

# 自己説明機能を備えた情報システムへ向けて ～情報システムの相互理解のためのゲームと物語の生成～ Toward Information Systems with Self-explanation Function --- Games and Story Generation for mutual understanding ---

飯島正<sup>†</sup>

Tadashi Iijima<sup>†</sup>

<sup>†</sup> 慶應義塾大学 理工学部

<sup>†</sup> Department of Science and Technology, Keio Univ.

## 要旨

情報システムの開発時ならびに開発後の長期における運用において、利用者のもつ変化し続ける要求と開発されたシステムの間ズレが問題を引き起こすことがある。その問題を解決し継続的に改善していくには変化し続けるズレを認識し理解することが必要である。しかし、開発者と利用者の間での立場や観点の相違が、そうしたズレを理解し共有する妨げとなっている。

要求獲得においてはシナリオに基づく手法が活用されているが、既に開発され運用されているシステムと利用者の求めているものとのズレを感覚的に把握することは難しい。本発表では、情報システムのコンポーネントに自己説明機能を組み込み、利用者の理解を促進する物語やゲームを生成する、いわば逆シナリオアプローチを提唱し、それに向けたアーキテクチャの構想を述べる。

## 1. はじめに

情報システムは、いまや社会基盤の核といってもよいくらいに社会に浸透している。その中で、利用者とするものと、実際に運用されているもののギャップが、近年は特に問題になっている。そうしたギャップの中でも、当初は、特に「違和感」としかとらえられなかったことが、後で考えると、その後の社会的に影響の大きいシステムトラブルの予兆であったと気付かされることも多い。しかし、後になって、「あの時の違和感を大事に、その原因を追及しておけばよかった。違和感を解消すべく対策を講じておけばよかった」と後悔しても後の祭りである。このことはユーザビリティに関しては、比較的的理解しやすい。たとえば違和感や使いにくさを感じるユーザインタフェースは、操作ミスを誘発することが多い。しかし、これはユーザビリティ（操作性や情報デザイン）に限らず、情報システムのレスポンス時間などの性能面にも及ぶ。これらは仕様として表現し設計実装へ反映させていくことは難しく、かつては要求獲得でも見過ごされがちなものであった。これは、機能要求ではなく主に非機能的な要求(NFR; Non-Functional Requirements)が、形式仕様として表現されにくいことに由来する。しかし、情報システムの利用者が感じ取り得る「違和感」は決して非機能面だけとは限らない。機能面に由来する「違和感」も少なくない。情報システム開発者が利用者の立場で、その違和感を実感することは重要である。

しかし、その違和感を感じるために、実システムに影響を与えたり、開発のコストが大きく上乗せされることは望ましくない。そこで、実システムそのものではなく、その構築の際に利用されるシナリオやモデル、実システムから得られる実行トレースやログを利用したアプローチが期待できる。但し、この枠組みは、実システム全体をブラックボックスとして適用するのではなく、粒度の比較的小さいコンポーネント単位で構成する。それにより、コンポーネント間の相互作用に関して、理解するのに役立つ。これは、障害発生後に、その原因究明にも役立てることができる。

従来、利用者主導、もしくは利用者とのコミュニケーションを円滑にし利用者に寄り添う形で、各種の要求を獲得・定義するシナリオに基づく方法論 (John M. Carroll による Scenario-based Design[1]や、Alan Cooper によるとされるペルソナ手法など) が提唱されてきた。シナリオに基づく要求獲得の考え方は、利用者の特徴付けをしたペルソナによる利用者指向の Web サイトのデザインにも使われている[2]。

その利用の方向性はシナリオから要求、モデル、さらに実装を生成していくというものであった。本研究では、シナリオを利用するとはいっても、その目的はシステムの理解・分析であり、どちらかという逆方向、すなわちシステムに関して観測できる情報（実行トレース/ログ）から、それを説明するシ

ナリオを生成するという方向であり、逆シナリオアプローチと呼ぶべきものである。もともと、従来よりシナリオはシステム評価にも適用されてきている（シナリオに基づくソフトウェアアーキテクチャ評価 [3]-[6]）。また、実行トレースやログだけからシナリオを構成することは極めて困難であるため、実際には、要求獲得時のシナリオ、システム設計時のモデルをベースに、実際に情報システムが出力した実行トレース/ログに対応するシナリオを生成する。

情報システムは、単にソフトウェアシステムというだけではない。情報システムと利用者とのインタフェースに加えて、利用者同士のインタフェースも重要である。たとえば、窓口に来た顧客と応対してソフトウェアを操作する場合、操作者のための利用者インタフェースをモデル化するだけでは、真に情報システムをモデル化しているとは言い難い。顧客と操作者の間で、どのように対話がなされ、そこでどのように情報が交換されるかを評価することにより、ソフトウェア開発者と要求者の間に存在した誤解や思い違いが明確化することもある。そうした人間同士の対話（相互作用）のモデル化には、発話行為理論(SPT; Speech Act Theory)に基づく LAP(Language Action Perspective)[7]-[9]をベースとする。表層的な対話の中に込められた、他者への働きかけの目標や意図を取り扱うためである。

まず、シナリオから、その対象とする情報システムと、その利用者とをよりよく説明しつつ、関係者に考えさせる物語を生成することを試みる。このために、同値分割や境界値といったテストケース生成の考え方が利用したり、入力データに対して案に仮定している意識されていない前提条件などを明らかにしていくためのストーリー展開を考える。

さらに、シナリオからさらにゲーム要素を取り入れたシミュレーションゲーム（ロールプレイングゲーム）を生成し、臨場感を持ってシステムに対する理解を促進することを試みる。教育工学において学習意欲の向上のためにゲーム（特にロールプレイングゲーム）を利用する事例は増えてきている。そうしたゲームはシリアスゲームと呼ばれ、ビジネスゲームや被災時のサバイバルといった意思決定シーンを体験させ得点を競うものから、シミュレーションで再現された対象を安全な立場で深く知るために操作してみるものまで、いろいろなタイプがある[10]-[14]。そうした様々なアプローチを、ソフトウェア工学ならびに情報システム学にも適用したい。また、従来、仕様の作成にあたっては、曖昧さを排除し正確な形式的な記述を、如何に創り出すかが大きな目標であったのに対し、本研究では、利用者や開発者にとって理解しやすい記述を創り出すことを目標とする点が特徴的である。ここで、「情報システム利用者にとって理解しやすさとはどういうものか」、「モチベーションを維持し臨場感を持って小さな違和感を拾い上げるにはどうすればよいか」といったことから追求する。

本研究が提唱する「逆シナリオアプローチ」は、ブラックボックス化されてきた情報システムを、利用者に理解しやすい形でホワイトボックス化する手法といえる。いわゆる情報システムに限らず、広くいろいろな社会システムの理解促進・デジタルデバイドの克服に役立てることができる。たとえば低年齢層や高齢層への情報システムの直観的理解と普及をも促すものとなる。

企業等において内部統制が注目されている昨今、企業監査にもあらたな手法を提供することになる。特にセキュリティ分野においては、ログや実行トレースなどの痕跡から、セキュリティ障害を追及するデジタルフォレンジックス(forensics)が注目されているが、本研究の目標の一つに、このセキュリティ障害のための説明生成があり、原因の特定については同種の障害の予防へつながる。

本稿は、以降で、逆シナリオアプローチの概略、現在、設計中のシナリオ記述言語 ISML に影響を与えている先行研究の紹介と、ISML の概要、どのような方針でゲームや物語を生成するか、について構想を述べる。

## 2. 逆シナリオアプローチ

一般にソフトウェアシステムは、ブラックボックスとなりがちである。また、情報システムは、単にソフトウェアシステムというだけではなく、それに関与する人が重要な位置を占め、流動的な側面を強めている。したがって、情報システムを構成する個々のソフトウェアコンポーネントに、ブラックボックスを破って、内部情報を外部に提示する機能を持たせることが必要である。また、人間的要素を説明するために、人間の振る舞いをモデル化する仕組みが必要である。前者に関しては、まず、ロギング機能をベースとするところから始め、いかなる情報をどのように出力すればよいかを検討し、小規模な情

報システムを構築したうえでそれを検証する。後者に関しては、状態遷移に基づくシナリオ記述言語（まず XML テキスト表現、次に図的表現を検討する）を定義し、その上に、発話行為理論に基づくコミュニケーション理論である **LAP (Language/Action Perspective)** を適用することで関与者同士の相互作用に意味づけをおこなう。そのようなビジネスプロセスを表現するワークフローをソフトウェアシステムにおけるビジネスロジック層と結び付けることで、動いているシステムの状態を説明する物語 (Narrative) を生成させることを試みる。余分な情報を省き、より自然で理解しやすい物語の生成に関して検討が必要である。

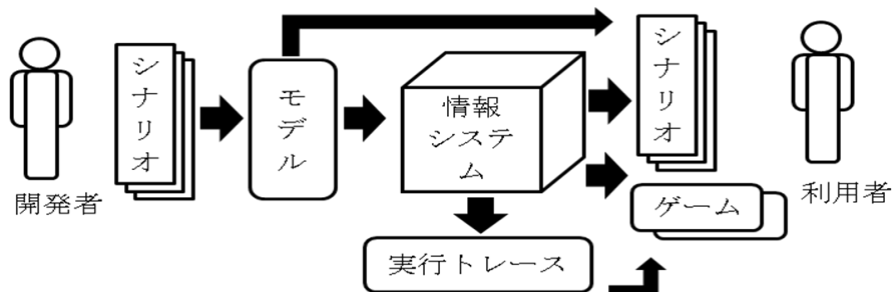


図1 逆シナリオアプローチにおけるシナリオとゲームの生成

### 3. シナリオとその記述言語

図1に示すように、本研究では開発時に与えられたシナリオをベースに、テストケース生成の手法を利用して、更なるシナリオを生成する。特にシナリオにおける登場人物（アクター）自体を強く特徴づけるキャラクタを明確にするために、開発時には、利用者をモデル化したペルソナを利用する、いわゆるペルソナシナリオ技法を採用する。こうしてつくられたシナリオを、実際に構築された情報システムとその実行トレースをもとに、調整してシナリオを生成する手法が逆シナリオ手法である。

本研究における**シナリオ記述言語(ISML; Interactive Scenario Modeling Language)**は、現在、設計中である。関連するシナリオ記述言語としてSDML(Scenario Description Markup Language)[13][14]と、Q[15]がある。SDMLはソフトウェア開発におけるシナリオに基づく要求工学を、Qはエージェントベースドシミュレーション(ABS; Agent-based Simulation)におけるエージェント間対話を指向している。また、振る舞い言語 ABL(A Behavior Language)[16]は、人格や感情、社会的相互作用などを取り入れた、いわゆる Believable Agent のためのものである。対話的に動的に物語生成(Storytelling)をおこなうには、状態遷移指向ではなく、タスク指向の構造を使う方が便利である。それには、HTN(Hierarchical Task Network)が有用である。ISMLは、**タスク指向の構造と、状態遷移指向の構造を、粒度に応じて交互に取り入れる**ことを考えている。

ISMLは、これらの性質を受け継ぐとともに、**Role 概念**を強く取り入れている。Scenarioは、Goal, Precondition, Casting, Pre-condition, Post-condition, Scenesから構成される。ScenesはSceneの集まりとなるが、順序関係を定義した状態遷移モデルとなっている。Castingでは、RoleとActorが対応付けられ(多対多関係)、Casting自体も状態背にモデルとなっており、Actorが対応付けられるRoleも変化する。

### 4. ゲームと物語の生成

ゲーム(いわゆるロールプレイングゲーム)は、ペルソナをもとにロールを構成し、ゲーム参加者を体験しながら、情報システムに対する理解を深めていく。ここで扱うゲームは物語性の強い**ロールプレイングゲーム**を基盤とするが、その上に、各コンポーネントに対して前提となっている条件や、それが本当に適切なものか、を考えさせる設問を含める。すなわち、ロールプレイングゲームに**クイズ形式のゲーム**も組み合わせた構造を持つ。物語の生成(storytelling)においては、**実システムの実行トレースやログから得られた情報**も活用する。これによって、**要求者の要求と実システムとのズレ(差異)**も可視化していくことを目指す。

### 5. おわりに

本稿では、自己説明機能を備えた情報システムの基本的な構想を述べた。ゲームや物語の形でサブシステム(コ

ンポーネント)は、それぞれ自身について説明する機能を持つ。また、情報システムでは、単なるソフトウェアシステムというだけでなく、関与する人が重要な位置を占める。そこで、人と人の直接の対話や、ソフトウェアを介した人と人の対話もモデル化し可視化の対象とする。この構想を具体化し、情報システム開発・運用にかかわる様々な人々の相互理解のためにゲームや物語を生成することは、今後の課題である。

## 参考文献

- [1] John M. Carroll: "Making Use -- Scenario-based Design of Human-Computer Interactions," The MIT Press, 2000;  
(訳書) 翻訳: 郷健太郎, "シナリオに基づく設計 -- ソフトウェア開発プロジェクト成功の秘訣," 共立出版, 2003.
- [2] Kazman, Abowd, Bass, and Clements: " Scenario-based analysis of software architecture," IEEE Software, Vol.13 , No.6, pp.47-55, 1996.
- [3] Kazman, Bass, Abowd, and Webb: "SAAM: a method for analyzing the properties of software architectures," in Proc. of International Conference on Software Engineering (ICSE), pp.81--90, 1994.
- [4] Kazman, Klein, Barbacci, Longstaff, Lipson, and Carriere: "The architecture tradeoff analysis method," in Proc of International Conference on Engineering of Complex Computer (ICECCS), pp. 68--78, 1998.
- [5] Buchgeher, Weinreich: "An Approach for Combining Model-Based and Scenario-Based Software Architecture Analysis," in Proc. of 2010 Fifth International Conference on Software Engineering Advances (ICSEA), pp. 141--148, 2010.
- [6] Steve Mulder, and Ziv Yaar: "The User Is Always Right: A Practical Guide to Creating and Using Personas for the Web," New Riders Press, 2006; (訳書) 翻訳: 奥泉直子, 監訳: 佐藤伸哉, "Web サイト設計のためのペルソナ手法の教科書," 毎日コミュニケーションズ, 2008.
- [7] Terry Winograd : "A language/action perspective on the design of cooperative work," Human-Computer Interaction , Vol.3, No.1, pp.3--30 , L. Erlbaum Associates Inc., 1987.
- [8] Medina-Mora , Winograd, Flores, and Flores: "The action workflow approach to workflow management technology," in Proc. of ACM conference on Computer-supported cooperative work(CSCW), pp. 281 --288, 1992.
- [9] Winograd,, and Flores: "Understanding Computers and Cognition: A New Foundation for Design," Ablex, 1986; (訳書) 翻訳: 平賀謙, "コンピュータと認知を理解する---人工知能の限界と新しい設計理念---," 産業図書, 1989.
- [10] 藤本徹: "シリアスゲーム -- 教育・社会に役立つデジタルゲーム," 東京電機大学出版局, 2007.
- [11] Marc Prensky: "Digital Game-Based Learning," Paragon House, 2007;  
(訳書) 翻訳: 藤本徹, "デジタルゲーム学習," 東京電機大学出版局, 2009.
- [12] David Michael, and Sande Chen: "Serious Games -- Games that Educate, Train, and Inform," Course Technology, Cengage Learning, 2006.
- [13] Bryan Bergeron: "Developing Serious Games," Charles River Media, 2006.
- [14] M. Zyda: " From visual simulation to virtual reality to games," IEEE Computer, pp.25--32, Sept.,2005.
- [12] Yessad, Labat, and Kermorvant: "SeGAE: A Serious Game Authoring Environment," in Proc. of International Conference on Advanced Learning Technologies (ICALT), pp.538--540, IEEE, 2010.
- [13] Giuseppe Della Penna, , Benedetto Intrigila, Anna Rita Laurenzi and Sergio Orefice: "An XML Definition Language to Support Scenario-Based Requirements Engineering, " International Journal of Software Engineering & Knowledge Engineering, Vol. 13 No.3, pp.237-256, World Scientific Publishing, 2003.
- [14] Giuseppe Della Penna, , Benedetto Intrigila, Anna Rita Laurenzi and Sergio Orefice: " An XML Environment for Scenario based Requirements Engineering," Journal of Systems and Software, Vol.79, No.3, pp.379--403, Elsevier, 2006.
- [15] Toru Ishida; " Q: A Scenario Description Language for Interactive Agents," IEEE Computer, pp.42--47, Nov., 2002.
- [16] Mateas, and Stern: "A Behavior Language for Story-Based Believable Agents," IEEE Intelligent Systems, Vol. 17 , No. 4, pp. 39--47, Jul./Aug. , 2002.
- [17] Cavazza, Charles, and Mead: " Character-Based Interactive Storytelling," IEEE Intelligent Systems, Vol. 17 , No. 4, pp.17-24, Jul./Aug. , 2002.