**１５．テーブルゲーム作成のためのライブラリー開発**

**［機械・電気システム工学専攻］ 油布　航一郎**

**［指　導　教　員］ 加藤　直孝　教授**

Abstract:Table games are inherently limited by the places, the number of people and objects. Therefore, it is not always easy to do them. Moreover, it takes time and effort to create a new board game. Therefore, I have developed a software for creating board games online on Unity, which makes it possible to create board games that are not limited by time, place, or object.

**１．緒言**

　本研究の目的はテーブルゲームをオンライン上で行うことができ、かつユーザーがゲームの定義をすることができるソフトウェアを開発することである。そのために、そのソフトウェアを開発するためのライブラリーを開発した。このゲームは、場所にとらわれずオンラインで行うことができる。　テーブルゲームとは、一つのテーブルに人間が集まってゲームを行うものである。たとえばチェスや将棋等の１対１で行うゲームやババ抜きや７並べなどの多数の人で行うゲームなどである。テーブルゲームを行うにはそのゲームを行うための道具、場所、そして同一の場所に集まることのできる人がいないといけない。現在、将棋やチェスのみを行えるサイトなど、特定のボードゲームのみを行えるオンラインサービス

（Board Game Arena[1]等）は存在している。しかし、そのゲームの種類は多くなく例えば～～～のような特定の有名なゲームに限られている。

**２．使用開発環境**

今回はグラフィック処理に長けている統合開発環境Unity[2]を用い、通信システム部分にはリアルタイムの通信同期サービスであるPhoton[3]を使用する。テーブルゲームはゲームにおける情報を視覚的に入手するためグラフィック処理を行うことが容易であることが望ましい。UnityはC#、JavaScript、Booなどのプログラミング言語で開発を行うことが出来、Unityでの開発による収益が一定額を超えていない場合無料で使用することが出来る。類似する開発環境にUnreal Engineがあるがこちらはよりグラフィックスに特化しており処理を行うのに大きなマシンパワーを必要とする[4]。本研究はテーブルゲームの基盤を作ることを目的にしているので、処理の軽さを重視してUnityを選定した。Unityは非常に多くのリファレンスが存在し開発を行うにあたって便利である。

Fig.1 テーブルゲームの分類

Photonは様々なプラットフォームに対応した同期通信用のSDK(Software Development Kit)とサーバーを提供しているサービスである。SDKを使用することで、Photonの提供するサーバーにネットワーク経由で接続し、Unity内のオブジェクトの状態や位置の同期をとることが可能になる。Photonは時間当たりの通信料が一定を超えなければ無料で使用できる。

Fig.2 クラス関係図

**３．テーブルゲームの概念**

　　テーブルゲームと言えば様々な種類のものが存在する。多くのテーブルゲームは複数の駒、複数のカード、全体の盤面で構成され、プレイヤーはそれらに対して、持つ、置く、動かす動作を行うという少ない要素でゲームが成立する。例えば将棋やチェスは数種類の駒を盤面に置きそれを二人のプレイヤーが盤面上を動かすことでゲームが進行する。現在存在するテーブルゲームの多くには次のような特徴がある。

　(1)駒：個体における情報は少なく対象物に付属することでチェッカーとして機能する

　(2)盤：対象物の位置関係を制限する

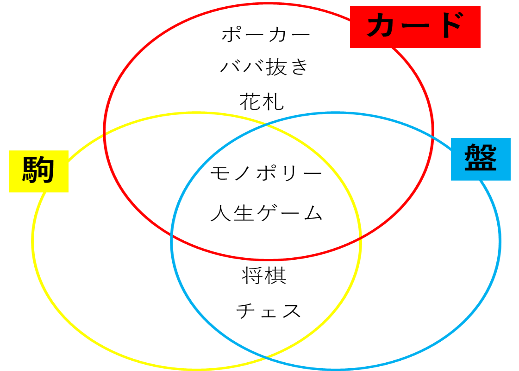
　(3)カード：情報を多く持ち

・自身のみに見える

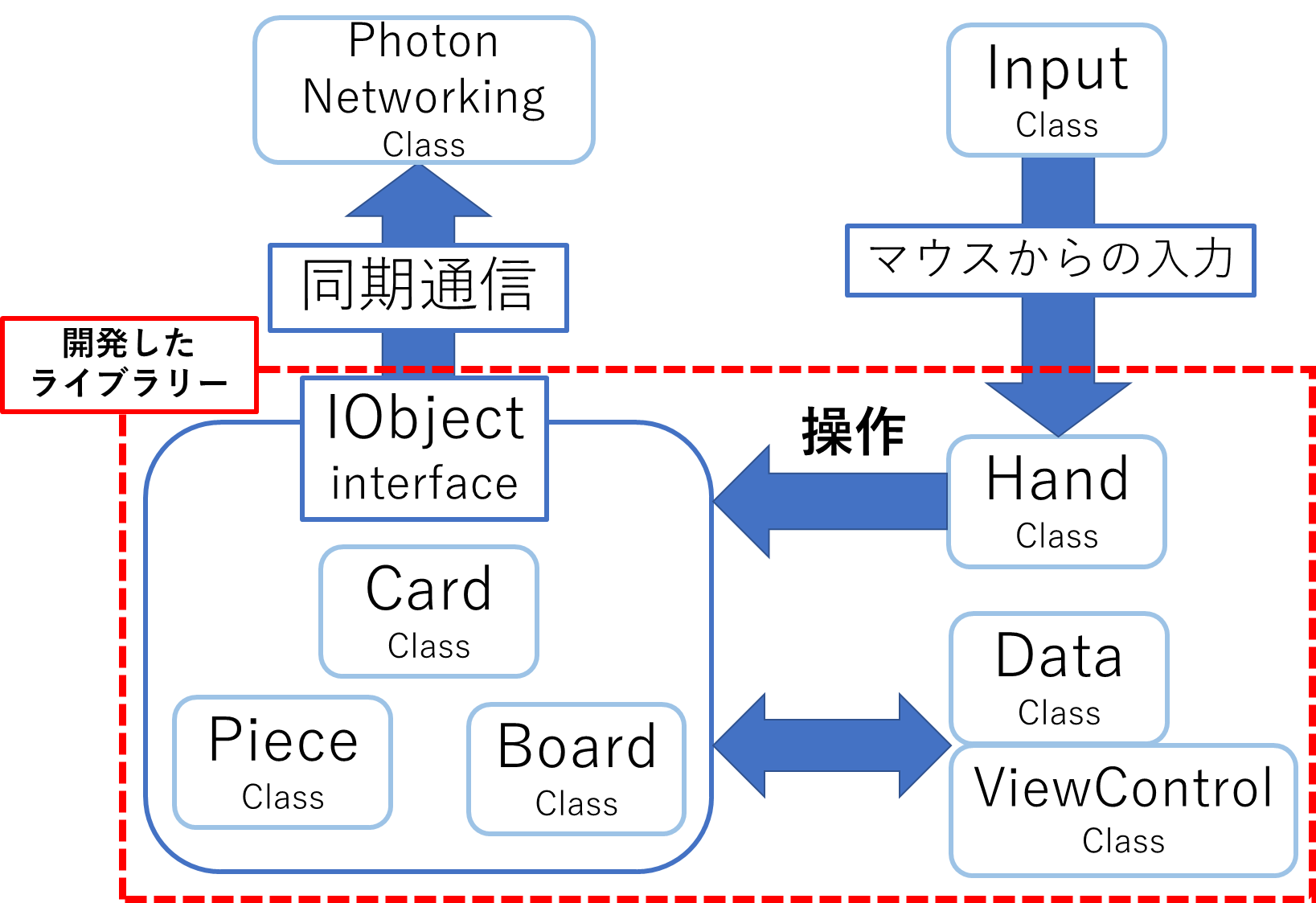
・他者のみに見える

・自身、他者の両方に見える

・自身、他者の両方に見えない

Fig1に駒、カード、盤の要素にゲームを分解し分類したものを示す。

**４．ライブラリー開発**

開発したクラスのライブラリー内の関係図をFig.2に示す。

テーブルゲームの概念として分類されたカード、駒、盤それぞれの要素を

・Card Class

・Piece Class

・Board Class

として定義する。これらのクラスはIObject　interfaceを継承している。

次にテーブルゲームでの物体の移動や変化を行うクラス

・Hand Class

を定義する。Hand Classはテーブルゲームを進行する際の手の役割を果たし、IObject interfaceを継承したクラスを管理するクラスである。マウスからの入力を感知し、その入力に応じてマウスカーソルの位置にあるIObject 　interfaceを継承した物体の状態を変化させる。

Hand ClassはIObjectに指定されている関数を呼び出すので、IObject interfaceを継承したクラスを新たに作ることでカードや駒、盤以外のふるまいを持つクラスも

Hand Classで操作可能である。

実際にライブラリーを使用するにはまず以下の手順を踏む

1. スクリプトとGame ObjectがフォルダーにまとまったデータのライブラリーをAsset->Import New Assetからインポートする。
2. Player\_prefabをScenesに追加する。
3. 駒やカードなどを出現させたい箇所で

obj=PhotonNetwork.Instantiate(“名前”);

または

obj=Instantiate(“名前”);

と記述すると駒やカードなどがGameObjectとして出現する。

1. 出現したGameObjectに初期状態を入れるため

obj.GetComponent<Data>().cloneData(json);

または

obj.GetComponent<Data>().setData(値);

と記述する。

上記の４手順でオブジェクトの生成、操作が行えるようになる。

**５．作成したソフトウェア**

　開発したライブラリーを用いてソフトウェア

「Table Game Creator」を開発した。

　Table Game Creatorはテーブルゲームを作成し、オンライン上でのプレイを可能にする。

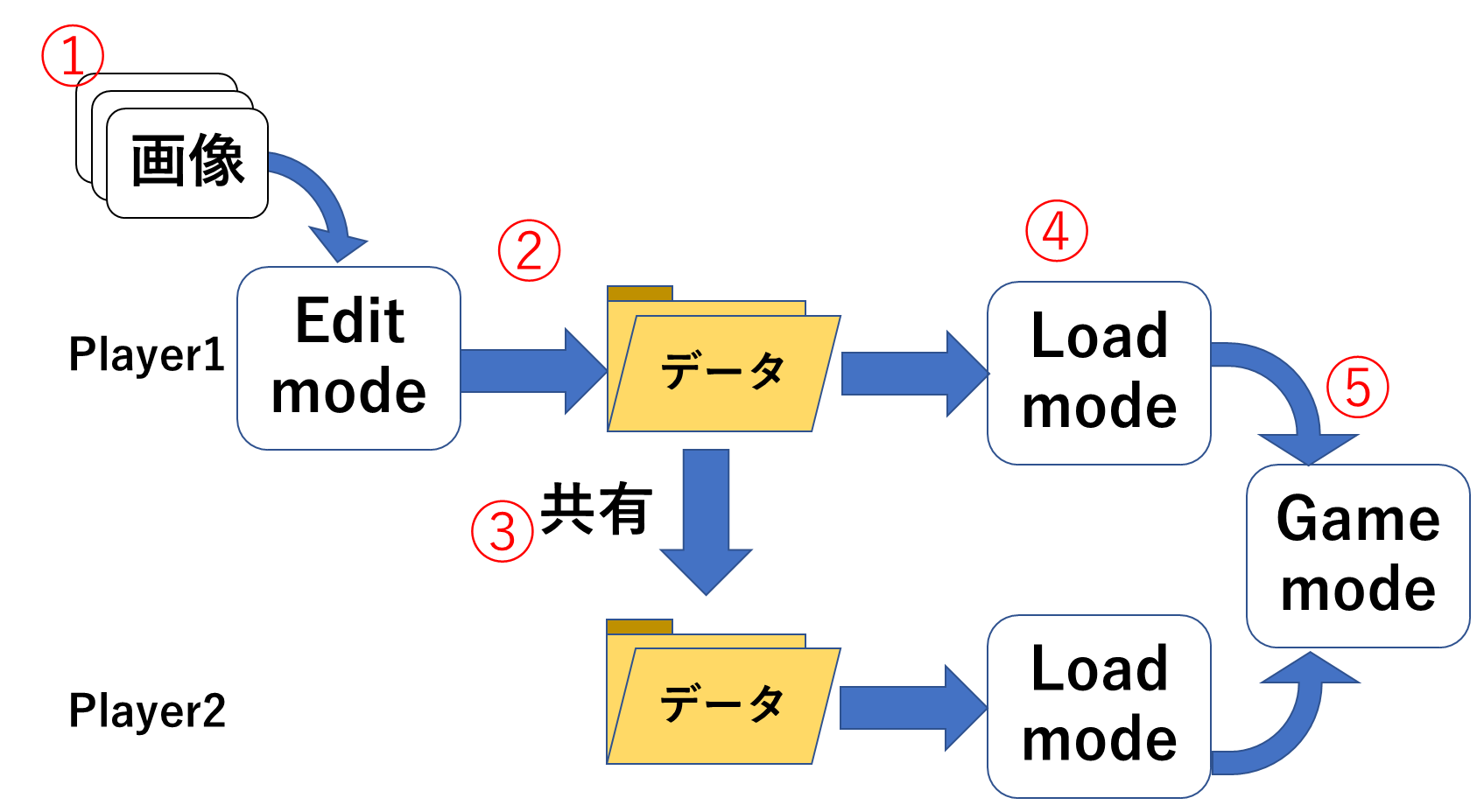
　ソフト全体の動作の流れをFig.3に示す。

Fig.3 Table Game Creator 動作手順

手順としては

1. 画像の用意

テーブルゲームで使用される画像（トランプの数字とマークや将棋の駒の文字等）を作成、または入手。

1. テーブルゲームの編集

Edit modeでゲームの名前を決め、テーブルゲームに必要なもの（使う物の種類や個数）を決める。

1. データの共有

編集で出力されたデータを、テーブルゲームをプレイしたい人間と共有する。

1. データの読み込み

Load mode でプレイを行うテーブルゲームを選択しオンラインにアクセス。

1. テーブルゲーム開始

　プレイを行いたい人全員がオンラインのGame modeにアクセスを行ったら初にテーブルゲームを始めた人が「Game Start」ボタンを押すと、Card、Piece、Board のそれぞれのクラスを持ったオブジェクトが生成される。

上記①～⑤の手順でテーブルゲームをオンライン上で行うことができる。Fig.4にGame mode の一例を示す。



Fig.4 (テーブルゲーム名)のプレイ画面

　テーブルゲーム内で使用するオブジェクトの設定を独自で行えるので、ユーザーは従来のルールにとらわれることなく自由にテーブルゲームをプレイすることが可能である。

**６．結言**

　本研究における開発によって、次のことが可能になった。

・独自のテーブルゲームをオンライン上で作成し、プレイを行う。

・Unityで新たにオンライン通信の同期を行うテーブルゲーム開発を行う際、今回開発したライブラリーを用いることで他のテーブルゲーム開発の負担を低減できる。

ライブラリー、ソフトウェアの公開はそれぞれUnityのAsset Store、SteamのStoreにて公開審査中である。

　今後の展望としてプレイ途中の状態の保存機能を追加することでより円滑にテーブルゲームを行える予定である。

**参考文献**

[1] Board Game Arena

https://www.photonengine.com/ja-JP/Photon

<https://boardgamearena.com/>

[2] Unity

https://unity.com/ja

[3] Photon

https://www.photonengine.com/ja-JP/Photon

[4] 【徹底比較】Unity vs Unreal Engine！

<https://www.geekly.co.jp/column/cat->

webgame/1903\_051/