

横浜国立大学におけるビジネスゲームプラットフォームの発展

田 名 部 元 成

1. はじめに

横浜国立大学経営学部では、2001年度からビジネスゲームを主体とした教育と研究を展開してきた。この背景には、同年に横浜国立大学に着任した白井宏明（現横浜国立大学名誉教授）のビジネスシミュレーションならびにビジネスゲームの研究と教育実践、そしてその普及に対する精力的かつ継続的な取り組みがある。本稿では、白井が着任した2001年4月から定年退職する2017年3月までの間に行われた多様なビジネスゲームに関わる取り組みを支えたビジネスゲームのプラットフォームであるYBG（Yokohama Business Game）に焦点を当て、その発展の経緯を概観するとともに、それぞれの時代区分における意味づけを行い、今後のビジネスゲームプラットフォームの展開の方向性を考察する。

情報システムとしてのYBGは、ビジネスゲームの設計や開発を行うゲーム開発者、開発されたゲームを教育や研究に用いるゲーム実施運営者、そして、実施されるゲームをプレイするゲーム参加者という異なる役割の利用者に対して、ゲームの開発、運営、参加を支援する機能を提供するオンラインシステムである。本稿執筆時点で利用されているYBGは、正式には後に説明するBSel（Business Simulation for e-Learning）と呼ばれるシステムであるが、BSelの多くの部分はYBGの考え方を引き継いだものとなっているため、YBGの一連として捉えられる。したがってBSelはシステム名でもありYBGのバージョン名でもある。このため、文脈上区別が必要ない場合は、YBGとBSelの総称としてYBGを用いる。

YBGを用いて、横浜国立大学で初めて授業の中でビジネスゲームが実施されたのは、2001年10月11日のことである。YBGという言葉が初めて使われたのは定かではないが、少なくとも筆者の所有する2001年12月6日付の電子メールのメッセージの中で、その言葉が使われていることから、2001年頃から定着したものと思われる。その意味では、2001年がYBG元年であると言える。白井は、2001年度に新設された大学院授業科目「ビジネス・モデリング特論」、翌年度新設の経営学部3年次向けの授業科目「ビジネスゲーム」において、YBGを積極的に用いた経営学教育を展開した。2004年度からは、経営学部1年生を対象とした専門科目「グループ思考システム論」が開講され、以来2017年度まで筆者がその科目を担当し、YBGを積極的に活用したゲーミングを教育に展開した。YBGは、文部科学省の現代GP（2004-2006年度）や特色GP（2007-2009年度）での採択や科学研究費による研究プロジェクト、大学内の公式な研究組織であるYNUビジネス

シミュレーション研究拠点の設置など、幾つかの重要な契機に機能強化が図られている。現時点でYBGは、国内120を超える教育機関にその利用環境が提供されており、我が国最大の教育機関向けのビジネスゲームのITサービス提供プラットフォームに成長した。本稿では、YBGのこれまでの発展を歴史的に概観したうえで、各時代区分におけるYBGの意味を見出しながら、今後のYBGの方向性を考える。

2. YBGの誕生

YBGの前身となるシステムは、筑波大学で開発されたビジネスゲーム生成システム (Game Generator; GG) である。GGは、ビジネスモデル生成システム (Business Model Development System; BMDS) とビジネスモデル記述言語 (Business Model Description Language; BMDL) から構成される。BMDSは、BMDL言語の文法に従って記述されたビジネスゲームのモデル (スクリプト) を入力として、ウェブサーバ上で動作するビジネスゲーム環境を出力として生成する生成器である。BMDLは、動作するビジネスゲーム環境を生成するためのビジネスゲームに特化したドメイン固有言語である。

GGは、オンラインビジネスゲーム環境の生成器という単純なものだったが、生成されたゲームの版管理や運用管理などのゲームの開発と実施に関わるビジネスプロセスを支援する機能は有していなかった。もともとYBGは、ゲーム開発がGGが動作するサーバへのアクセスが遠隔から行えないという当時の利用環境の制約を解消するために、ゲーム開発をインターネット経由で行えるよう、そのインタフェースを標準的なウェブブラウザのみで行えるようにするために開発されたという経緯を持つ。筑波大学で開発されたGGの動作環境はUNIX系OSであったが、2000年当時、リモートアクセスの制約の問題を解決するために、白井はGGがPC上で動作するように、Windowsへの移植を考えていたようであるが、当時のWindows環境ではUNIX互換のソフトウェアを導入し、GGを動作させるのは不可能ではなかったにせよ、気軽に誰でもがゲー

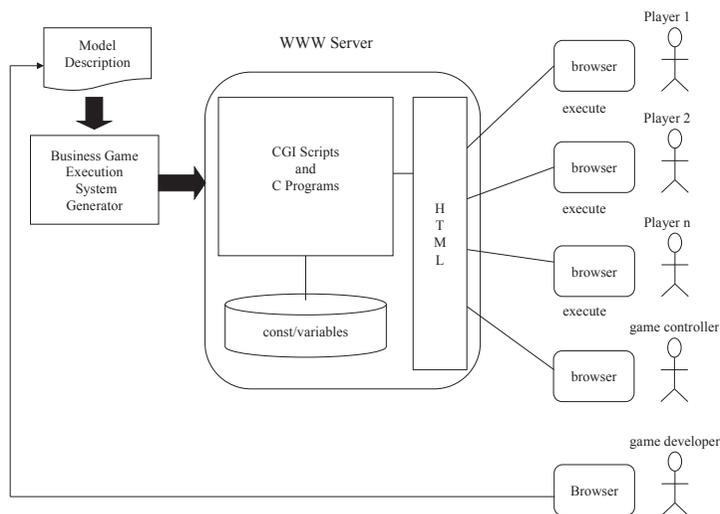


図1 開発当初のYBGのアーキテクチャ

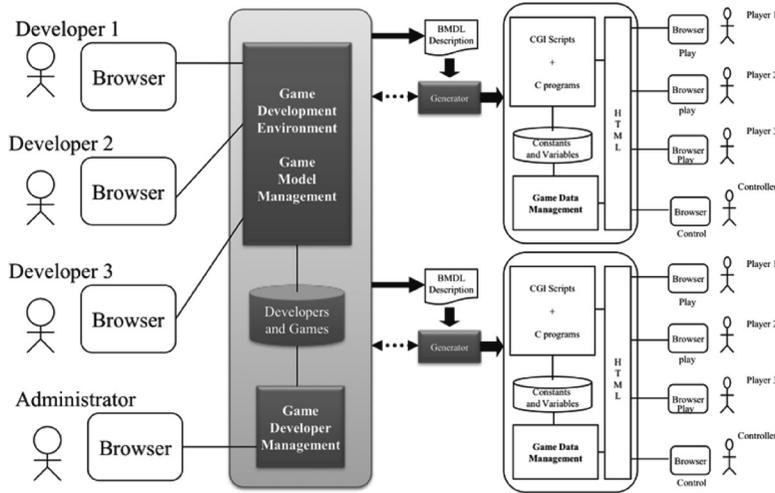


図2 YBG1.0のアーキテクチャ

ムを開発できる環境を容易に構築することは難しかったため断念せざるを得なかったようである。筆者は白井から、ウェブブラウザを通じてサーバにアクセスして独自のビジネスゲームを開発できるビジネスゲーム開発運用のシステムをGGをベースに構築できないかとの相談を受けYBGの開発に着手した。GG開発者の久野靖氏（筑波大学・当時）より、Perl言語によって記述された1420行（51,174バイト）という簡潔にまとめられたソースコードを送ってもらい、それをもとに数ヶ月かけて完成したのが、YBGの原型である。図1に開発当初のYBGのアーキテクチャを示す。

その後、筆者は、図1に示すYBGの基本環境を複数生成できるアーキテクチャを考案し、その実装を行なった（図2）。新しいYBGシステムによって、システム管理者がゲーム開発者を複数登録でき、それぞれのゲーム開発者が複数のゲームを自由に開発できるようになった。当初のYBGの仕様書を付録に示す。

3. YBGのスクラッチ開発

YBGにおける開発環境のウェブ化や複数開発者の管理機能は、ゲームの開発と実施の機会を増やすことに貢献した。この結果、ゲーム実施にかかるプレイヤーやデータの管理、あるいは開発中のゲーム自体の管理など、運用面と開発面での強化の必要性が高まり、それを受けて、白井と筆者は、2003年頃からYBG2系列の開発に着手している（田名部, 2011）。YBG1.0は、GGのソースコードの改変と追加によって実装されていたが、YBG2.0の開発では、BMDLの言語処理の方法を抜本的に見直し、その実装は、ゼロからソースコードを記述することで実現している。追加された機能は、ゲーム開発者が開発中ゲームの動作確認を行える機能、開発されたゲーム実施環境をセッションと称して個別に管理できる機能、YBG利用者をユーザIDで管理し、ユーザ毎にプレイヤー、コントローラー、開発者という権限をそれぞれ与えられるようにする機能などである。その後、白井と筆者は、YBG2.0の本格運用に向けて、大学予算を獲得してYBG2.1

の開発を進めている。

YBGは、「グループ思考システム論」(学部1年向け)、「ビジネスゲーム」(学部3年向け)、「ビジネス・モデリング特論」(大学院)、「ビジネス・シミュレーション」(ビジネススクール)において積極的に活用され、このような取り組みが評価され、横浜国立大学経営学部は、平成16年度文部科学省現代的教育ニーズ取組支援プログラム(現代GP)(2004~2006年度)に採択された。この取り組みは、いつでもどこでも学習を可能とする経営学eラーニングの開発と実践をテーマとするもので、eラーニングによって、学習機会の多様化と社会人などの学外からの学習機会を増加させ、さらに学生1人1人の学習状況を管理することによってきめ細かい学習支援を目指すというものであった。この採択は、YBGを新しい領域へと導くことになった。白井は、複数の学習者が参加することのできるマルチユーザ型eラーニングという概念を提案したが、新しく開発されたYBG3.0, 3.1はその概念実証としての役割も果たしている。そしてYBG3.xは、eラーニング型授業の設計と実践に用いられた。現代GPにおける取り組みは、白井が主査を務める日本シミュレーション&ゲーミング学会ビジネスシミュレーション研究部会の活動と連動して、YBGユーザコミュニティの基盤形成にも繋がった。

4. YBGとユーザコミュニティ

YBGをさらに別の方向へと導いたのは、平成19年度文部科学省特色ある大学教育支援プログラム(特色GP)(2007~2009年度)での採択である。この取り組みは、ビジネスゲームを活用した体験型経営学教育を実施できる教員を育成するためのFD(Faculty Development)プログラムを開発するもので、(1)教員育成のための標準プログラムとして、ビジネスゲームの利用・改造・新規開発の3段階教育プログラムを開発する、(2)全国各地域に拠点となるノード校を育成し、ここを通じて地域内の大学に人的支援サービスを提供できるビジネスゲーム・コンソーシアムを構築する、(3)ビジネスゲーム・ネットサービス(インターネット上のコミュニティサイト)を構築し、このサイトを通じてビジネスゲームの開発ノウハウや授業ノウハウの共有・交換、教材の流通・再利用促進を支援する、という3つの柱を実現しようとするものであった。特色GPを契機に、現在、年2回開催しているYBGユーザ会議の前身に位置づけられる特色GPセミナーや特色GPシンポジウムが開催された(表1)。

この間、YBG2007, 2008, 2009がリリースされ、ゲーム中にプレイヤー同士がテキストによるチャットができるコミュニケーション機能やビジネスゲームモデルやシナリオなどの教材流通の仕組みが新たに実装されている。YBG2009シリーズでは、内部のデータ管理にMySQLというデータベース管理システム(DBMS)を採択したYBG2009DBがリリースされている。

特色GPにより、YBGユーザコミュニティの形成がなされたが、このことは、YBGがコミュニティで利用されるプラットフォームとしての役割を確立したと言える。コミュニティの形成は、YBGの機能強化によるものというよりは、ユーザIDの発行や役割変更、ゲームモデリングにおける問い合わせや相談への対応、ゲーム実施における障害や不具合への対応、ユーザが安心してYBGを授業などで用いることができるようなバックアップなど、白井によるYBGユーザに対する充実した支援活動によるところが大きい。

その後、YBG2009DBから、教材流通とゲームプレイヤー間コミュニケーションの機能をなくし、システムの文字コードをUnicodeに変更したYBG2010DBがリリースされている。文字コー

表1 特色GP関連イベント年表

日付	イベント名	場所
2007年10月20日	横浜国立大学特色GPセミナー in札幌	ホテルサンルート札幌
2007年10月28日	横浜国立大学特色GPセミナー in秋田	秋田大学
2007年11月9日	横浜国立大学特色GPセミナー in博多	ホテルセントラザ博多
2007年11月20日	横浜国立大学特色GPセミナー in大阪	大阪国際交流センター
2008年3月28日	横浜国立大学特色GPシンポジウムin横浜	横浜エクセルホテル東急
2008年8月21日	横浜国立大学特色GPセミナー in金沢	金沢都ホテル
2008年9月9-11日	横浜国立大学特色GPセミナー in東京	ホテルパークサイド
2008年10月5日	横浜国立大学特色GPセミナー in高松	全日空ホテルクレメント高松
2008年10月15日	横浜国立大学特色GPセミナー in名古屋	ホテルサンルートプラザ名古屋
2008年12月10日	横浜国立大学特色GPセミナー in仙台	東北大学
2009年2月27日	横浜国立大学特色GPセミナー in広島	ANAクラウンプラザホテル広島
2009年3月5日	横浜国立大学特色GPセミナー in京都	ホテル京阪京都
2009年8月3日	横浜国立大学特色GPセミナー in東京	東京国際フォーラム
2009年9月20-22日	横浜国立大学特色GPセミナー in京都	ホテル京阪京都
2009年12月4日	横浜国立大学特色GP成果報告会	横浜国立大学経営学部

ドの変更は、ゲームのインタフェースやゲーム自体の国際化への対応である。

5. BSEL

YBGは、ラウンド進行型のビジネスゲームを想定しているため、オンラインで利用できるとは言っても、基本的には各ラウンドに定められた期限までにプレイヤーが入力を行なう必要がある、いつでもゲームに参加でき、また、一方で、毎ラウンド参加しなくてもよいタイプのゲームは想定していない。すなわち完全eラーニング型のビジネスゲームを当初から前提としていない。この問題に対処すべく、白井と筆者は、2011年頃から、多人数型完全eラーニングや独習型eラーニングに対応した新しいビジネスゲームプラットフォームBSEL (Business Simulation for e-Learning) の開発に着手した。最初のリリースは、2014年のBSEL2014である。現行のオンラインで提供されているYBGのバージョンは、BSEL2014から従来YBGの機能を削減して、ビジネスゲームの開発と運用が最低限行える軽量版であるBSEL2014Liteである。

現行のYBGは、ビジネスゲームコミュニティに対するビジネスゲームの開発、実施、管理サービスの提供と利用支援、ビジネスゲーム教材やビジネスゲームを使った教育と研究の実践ノウハウの共有といった様々なビジネスゲームに関連する活動を支える中核的な基盤となっている。YBGは、もはや単なるソフトウェアシステムではなく、ITサービスを提供するサービス資産であり、YBGによって提供されるITサービスは、ビジネスゲーム設計開発者、実施運営者、プレイヤーという「顧客」に対して、それぞれの目的の達成に資する価値をもたらす必要がある。すなわち、ITサービスマネジメントの視点でYBGを考える必要がある。このようなITサービスを管理する手法は、ITサービスマネジメント (ITSM) と呼ばれる。ITSMの文脈でサービスとは、顧客が特定のコストやリスクを負うことなく、期待する成果を実現することを促進することによって、顧客に価値を提供する手段のことを言う (田名部, 2017)。ITSMに対して広く利

用されているアプローチに、1980年代に英国政府がまとめた、IT提供者に求められるサービス機能に関する成功事例集に端を発しているITIL (IT Infrastructure Library) があるが、YBGをITSMの視点で考える際には、ITILに準拠したマネジメントの導入が効果的であると筆者は考えている。

このような背景から、2017年度からYNUビジネスシミュレーション研究拠点では、その中にYBG/BSel開発運用チームを設置し、YBG/BSelの開発および保守に携わってきた業者とYBG/BSelが提供する各種サービスの品質水準の維持および向上に資する、関連システムの修正、新規開発、運用、保守等に係る関係者との情報交換と作業調整を行うことを目的としたYBG/BSel開発運用会議を定期的で開催している。この会議で決定した2017年度の重点化領域は以下の通りである。

(1) 情報セキュリティ対応

情報セキュリティリスクアセスメントの実施、情報セキュリティリスク対応プロセスの策定、脆弱性診断と診断結果に基づく技術的対応、個人情報への取扱いに対する検討

(2) 安定的なサービスの提供

サービスレベル管理、インシデントおよびサービス要求管理、問題管理、構成管理、変更管理、リリース及び展開管理など、ITILに準拠したプロセスの導入

(3) YBG/BSel利用体験の機会拡大

YBGを利用したことのない人でも、ビジネスゲームに随時参加でき、YBGのビジネスゲームの面白さや教育や研究に対する有用性を体感してもらえらる仕組みの導入

(4) 言語的定性的ビジネスゲームへの対応

これまで教育実践研究として行った言語的定性的ビジネスゲームの理論的展開に対応するあらたなYBG/BSelの高度化

YBGをITサービス資産と考えたとき、ビジネスゲームに関連する活動をシームレスに支援するサービスが、YBGによってすべて提供されている訳ではない。例えば、新規に授業にYBGを使いたいという顧客が現れたとき、ユーザIDを発行するというサービス要求 (service request) に対しては、現状ではマニュアルでユーザIDを登録するという対応をとっている。授業中に学生が開発したゲームを実施させたいときも、プレイヤー権限のユーザIDを複数発行する必要があるが、これもYBGシステム管理者の対応が必要となる。筆者は、ITサービスマネジメントの観点から、このような顧客からのサービス要求に対するプロセスを定式化してYBGの新しいITサービスによって支援することを検討している。

6. YBGの今後

これからは、Cloud Computing, Artificial Intelligence, Mobility, Big Data, Robotics, Internet of Things, Cybersecurityという技術が結合しながら急速に社会に浸透する時代に入ると考えられる。これら7つの技術トレンドは、それらの頭文字を取ってまとめてCAMBRICと称す動きもある。このような造語が作られた背景には、これらの技術がシステムに組み込まれて製品やサービスとして社会に実装されていくと、従来の規範では対処できない事態を招き、それらは、技術的側面ではなく、哲学的、倫理的、法制的な規範の見直しを根本から迫るもので、個々

の要素技術で物事を捉えず全体を俯瞰して包括的に考える必要性の高まりがある。その意味では、ビジネスゲームが利用される教育や研究の対象とするビジネスやマネジメントあるいは経済社会システムの規範、さらには、そのような環境下におけるゲーミングのあり方も再考する必要がある。

YBGは、BSelの開発時からクラウド上でのサービス提供を意識したものとなってきているが（田名部, 2011, 2012）、その他の技術動向は、あまり意識されていない。まず、今後ビジネスゲームが、様々なサービスと連携してサービスを提供していくことを考えると、YBGからユーザーインタフェースの部分を切り離し、API（Application Programming Interface）化を図る必要があるだろう。このことにより、例えばモバイル端末上の専用アプリケーションが、APIを介してYBGに接続して、ビジネスゲームに対する意思決定入力やコントローラによるラウンド進行などが可能となるばかりではなく、知的エージェントがYBGにアクセスして、プレイヤーとしてゲームに参加することも可能となる。簡易的なエージェントであれば、現行のYBGでも実装事例があるため、人工知能を取り入れたゲーミングは十分に実現可能であると考えられる。従って、ゲームの実施データを大量に蓄積して、プレイヤー行動やシステムの特性を学習させ、その学習結果に基づいてマシンエージェントを振る舞わせることによって、人間自身を強化するのに使えるかもしれない。また、ロボットとの身体的対話が要素となるビジネスゲーミングがこれからの社会での教育訓練に用いられるようになるかもしれない。後半の例は、近未来のことであるため、あくまでも想像上の話だが、人工知能やロボットの登場により、YBGも新しい教育と研究のニーズに沿って、役割が変化していくことだろう。

7. おわりに

YBGは、ビジネスゲームを生成するシステムからスタートし、情報技術の発展、ユーザコミュニティの発展とともに、ビジネスゲームの活動の諸側面において価値を提供するITサービスに進化してきた。従って、今後は、ビジネスゲームの開発者や利用者からなるコミュニティに対して、すべてのビジネスゲーム関連活動に価値を提供するITサービスのマネジメントが求められる。さらに、CAMBRICに代表される技術トレンドの社会への浸透を踏まえて、哲学的、倫理的、法制的な視点からのシミュレーション&ゲーミングのあり方を検討し、その検討結果をYBGというITサービスに反映させていくことが求められる。すなわち、今後のYBGの発展においては、技術的な高度化は主要な目的ではなく、ビジネスゲームを包括するシミュレーション&ゲーミングがもつ本質的部分を人間や社会といった側面から探求し、その知見を新しい社会環境においてビジネスゲームとして適用するための枠組みの進化が重要な目的となるだろう。

参 考 文 献

- 田名部元成（2011）ビジネスゲームのための言語の設計と実装，横浜経営研究，32（2），pp.53-78.
- 田名部元成（2012）クラウド型ビジネスシミュレーションアーキテクチャ，第2回社会システム部会研究会資料，計測自動制御学会，pp.17-18.
- 田名部元成（2017）ビブリオ・トーク -私のオススメ-：『アポロ13』に学ぶITサービスマネジメント，情報処理，58(7)，pp.634-635.

<付録 YBGの仕様書>

以下は、YBG1.0の開発版であるYBG Version0.61の仕様書（2003年7月8日、仕様書バージョン0.1、田名部作成）からの抜粋である。

用語

本仕様書では、以下の用語を定義された用語として用いる。

1. YBGユーザ

インターネットからYBGシステムにアクセスをするユーザ。以下の種類がある。

- a. プレイヤー：ゲームをプレイするユーザ
- b. コントローラー：ゲームの進行を制御する役割をもつユーザ
- c. ゲーム開発者：個別のゲームを開発するユーザ。ゲーム開発者によって作られたゲーム実行環境は、プレイヤーやコントローラーによって利用される。単に開発者と呼ぶこともある。
- d. ゲーム開発者管理者：ゲーム開発者のアカウントの管理を行うユーザ。

2. YBGシステム開発者

YBGシステムを開発する者。UNIXアカウントybgadminの所有者であり、root権限も持つ。この仕様書はYBGシステム開発者を対象に書かれている。ゲーム開発者として default というユーザ名を持つ。

3. ジェネレータ

プレイヤーに実行させるためのビジネスゲーム実行個別システムを生成するシステムコンポーネント。ゲーム開発者によって利用される。

4. ビジネスゲーム実行個別システム

ゲーム開発者が開発する、プレイヤーに実行させるための個別システム。ジェネレータによって生成される。個別システムと呼ぶときもある。

Y B Gシステム

YBGシステムは以下のプログラムモジュールから構成される。個別のビジネスゲーム実行系は、すべてジェネレータ (index.cgi) から生成される。

名称	ファイル名	場所	説明
ジェネレータ	index.cgi	\$YBGHOME/ybg/cgi-bin/usr/*/dev/	個別ビジネスゲームを生成するシステム
ゲーム開発者管理システム	psmanager.cgi	\$YBGHOME/ybg/cgi-bin/admin/	ゲーム開発者のアカウント管理を行うシステム
入出力用共通ライブラリ	stdio.pl	\$YBGHOME/ybg/cgi-bin/lib/	CGI入出力用共通ライブラリ
ジェネレータ更新	update-generator.pl	\$YBGHOME/ybg/cgi-bin/admin/	\$YBGHOME/cgi-bin/usr/default/index.cgiを各開発ユーザディレクトリへコピーする。(保守用)

※ \$YBGHOME は/home/ybgmgr を仮定している。

ジェネレータ

ジェネレータとは、ビジネスゲーム実行個別システムを生成するサブシステムであり、各ゲーム開発者のディレクトリのdev ディレクトリに配置される。例えばゲーム開発者 bgdev のホームディレクトリは

```
$YBGHOME/ybg/cgi-bin/usr/bgdev
```

となり、ゲーム開発者bgdev用のジェネレータの位置は

```
$YBGHOME/ybg/cgi-bin/usr/bgdev/dev/index.cgi
```

となる。各ゲーム開発者のホームディレクトリ以下の環境は、ゲーム開発者管理者がゲーム開発者管理システムpsmanager.cgi でゲーム開発者のユーザアカウントを作成した際に、YBGシステム開発者用アカウント“default”の環境の一部をコピーして作成される。したがって、YBGシステム開発者は、YBGシステム自身を改良、変更する場合は直接 shell から

```
$YBGHOME/ybg/cgi-bin/usr/default/dev/index.cgi
```

へ修正を加える。そして最終テストが終わった段階で、変更されたindex.cgiをリリースする。各ゲーム開発者のジェネレータを変更するには、update-generator.plを用いる。

上述した通り、ジェネレータはビジネスゲーム実行個別システムを生成する。個別システムは、各ゲーム開発者のホームディレクトリの bg ディレクトリ以下に配置される。例えば、default というゲーム開発者の作成したtest01 という名称のゲームであれば

```
$YBGHOME/ybg/cgi-bin/usr/default/bg/test01/
```

以下に個別システムのファイルが配置される。

新規にゲームを作成した直後は、ゲームシナリオテンプレートファイルinit.ggのみが該当ディレクトリにコピーされる。

```
コピー元：$YBGHOME/ybg/template/default.gg
```

```
コピー先：$YBGHOME/ybg/cgi-bin/usr/default/bg/test01/init.gg
```

このinit.ggはゲーム実行個別システムを生成するときに参照される。

ジェネレータindex.cgi は上述のinit.ggから個別システムのすべてのファイル群を生成する。以下は、デフォルトのinit.ggを利用して個別システムを作成した直後のファイルのリストである。(ls -lRの出力結果の一部)

```
-rw-r--r--  1 nobody  nobody           0  7月  8 15:53 AGENT
drwxrwxrwx  2 nobody  nobody        4096  7月  8 15:53 DATA
-rw-r--r--  1 nobody  nobody         325  7月  8 15:53 Makefile
```

-rw-r--r--	1	nobody	nobody	70	7月	8	15:53	PASSCRYPT
-rw-----	1	nobody	nobody	34	7月	8	15:53	PASSWORD
-rw-rw-rw-	1	nobody	nobody	3	7月	8	15:53	ROUND
-rwxr-xr-x	1	nobody	nobody	24804	7月	8	15:53	compute
-rw-r--r--	1	nobody	nobody	5550	7月	8	15:53	compute.c
-rwxr-xr-x	1	nobody	nobody	2918	7月	8	15:53	control.cgi
-rw-r--r--	1	nobody	nobody	1925	7月	8	15:53	control.html
-rw-r--r--	1	nobody	nobody	9593	7月	8	15:53	dispcsv.c
-rwxr-xr-x	1	nobody	nobody	27180	7月	8	15:53	dispcsv.cgi
-rw-r--r--	1	nobody	nobody	10321	7月	8	15:53	dispcsv.pl
-rwxr-xr-x	1	nobody	nobody	14177	7月	8	15:53	dispgen
-rwxr-xr-x	1	nobody	nobody	368	7月	8	15:53	dispgen.c
-rwxr-xr-x	1	nobody	nobody	27148	7月	8	15:53	display
-rw-r--r--	1	nobody	nobody	10898	7月	8	15:53	display.c
-rwxr-xr-x	1	nobody	nobody	1855	7月	8	15:53	display.cgi
-rwxr-xr-x	1	nobody	nobody	7237	7月	8	15:53	fileman.cgi
-rwxr-xr-x	1	nobody	nobody	1702	7月	8	15:53	ig-price.cgi
-rwxr-xr-x	1	nobody	nobody	1825	7月	8	15:53	index.cgi
-rw-r--r--	1	nobody	nobody	257	7月	8	15:53	index.html
-rw-r--r--	1	nobody	nobody	2934	7月	8	15:35	init.bak
-rw-r--r--	1	nobody	nobody	2934	7月	8	15:53	init.gg
-rwxr-xr-x	1	nobody	nobody	1525	7月	8	15:53	ir-price.cgi
-rw-r--r--	1	nobody	nobody	5053	7月	8	15:53	variables.c

各ファイルの意味や役割は以下の通りである。

File/Directory	説明	備考
AGENT	(不明)	
DATA	ゲームの状態変数の値を格納するディレクトリ	init.ggを用いて個別システムを生成した直後はこのディレクトリには何もファイルは置かれない。
Makefile	C言語のコードをコンパイルするための指示ファイル	
PASSCRYPT	プレイヤーやコントローラーがhttpでアクセスした際の基本認証時に参照されるファイル。パスワードは暗号化されている。	
PASSWORD	PASSCRYPTの元になるプレーンテキスト	
ROUND	ゲームのラウンド番号を格納したファイル。中身はテキストでラウンド番号が記載されている。	
compute	compute.cをコンパイルした実行形式	
compute.c	モデルの計算を実行するC言語コード	
control.cgi	コントローラーの各種操作を実行するためのCGIスクリプト	
control.html	コントローラー制御フォームのWebページ	
dispcsv.c	ゲームの状態をCSV形式で表示するC言語コード	
dispcsv.cgi	dispcsv.cをコンパイルした実行形式	
dispcsv.pl	dispcsv.cgiに相当するPerlスクリプト	作成中、未完成
dispgen	dispgen.cをコンパイルした実行形式	
dispgen.c	dispcsv.cgiにパラメータを渡すためのラッパーC言語コード	
display	display.cをコンパイルした実行形式	
display.c	変数の表示出力を行うC言語コード	

display.cgi	入口ページの表示ボタンを扱うCGIスクリプト	
fileman.cgi	DATAディレクトリのファイル管理CGIスクリプト	筑波版には無い
ig-price.cgi	各入力ページの内容を生成するCGIスクリプト	定義ファイルinit.ggに応じてig-○○.cgiとなる.
index.cgi	ゲームの入口となるCGI	筑波版には無い
index.html	ゲームの入口となるWebページ. index.cgiへリダイレクションする.	筑波版のindex.htmlがSSIだったのでindex.cgiを作成してCGI化を図ったため, オリジナルは異なる. (最終的には不要になるファイルである)
init.bak	ゲーム開発者がWebブラウザからサーバにシナリオ定義ファイルを送った際に作成されるinit.ggのバックアップファイル.	
init.gg	個別ゲームシステムを定義したファイル.	ゲーム開発者がWebブラウザからサーバにシナリオ定義情報を送る度に書き換えられる.
ir-price.cgi	各入力ページのデータを受け取り, 値をチェックした上でデータファイルに書き出すCGIスクリプト	
variables.c	C言語の変数定義と読み書き用関数のコード.	幾つかのCソースファイルで利用される.

ジェネレータindex.cgiの処理

ジェネレータはwebクライアントからパラメータを受け取り, 一定の処理を行った後にクライアントに情報を渡す. クライアントから受け取る情報は, ゲーム開発者の開発状態を表すモード情報や, 個別システムの定義情報 (最後にはinit.ggとして保存される) である.

モード情報は, webクライアントからmodeという変数の値として渡される. modeの値は次の5種類が定義されている. このmodeの値によって処理は分岐し, それぞれの処理を行う.

NULL	初期画面の表示
Creategame	新規ゲームの作成を行う (サブルーチンCreateGame () を実行)
selectgam	既存ゲームの選択を行う (サブルーチンSelectGame () を実行)
editconfirm	サーバへ送信したシナリオファイルの確認 (EditConfirm () の実行)
compile	個別システムを生成する (Compile () の実行)

[たなぶ もとなり 横浜国立大学大学院国際社会科学研究院教授]

[2017年12月21日受理]