアの機能要求を、「あるアクタがある状況の下で、何かのイベントに対して、指定された動作をソフトウェアが実行することにより、期待する結果をアクタにもたらす」ことだと考える[3]。機能要求をアクタ、事前状況、イベント、入力、動作、出力、応答、事後状況というように構成要素に分類して理解しておくことで、これらの要素の内容を明確化できるだけでなく、相互関係を分析することができるようになり、それに基づいて機能要求の完全性を確認できることが分かる。すなわち、アクタ、事前状況、イベント、入力、動作、出力、応答、事後状況について必要な要素とその組合せがすべて抽出されていることにより、要求が完全だと確認できることになる。

2.3 ユースケース

ユースケースでは、ある目的に従ってシステムやサブシステムなどとアクタ間で交換されるメッセージ系列の集合を記述する[4]。

2.4 シナリオ

シナリオではシステムがどのように利用されるかを説明するために、シナリオを開始するときの事前状態、終了したときの事後状態、通常のイベントの流れと例外時の流れなどを記述する[4]。

2.5 CATWOE

CATWOE では、システムのミッションを顧客 (C) 運用者 (A) あるべき姿への変換 (T) 世界観 (W) 所有者 (O) 環境制約 (E) によって定義する[5][10]。

2.6 活動理論

Leont'ev は、人間活動を、活動、行動、動作からなる 3 階層モデルとして体系化した[6]。活動 (activity)は複数の意識的な行動 (action) に分解される。行動はさらに条件に応じて無意識に実行できる動作(operation)に分解される。活動を方法づけるのは活動対象への動機である。行動を方向づけるのは目的である。動作を方向づけるのは外部からの条件である。また Engeström [7]は、活動システムを、活動の主体、活動の対象、活動の帰結、活動を支援する人工物、活動の主体とともに協働する他の参加者からなる共同体、活動に関する規則、共同体の構成員の役割分担から構成する活動システム図を提案した。

2.7 ATRE

活動理論に基づいて、情報システムの人的コンテキストを形成する「社会的な特性」に関する知識リポジトリを構築することによって、要求抽出を支援するフレームワークが ATRE (Activity Theory for Requirements Engineering)である[8]。ATRE では社会科学専門家、顧客、要求工学専門家からなるチーム・コミュニケーションに基づいて要求抽出を支援する。なお ATRE の知識リポジトリでは、活動理論の構成要素を記述するために定義された UML プロファイルである UML-AT を用いている。

2.8 CORE

英国国防省のための要求分析手法として開発された手法が CORE(Controlled Requirements Expression)である[9]。CORE ではシステムを階層的に構成するビューポイントに従って、①入力を発信するビューポイント、②入力、③動作、④出力、⑤出力を受信するビューポイントを記述する。CORE ではビューポイントにより構成要素間の入出力関係を記述できるので、複雑なシステムトランザクションを分析することができる。

3 アクタ連携モデル

アクタ連携モデル (Inter Actor Model, IAM) では、3つの運用問題を解決するために、活動理論

を考慮してアクタ状況モデルに、規則、関係者、役割分担を拡張した。すなわち、運用についてのアクタ相互作用モデルでは、①運用主体、②事前状況、③運用対象、④事後状況、⑤契機（イベント）、⑥応答（レスポンス）、⑦運用手順、⑧入力、⑨出力、⑩運用規則、⑪関係者、⑫役割分担に基づいて運用知識を記述する。

この結果、IAM では、運用主体と生産物としての運用対象を運用手順と対応付けて管理できる。また運用手順の内容を、運用契機、応答、入出力、運用規則、運用関係者、役割分担まで含めて記述できるようにしたことで、従来は曖昧になりがちだった運用規則や役割分担を明確化できる。

なお、本稿では運用知識抽出に限定して議論するがアクタ相互作用モデルは運用要求だけではなく、一般的にシステム要求を記述できる。

(1) 運用主体

運用手順を実施するアクタが運用主体である。運用者が運用主体である。

(2) 事前状況

運用手順が開始される前に成立していなくてはならない条件を事前状況として記述する。

(3) 運用対象

運用主体が実施する運用手順に従って運用されるシステムとしてのアクタが運用対象である。

(4) 事後状況

運用手順が完了した後で成立していなくてはならない条件を事後状況として記述する。

(5) 契機（イベント）

運用主体が運用を実施する契機を記述する。契機の記述では、複数の契機をまとめて書かないようにそれ以上分解できない単位で記述することに注意する。

(6) 応答（レスポンス）

運用主体が運用手順を実施した結果として生成すべき報告やメッセージを記述する。

(7) 運用手順

運用対象に対して実施すべき運用手順の内容を記述する。

(8) 入力

運用手順を実施する際に必要となる入力情報を記述する。

(9) 出力

運用手順を実施した結果、運用対象から出力される出力情報を記述する。

(10) 運用規則

運用手順の前提となる規則を記述する。

(11) 関係者

運用手順に関連する関係者を記述する。

(12) 役割分担

運用手順に関連する関係者間の役割分担を明確に記述する。

IAM に基づいて運用知識を簡潔に記述するために考案した運用活動票の例を表 1 に示す。

表 1 運用活動票

要求ID					
主体	契機	運用手順		応答	対象
事前状況		入力	出力		事後状況
運用規則		関係者		役割分担	

4 運用知識抽出法

運用活動票を用いて現行システムの運用手順を抽出する手順を以下に示す。

【手順1】現行の運用活動を列挙して運用活動リストを作成する

【手順2】運用活動リストに挙げられた運用活動ごとに、運用活動票を作成する。

【手順3】運用活動票の内容について妥当性を確認する。

現行システムの運用では、曖昧な点や不明点が多い可能性があるため、運用活動票の各項目をすべて記述できない場合が想定される。このような場合には、無理に各項目を埋める努力を必要はない。むしろ、現行システムの運用上の問題点が明確にできたことを評価すべきである。もし運用活動票の作成者が、現行システムの運用では存在しない内容を無理に憶測で記述してしまうと混乱を生じるので、こうしたことが起きないように留意することが重要である。

5 適用評価

5.1 適用対象

サービス提供者が外部に運用を委託しているあるシステムについて、委託先の運用担当者に依頼して運用活動票を作成した。今回の適用に先だって、このシステムの運用手順数を筆者が聞いたところ、運用担当者から明確な回答を得ることができなかった。この事実は、活動理論の紹介で述べたように、「システム運用行動が無意識に実行できる動作(operation)に」なっていて、聞かれても答えられない状況だったことを示している。また、このシステムについての運用手順が明文化されていれば、それを参照すれば運用手順数を答えることもできたはずだが、そういう文書もなかったということになる。

5.2 適用結果

まず運用活動票の記述内容の意識を合わせるため、数件の運用手順を選んで運用活動票を1週間

程度で作成してもらい、それを筆者と運用担当者でレビューすることで運用活動の記述内容について意識のずれがないことを確認した。その後、2週間で50件の運用活動票が完成し、現行システムの運用内容を確認した。運用内容の確認結果は次のようになった。

- 1) 運用主体 訂正なし
- 2) 事前状況 記入もれ5件
- 3) 運用対象 記述予定1件, 不明3件
- 4) 事後状況 記入もれ2件
- 5) 契機 訂正なし
- 6) 応答 不明2件
- 7) 運用手順 不明1件
- 8) 入力 記入もれ2件
- 9) 出力 不明4件
- 10) 運用規則 曖昧8件, 不明18件
- 11) 関係者 不明3件
- 12) 役割分担 不明2件

このうち、曖昧項目の内容は、システム監視周期などの具体的な設定値であり、明確化できるものである。記入もれについては、確認時に指摘することでその都度訂正した。これに対して不明点については、詳細な調査の必要性が明確になった。

このように、運用手順の個数と不明点が明確化でき、運用手順の管理が可能になった。運用担当者からも「システムの運用現場ですぐ使えるのでシステム運用知識の抽出が容易化できた」との好評価を得た。

5.3 留意点

運用担当者による試行評価では、以下の2点について運用活動票を記述する上での分かりにくさが判明した。

(1) イベントの分解

たとえば、「システムの起動・終了」というように、運用担当者は、運用上、対になる活動を1個のイベントとして、記述してきた。しかし、実際

にはシステムの起動と終了は異なる契機で発生するイベントであるとして、2個の運用活動票に分解して記述することを指摘した。

(2) 事後状況の記述内容

運用担当者には、運用作業を実施した後に、どんなことが成立していなくてはならないかということ認識することが難しかったようである。つまり、運用作業の目的が理解されておらず、運用作業を実施することが目的化している現状を反映した結果となった。しかし、あえて事後状況を考えさせることで、運用作業の目的は何であるかを明確にできることに気付いてもらうことができた。

この2点を試行確認の打合せで説明することで、運用担当者らは問題なく現行システムの運用手順を記述できた。

重要な点は、現行システムの運用活動を記述した結果、運用活動票の項目が未記入となる不明点が多数抽出されたことである。今後、これらの不明点を解決し、現行システム運用上の不明点を明確化していく必要がある。逆にいえば、運用活動票は、運用上の不明点の抽出に大いに役立ったことになる。

6 議論

これまでのシステム移行では、現行システムの Asis モデルを分析し、次いであるべきシステムの Tobe モデルを作成することが前提だった。しかし、実際には、Asis モデルと Tobe モデルの関係が不明確で、異なる2つのモデルが断絶し易いという問題がある。また Asis モデルをせっかく作成しても、Tobe モデルに基づいて次期システムを開発することから、Asis モデルが無駄になるという問題もあった。また、Asis モデルから Tobe モデルへどのように具体化すればいいかが明確ではないという問題もあった。

本稿では、アクタ相互作用モデルによって現行システムの不明点や欠陥を発見し、それを解消することで Tobe モデルを作成できる可能性があることを明らかにした。

このことから、図1に示すようなアクタ相互作用モデルに基づくシステム移行モデル作成プロセスを構築できることが分かる。このプロセスでは、①現行の運用モデルの抽出、②運用モデルの洗練、③Tobe モデルからの新規モデルの抽出、④現行の運用モデルの改善、⑤次期モデルの生成という5つの活動を具体化している。

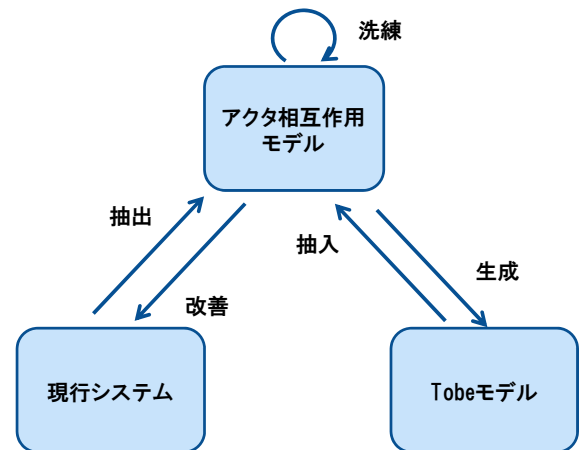


図1 システム移行モデル作成プロセス

このような循環型の移行モデル作成プロセスを用いることで、システムの運用モデルを、アクタ相互作用モデルによって常に最新化できることから、継続してシステム移行を実施する場合にも、これまでのように、毎回 Asis モデルの作成と Tobe モデルの作成を繰り返す必要がなくなることも重要である。

また、アクタ相互作用モデルを現行システムに対する Asis モデルと考えることもできるが、現行システムに対して Asis モデルを作成せずに、現行システムについてのアクタ相互作用モデルを Tobe モデルの種となる初期モデルと考えて洗練することで Tobe モデルだけを作成すると考えることもできる。

運用活動票では、運用活動を契機と応答に着目して運用手順を記述しているため、運用手順の全体的な流れを理解するためには、運用活動票の事前状況と事後状況の依存関係を明らかにする必要がある。このためには、運用活動票を要素とする運用フロー図を作成すればよい。通常は運用

フロー図が用意されていることが多いと思われるが、運用フロー図の問題点は、運用活動について、運用活動票で明らかにしたようなアクタ間の役割分担や運用規則が明確にされないことである。したがって運用フロー図だけでは運用上の欠陥が発見されたときに原因の所在を究明することが困難になると考えられる。

また、運用活動票をそれ以上分解できない運用契機ごとに作成する点について、細かくなりすぎて煩雑になるからまとめて管理すればいいのではないかという意見があるかもしれない。しかし、もしそういう管理をしていたら、運用上の欠陥が発見されたときに運用管理単位が原子化されていないため、大まかな運用内容から欠陥分析を実施することになり時間かかるだけでなく、欠陥原因を特定できない可能性もある。さらに運用上の危険個所の分析もあいまいになるので、精密な危機管理がそもそもできない。つまり運用契機単位に運用活動を分析しないのは、運用に関する危機管理を放棄するだけでなく、運用についての説明責任を果たせないことになる。

7 まとめと今後の課題

本稿ではシステム運用知識表現の一例としてアクタ相互作用モデルに基づく運用活動票を考案し、これを用いた運用知識抽出手順を提案した。運用活動票を実際のあるシステムに適用した結果から、アクタ相互作用モデルがシステム運用知識抽出を効率化できるだけでなく、運用知識の欠落を指摘できることを明らかにした。これにより運用手順の個数を計数可能にただけでなく、運用知識の可視化手段を提供できることも示した。

また、運用目標の展開と、展開された運用目標を保証することが重要である。これらについても簡便な記述方法を考案しており、別途報告する予定である。

参考文献

- [1] 新谷勝利, 長谷部武, 倉持俊之, ビジネス・プロセス改善領域, SEC journal, vol.6, No.2, Jun., pp.76-78, 2010.
- [2] Shuichiro Yamamoto, Komon Ibe, June Verner, Karl Cox, and Steven Bleistein, ACTOR RELATIONSHIP ANALYSIS FOR i* FRAMEWORK, J. Filipe and J. Cordeiro (Eds.): ICEIS 2009, LNBIP 24, pp. 491–500, 2009.
- [3] Noboru Hattori, Shuichiro Yamamoto, Tsuneo Ajisaka, and Tsuyoshi Kitani: Proposal for Requirement Validation Criteria and Method based on Actor Interaction, IEICE TRANSACTIONS on Information and Systems, Vol.E93-D, No.4, pp.679-692, 2010.
- [4] 山本修一郎, 要求定義・要求仕様書の作り方, ソフト・リサーチ・センター, 2006
- [5] チェックランド著, 根来龍之監訳, システム仕様の分析学-ソフトシステム方法論-, 共立出版, 1996
- [6] Leont'ev, A.N., Activity, Consciousness, and Personality, Prentice-Hall (1978)
- [7] Engeström, Y. (1987) Learning by expanding: An activity-theoretical approach to developmental research. Helsinki: Orienta-Konsultit. ユーリア・エンゲストローム, 山住勝弘ほか訳, 拡張による学習, 新曜社, 1999
- [8] Ruben Fuentes-Fernandez, Jorge P. Gomez-Sanz, Juan Pavon, Understanding the human context in requirements elicitation, Requirements Engineering, vol.15, No.3, pp.267-283, 2010
- [9] M. Elizabeth C. Hull, Ken Jackson, and A. Jeremy J. Dick, Requirements Engineering, Springer-Verlag, 2002.
- [10] 山本修一郎, ゴール指向によるシステム要求管理, ソフト・リサーチ・センター, 2007