

# スクウェア・エニックスのAI

推薦コメント

**岩谷 徹**

ゲームクリエイター  
ゲームAIの父

日本のおもてなし精神と  
ゲームAIの共鳴！

**栗原 聡**

人工知能研究者  
慶應義塾大学理工学部教授

これから訪れるメタバース時代のための  
実践AI教科書である。

**真鍋大度**

アーティスト・DJ・プログラマ

ゲームにはAIの全てが詰まっている。  
理論と実践で多大なインスピレーションを得るだろう。

スクウェア・  
エニックスの

AI

2024年 7月末発売予定

#### ■ ご注意

本書は著作権上の保護を受けています。論評目的の抜粋や引用を除いて、著作権者および出版社の承諾なしに複製することはできません。本書やその一部の複写作成は個人使用目的以外のいかなる理由であれ、著作権法違反になります。

#### ■ 責任と保証の制限

本書の著者、編集者および出版社は、本書を作成するにあたり最大限の努力をしました。但し、本書の内容に関して明示、非明示に関わらず、いかなる保証も致しません。本書の内容、それによって得られた成果の利用に関して、または、その結果として生じた偶発的、間接的損傷に関して一切の責任を負いません。

#### ■ 著作権と商標

本書に記載されている製品名、会社名は、それぞれ各社の商標または登録商標です。本書では、商標を所有する会社や組織の一覧を明示すること、または商標名を記載するたびに商標記号を挿入することは、特別な場合を除き行なっていません。本書は、商標名を編集上の目的だけで使用しています。商標所有者の利益は厳守されており、商標の権利を侵害する意図は全くありません。

FINAL FANTASY VII REMAKE INTERGRADE© SQUARE ENIX CHARACTER DESIGN: TETSUYA NOMURA / ROBERTO FERRARI

ドラゴンクエストビルダーズ2 破壊神シドーとからっぽの島© ARMOR PROJECT/BIRD STUDIO/SQUARE ENIX

SQUARE ENIX Tech Preview: THE PORTOPIA SERIAL MURDER CASE© SQUARE ENIX

ドラゴンクエストX 目覚めし五つの種族 オフライン© ARMOR PROJECT/BIRD STUDIO/SQUARE ENIX

ヴァルキリーエリュシオン© SQUARE ENIX

Unreal® is a trademark or registered trademark of Epic Games, Inc. in the United States of America and elsewhere. Unreal® Engine, Copyright 1998 – 2024, Epic Games, Inc. All rights reserved.

ROBOTO Apache-2.0 license <https://www.apache.org/licenses/LICENSE-2.0>

## 謝辞

本書を手にとってくださった皆様に深く感謝申し上げます。

出版元である株式会社ポーンデジタルの皆様に感謝します。

桐生隆司社長、スクウェア・エニックスのすべてのスタッフ、  
本書の制作に関わったすべての方々に感謝いたします。

荒牧岳志さん、長谷川勇さん、青池恵美子さん、そして、  
原稿のとりまとめをされた内藤ゆみさんに感謝いたします。

執筆者代表 三宅 陽一郎

## |ごあいさつ

皆さん、こんにちは。スクウェア・エニックスの三宅陽一郎です。この本は、なるべく多くの人に、平易に、我々が日々取り組んでいる「デジタルゲームのAI」について知ってほしい、という思いから始まりました。「デジタルゲームのAI」は、最高に面白い分野です。とても面白くて私は2004年から20年ぐらいつとこの仕事をしています。飽きるということがなく、発展も速く、毎日新しい気持ちで取り組んでいます。

最初に「デジタルゲームのAI」を簡単に説明すると、「**一緒に冒険する仲間キャラクターをより賢くする・敵キャラクターをより生き物っぽくする**」ことです。本書で紹介するAIは他にも様々ありますが、まずはそう考えて頂いてかまいません。そこがすべての出発点になります。

## |この本はどんな本？

でも、AIって目に見える分野じゃないから、なかなかわかりづらいところがあります。この分野の面白さや広がりやを伝えたいのですが、ひとりの力には限界があるため、僕の信頼するスクウェア・エニックスのAIに取り組むメンバーに声をかけ、それぞれの開発・研究分野を解説してもらうことにしました。目標としたのは、「**日本で最もわかりやすい、デジタルゲームAIの入門書**」です（実際、そうなっていると思います）。

各メンバーの執筆でお願いしたのはひとつ、「とにかく、わかりやすく！」です。具体的には、「解説の半分までは誰でもわかりやすいように書いて！」ということです。だから、各解説の前半は誰もわかるようになっていきます。そして、後半から解説の終わりに向かって、ほんの少し専門的になることもあります。しかし、その坂もなるべく緩やかになるように書かれています。そのため、読者の皆さんは、なんの前提知識もなく、手ぶらで読み始めても問題ありません。

文系／理系も関係ありません。数式も、どうしても必要な個所を除いて、かぎりなく省いてあります。また、どの章からでも読み始められるので、興味の赴くまま、心の任せるままに、この分野を散歩してみてください。きっと読者の皆様は、これまで見たことのない、いろんな新しい知識や知見を得ることができるでしょう。

## デジタルゲームのAIの風景

一つの山には木立もあり、草原もあり、花々もあり、茂みもあります。同じように、「デジタルゲームのAI」という山にも、様々な場所や風景があります。本書はその山をゆったりと散策する本です。なるべく平坦な道になるように、それぞれの解説が工夫されています。難しいところは「ここからはこんな風景が見えるよ」という程度におさめています。ここでは、これから散策して頂くコースのあらましを紹介していきましょう。

PART1の「**ゲームAI**」は、本書の入り口になります。先ほど言及した「キャラクターを賢くする・生き物らしくする」ことが目標になります。AIで動くキャラクターが、ゲームステージの中で複雑な地形の中で岩を避けたり、川を飛び越えたり、うまく動き回れるようにするための技術を「**ナビゲーション**」と言います。この「ナビゲーション」の専門家のフランソワさんが、飛び回るキャラクターまで含めた「**3Dナビゲーション**」の世界を案内します。次に、キャラクターたちは自分で考えて行動する必要があります。こういった技術を「**意思決定**」と言い、ゲーム産業では長年、この分野を世界的にリードしてきました。中でも「自分で計画を立てて行動するAI」を「**プランニング**」と言い、この数年、プランニングの開発・研究を進めてきた並木さんが解説します。さて、キャラクターである以上、自分の身体は自分で動かす必要があります。身体を動かす技術を「**アニメーション**」と言い、森寅嘉さんは、この分野に修士課程の学生の頃から興味を持っていて、ずっと開発・研究を続けています。本PARTでは、その成果を解説します。

PART2は「**キャラクターインタラクション**」を扱います。ゲームは大きく捉えると、ユーザーとキャラクターのインタラクション（やり取り）からなります。インタラクションの形にもいろいろあり、動作である場合もあれば、**言葉**である場合もあります。また、キャラクターがただ命令に従う場合もあれば、**キャラクターが感情をもって接する**場合もあります。「キャラクターに感情を持たせるには？」というテーマについてゴティエさんが、「言葉によってユーザーと対話する技術」について森友亮さんがそれぞれ解説します。

PART3は「**メタAI**」を扱います。この言葉を耳にしたことがない方も多いかと思います。PART1～2は「キャラクター」を扱っていたので直感的にわかりやすいですが、「メタAI」というのはゲームシステムがAIとなったものです。つまり、ユーザーを神視点から見て、ゲームを変化させるのです。詳しい解説は本PARTを読んで頂くとして、スクウェア・エニックスでは『FINAL FANTASY XV』（2016年）にも搭載するほど、メタAIに力を入れています。この技術は、ゲーム全体のクオリティをぐっと引き上げる力があります。或いはゲーム開発でどうしても埋まらない「かゆいところ」を補完する力があります。里井さんは、この技術を使ってユーザー体験（UX）を引き上げる開発を主導しており、水野さんは、この技術をオムロン社との共同プロジェクトでロボットに応用し実空間へと応用しました。宋さんは、本技術をブラッシュアップしたコア技術とするプロジェクトを担当しています。

PART4は「**ディープラーニング**」です。AIとはディープラーニング（深層学習）という認識を持つ方も多いですが、それ以外にもAI技術は様々あり、ディープラーニングは広大なAI技術の一つにすぎません。にもかかわらず、ディープラーニングはやはり重要で、これから「デジタルゲームのAI」分野の変革を担う技術です。それぞれ基礎技術（ディープニューラルネットワーク）は共通していますが、音声、画像、アニメーション分野への応用をレアンドロさん、エドガーさん、遠藤さんがそれぞれ解説します。

PART5は「**AIによる品質保証の自動化**」です。デジタルゲームは大規模かつ複雑になっていて、作成後にバグがないか、ミスがないか、確認する作業がとて大変です。そこで、AIが自動的にプレイして、作品に残っている穴を見つける技術がとて大切になります。ここでは、長年このテーマに取り組んでいるファビアンさん、太田さん、ジョシュアさんが解説します。実際のタイトルにもとづいた原稿で、読み応えのある内容になっています。

## 本書の読み方

本書のあらましをさっと紹介してきました。「デジタルゲームのAI」っていろいろあるんだなあ、と思って頂けたのではないのでしょうか。ここからは、気の赴くままに、読んで楽しんで頂けたら幸いです。気になる部から読んでも、気になる章から読んでも良いでしょう。また本書全体に渡って、「注釈」を入れています。注釈から読んでもみても面白いでしょう。

本書をお読みいただき、楽しんで頂くのも良いですし、ゲームをプレイするときに思い出して、別視点でゲームを楽しんで頂くのも良いです。また、研究分野にするのも良いでしょう。本書を通じて「自分の世界が少し広がった」と思って頂ければ幸いです。では、「デジタルゲームのAI」の世界の冒険へと旅立ちましょう。

### ■注釈について

正しそうで正しくない少し正しい、かもしれない注釈。難解な語句が多い技術書のなかで、初学者の方でも読み進める手助けになればとご用意しました。PART1は本文執筆者の森寅嘉さんと並木幸介さんが、PART2～5は坂田新平さんが解説をしています。

### ■お知らせ

2024年4月の組織変更により、旧AI部を含む部門名・職種等に変更がありました。本書では、執筆時点の表記を用いております。

SAMPLE



三宅 陽一郎 Youichiro Miyake

スクウェア・エニックス AI部  
ジェネラルマネージャー

2022年よりAI部ジェネラル・マネージャー。2011年入社より、リードAIリサーチャー。京都大学で数学を専攻、大阪大学大学院（物理学修士）を経て、博士（工学、東京大学）。2004年よりデジタルゲームにおける人工知能の開発・研究に従事。ゲームAI技術の黎明期からAI技術を収集し、また独自の技術を開発しながら、ゲームAIの体系化とその発展を推進している。「大規模デジタルゲームにおける人工知能の一般的体系と実装 - FINAL FANTASY XVの実例を基に -」にて2020年度人工知能学会論文賞を受賞。『FINAL FANTASY XIV』AIテクニカルアドバイザー、『FINAL FANTASY XV』リードAIアーキテクト、『KINGDOM HEARTS III』AIテクニカルディレクター、『FINAL FANTASY VII REMAKE』QA 自動化AIテクニカルアドバイザー、などを手がける。日本デジタルゲーム学会理事、人工知能学会編集委員会副委員長、ゲーム情報学研究会運営委員。著書に『FINAL FANTASY XV の人工知能』（ポーンデジタル）など多数。CEDEC、GDC (Game Developers Conference)、IEEE CoG(Conference on Game)、SIGGRAPH、SIGGRAPH Asia、人工知能学会など国内外の学会・産業カンファレンスで講演・発表を多く行なっている（肩書、所属は2024年3月時点のもの）。



# もくじ

ごあいさつ	4
この本はどんな本?	4
デジタルゲームのAIの風景	5
本書の読み方	6

## PART 1

### ゲームAI

16

はじめに	18
------	----

<b>キャラクターの定義</b>	<b>20</b>
1. キャラクターの作成	21
2. 操り人形師	22
3. データ、データ、データ!	23

<b>ナビゲーションメッシュ</b>	<b>25</b>
1. 環境の認識	25
2. マッピング (地図の作成)	26
3. ボクセル化	28
4. 接続	32
5. 三角形分割	32
6. 侵食	35

<b>経路探索、ステアリング、回避</b>	<b>37</b>
1. 簡単なクエリ	37
2. 経路探索	39
3. 操舵	44
4. 回避	45
5. まとめ	46

<b>ナビメッシュの拡張</b>	<b>47</b>
1. オフメッシュリンク	47
2. コスト	49
3. 好みとその他の要因	50
4. ナビメッシュを改造する	51
5. ランタイム生成	52
6. 塗装と補修	53
7. まとめ	56

<b>3Dナビゲーション</b>	<b>57</b>
------------------	-----------

1. フルワールド ボクセル化	57
2. 課題	58
3. 疎なボクセル八分木 (スパース ボクセル オクトリー)	60
二分法	62
構築	63
4. その他の課題	65
接続	65
剪定	65
最適化	65
ストリーミング	65
経路探索	66
5. 今後の用途	66
破壊と障害	66
新しい環境	67
より優れたAIの動作	67
音の伝播	67
<b>結論</b>	<b>67</b>
<b>プランニングを用いた意思決定システム</b>	<b>68</b>
1. 計画的なAI	68
2. プランニングAIの弱点	69
3. プランニングAIの実用例	71
リアルタイムの変化への対応	74
状態空間の広さへの対応	74
4. プランニングと機械学習	75
5. 新しい技術開発の難しさ	75
6. 開発支援部門の仕事	76
<b>キャラクターAIとアニメーション</b>	<b>78</b>
1. 私たちはどのように身体を動かしているのか？	78
2. キャラクターAIの問題点	79
3. キャラクターAIの導入事例	80
4. MULSアニメーション生成システム	82
5. まとめ	90
<b>PART 2</b>	<b>92</b>
<b>キャラクターインタラクション</b>	<b>92</b>
キャラクターインタラクションとは？	94
<b>感情AIを巡る旅：AIは実際に感情を感じているのか？</b>	<b>96</b>
1. 今までのAIに感情はあったのか？ 感情AIの歴史	96

気持ち、感情、気分	96
AIで動くキャラクター	97
作ろうとしているもの	99
2. 関連研究	100
感情モデル	101
気分モデル	102
モデルについての最終的な決定	103
3. ゲームのための感情要素	105
エモーショナルコンポーネント	106
感情モジュール	107
イベント評価システム	108
気分モジュール	110
性格とは?	111
感情、気分、性格を表現する	111
デザイナーがキャラクターを自分好みに調整するための施策	114
4. エモーショナルコンポーネントの開発から学んだこと	117
5. 付録：ペーパー・研究	121
感情モデル	121
気分モデル	122

<b>自然言語処理</b>	<b>124</b>
1. ゲームキャラクターたちと自由にしゃべろう	126
2. AIで「言葉」を扱う技術	129
3. 開発事例：NLPアドベンチャー	134
アドベンチャーゲームの方式：「コマンド入力式」と「コマンド選択式」	134
「新世代コマンド入力式」の実現に向けて	136
テックプレビュー「SQUARE ENIX AI Tech Preview : THE PORTOPIA SERIAL MURDER CASE」のリリース	139
「デジタルゲームのためのNLP」に必要なこと	140
4. ChatGPT以降の世界	141
ゲームと大規模言語モデル	143

## PART 3

### メタAI

146

メタAIとは	148
--------	-----

<b>「感情を揺さぶる」メタAI</b>	<b>152</b>
1. メタAIで達成したかったこと	154
2. 2次元感情マップ	155
3. 2次元感情マップに基づくメタAI	158
4. Current EPの推定	158
5. Goal EPのプランニング	160

6. Next EPの更新とゲームの調整	161
7. メタAIとの接点になるゲーム要素	162
8. 感情を可視化するゲームプレイ分析ツール	163
9. 感情の強さのタイムライン表示	164
10. メタAIの評価	165
Current EPの推定結果とプレイヤーの感情は一致したのか？	165
メタAIによって、プレイヤーの感情を揺さぶることができたのか？	166
難易度が許容範囲か？	166
ゲームデザイナーがバランス調整しやすいと感じたか？	166
11. まとめ	167
<b>シューティングゲームのメタAI</b>	<b>168</b>
1. シューティングゲームの難易度の作り方	168
一般的な手法	169
動的難易度調整	170
2. シューティングゲームにメタAIを入れてみた	170
2-Layer危険度コントロール	172
3つのAI	174
3. 動的難易度調整の注意点	177
4. 将来の課題	177
5. まとめ	178
<b>卓球ロボットのメタAI</b>	<b>179</b>
1. メタAI、ゲームの世界を飛び出す	179
2. 楽しく上達したい！	180
3. 卓球がわからない…	183
4. 卓球とアクションゲームは何が違うのか	184
5. みんなで汗をかいて仮説を検証するぞ	186
6. プレイヤーのモチベーションをコントロールする作戦	188
7. 現実世界でメタAIが動いた！卓球ロボットのメタAI	190
8. ゲームAIよ、ゲームの世界を飛び出そう	191
<b>メタAIの効果検証</b>	<b>193</b>
1. 相関関係と因果関係	193
相関関係	193
因果関係	194
2. 因果関係を検証する：対照実験と盲検化	194
3. まとめ	196
<b>メタAIとゲームUX</b>	<b>197</b>
1. メタAIが使えるゲーム、使えないゲーム	197
2. ゲームUXとは	197
3. 感情の「流れ」を分析する	198

4. ゲームUXの共通言語化 - パターンランゲージ	203
5. まとめ	204

**メタAIの展望** ..... **205**

1. メタAIの使われる場がより広がる	205
より多くのゲームでメタAIが導入される	205
他のゲームAIと組み合わせる	206
非ゲームのインタラクションに応用する	206
2. メタAIで実現する体験がより深まる	207
メタAIがプレイヤーの前に登場する	207
メタAIがプレイヤーの成長を支援する	207
3. メタAIの発展を支える	208
ゲームUXがより可視化され、より客観的に検証されるようになる	208
4. まとめ	209

**PART 4**

**ディープラーニング**

**210**

機械学習とは	212
--------	-----

**スタイルトランスファー** ..... **220**

1. ゲームAIとその感情	220
2. AIでも絵を描けるよ！	220
スタイルトランスファーと感情表現	221
3. 画像生成AIの技術の裏	222
AIってどうやって画像を描くの？	222
画像データの構造	222
CNNの基礎	223
画像生成におけるCNN	225
スタイルトランスファーモデルの概要	226
スタイル情報の注入	227
スタイル概念を獲得せよ：スタイルトランスファーの逆伝播	229
学習ハードウェア	230
推論	232
余談	232
4. AIは絵を描く	233
感情と絵	233
エモーショナルコンポーネント	233
スタイルの選択と出力	234
スタイルトランスファーによるAIの感情表現	235
5. これからの絵による感情表現	236

**機械学習による自動リップシンク アニメーション** ..... **238**

1. データとしての音の表現	238
2. 音声を画像へ変換する	241
3. 絶対音感と相対音感	243
4. 音声の時間経過フローを学習する	248
5. アニメーション生成	251

## 深層学習によるキャラクターアニメーションの生成 256

1. アニメーションの作成工程	256
キーフレームを打つ	257
自動補間する	258
ベイクする	258
2. キャラクターアニメーション	259
キャラクターアニメーションの作成前の工程	259
キャラクターアニメーションの作成	260
モーションキャプチャの利用	260
3. 深層学習によるアニメーション生成のモチベーション	261
4. 深層学習によるキャラクターアニメーション生成の基本	262
5. 研究内容	264
研究の前提	264
研究対象	265
手法の概略	265
モデル	266
6. 現状の課題と今後の研究について	268
7. 深層学習によるキャラクターアニメーション生成の展望	268
自然言語を用いたアニメーション生成	269
リアルタイムアニメーション生成	269
演技をAIに教える	269
8. まとめ	270

## TALK 1 スクウェア・エニックスAI部 座談会 若手メンバー編

AI部から生まれる、新しいゲームの可能性	272
----------------------	-----

## PART 5 AIによる品質保証の自動化

品質保証の概要	282
QAにおけるゲームAIの使用	288
1. シミュレーションテスト	289
2. 探索テスト	290

3. スクリプトテスト	290
4. 非侵入型と侵入型テスト	291
5. ACREの趣旨	292
6. 操作記録とリプレイ	293
7. ゲームステート	295
8. 同期	296
9. マップと経路探索	299
10. 探索	301
11. 今後の課題	302
<b>バックエンド システム</b>	<b>303</b>
1. 開発の背景	303
2. ワークフロー	304
3. テストスケジューラー	306
4. スライスパッケージと固定パッケージ	307
5. 機材使用率配分	308
6. 機材の共有と割り当て	309
7. レジリエンス (回復機能)	310
8. 停止時間とトイレの削減	311
優先度付きテスト結果削減	311
セッションのマージとコミット	311
Windows Updateの自動化	312
メトリクス (計測して取得されたデータ) の取得と計測	312
<b>バグの分類とデータ分析</b>	<b>313</b>
1. バグの分類	313
2. フィンガープリントを用いたバグの分類	314
3. バグの再現率と解決	314
4. 通知	316
5. データ分析	318
Metabase™	318
Sentry™	319
<b>結論</b>	<b>321</b>

**TALK 2 スクウェア・エニックスAI部 座談会 セクションリーダー編**

<b>研究者たちが描く「ゲーム×AIの未来像」</b>	<b>322</b>
-----------------------------	------------

あとがき	332
用語集	334

**SAMPLE**





# PART 1

## ゲームAI

PART 1は、「AIで動くキャラクターがいきいきとゲーム世界の中で活躍するために必要なAI技術」を解説します。皆さんは、キャラクターがどんなことをしたときに、「ゲーム世界の中でいきいきとしているなあ」と思われますか？ 私はこのテーマにもう20年以上取り組んでいるのですが、それは、ざっとこんなときではないでしょうか？

- > キャラクターがゲーム世界の状況をきちんと感じ取っているとき
- > キャラクターがゲーム世界できちんと判断しているとき
- > キャラクターがゲーム世界できちんと移動しているとき
- > キャラクターがゲーム世界でたくみに身体を動かしているとき

上から順に専門的に言うと、最初の2つが「センシングと認識」「意思決定」、3つめが「ナビゲーション」、最後が「アニメーション」となります。本PARTの3つの章がこれらに対応し、それぞれのエキスパートであるフランソワさん、並木さん、森真嘉さんが解説します。

キャラクター1体を動かすのに、これほど多くのAI技術が集結していることに、驚かれるかもしれません。しかし、私たち人間が現実世界で活動するためには、それだけ深く膨大な知能が動いている、ということでもあります。ぜひ、本PARTで、知能の深みというものを感じ取ってください。

三宅 陽一郎

## はじめに

PART 1では、スクウェア・エニックス AI部による「ゲームAI」への取り組みについて、ご紹介します。今日のゲームで使われているキャラクターAIは、「センサー」「意思決定」「エフェクター」の3つのメインモジュールで構成されています。センサーが環境から情報を取得し、意思決定システムで行動を決め、エフェクターで動作を生成します。これに共有情報を記録する「ブラックボード」システム、環境情報を処理するための「ナビゲーション」システムを加えたものが、現代の基本的なキャラクターAIの構成となっています（図A）。

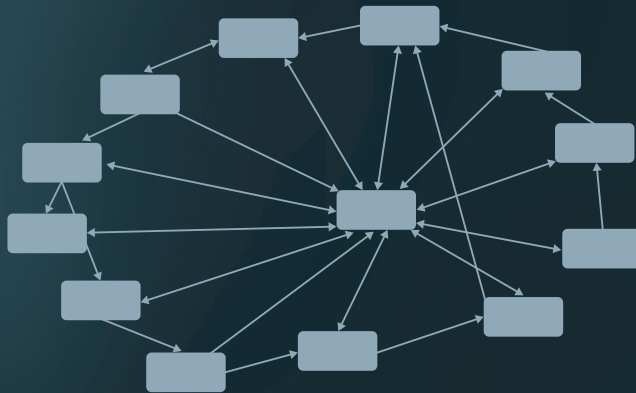


図A キャラクターAIの構成

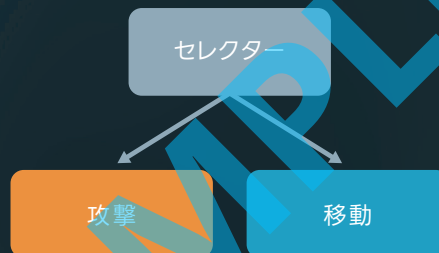
このPARTでは、最初にドゥリス ジャンフランソワさんが一般的なAIについての説明と、AI部で開発を進めているナビゲーションシステムの取り組みをお話します。次に並木さんがプランニングを用いた発展的な意思決定システムを、最後に森 寅嘉さんがゲームAIとアニメーションシステムを接続するプロシージャルアニメーション技術について解説します。ここでは、PART 1で共通して出てくる用語をいくつかご紹介します。

### FSMとビヘイビアツリー

現代のゲームで使われている代表的な意思決定の手法に、「FSM（有限状態マシン）」と「ビヘイビアツリー」があります。FSMはゲーム業界で古くから使われてきた手法で、状態間の遷移でAIの振る舞いを記述するシステムです。FSMは現代でもよく使われていますが、状態が増えるにつれて状態間の遷移が爆発的に増えてしまうという問題を抱えています（図B）。



図B 状態遷移が爆発してしまったFSMデータの例。こうなってしまうとデータの管理がとても大変に



図C ビヘイビアツリーの例

この問題に対処して一定の成功を収めたのが、ビヘイビアツリーと呼ばれる手法です（図C）。これはAIが実行するタスクを記述するシステムで、コンポジットタスク（合成タスク）とプリミティブタスク（原初タスク）という2種類のタスクを使って振る舞いを記述していきます。FSMで状態間の遷移として記述していた制御ロジックが、ビヘイビアツリーではコンポジットタスクに置き換わっています。これにより、状態間遷移の増加を抑制しようというアイデアです。

## ゲームエンジン

ゲームを開発するために必要なソフトウェア、ツール、描画システムが揃った開発パッケージのことです。テクノロジーの進歩に伴ってゲームは高度化し、ゲームを構成する最低限の要素を準備するだけでも膨大な労力が必要とされるようになりました。そのため、現代のゲームは基本的に「ゲームエンジン」と呼ばれる一定の開発環境を使って、開発・制作されています。有名なものでUnreal Engine、Unity、Godot Engineなどの商用エンジン、社内のみで使用する内製エンジンがあります。



## ドゥリス・ジャンフランソワ Jean-Francois Durris

フランス出身で、ゲーム業界で15年の経験を持つゲームプレイ/AIプログラマー。『Far Cry 4』『Dishonored 2』『Deathloop』などのAAAタイトルに携わり、ゲームプレイの仕組みやキャラクターの能力を実装してきた。専門はキャラクターのナビゲーションで、NPCがジャンプしたり、梯子を使ったり、泳いだり、ドアを使ったりできるシステムを開発。現在は、未発表タイトルで使用する自社製3Dナビゲーションミドルウェアの開発を推進している。

## キャラクターの定義

新しいロールプレイング ゲームを手に入れたところを想像してください。ゲームパッドを握りゲームを起動して、世界の探索を始めます。とある街から冒険が始まり、あなたは物語の主人公である「ヒーロー」を操作します。街を探索すると、村人、商人、動物、衛兵、そして魔王を退治せよと命令する王様など、さまざまな人たちに会います。任務を果たすために町の外に出ると、モンスターに遭遇し、倒す必要があります！

このような定番のシーンには、さまざまなキャラクターが登場しますが、大きく2つに分類することができます。1つは主人公など、プレイヤーであるあなたが操作するキャラクターです。これはよく「**プレイヤーキャラクター**」と呼ばれています。一方、「ゲーム」にコントロールされているキャラクターはプレイヤーがコントロールできないため、「**NPC**」(Non-Player Characters)と呼ばれています。

あなたが操作するプレイヤーキャラクターは、歩く/走る/ジャンプする/物を拾う/攻撃する/防御するなど、ボタン1つでさまざまなアクションを実行できます。それぞれのアクションは、プレイヤーであるあなたが画面上の状況を見て理解し、自身で決定します。例えば、村人を見かけたら話しかけ、たどり着けなさそうな場所に宝箱があれば、周囲を見渡して環境を確認し、たどり着く方法を探します。モンスターと戦って、ファイヤーボールが迫ってきたら、それを避けようとするでしょう。これらの行動(つまり「何を」するか)は、すべてあなたが操作するキャラクターが実行することです。これらの行動を「いつ」「どこで」「どのように」「なぜ」発動させるかは、コントローラを持っているあなた(人間)が決めます。状況を分析し、自分の判断でボタンを押して実行させるのです。

一方、NPCはどうでしょうか？人間が操作しているわけではないので、必然的に「ゲーム」が支配しているはずですが。実際にNPCの行動を見てみると、制限されているとはいえ、プレイヤーである主人公ができることに近いものがあります。また、シンプルでありながら違和感のない行動をとり、状況に応じて行動を起こします。では、彼らの行動、つまり「何をするか」がプレイヤーと同じなら、「いつ」「どこで」「どのように」「なぜ」はどのように決まるのでしょうか？NPCは人間と同じように、見て、聞いて、環境を理解し、周りで起こっていることに反応できるのでしょうか？

ここでは、NPCが「どのように」移動できるかに焦点を当てて、その仕組みを説明します。まず、NPCの作り方から見ていきましょう！

### 【NPC】

もともとはテーブルトークRPGに由来した言葉で、プレイヤーではなく、ゲームマスターや審判によって操作されるキャラクターのこと。コンピュータゲームが主流になってからは、ゲームマスターの役割をコンピュータが司るようになり、自然とノン・プレイヤー=コンピュータに操作されたキャラクターの意味が変わっていった。

## 1. キャラクターの作成

すでにご存知かもしれませんが、ゲーム内で目に見えるものやアクセスできるものは、すべて「**ポリゴン**（通常は三角形）」で構成されており、「物理的」な要素で作られています。多くの三角形をつなげて「形」を作り、「メッシュ」と呼ばれるものになります。

例えば、洋服を作る場合、細い糸を組み合わせる「網目」を作ります。ゲームの「メッシュ」もこれと同じです。テーブル、木、橋、地面など、ゲームに登場するすべてのものは複数のメッシュで構成されており、主人公やNPCも同様です。そのメッシュは、服や髪もすべて含めて「キャラクターメッシュ」と呼ばれています。武器は一般的にキャラクターメッシュと分けられ、付属された別のメッシュになります。

この時点で完成したものが、色のないポリゴンで構成されている「3Dモデル」になります。キャラクターの形ができたなら、「テクスチャ」（三角形に画像を貼り付けて色などをつけるもの）で色をつけます。これでキャラクターの見た目は良くなりますが、もう少し工夫が必要です。歩く／しゃがむ／ジャンプする／攻撃するなどの動作を実行をさせるには、キャラクターメッシュに「スケルトン」を組み込む必要があります。これにより、スケルトンの周りにあるポリゴンを動かせるようになります。スケルトンを入れたら、最後に「アニメーション」を作成して、ポリゴンを時間経過とともに動かします。これで、キャラクターが動いて見えるようになります！（**図1**）

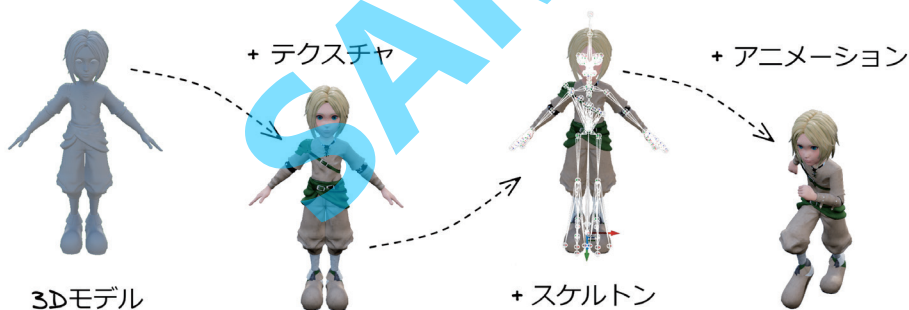


図1 キャラクターができるまで

モデリング（メッシュの作成）、テクスチャリング（テクスチャの作成と貼り付け）、アニメーション（動きの付与）の3ステップが、キャラクター作りの主な工程です。他にもさまざまな手法がありますが、上記のステップは最低限の必須項目になります。ここまでは、プレイヤーキャラクターやNPCを問わず、すべてのキャラクターに共通する作り方です。

このようなキャラクターは、ゲームエンジンでなく専用の3Dモデリング&アニメーションソフトで作成します。テクスチャリングに使う画像も同様で、専用の画像編集ソフトで作成します。キャラクターをゲームで使うためには、これらの外部ソフトからゲームエン

## 【アセット】

一般的な英語では財産や資産のことを指すことが多いが、ゲーム開発ではゲームを構成するファイルのことをアセットと呼んでいる。アセットには「目的達成のために役に立つもの」という意味もあり、普段の開発では、ゲームを動かすために「役に立つ」「貴重なもの」というニュアンスで使っている。

ジンにデータとしてエクスポートする必要があります。ゲームエンジンで読み取れるように、キャラクターのメッシュ情報は三角形のリストとして、テクスチャは色のリストとしてファイルに保存されます。通常、メッシュは1ファイル、テクスチャは1ファイル以上、アニメーションは1ファイル（大量になることもあります）です。ゲーム内で使用されるデータのことを「アセット」と呼びますが、これはキャラクターに限ったことではありません。他にも、サウンドアセット、VFXアセット、AIアセットなどがあります。

キャラクター作成プロセス全体が「オフライン」で行われ、データはゲームが実行される前に作成されるため、静的なものとなります。ゲームではファイルを読み込んでハードウェアのメモリに格納し、そのまま使用するだけです。これに対して「リアルタイム」（または「ランタイム」）作成は、ゲームの実行中に動的にデータを作成することを指します。例えば、周りの環境に応じてその場でアニメーションを作成する「プロシージャルアニメーション」があります（詳しくは、森寅嘉さんの章「キャラクターAIとアニメーション」をご覧ください）。

## 2. 操り人形師

私たちのキャラクターは、現時点で一連のファイルとして存在しています。これらを使って、キャラクターをゲーム「エンジン」に入れましょう。ここでエンジンと呼ばれているものは、画像やメッシュを読み込んだり、音を鳴らしたり、ネットワークで情報を送ったり、画面に何かを描画したり、ゲームコントローラを扱ったりと、いろいろなことができる特殊なソフトウェアです。エンジンには、ゲームを作るために必要なものがすべて備わっています。

キャラクターメッシュについては、事前に作成された一連のファイルを読み込み、キャラクターのポリゴンとテクスチャと一緒に画面に描画します。ゲームエンジンには、重力をシミュレートする「物理システム」も搭載されているため、キャラクターは地面の上にいることになります！ そのキャラクターを主役にする場合は、ゲームパッドのボタンとアニメーションを関連付けます。例えば、ジョイスティックを前に倒すと、キャラクターがその方向に歩くアニメーションをゲームエンジンが「再生」します。大まかではありますが、これがゲーム内でキャラクターを動かす基本的な方法になります！

ジョイスティックで操作する直前まで、プレイヤーキャラクターとNPCに違いはありません。では、NPCでジョイスティックの代わりとなるものは何でしょうか？

現在のキャラクターは、物理的な形に関するデータの断片に過ぎず、意思決定の能力が備わっていません。手足や頭はあっても、自分で動いたり、環境を認識したり、考えたりすることのできない、ただの人形です。この操り人形は、与えられた指示にしか反応しません。人間がすべてのNPCをコントロールすることはできないので、NPCに自律性を与えなければいけません。そのためには、「データ」を操作する「ロジック」を追加する必要があります。

これは「プログラミング」と呼ばれ、あるデータをプログラム（ロジック）に与えると、そのプログラムが何らかのアクションを起こしたり、新しいデータを生成したりします。つまり、正しいロジックが実装されていれば、人形に知能を持たせることができるわけです。ここで、AIエンジニアが活躍します。私たちは、「キャラクターに生命を吹き込む操り人形師」なのです！

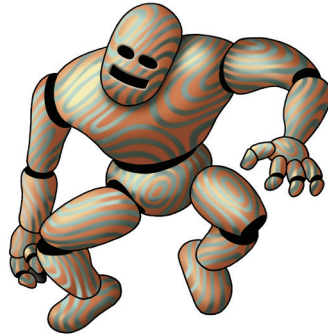


図2 AIエンジニアは操り人形師？  
©ARMOR PROJECT/BIRD STUDIO/SQUARE ENIX

では、脳はどのように働くのでしょうか？ といっても、これは非常に複雑なトピックなので、本書に関連する部分のみ説明しましょう。人間として、何かを聞いたり、見たり、感じたりするのは、一般的に「刺激を受ける」と表現できます。いわば、これらの刺激は、反応を生み出すために脳が処理しなければならない「データ」です。私たちは突然大きな音が聞こえたらびっくりするし、かわいいものを見れば微笑みます。足の指をテーブルにぶつければ、痛みを感じ、飛び跳ね、もしかしたら泣くかもしれません！これが「作用・反作用」のメカニズムです。

ゲームの中で置き換えるなら、「入力」（周辺にあるさまざまな種類のデータ）を抽出して何らかのロジックで処理し、それに対して適切なアクションを「出力」する、というものです。これは人間がやっていることを非常に単純化したものですが、NPCに**知的な印象**を与えるには十分であり、それがゲームにおける「AI」の本質です（AIがどのように入力を処理して出力を作るかは、並木さんの章「プランニングを用いた意思決定システム」をご覧ください）。

### 3. データ、データ、データ！

NPCをより賢く見せようと思えば思うほど、より多くの入力データが必要になります。そして、多くの入力を処理するには、より多くの処理能力が必要になりますが、それによって、NPCが行えるアクション（出力）の数が増えます。

例えば、ゲームの中でNPCが自分の行動にどう反応するか試すとしましょう。もしNPCを突き飛ばしたり、進路にずっと立ちふさがったりしたら、NPCは怒るでしょうか？ 人間であれば怒るでしょうが、ゲームのNPCにそのような行動を何時間続けても怒りません。あなたの周りを回るか、その場に留まろうとするだけです。なぜでしょうか？

まず、NPCが「怒る」ようにできていないからです（この反応はゲームにプログラムされていません）。キャラクターのタイプによっては、怒鳴ったり、泣いたり、逃げたり、あるいは喧嘩を始めたりと、違う反応を求められるかもしれません。単純な刺激を扱うだけで

**【知的な印象】**  
一般に、ゲームAIはなにかの動きをしないと、それに知性があることがプレイヤーにはわからない。とても賢いが座っているだけで何もしないNPCは、プレイヤーから見ると、見た目だけのモデルデータと見分けがつかない。

そこで、プレイヤーが持ち上げて動かそうとした時に「うわっ、何ですか!？」と叫ぶようなリアクションを加えると、プレイヤーは、このNPCが知性ある存在だと認識できるようになる。



も、さまざまな反応を引き起こす可能性があり、それを現実的に見せるのは難しく、NPCの行動全体の複雑さが増します。NPCが逃げると決めたら、どこへ行くべきでしょうか？ 自分の家？ それとも、警備員の居場所？ 状況やキャラクターの好みによって反応は様々でしょうが、私たちにとって入力データはすべて同じです。

次に、NPCはプレイヤーが嫌がらせをしていることを「理解」しません。NPCが知っているのは、「行く手に障害物がある」ということなので、それを避けようとするだけです。NPCが「怒る」ためには、プレイヤーの意図を分析しなければなりません、それを察知させるのはかなり難しいです。PART3で説明するメタAIは、このような状況から情報を引き出すことを可能とします！

この例からわかるように、NPCをそれらしく行動させるには、世界に関する多くの情報を抽出してNPCに送り、彼らが正しい判断を下せるようにする必要があります。人間の判断のほとんどは、五感から得た知識に基づいています。見たり、聞いたり、感じたり、触ったり、嗅いだりするものは、周囲の環境に関する情報となり、私たちはそれに基づいて判断を下します。ですから、もし、NPCに私たちと同じような反応をさせたいのであれば、何らかの方法で感覚を与え、周囲の空間から情報を引き出せるようにする必要があります。次は、NPCにどのように空間感覚を与えるか、考えてみましょう！

### 開発日記：レベル1の冒険者

AI研究者 A

- 20XX年某月某日 春先にスクウェア・エニックスに転職した。  
異業種からの転職であったため、ゲーム開発者レベルが1の状態の自分が、ラスボス級のゲーム会社に入って良いのだろうか？ という不安もあったものの、なぜか入ってしまった。
- 20XX年某月某日 入社後オフィスで業務中、あの有名ゲームタイトルや、聞き覚えのあるゲームクリエイターの名前を耳にはさむことがあり、自分としては、まるでゲーム序盤で間違えて最終ダンジョンに入ってしまったような気分である。
- 20XX年某月某日 考えてみれば当たり前のことではあるが、ゲーム会社とはいえ、すべての社員がゲーム開発をするわけではない。腕に覚えがあれば、他の分野の技術であっても、ゲーム会社に入社するチャンスがあるということなのだ。