

B-005

エンドユーザ向けゲーム開発ツールの試作：  
トランプゲームを対象としたオブジェクトベースの解釈実行方式  
A game development tool for end-users:  
An object-based interpretation method for card games

米澤 剛†

中所 武司†

Tsuyoshi Yonezawa

Takeshi Chusho

あらまし ゲーム作りの需要は多く、エンドユーザ向け、非エンドユーザ向け問わず、いくつものゲーム作成ツールが存在するが、ある程度以上の規模のゲームの作成となると大量の情報を扱う必要があり、ゲーム作りの知識や技術なしでは難しい。そこで、本研究では、トランプゲームを対象に、「カード」「山札」「場」などのオブジェクト毎にレイアウトと動作を定義するだけでゲーム作りを可能とするツールを試作した。本方式では、ユーザが内部処理を意識する必要はないので、ゲーム作りの知識や技術がなくてもゲームを組み立てることができる。

## 1. はじめに

ゲーム作りの知識や技術はないが、ゲームを作ってみようという人は多く、実際にエンドユーザ向け、非エンドユーザ向け問わず、いくつものゲーム作成ツールが存在し[1]、アクションゲーム、シューティング、ロールプレイングゲームなど、その気になれば一般的なジャンルのゲームの作成は可能という状態である。しかし、それでもそういったゲームを作るためには大量の情報を扱う必要があり、ある程度以上の規模のゲームの作成となるとゲーム作りの知識や技術なしでは難しい。

そこで、本研究では他の一般的なビデオゲームに比べ扱う情報量の少ないトランプゲームに注目し、オブジェクトベースの解釈実行方式を用いることで、ゲーム作りの知識や技術を持たないユーザが扱うことのできるゲーム開発ツールの試作を行う。

本稿で開発の対象とするトランプゲームは、相手プレイヤーの動作の設定に関する複雑な設定が必要になることが考えられる複数人用のトランプゲームではなく、フリーセルやソリティア(クロンダイク)、スパイダーといった、決められたルールに従って順不同に配布されたカードを特定の場所に正しい順番に並び替えるといった、一人遊びのゲームに限定する。

## 2. トランプゲームのオブジェクトベース解釈

### 2.1. ルールの考察

本稿で開発するツールで作成可能とするトランプゲームの中から、ソリティアを例にとりルールを表1にまとめた。

ソリティアには、まず「カード」と、「山札」、「山札用の場」、「同スタート用の場」、そして「並び替え用の場」が存在していて、「山札」は“クリック時の挙動”のルールを、「山札用の場」、「同スタート用の場」、「並び替え用の場」はそれぞれ“場へのカードの置き方”と、“場からのカードの移動”のルールを持っている。そして、これらを定義することが出来れば、ソリティアを定義することは可能である。

これは他の既存のトランプゲームである、フリーセルやスパイダーの場合も同じで、これらのゲームは、『初期状態』と『ゲームクリア条件』、そして『カード』と『山札』と数種類の『場』によって出来ていると言える。

表1：ソリティア ルールまとめ

		ソリティア
初期状態	使用カード	一組(52枚)
	場札	左から1枚、2枚、3枚、4枚、5枚、6枚、7枚を裏返しで置く
山札	枚数	24枚
	クリック時の挙動	上から3枚を専用の場にあるカードに重ねて表で置く
山札用の場	場へのカードの置き方	他の場からのこの場への移動は出来ない
	場からのカードの移動	一番上のカード一枚のみ移動可能
同スタート用の場	場へのカードの置き方	同じスタート、番号が昇順の場合のみ重ねておくことが出来る
	場からのカードの移動	一番上のカードのみ移動可能
並び替え用の場	場へのカードの置き方	赤と黒の交互、番号が降順の場合のみ並べることが出来る
	場からのカードの移動	赤と黒が交互に並び番号が降順であれば、一組として移動することが出来る
ゲームのクリア条件		特定の場にそれぞれ一組ずつ、計4組の同スタートのカードをそろえる

### 2.2. オブジェクトの定義

2.1 を基にしてカードゲームにおける、一般的なオブジェクトの定義を行う。

#### ①カード

ゲームで用いられるトランプカードのオブジェクト。

通常、トランプで扱われる52種類のカードを扱い、それぞれの種類を識別することが可能である。

なお、トランプゲームにおいてジョーカーはゲームごとに様々な異なる役割を持っていて、ゲームによって様々な扱いをされるため、ジョーカーを含むトランプゲームを開発する場合、ゲーム開発者はゲームごとにジョーカーの複雑なルールを設定しなくてはならず、それを設定するための開発ツールの実装は困難である。そのため、本稿ではジョーカーは扱わないものとする。

†明治大学大学院理工学研究科基礎理工学専攻情報科学系

ソフトウェア工学研究室

## ②デッキ

山札のオブジェクト。

ゲーム開始時、ゲームで使用する全てのカードオブジェクトを所持していて、「初期配布ルール」に従ってそれぞれのスポット（後述）にカードを配布する。

デッキをクリックしたときに行われる配布のルールである「通常配布ルール」と、デッキをクリックしたときにデッキにカードが存在しない場合に行われる動作の内容を記した「無所持ルール」を有している。

図1の上段の左から2番目がこれに対応する。

## ③スポット

カードを置く事が出来、カードの移動に用いる場のオブジェクト。

スポットに別のスポットから動かしたカードを置く時に、カードを置く事が可能かどうかの判定を行う「プットルール」と、スポットからカードをつかんで動かす時に、カードを動かすことが可能かどうかの判定を行う「テイクルール」、そしてゲームの終了判定を行う「クリアルール」を有している。

ソリティアの場合、表1の山札用の場、同スート用の場、並び替え用の場のそれぞれのインスタンスがこれに対応する。図1では、上段のデッキ以外の5か所と、下段の7か所がスポットである。

## ④フィールド

同じルールを持った1つ以上のスポットの集合。

フィールドに存在する全てのスポットは共有するルールを持つ。

ソリティアの場合、山札用の場の1つ、同スート用の場の4つ、並び替え用の場の7つはそれぞれ同じルールを持っているため、これらのそれぞれの集合がフィールドに対応する。図1では、上段左端のスポット1個、上段右端のスポット4個組、下段のスポット7個組の3種類がフィールドである。



図1：オブジェクト説明図（ソリティア）

波線：デッキ， 実線：スポット， 破線：フィールド

## 3. トランプゲーム開発ツールの制作

### 3.1. ツールの全体像

トランプゲーム開発ツールは、図2に示すように大きく分け3つの部分，“ゲーム開発部分”，“ゲームプレイ部分”，“作成したゲームのデータ（以下‘ゲームデータ’と記述）”の3つの部分から構成される。ゲーム開発部分でユーザはゲームを開発し、ユーザが入力したゲームの仕様からゲームデータを作成し、そのファイルを出力する。ゲームプレイ部分では、ユーザのゲーム起動指示に基づき、ゲームデータのファイルを読み込み、トランプゲームを生成、動作する。

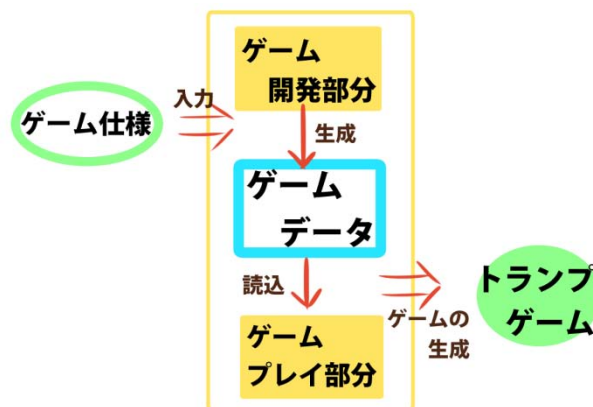


図2：ツールのシステム構成図

### 3.2. オブジェクトの実装

2.2 で記述した各オブジェクトの実装定義を示し、開発ツールとして実装するためのデータの形式を考え、ゲームデータへの記述内容を記す。

#### ・カード

##### 所持データ

識別番号（1～52），表裏判定

#### ・デッキ

##### 所持データ

座標，所持カードリスト

##### ー初期配布ルール

全てのスポットにおける，配布する枚数，配布する全てのカードの表裏情報。

ゲームデータにはこれらの情報を記述する。

##### ー通常配布ルール

初期配布ルールと同様

##### ー無所持ルール

特定のスポットにあるカードを全てデッキに回収する。この回収の対象となるスポットの情報。

ゲームデータは全てのスポットを対象とする真偽を持ち，真であるスポットを回収の対象とする。

#### ・スポット

##### 所持データ

座標，所持カードリスト

##### ープットルール

カード2枚，あるいはカードとカードの無い状態（以下“空”と記述）の関係を記憶する。

スポットの一番上のカード（スポットにカードが存在し

ない場合は空状態)と置きたいカードの東の一番下のカードの関係, 更に, 置きたいカードの東のすべてのカードにおける連なるカードの関係が真である場合のみスポットにカードを置くことが可能となる。

例えば特定のスポット A に下から a, b の順にカードが置かれている時に, 別のスポット B から動かした下から c, d, e の順のカードを置こうとする場合, スポット A における b と c, c と d, d と e の関係がすべて真である場合のみスポット A に c, d, e のカードを置くことが可能となる。

ゲームデータは, ジョーカーを除く全種類のカード 2 枚の関係と, カードと空の関係を保うため,  $(5 \times 2 \times 5 \times 2) + (5 \times 2 \times 1)$  の表に真偽を記述する。

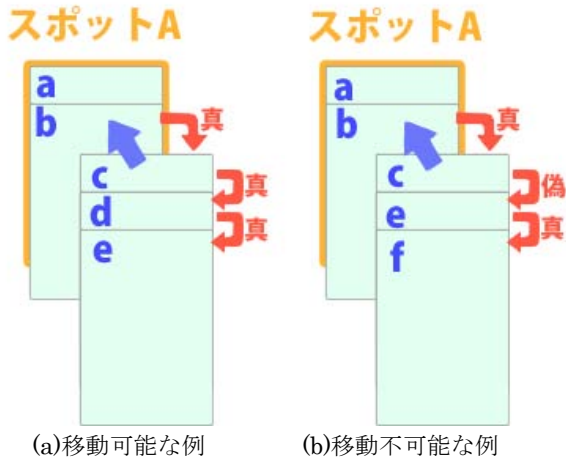


図 3 : プットルール説明図

一テイクルール

カード 2 枚, あるいはカードと空の関係を記憶する。

掴みたいカードより上に存在しているカードすべてにおける連なるカードの関係が真であり, かつ一番上に存在しているカードと空の関係が真である場合のみ移動可能となる。

例えば特定のスポット A に下から a, b, c, d, e の順にカードが置かれている時に, c のカードをつかんで移動しようとする場合, スポット A における c と d, d と e, e と空の関係がすべて真である場合のみスポット A から c, d, e のカードが移動可能となる。

ゲームデータへの記述はプットルールと同様。

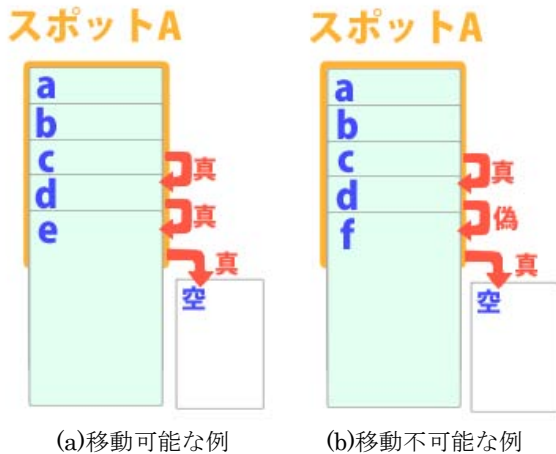


図 4 : テイクルール説明図

ークリアルール

スポットに存在するカードの枚数によって真偽を返すルールであるため, 比較するカードの枚数と, “=”や“<”といった関係演算子を記憶する。

ゲームに存在する全てのスポットに設定されているクリアルールが真となった場合にゲームがクリアとなる。

ゲームデータにはフィールド毎に比較するカードの枚数と, 関係演算子を記述する。

・フィールド

所持データ

所持スポットリスト

3.3. ゲームデータ形式

前節で述べたオブジェクトの実装定義内容は, オフィスソフトの Excel の保存形式である “.xlsx” 形式を用いてゲームデータとして保存する。

これは, ゲームデータにはルールのテーブルなどの表表現を多く含むため, ゲームデータの形式も表表現が可能なのが好ましいためである。また, Excel データは, 文字列, 数値, 真偽, といった型を用いることが可能であり, 更に, アプリケーション開発環境である Java における Excel データを扱うライブラリ [2] の機能も充実しているので実装を容易にすることが可能である。

3.4. ゲーム定義の設定方法

①ゲームレイアウト

デッキの座標の設定, 使用する全てのスポットの座標の設定を行う。

ソリティアの場合, それぞれのオブジェクトを図 5 の様に配置する。

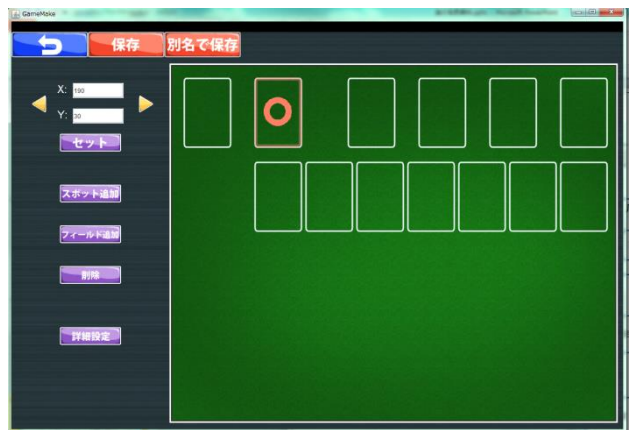


図 5 : ソリティア レイアウト設定画面

②使用カード設定

使用するカード 52 種類それぞれに使用する枚数を設定する。ソリティアの場合, すべての種類のカードを一枚ずつ使用するため, カード 52 種類それぞれに “1” を設定する。

③デッキ初期動作設定

全てのスポットに対し, ゲーム開始時にデッキから配布されるカードの枚数を設定する。また, 同時に配布されるカードの表裏も設定する。

ソリティアの場合、初期動作は場に左から1枚、2枚、…、7枚と言うようにカードを裏で配布し、それぞれの場の一番上のカードのみを表にする。その為、山札用の場、同スート用の場は全て設定なし、並び替え用の場へは左から表、裏表、…、裏裏裏裏裏裏表、の様に設定する。

#### ④デッキ通常動作設定

設定内容は③と同じである。

ソリティアの場合、通常動作は山札用の場にカードを3枚表で配布する。その為、山札用の場に、表表表、の様に設定し、山札用の場以外のスポットへの設定は行わない。

#### ⑤デッキカード無所持動作設定

全てのスポットに対し、真偽を設定し、この真偽が真である全てのスポットをカード無所持動作の対象とする。

ソリティアの場合、山札にカードが無いときに山札をクリックすると、山札用の場にあるカードを回収し、山札とする。その為、山札用の場に真を設定し、それ以外のスポットへは偽を設定する。

#### ⑥プットルール設定

ジョーカーを除く4種類のスート、13種類の数のカード、2枚の全ての組み合わせと、すべての種類のカードと空の関係を真偽で設定する。

カード同士の関係を設定する場合、4[スート]×4[スート]の表と、13[数]×13[数]の表が用意されていて、先に4×4の表でどのスートの関係を設定するかを選択した後に、13×13の表で数字の関係の真偽を設定する。

例えば、ハートの5に対するスペード6の関係を真に設定する場合、先に4×4の表でハートに対するスペードの関係を示す箇所を選択し、13×13の表で5に対する6の関係を示す箇所を真に設定すればよい。

ソリティアの場合、山札用の場へのカードの移動は不可能、同スート用の場へのカードの移動は同じスートであり、昇順で連続するカードを一枚のみ可能、並び替え用の場へのカードの移動は赤黒が交互に並び降順で連続するカードを複数枚可能である。

この中で、並び替え用の場への設定を例にとって考える。赤黒が交互に並ぶように設定するため、ハートに対するスペード、ハートに対するクラブ、ダイヤに対するスペード、ダイヤに対するクラブ、スペードに対するハート、スペードに対するダイヤ、クラブに対するハート、クラブに対するダイヤの計8つの組み合わせに対して、降順で連続に並ぶように、13に対する12、12に対する11、…、2に対する1の真偽を真に設定する。また、並び替え用の場の空に対するカードの置き方は、スートを問わず13のカードのみを置く事が出来るため、空に対するハート、空に対するスペード、空に対するダイヤ、空に対するクラブの計4つの組み合わせに対して、13の真偽を真に設定する。

図6はソリティアにおけるハートに対するスペードのプットルールの設定画面である。

#### ⑦テイクルール設定

設定内容は⑥と同様である。

⑥と同様にソリティアの並び替え用の場への設定を例にとって考える。ソリティアの並び替え用の場からのカードの移動は、プットルールと同様に赤黒が交互に並び降順

で連続するカードを複数枚可能である。その為、カードに対するカードの関係の設定方法はプットルールの場合と同様である。ただし、テイクルールでは、プットルールで設定した空に対するカードの関係の代わりに、カードに対する空の関係を設定する必要がある。これは動かしたいカードの上にカードが無い時にカードを動かすことが可能かどうかを表す。ソリティアの並び替え用の場の場合、全てのカードが移動可能であるため、空に対するカードの設定は全て真を設定する。

#### ⑧クリアルール設定

クリア判定時に比較するカードの枚数を設定し、比較に用いる関係演算子の選択を行う。

ソリティアの場合、同スート用の場4つにそれぞれカードが13枚集まるとゲームがクリアとなる。その為、同スート用の場には比較するカードの枚数を13、比較に用いる関係演算子には“=”(等号)を設定し、それ以外のスポットには比較するカードの枚数は0、比較に用いる関係演算子には“≥”(以上)を設定する。



図6：ソリティア プットルール設定画面

## 4. おわりに

アプリケーションとして、トランプゲームの開発にゲーム開発知識、技術が必要なものとなっていて、本稿の目標は達したと言える。

ただし、このツールを試用したユーザから「デッキを二つ以上用いたゲームを作りたい」といった声があったが、開発初期段階からデッキを複数用いることを想定していなかったため、細部の変更のみでデッキを複数扱うことのできるよう変更することは困難となってしまっている。また本稿では触れていないが、ユーザインターフェイスも稚拙であり、ユーザがトランプゲームを開発するには、まず、このアプリケーションにおける開発の流れを理解する必要がある。

ユーザが利用するツールの開発においては、ツールの評価後にユーザインターフェイスの改善や機能拡張が行いやすいように、拡張性を重要視したシステムアーキテクチャの設計が重要である。

## 5. 参考

[1]ツクール web (<http://tkool.jp/>)

[2]Apache POI-the Java API for Microsoft Documents (<https://poi.apache.org/>)