

Web サービス連携に基づくゲーミングシミュレーション 作成フレームワークの提案

鈴木 格† 菱山 玲子†

† 早稲田大学理工学術院 創造理工学研究科経営システム工学専攻

1 はじめに

近年、複雑な社会問題に関する議論の必要性が叫ばれている。そうした問題を議論するための未来の言語 [1] としてゲーミング・シミュレーション (以下, GS) が注目されている。一方, Web 上のサービス連携が活発になり, 連携による新たなサービスが作られている。

そこで本研究では, Web サービスを利用し, また Web サービスとして利用される GS を作成するためのフレームワークを提案したい。

2 関連研究

GS を定義するために必要な要素として, Greenblat [2] は, シナリオ, 役割, 外生要因, 手続とルール, 視覚イメージとシンボル, 計算システムの6つが必要であるとしている。

GS のシステムを構築するための方法として Shirai ら [3] のフレームワークなど, ルール記述のためのドメイン特化言語を用いてドメイン専門家によるルール記述を行い, それに基づいて GS を実装できるフレームワークが提案されている。

3 提案

本研究では, Web サービス連携に基づく GS を作成するためのフレームワークを提案する。具体的には, まず GS のルールを記述するためのドメイン特化言語 (以下, 提案言語) を定義し, 次に, 提案言語で記述されたルールに基づく GS サービスを作成し提供する Web サービス (以下, 提案サービス) を実装する。最後に評価として, 実際の GS を実装し, Web サービス連携に基づく GS の実装例を示す。

4 提案言語

提案言語は Common Lisp を拡張したものであり, この記述はボードゲームのメタファによる GS のモデル化に基づいて GS のルールを記述する。

具体的には次のようになる。まず, GS の状況は, 「ゲーム盤」上にあるどの「フィールド」あるいは「スタック」に, どの「トークン」がいくつ乗っているかによって表現される。ここで言う「トークン」とは, 現実の事物を象徴する用具のことを指し, 「フィールド」と「スタック」とはゲーム盤上の枠である。フィールドの上にあるトークンは順序を考慮しないが, スタックの上にあるトークンは順序づけられる。

参加者 (プレイヤー) が進行役 (GM) に対して意思決定を伝えるには, 「カード」を使う。プレイヤーは所定のカードに意思決定の詳細を書いて GM に提出する。GM はカードを受け取ってゲーム盤を操作する。また, GM はカードによる意思決定以外に, プレイヤーの「発言」に対するリアクションとしてゲーム盤を操作することがありうる。これは参加者に知らされることもある。知らされないこともある。これらを表したのが図1である。

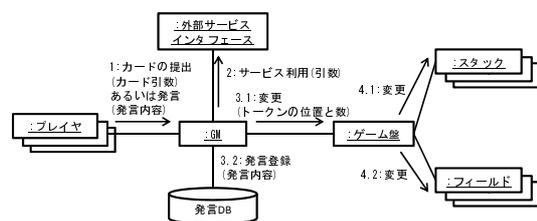


図1: モデル各要素のコラボレーション図

ルール記述は, ゲーム名, ゲームの説明, プレイヤーの定義, トークンの種類の定義, フィールドの名前, スタックの名前, カードの処理, 発言の処理, および初期状態のトークン配置を記述する。

また, 外部の Web サービスへの呼び出しを関数呼び出しのように記述することもできる。

A proposal of a framework for gaming simulations based on web service collaboration

Itaru SUZUKI†, Reiko HISHIYAMA†

†Department of Industrial and Management System Engineering, Graduate School of Creative Science and Engineering, Waseda University

itaru-suzuki@asagi.waseda.jp, reiko@waseda.jp

5 提案サービス

提案言語にもとづいて動作するGSをWebサービスとして実装した。提案サービスはAPIを通して現状の取得、カードの提出、発言を行うことができる。結果の成否はHTTPステータスコードで表され、その他の情報はS式などの機械可読な形式で返される。

このAPIを利用し、結果を自由な視覚イメージとシンボル表現するサービス、ないしクライアントと組み合わせることで、GSを構築できる。

また、APIを通してGSの状況を取得し、状況判断を行い、APIを通して行動するプレイヤーエージェントを作成することも可能である。

6 評価

提案言語によって実際のGSを記述した。取り上げるのは、Yamaguchi[4]らによる電力問題ゲームである。これは社会的ジレンマ状況を表現したGSであり、4人のプレイヤーはそれぞれ都市の指導者として街を経済的に発展させるとともに社会貢献をせねばならない。経済を発展させるには電力が必要であるが、獲得できる電力は限られており、入札によって高額を提示した街により多く配分される。また、出資して発電所を改良しなければ、ターンの経過にしたがって環境が悪化し、ペナルティが課せられる。発電所の改良およびペナルティの支払いが社会への貢献とみなされる。

それぞれの都市と発電所をフィールドとし、お金、借金、貢献度、経済力、ペナルティ、発電所のレベル、電力、および入札を助けるトークンを用意することで、電力問題ゲームの状況を表現する事ができる。この定義をリスト1に示す。

リスト 1: 電力問題ゲームの用具

```
(define-game Yamaguchi-Game
  "Yamaguchiらによるゲーム"
  :player-list ((p11 "1人目" ja) (p12 "2人目" en)
               (p13 "3人目" ja) (p14 "4人め" en))
  :token-list
  ((money "予算") (dept "借金") (contrib "貢献度")
   (economy "経済力") (penalty "ペナルティ")
   (e-power "電力") (e-level "発電所のレベル")
   (turn-marker "ターンの経過を表すチップ")
   (phese-marker "フェーズ進行を表すチップ")
   (bit-prior-notice "入札予告のためのチップ")
   (bit-marker "入札のためのチップ"))
  :field-list
  ((p11 "p11の街") (p12 "p12の街") (p13 "p13の街")
   (p14 "p14の街") (power-plant "発電所"))
  :stack-list ())
```

GSの初期化は、ゲーム盤の操作の列挙である。フィールドおよびスタックはsetf可能な場所として表現されるため、Common Lispの関数で操作できるが、操作によっては専用の関数が提供される。

カード定義の例をリスト2に示す。カードの定義はdefine-cardで行われ、これはdefunに似た記法を持つ。APIに対してカードの仕様を求められると、サービスは第二引数で表される型宣言に基づいて引数のチェックを行った後、第三引数以降の本体部分を実行する。

リスト 2: カードの例

```
(define-card financial-aid ((to-whom as a player)
                           (how-much as a number)
                           (by-who as a player))
  "誰かから誰かにお金を渡す"
  (if (>= (amount-of 'money by-who) how-much)
      (progn (put how-much 'money to-whom)
             (pick-up how-much 'money by-who))
      (remark :nobody "資金が足りません")))
```

さらに、このGSは多言語のGSであり、エージェントとの会話が可能であるが、このことは、プレイヤーの発言を翻訳サービスを利用して翻訳することと、APIを利用するエージェントプログラムを実装することで再現できる。

7 おわりに

提案手法により、実際にWebサービスを利用し、またWebサービスとして利用されるGSが実装できることを示した。今後の課題は、画面の記述を行うことができるフレームワークや、ルール記述からのエージェントの自動生成フレームワークの作成である。

謝辞

本研究は、科学技術融合振興財団 (FOST) 平成23年度補助金助成「多言語ゲーミング・シミュレーション作成フレームワークの開発および評価」(研究代表者: 鈴木 格) の成果によるものである。

参考文献

- [1] Richard D. Duke. ゲーミングシミュレーション 未来との対話. アスキー, 2001.
- [2] Cathy S.Greenblat. ゲーミングシミュレーション作法. 共立出版, 1994.
- [3] Hiroaki Shirai, Motonari Tanabu, et al. Game development toolkit for business people in japan. *Simulation Gaming*, Vol. 34, No. 3, pp. 437-446, September 2003.
- [4] Akihiro Yamaguchi, Keisuke Tsunoda, and Reiko Hishiyama. Player agents for langrid gaming. *Proceedings of the 29th ACM international conference on Design of communication*, pp. 51-58, 2011.