

「デジタルゲームにおける人工知能技術・ プロシージャル技術入門」

三宅 陽一郎

y.m.4160@gmail.com

Twitter: miyayou

2010.10.28

東京工芸大学



自己紹介

1999年京都大学総合人間学部基礎科学科卒業.

2001年大阪大学理学研究科修士課程物理学専攻修了.

2004年東京大学工学系研究科博士課程(単位取得満期退学)

同年、株式会社フロム・ソフトウェア入社.

「デジタルゲームにおける人工知能の研究・開発」

Contact Information

Youichiro Miyake

- Mail: y.m.4160@gmail.com
- Twitter: @miyayou
- Blog: <http://blogai.igda.jp>
- LinkedIn: <http://www.linkedin.com/in/miyayou>
- Facebook: <http://www.facebook.com/youichiro.miyake>

[講演]

全講演資料を公開しています。

2006年

CEDEC2006

「クロムハウズにおける人工知能開発から見るゲームAIの展望」

<http://blogai.igda.jp/article/33936286.html>

2007年

AOGC2007 招待講演「人工知能が拓くオンラインゲームの可能性」

<http://www.bba.or.jp/AOGC2007/080/>

CEDEC2007 招待講演「エージェント・アーキテクチャーから作るキャラクターAI」

<http://blogai.igda.jp/article/33936286.html>

Korea Game Conference 2007招待講演

2006 年～2007年

IGDA日本、ゲームAI連続セミナー「ゲームAIを読み解く」全6回

<http://blogai.igda.jp/article/33936286.html>

2008年

CEDEC2008 招待講演「ゲーム開発のためのプロシージャル技術の応用」

http://cedec.cesa.or.jp/2008/archives/archive_1.html

DiGRA JAPAN 公開講座「Spore におけるゲームAI技術とプロシージャル」

<http://www.digrajapan.org/modules/mydownloads/viewcat.php?cid=10>

IGDA日本 GDC報告会「GDCに見る最新AIとプロシージャル技術」

<http://blogai.igda.jp/article/33936286.html>

2009年

IGDA日本 GDC報告会「これからのゲームAIの作り方」

<http://www.digrajapan.org/modules/mydownloads/>

自己紹介

[特別論文]

人工知能学会誌 Vol. 23 No. 1 (2008 年 1 月) 「ゲームAI特集」
「デジタルゲームにおける人工知能技術の応用」(三宅)

[報告書]

デジタルコンテンツ協会

2007年度 第3章「ゲームAI」

「デジタルコンテンツ制作の先端技術応用に関する調査研究報告書」

2008年度 第3章「プログラミングAI」

「デジタルコンテンツ制作の先端技術応用に関する調査研究報告書」

ゲームAIの情報について、総合的にまとめてあります。

<http://www.dcaj.org/report/index.html> よりPDFダウンロードできます。

[インタビュー]

大学からゲームメーカーへ——AI研究で広がるステキなゲームの世界とは？

http://gamez.itmedia.co.jp/games/articles/0901/08/news129_3.html

<http://gamez.itmedia.co.jp/games/articles/0901/09/news075.html>

(ブログ) y_miyake のゲームAI千夜一夜 (IGDA日本)

<http://blogai.igda.jp/>

ゲームAIセミナー群

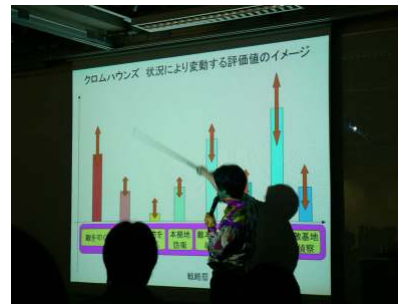
期	回	タイトル	日	講師	リンク
第1期	87	ラッキョ王国のアルゴリズム	8/22	WYGIWYG(ワイジワイ)	返す
	86	デモンストラのAI	8/8	アソシエーション	返す 返す
	85	ゲームAIのあつちの組	8/1	ゲームシステムA	返す
	84	ボードゲームの解法アルゴリズムと3DアクションゲームのAI	7/25	モナリザ	返す
	83	ゲーム空間における最適化と進化	7/18	ゲーム空間における最適化	返す
	82	生命のコーディング	7/11	ゲームのコーディング	返す
	81	日本のAIの産業を考える	7/4	ゲームAIのアーティファクト	返す
第2期	80	同僚は皆機械	6/20	同僚は皆機械	返す
	79	801 アーキテクチャー	6/13	801 アーキテクチャー	返す
	78	Synthetic Creature	6/6	Synthetic Creature	返す
	77	ノバール「AI」コミュニケーション	5/30	ノバール「AI」コミュニケーション	返す
	76	群知能とレベルデザイン	5/23	群知能	返す
	75	階層型システム/自動生成	5/16	階層型システム/自動生成	返す
	74	クロムハウズにおけるチームAI	5/9	7/9 チームAI	返す 返す
	73	異端モデル	5/2	異端モデル	返す
	72	Halo2における階層化バリエーション	4/25	階層化バリエーション	返す
	71	アスロノ「AI」における最適化アルゴリズム	4/18	最適化アルゴリズム	返す
第3期	70	Forza Motorsport 2 におけるベイズアンテナワーク	4/11	ベイズアンテナワーク	返す
	69	Thinking about Armored Core (VI) [アーモードコアのAIの最適化を考える]	4/4	最適化に関する知識	返す
	68	ロケッパにおける人工知能 (Review & Perspective 8)	3/28	ローによるマルチエージェント	返す
	67	Thinking about Armored Core (IV) [アーモードコアのサブシステム/アーキテクチャー]	3/14	サブシステム/アーキテクチャー	返す
	66	レベルデザイン	3/8	レベルデザイン	返す
	65	Thinking about Armored Core (III) [アーモードコア AI の最適化]	2/28	円柱を基本とする世界表現	返す 返す
	64	人工知能が「オンラインゲーム」の可能性	2/21	マルチエージェント	返す
	63	Thinking about Armored Core (II) [アーモードコアのAIの最適化]	2/14	行動の最適化	返す
	62	F.E.A.R. におけるゴール指向型AIの最適化	2/7	プランニング	返す
	61	Thinking about Armored Core			
第4期	60	データマイニング			
	59	進化と生態の仕組み			
	58	対戦レビュー (Review & Perspective)			
第5期	57	Killzone におけるNPCの行動最適化			



社内セミナー
(週1回/～190回)

FROM SOFTWARE

- 第4回 Halo2 有限状態機械
- 第3回 ChromeHounds マルチエージェント
- 第2回 F.E.A.R プランニング
- 第1回 Killzone 知識表現(世界表現)



IGDAゲームAI連続セミナー
(2ヶ月1回/全6回)

igda international game developers association Japan chapter



日本最大級のゲーム開発者カンファレンス
CEDEC 2007
CESA DEVELOPERS CONFERENCE

研究・勉強の仕方

- (1) 言葉を大切にする
- (2) コツコツと知識を積み重ねる
- (3) 小さなことを積み重ねていると、他人には追いつけないとんでもないところへ行けてしまう。

(By イチロー)

本講演の主旨

- ① 「デジタルゲーム」を人工知能から眺める。
- ② 過去から現在、そして未来へ向かう
「デジタルゲームの流れ」を体感する。

本授業の指針

4つの言葉を覚えて帰りましょう

- (1) 自律型エージェント (Autonomous Agent)
- (2) プロシージャル (Procedural)
- (3) メタAI (Meta-AI)
- (4) マルチエージェント (Multi-agent)

ここから眺めると、「デジタルゲームのAI」がよく見えます。
勉強するときに大切なのは、言葉を大切にして、
一日の中で一つ一つ覚えて行くことです。

コンテンツ

第1部 ゲームとは何か？

第2部 デジタルゲームにおける人工知能技術

4つの言葉を覚えて帰りましょう

第一章 自律型エージェント (Autonomous Agent)

第二章 プロシージャル (Procedural)

第三章 メタAI (Meta-AI)

第四章 マルチエージェント (Multi-agent)

質疑応答

コンテンツ

第1部 ゲームとは何か？

第2部 デジタルゲームにおける人工知能技術

第一章 自律型エージェント (Autonomous Agent)

第二章 プロシージャル (Procedural)

第三章 メタAI (Meta-AI)

第四章 マルチエージェント (Multi-agent)

質疑応答

第1部 ゲームとは何か？

ゲームとは何か？

ゲームとは何でしょうか？

考えたことがありますか？

今日は一緒に考えてみましょう。

ゲームの分類

ゲームには
どんな種類のものがあるのでしょうか？

ゲームの種類

	ゲームの種類	詳細な分類	さらに詳細な分類
①	アナログゲーム	カードゲーム ボードゲーム ギミックゲーム	ウォーゲーム 積み木ゲーム シミュレーションゲーム トレーディングカードゲーム



カタン



スリードラゴンアンティ



人狼



？



ゲームの種類

	ゲームの種類	詳細な分類	さらに詳細な分類
②	デジタルゲーム	商業ゲーム インディーズ・ゲーム シリアスゲーム カジュアルゲーム ハードコアゲーム オンラインゲーム オフラインゲーム	カードゲーム シミュレーションゲーム アクションゲーム 戦略ゲーム アドベンチャーゲーム 恋愛シミュレーション・ゲーム パズルゲーム(落ちゲー) コンストラクション・ゲーム RPGゲーム スポーツゲーム 音ゲーム



シムシティ4
(コンストラクション)



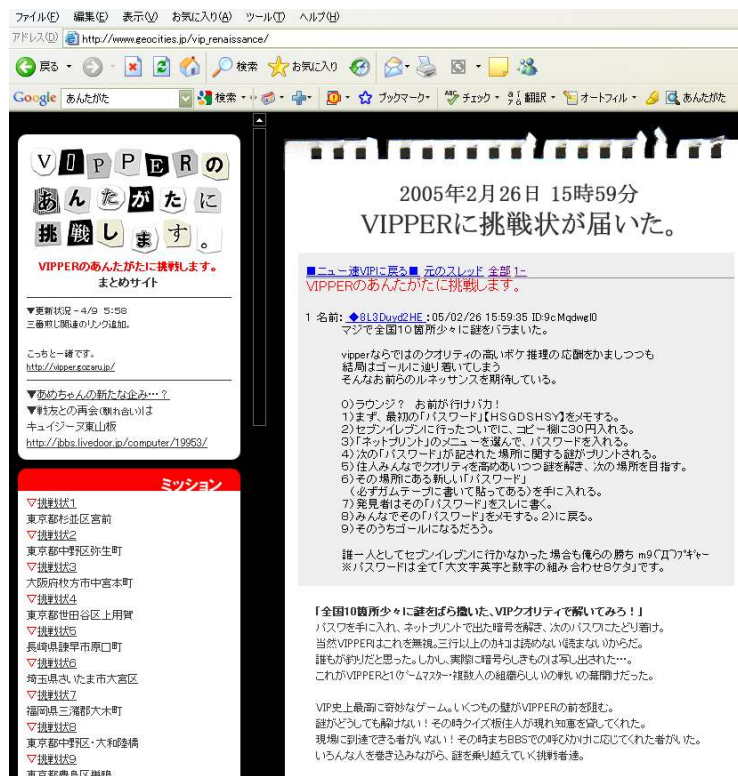
ハーツオブアイアン2
(シミュレーション)



パラッパラッパー(音ゲー)

ゲームの種類

	ゲームの種類	詳細な分類	さらに詳細な分類
③	代替現実ゲーム (ARG)	ネット カード 現実の施設 ファックス	Perplex City I love bees あんたがた



ネット、架空サイト、2ch、
TV番組、電話、ファックス、
セブンイレブンのサービスなど、
電話ボックス、看板、など、
現実にあるギミックを使用して、
ヒントを点在させながら、
謎解きをさせるゲーム

	ゲームの種類	詳細な分類	さらに詳細な分類
①	アナログゲーム	カードゲーム ボードゲーム ギミックゲーム	ウォーゲーム 積み木ゲーム シミュレーションゲーム トレーディングカードゲーム
②	デジタルゲーム	商業ゲーム インディーズ・ゲーム シリアスゲーム カジュアルゲーム ハードコアゲーム オンラインゲーム オフラインゲーム	カードゲーム シミュレーションゲーム アクションゲーム 戦略ゲーム アドベンチャーゲーム 恋愛シミュレーション・ゲーム パズルゲーム(落ちゲー) コンストラクション・ゲーム RPGゲーム スポーツゲーム 音ゲーム
③	代替現実ゲーム (ARG)	ネット カード 現実の施設 ファックス	Perplex City I love bees あんたがた
④	スポーツ	球技・アスリート	サッカー、走り高跳び...
⑤	伝統的ゲーム	身体を使う	おにごっこ、かくれんぼ、 缶蹴り、タッチおに

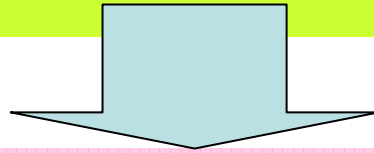
	ゲームの種類	詳細な分類	さらに詳細な分類
①	アナログゲーム	カードゲーム ボードゲーム ギミックゲーム	ウォーゲーム 積み木ゲーム シミュレーションゲーム トレーディングカードゲーム
		商業ゲーム	

ゲームには、実にいろいろな種類のゲームがある。

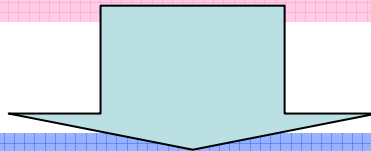
③	代替現実ゲーム (ARG)	ネット カード 現実の施設 ファックス	Perplex City I love bees あんたがた
④	スポーツ	球技・アスリート	サッカー、走り高跳び...
⑤	伝統的ゲーム	身体を使う	おにごっか、かくれんぼ、 缶蹴り、タッチおに

ゲームを定義できるか？

① ルール ② プレイヤー ③ フィールド
...



我々はゲームという概念をよく知っているつもりでいる。
しかし、厳密にゲームを定義することは、
それほど簡単なことではない。



ゲームという広大なフィールドは
研究するに値する深さと広がりを持っている

ゲームを研究する(研究者)



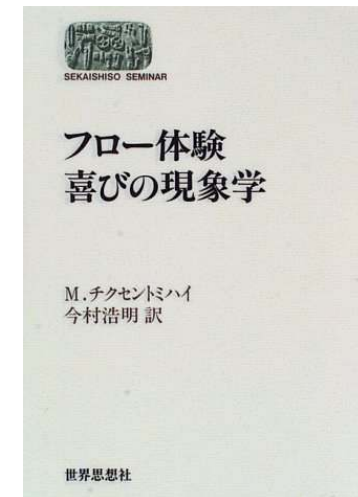
- 『遊びと人間』
ロジェ＝カイヨワ



『Rules of Play』
Katie Salen,
Eric Zimmerman



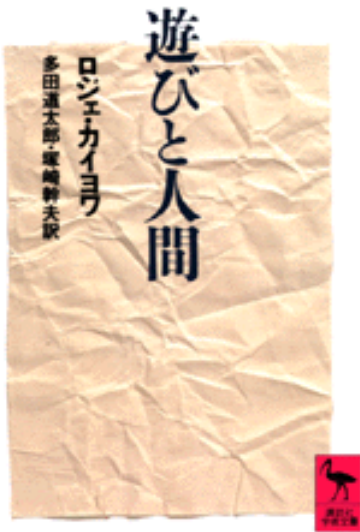
『Ludology』
(ゲーム学)
Gonzalo Frasca



『フロー体験』
チクセントミハイ



ゲームを研究する(研究者)



『遊びと人間』
ロジェ＝カイヨワ
(1958)

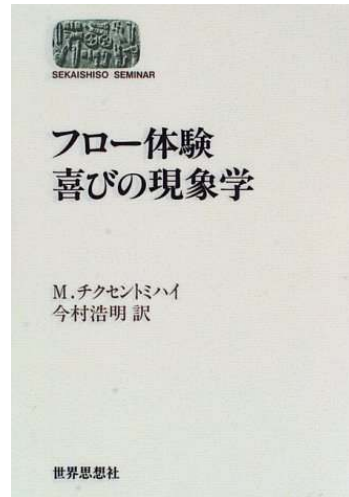


遊びの快楽を4つに分類

アゴーン(競争)、アレア(偶然)、
ミミクリ(模倣)、インクリンクス(賭)

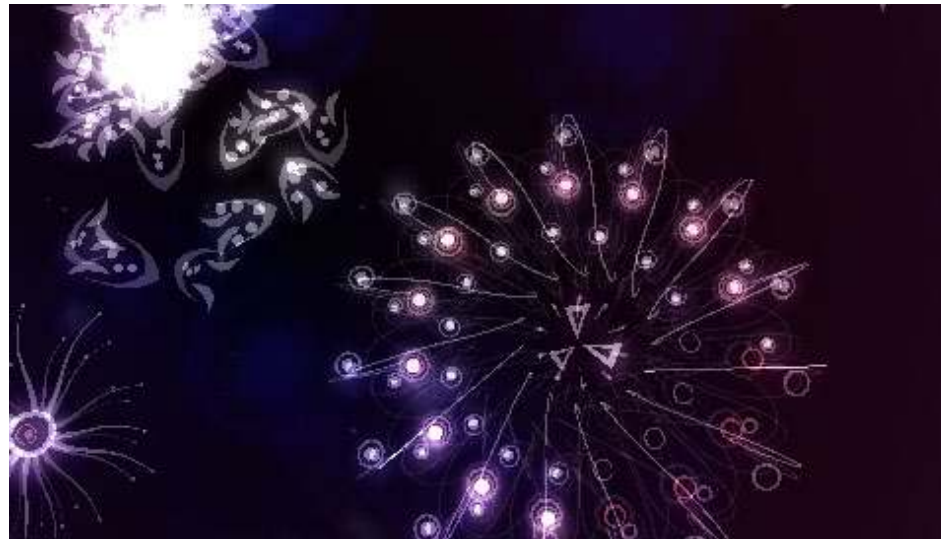
人類学から『遊び』を研究する

ゲームを研究する(研究者)



人が心地よくなる体験とは何か？（フロー体験）

『フロー体験』
チクセントミハイ



<http://game.watch.impress.co.jp/docs/20080410/ps3on.htm>

ゲームを研究する(研究者)



『Ludology』
(ゲーム学)
Gonzalo Frasca
(1999)



ゲームを物語論(narratology)ではなく
ゲームそのものとして扱うべき、という考えから、
ゲーム学(ludology)という言葉を作って、
ゲーム学の端緒を作る。

※ludoは、ラテン語で「遊ぶ」(I play)を意味する

ゲームを研究する(研究者)



『Rules of Play』
Katie Salen,
Eric Zimmerman
(2002)



可能性空間

(=ユーザーのアクションがゲーム空間内で
影響・意味のある範囲)

マジックサークル

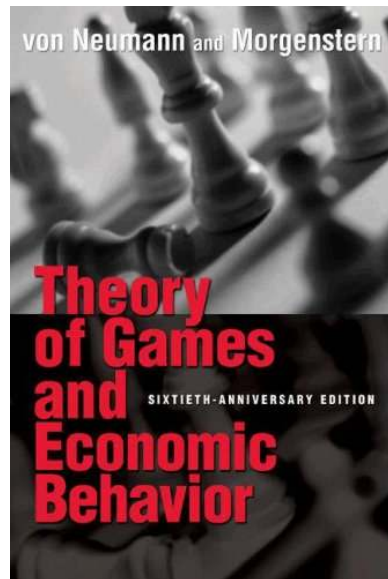
(= ゲームの面白さを味わう行為のループ)

ゲームで研究する

世の中の現象をゲームとして捉えて研究する

経済

フォン・ノイマン、
モルゲンシュタイン



生物進化

メーナード・スミス



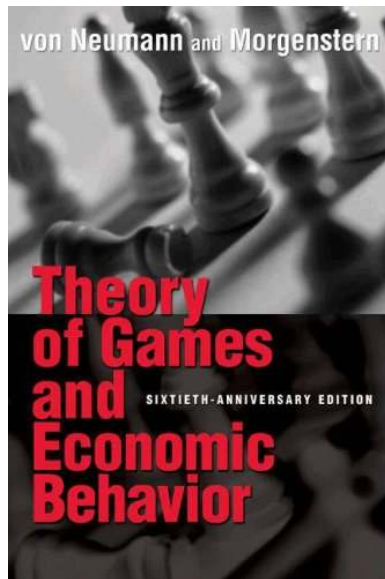
ESS(安定な進化戦略)

ゲームで研究する

世の中の現象をゲームとして捉えて研究する

経済

フォン・ノイマン、
モルゲンシュタイン



経済活動をゲームとして捉えて、
ミニマックス定理など、
ゲーム原理上の定理を用いて、
経済現象を定式化した、革命的な仕事。

ゲームで研究する

世の中の現象をゲームとして捉えて研究する

生物進化

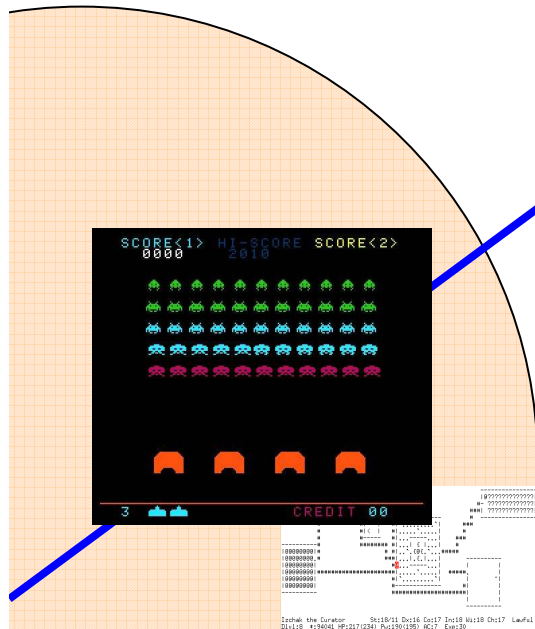
メーナード・スミス



ESS (安定な進化戦略)

進化をゲームとして見立てて、
生物がどのような戦略のもとに進化して
行くかを説く。

ゲームを創作する



[illegible]

ゲームを創作する

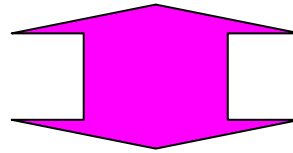
(ゲームはゲームを製作する者によって進化する)



ゲームを創作する

学問としてのゲーム

ゲーム研究者、ジャーナリスト、ゲームファン
「ゲームとはこういうものだ」と定式化する。



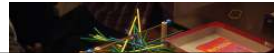
芸術（創作）としてのゲーム

ゲーム開発者

「ゲームとはこういうものだ」と固定観念を壊して、
新しいものを創造して行く

ゲームを創作する

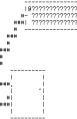
(ゲームはゲームを製作する者によって進化する)



ゲームは進化する。

ゲームを作る人の手によって。
ゲームをプレイする人によって。
ゲームを作り出す社会によって。

ゲーム開発者になるとは、
ゲームの進化の流れの中に身を置いて、
ゲームの進化に貢献する人間となること。



Izchak the Curator St:18/11 Dx:16 Co:17 In:18 Wa:18 Ch:17 Lauf:
Div:18 #194041 HP:217(234) Pu:190(195) BC:7 Ex:10

ゲーム作りはプロ開発者に限らない。
誰が行なってもよい。面白ければ勝ち。

ゲームの未来

ゲームの進化の果てには、
何があるのでしょうか？

...誰もわからない。

100年、200年後のゲームは、
どうなっているのか？

ほっておいても進化しない。
ゲーム開発を続けることで、
過去から未来へつないで行く。

第一章まとめ

- ① ゲームを研究する。
- ② ゲームで研究する。
- ③ ゲームを創作する。

「ゲーム」という概念の中には、人間の知識、生活にとって、奥深い認識、まだ発見されていない機能が隠されている。

たとえ、ゲーム開発者にならなくても、
ゲームから物事を捉えること、
ゲームとして物事を捉えること、
ゲームを実際に作ってみることは、
新しい世界の見方を教えてくれる。

第一部まとめ

- ①デジタルゲームは既に単なるピコピコゲームではない。
30年をかけて、技術と共に進化されて来た。
- ②デジタルゲームは既に科学的/哲学的研究の
対象である。(学会: DiGRA デジタルゲーム学会)
- ③ゲーム開発もそういった研究の成果を取り入れて、高
いレベルでデザインして行く時代になりつつある。

コンテンツ

第1部 ゲームとは何か？

第2部 デジタルゲームにおける人工知能技術

4つの言葉を覚えて帰りましょう

第一章 自律型エージェント (Autonomous Agent)

第二章 プロシージャル (Procedural)

第三章 メタAI (Meta-AI)

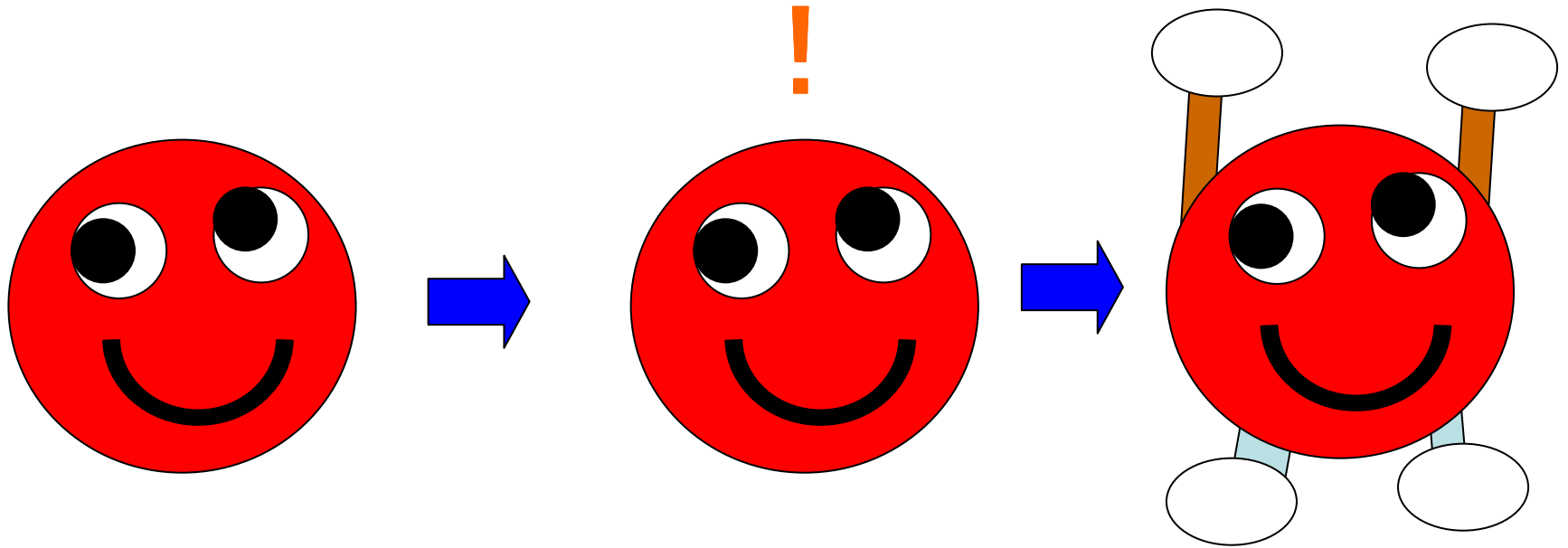
第四章 マルチエージェント (Multi-agent)

質疑応答

第1章 自律型エージェント (Autonomous Agent)

エージェントとは？

- ① 環境に対して情報を集める感覚(センサー)を持つ
- ② 自ら判断する能力を持つ。
- ③ 環境に対して働きかけることができる能力を持つ。

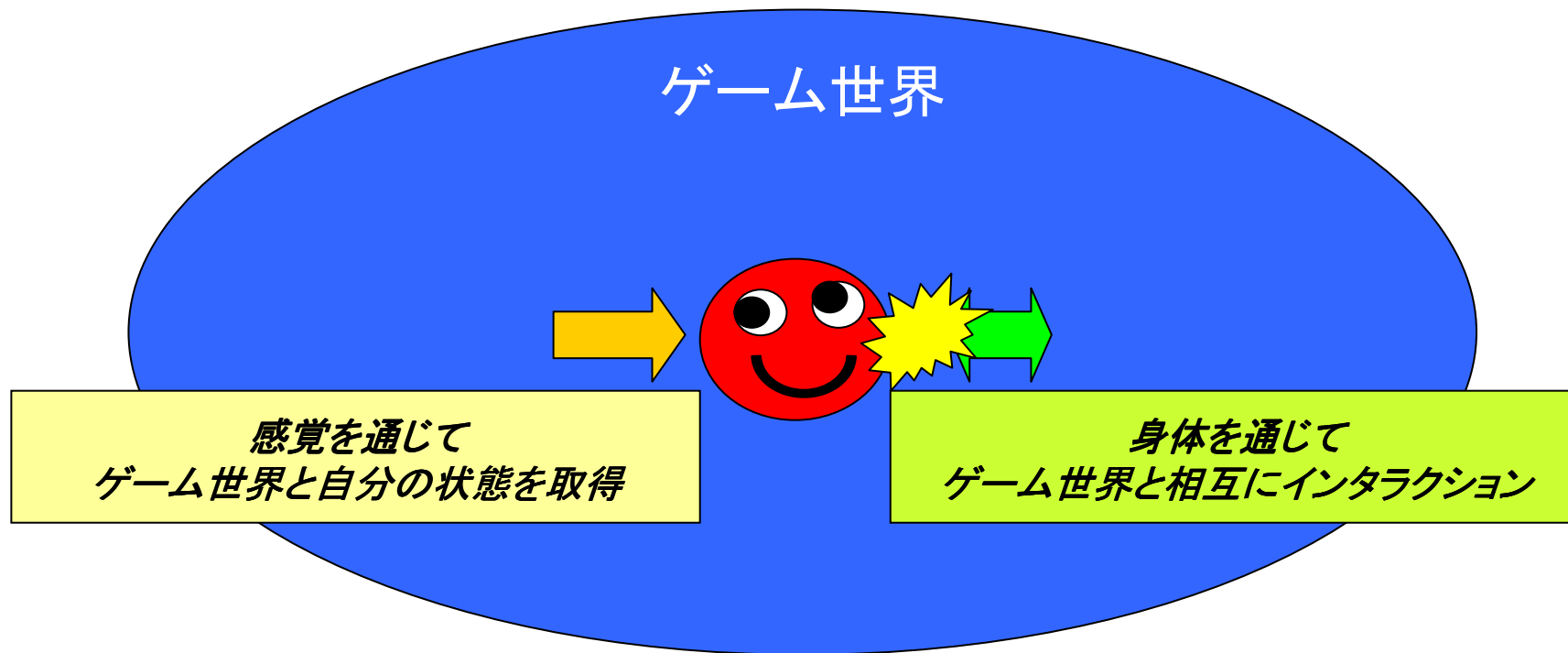


感覚を持ち

自ら判断して

世界に働きかける
能力を持つ

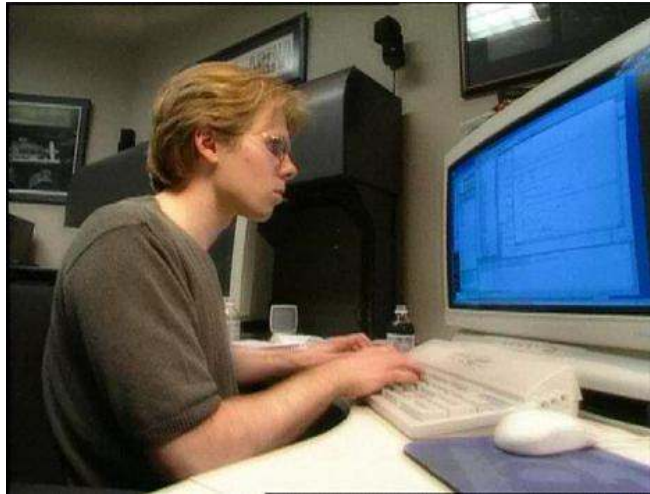
ゲームにおけるエージェント



エージェント = キャラクターAI

エージェントを作ろう！

プログラマーとエージェント

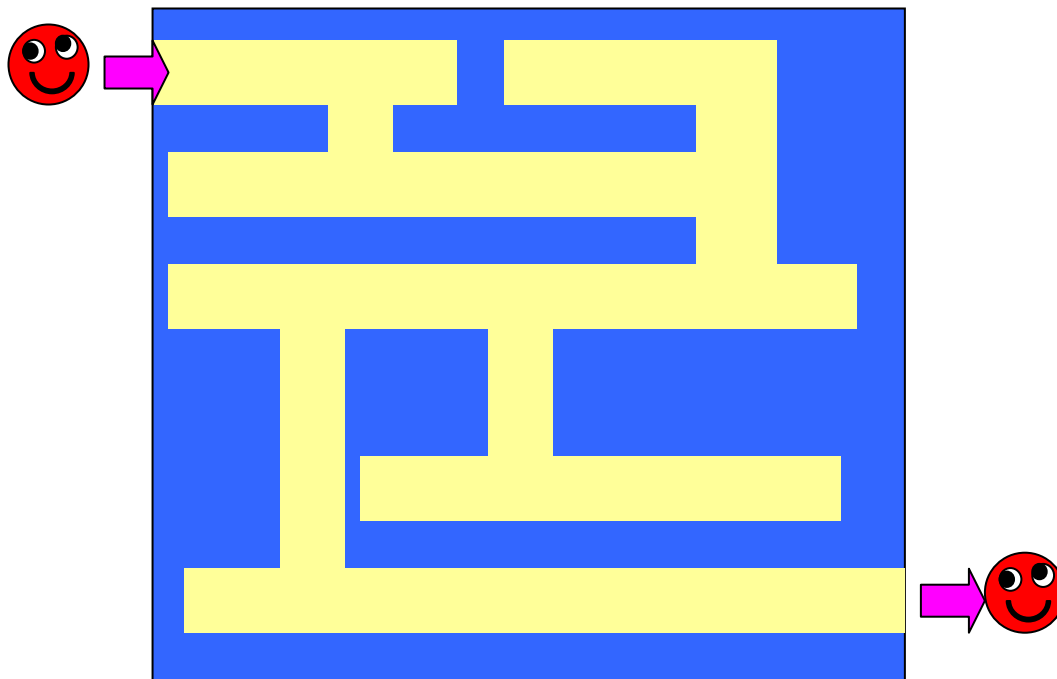


プログラミング



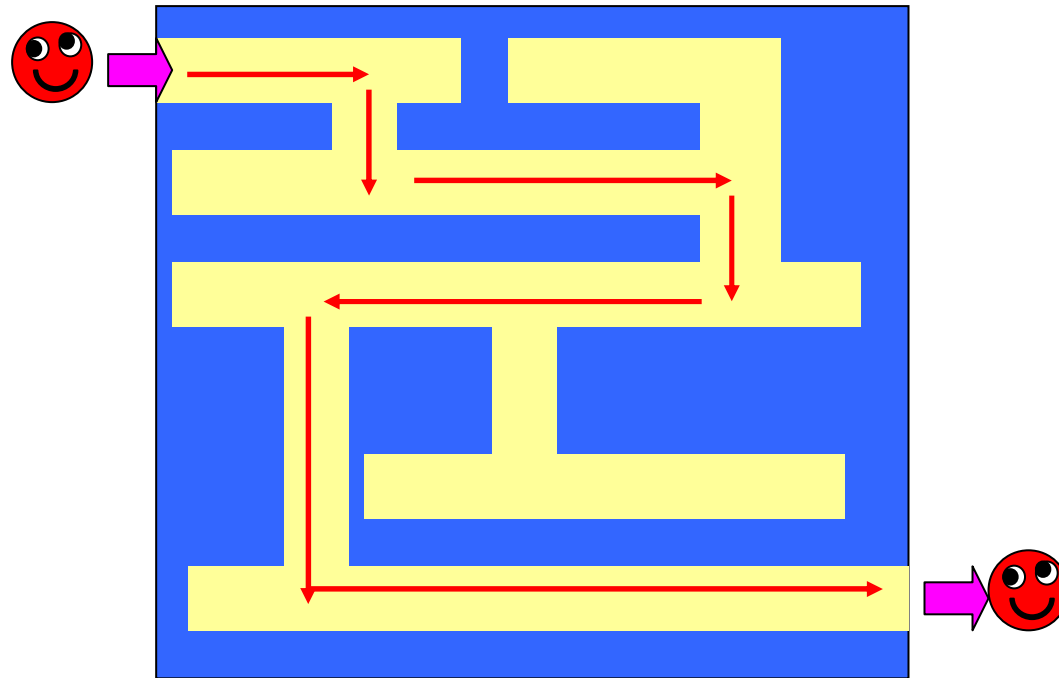
エージェントの自由な移動

Q1: 与えられた迷路を抜けることが出来るAIを考えてみよう！
どのように実装すればよいか？



エージェントの自由な移動

A1: プログラマーが迷路を解いて答えを実装する



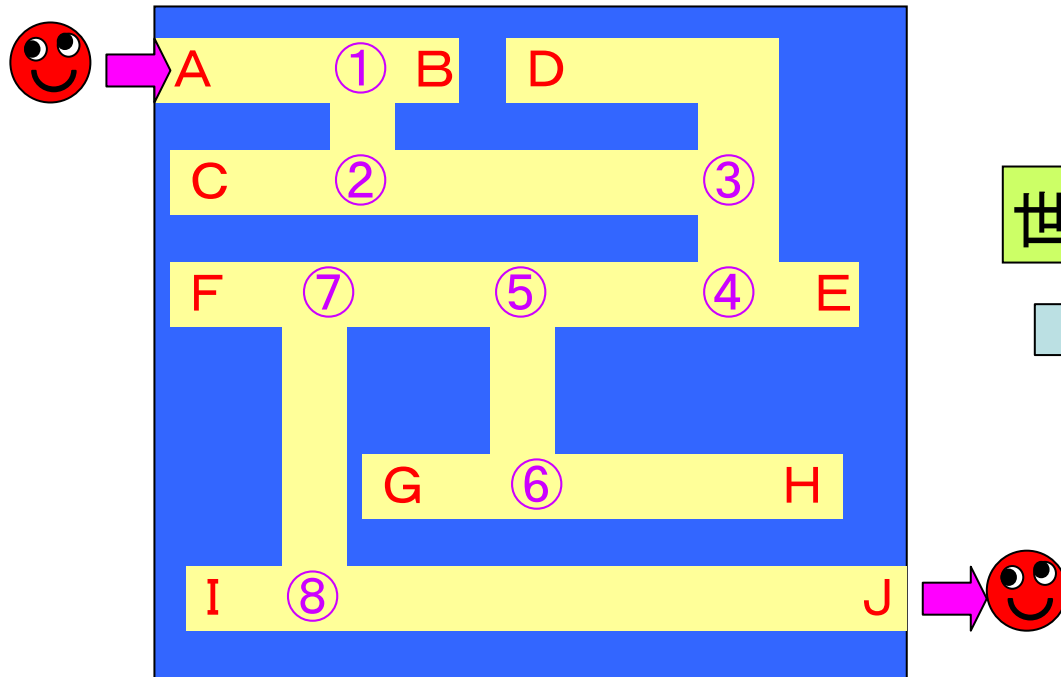
プログラム: ① 5m直進
② 右折
③ 行き当たりまで直
④ 左折
⑤ 行き当たりまで直
⑥ 右折

⑦ 10m直進
⑧ 左折
⑨ 行き当たりまで直進
⑩ 左折
⑪ 直進
⑫ ゴール

エージェントの自由な移動

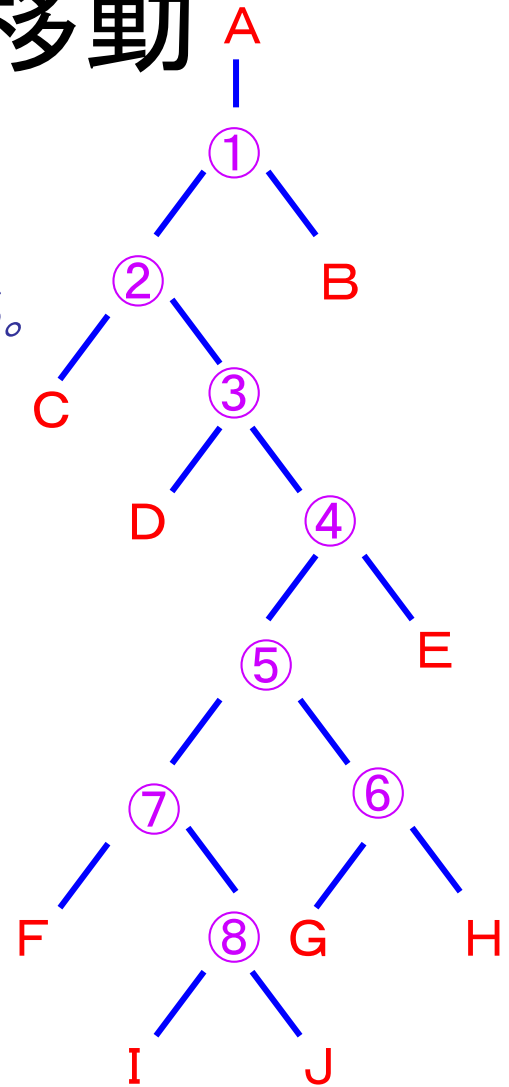
A2: AI自身に迷路を解かせる

「迷路内の移動＝グラフ上の移動」と問題が簡単になる。
グラフ検索のアルゴリズムでAIを動かすことが可能になる。



最大3分岐の合流しない迷路は
2分木(binary tree)と等価である

世界表現

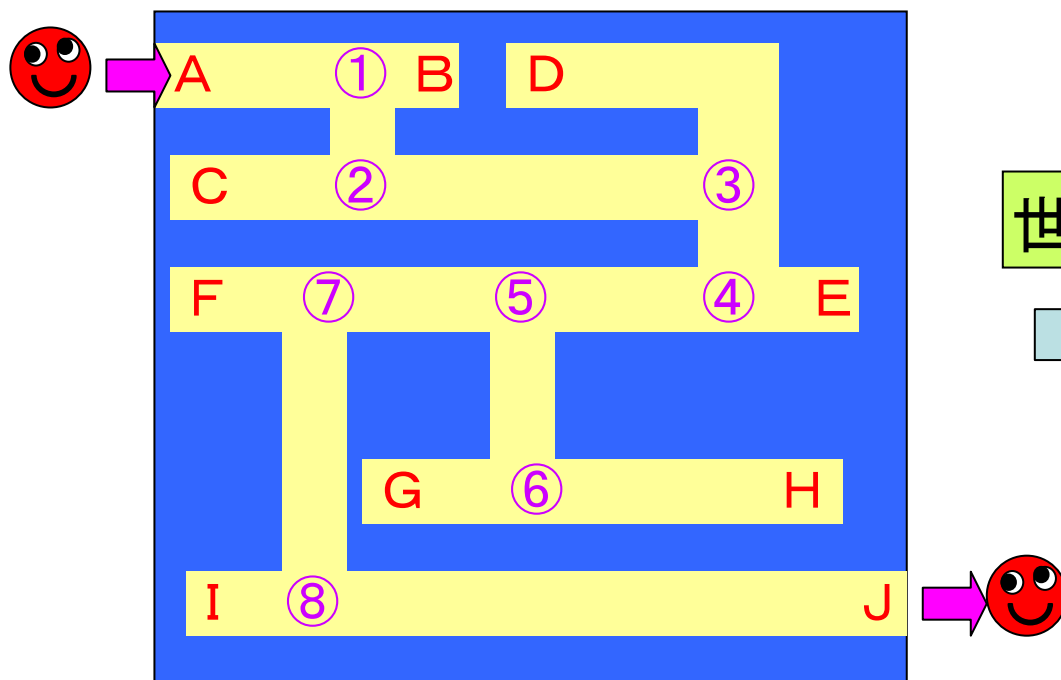


ゲーム世界をAIの制御のために表現したものを世界表現という。

NPCの自由な移動 ①

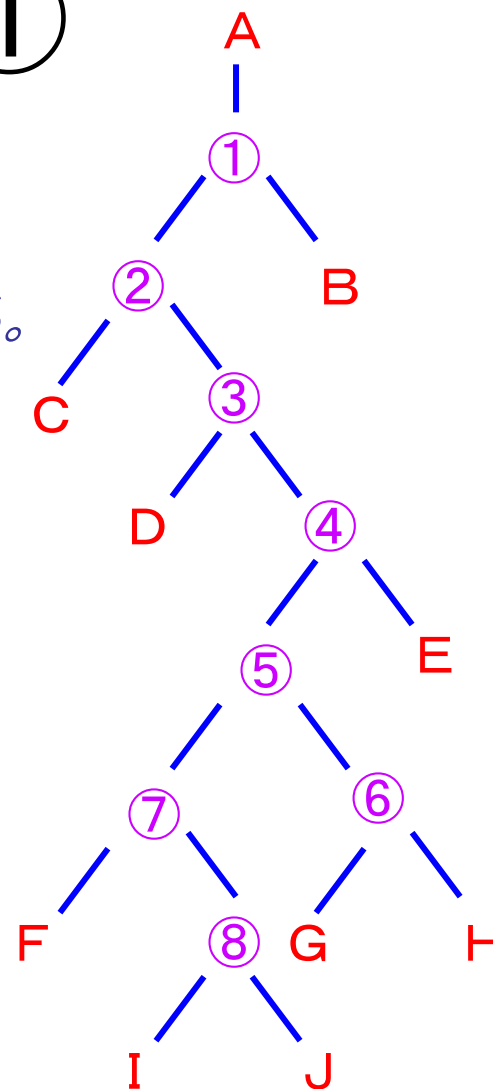
A2: AI自身に迷路を解かせる

「迷路内の移動＝グラフ上の移動」と問題が簡単になる。
グラフ検索のアルゴリズムでAIを動かすことが可能になる。



最大3分岐の合流しない迷路は
2分木(binary tree)と等価である

世界表現



迷路を世界表現(この場合はグラフ表現)して
探索アルゴリズムによって、AI自身に問題を解かせる

Counter Strike

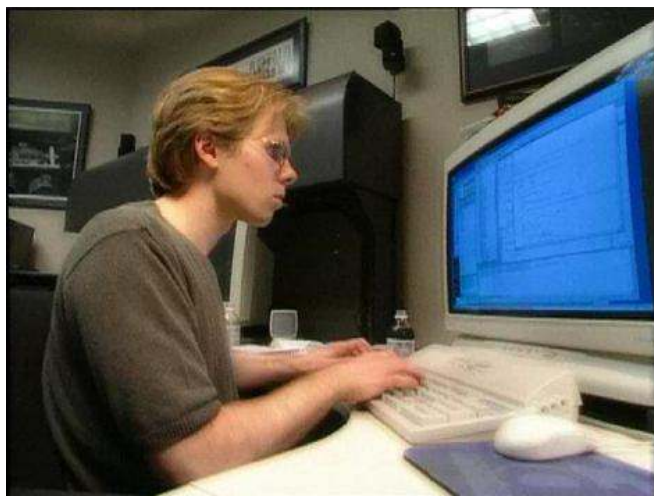


Aaron Earle, "The Making of the Official Counter-Strike Bot" (GDC2004)
<http://www.gdcvault.com>

プログラマーとエージェント

ハードコーディング型実装

プログラマーがステージごとに考えた思考を実装する



プログラミング
実装

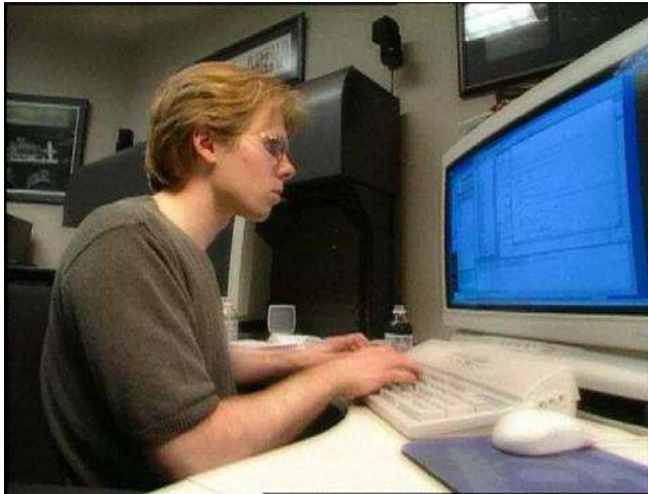


プログラマーが答えを解いて
それをAIに実装する

プログラマーとエージェント

自律型エージェント実装

どんなステージでもエージェント自身が考えて実装する



プログラミング
実装

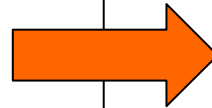


プログラマーはAI自身が問題を
解くように実装する

AI自身が答えを解く

現代のエージェント技術の流れ

ハードコーディング型実装



自律型エージェント実装

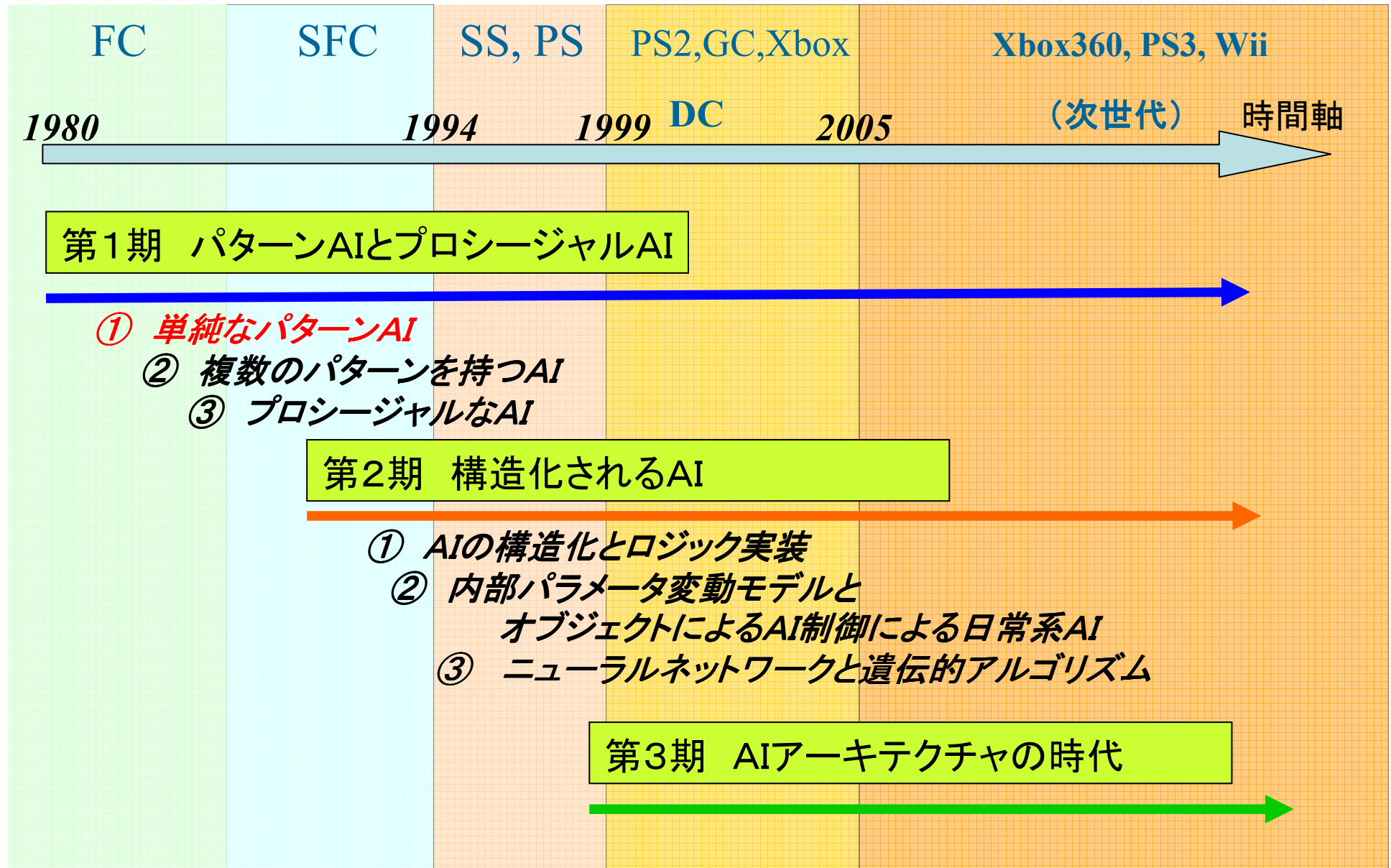
デジタルゲームのエージェントは、
「プログラマーがステージについて考えたことを実装する」方式から、
「エージェント自身に考えさせる」という方式に以降しつつある。

自律型エージェント (autonomous agent)

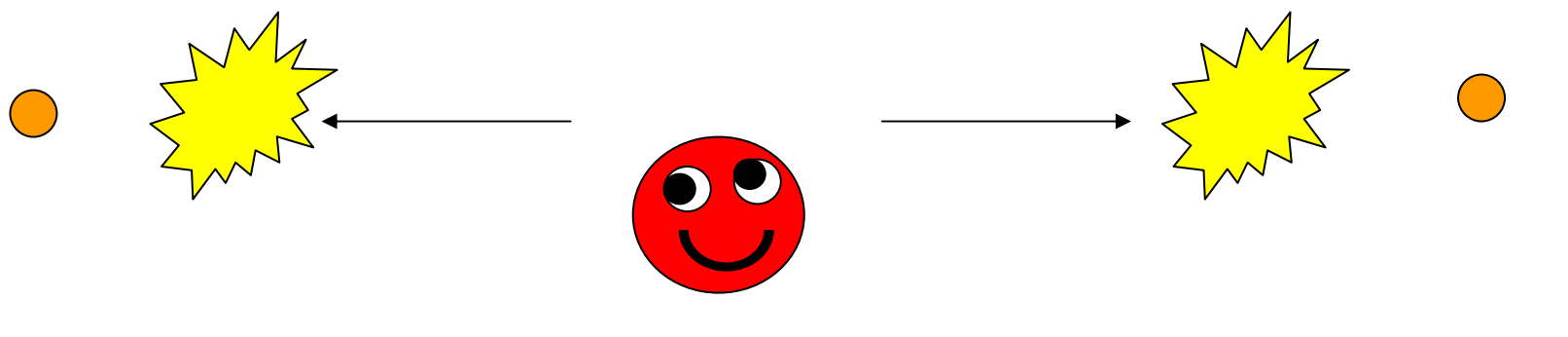
だんだんと AIがプログラマーの
手を離れて自律する

ここでエージェントの実装のされ方の
歴史を追ってみましょう！

ゲームにおけるエージェントの歴史



第1期 ①単純なパターンAI



同じパターンをくり返すだけ

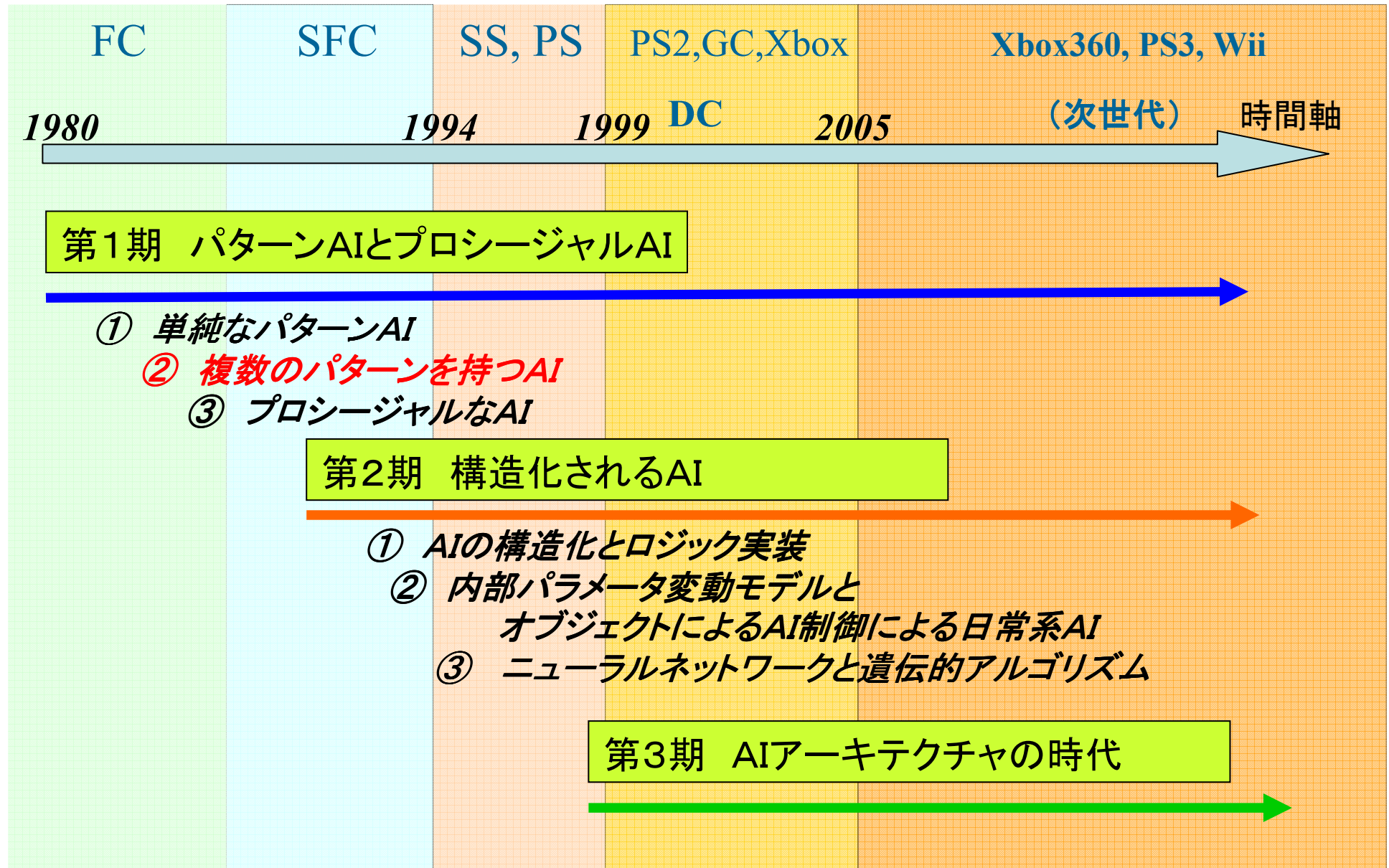
(例) スペースインベーダー (1978)



プレイヤーの動きに関係なく、決められた動きをする

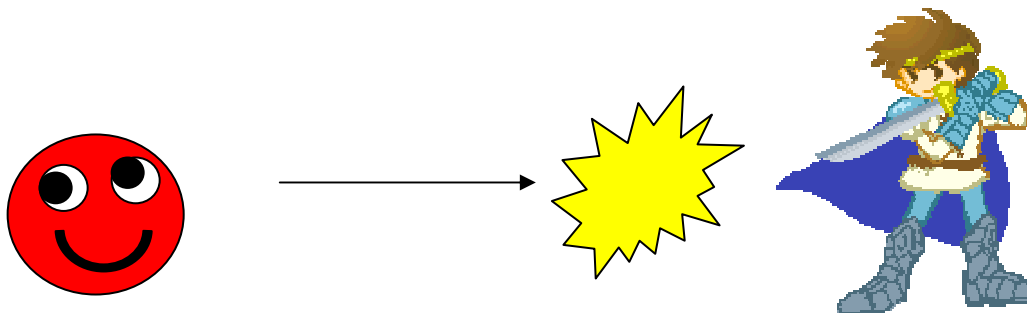
スペースインベーダー

ゲームにおけるエージェントの歴史



第1期 ②複数のパターンを持つAI

Interactive



あらかじめ決められた行動を、
状況によって使いわけAI

「プリンス・オブ・ペルシャ」など、
スプライトアニメーションを用意する必要がある場合、
数パターンに限られる。

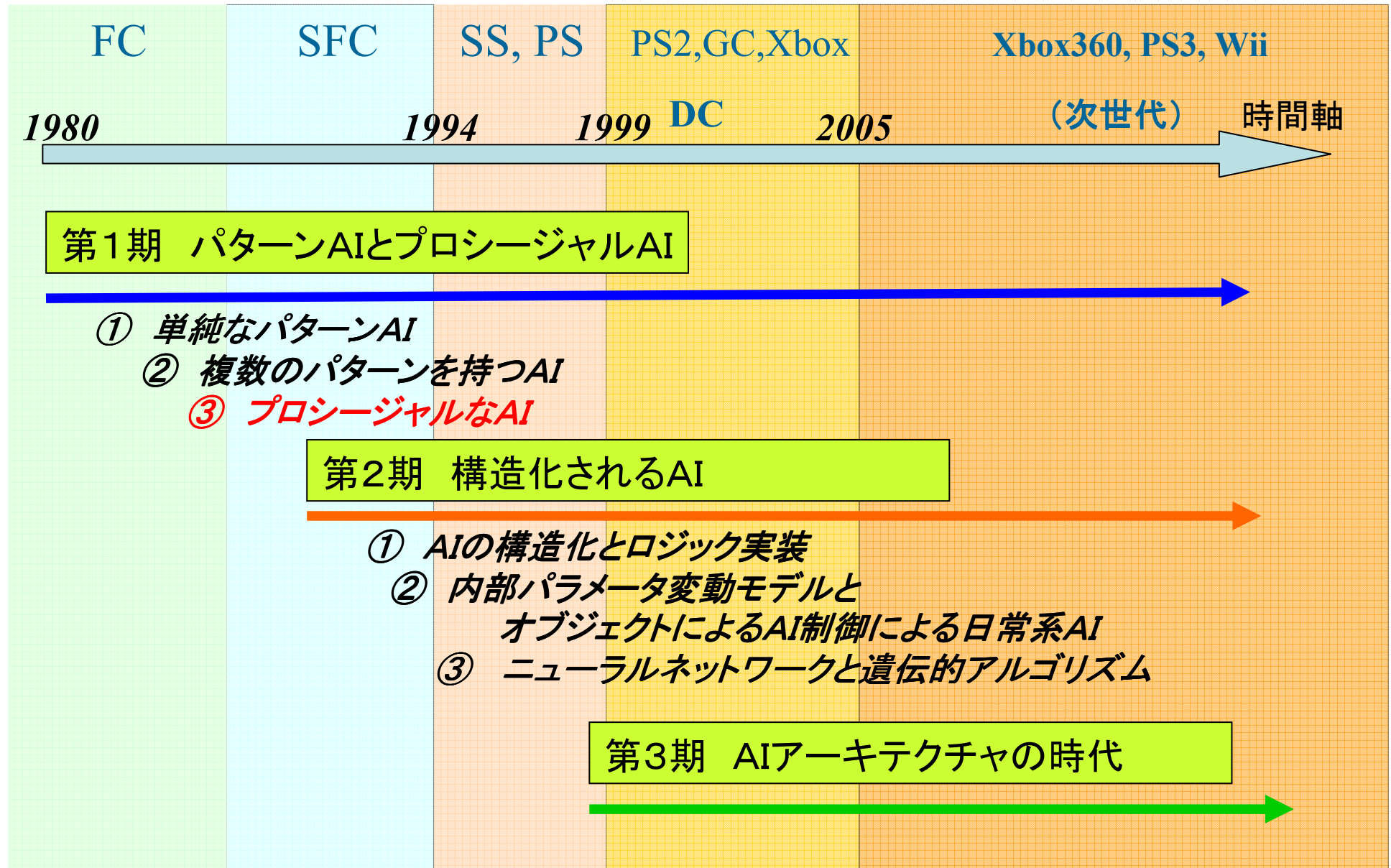
(例) プリンス・オブ・ペルシャ



「プリンス・オブ・ペルシャ」など、
スプライトアニメーションを用意する必要がある場合、
必然的にこういった制御となる。

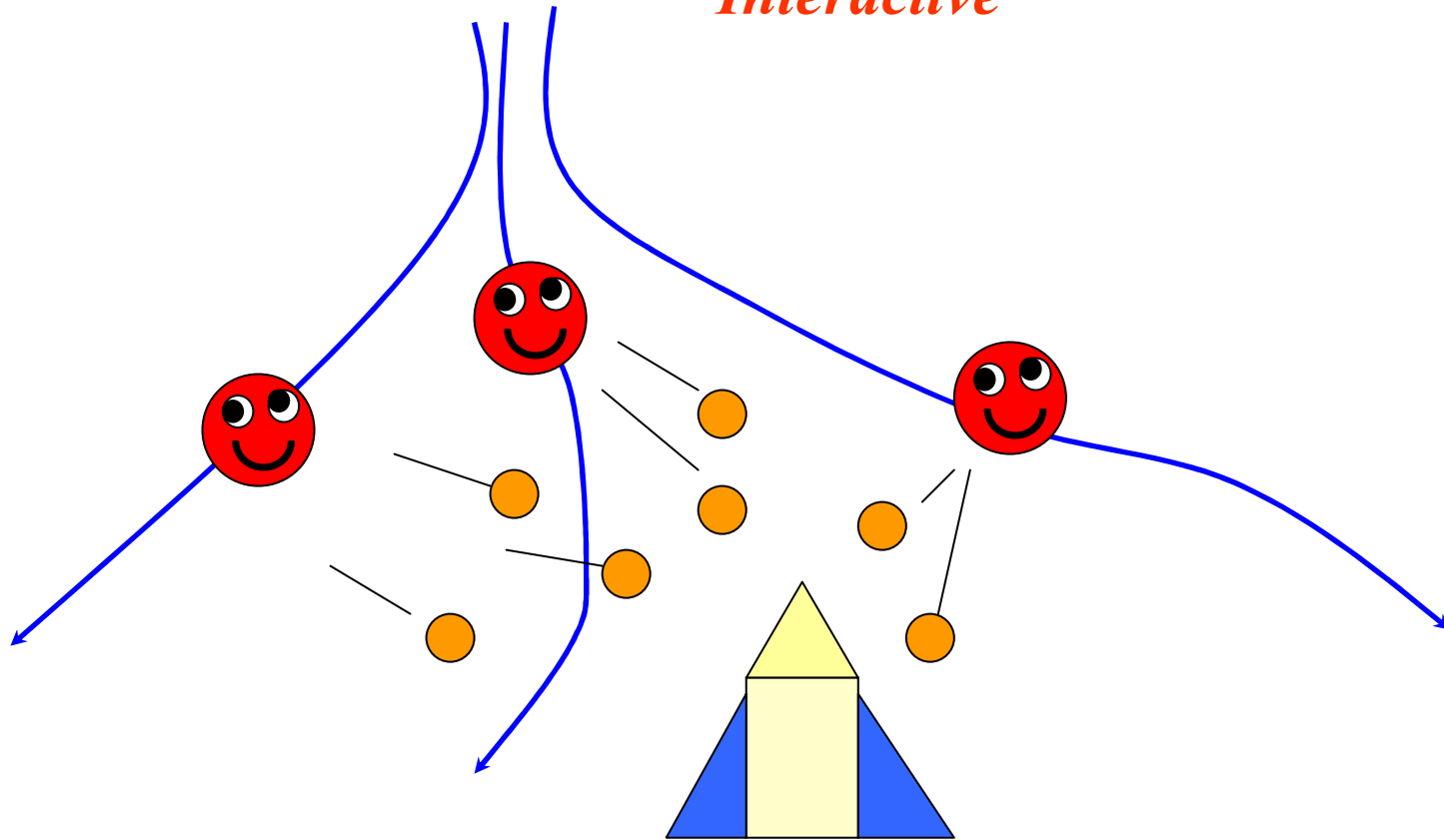
プリンス・オブ・ペルシャ

ゲームにおけるエージェントの歴史



第1期 ③プロシージャルなAI

Interactive



シューティングゲームなど、機体の軌道や弾道を、
逐次的に関数の計算で行なう。
(例) 数値列を用意する場合もある。

(例) ゼビウス？



ゼビウス

遠藤雅伸氏 あと面白い機能なんですけれど、ゼビウスには非常に簡単なAIが組み込まれています。

「プレイヤーがどれくらいの腕か」というのを判断して、出てくる敵が強くなるんです。強いと思った相手には強い敵が出てきて、弱いと思った相手には弱い敵が出てきます。そういったプログラムが組み込まれています。

ゲームの難易度というのは「初心者には難しくて、上級者には簡単だ」ということが、ひとつの難易度で(調整を)やっていくと起きてしまうので、その辺を何とか改善したいな、ということでそういったことを始めてみたのですけれど、

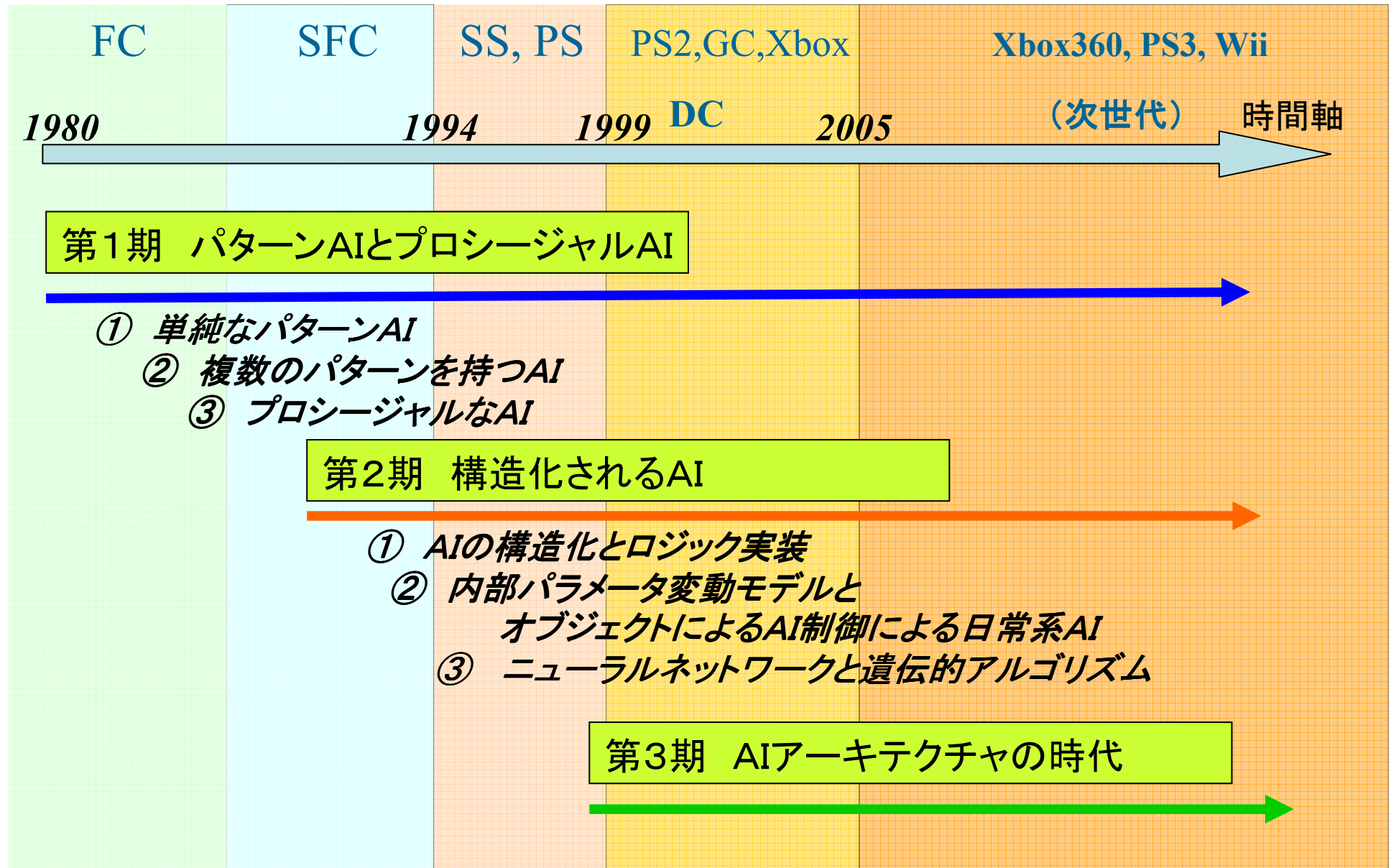
お陰で割合にあまり上手くない人でも比較的長くプレイできる、

うまい人でも最後のほうに行くまで結構ドラマチックに楽しめる、そういった感じになっています。

— ゼビウスセミナー —

<http://spitfire.client.jp/shooting/xvious2.html>

ゲームにおけるエージェントの歴史

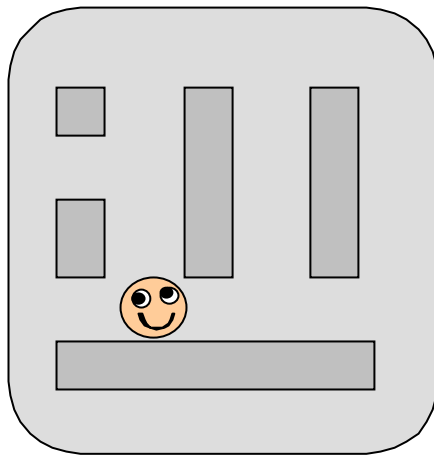


第2期 構造化されるAI

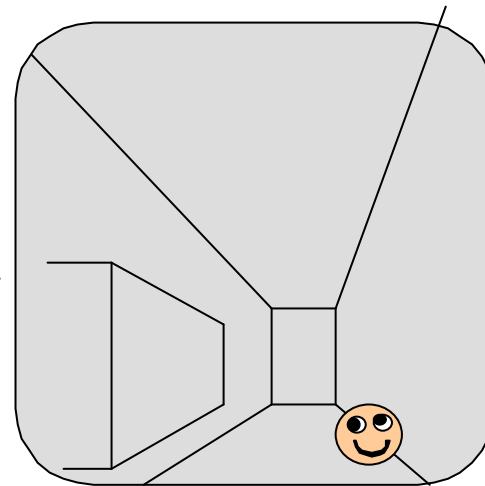
アセンブラからC言語への移行

2Dから3Dが主流へ → AIにとって爆発的な情報量の増大

80年代のAI技術の盛り上がりゲームへスピンオフ

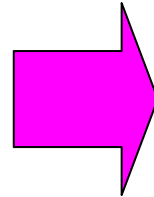


グリッド上のロジック
俯瞰制御



無数のレイキャスト(射線計算)
主観制御

2D・3DのAIをムービーで比較



Halo

マリオ

Halo

第2期 構造化されるAI

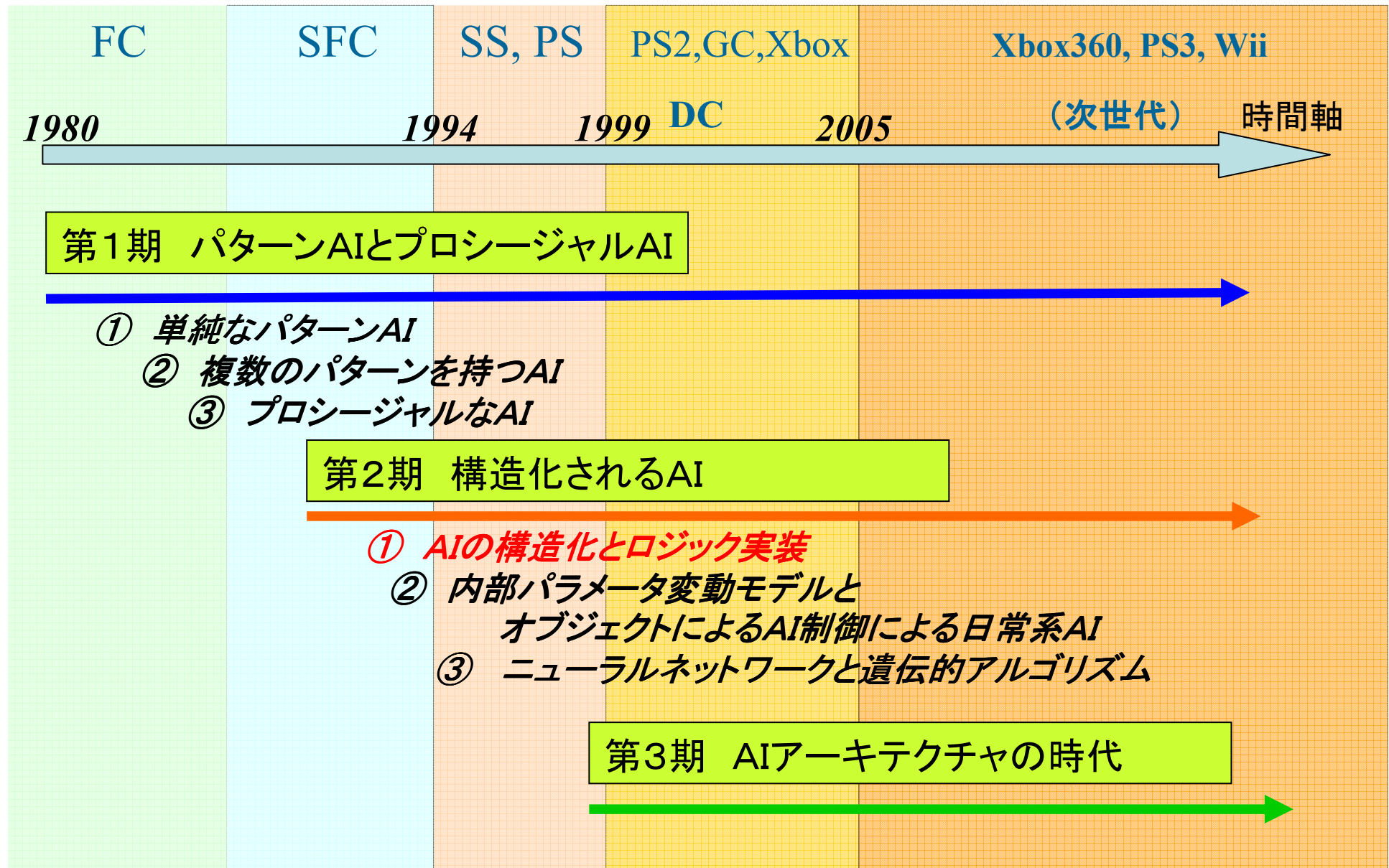
AIが処理すべき膨大な空間情報

→ 結局、対処出来ませんでした
(複雑すぎる、1994年から10年の課題となる)。

暫定的解決策

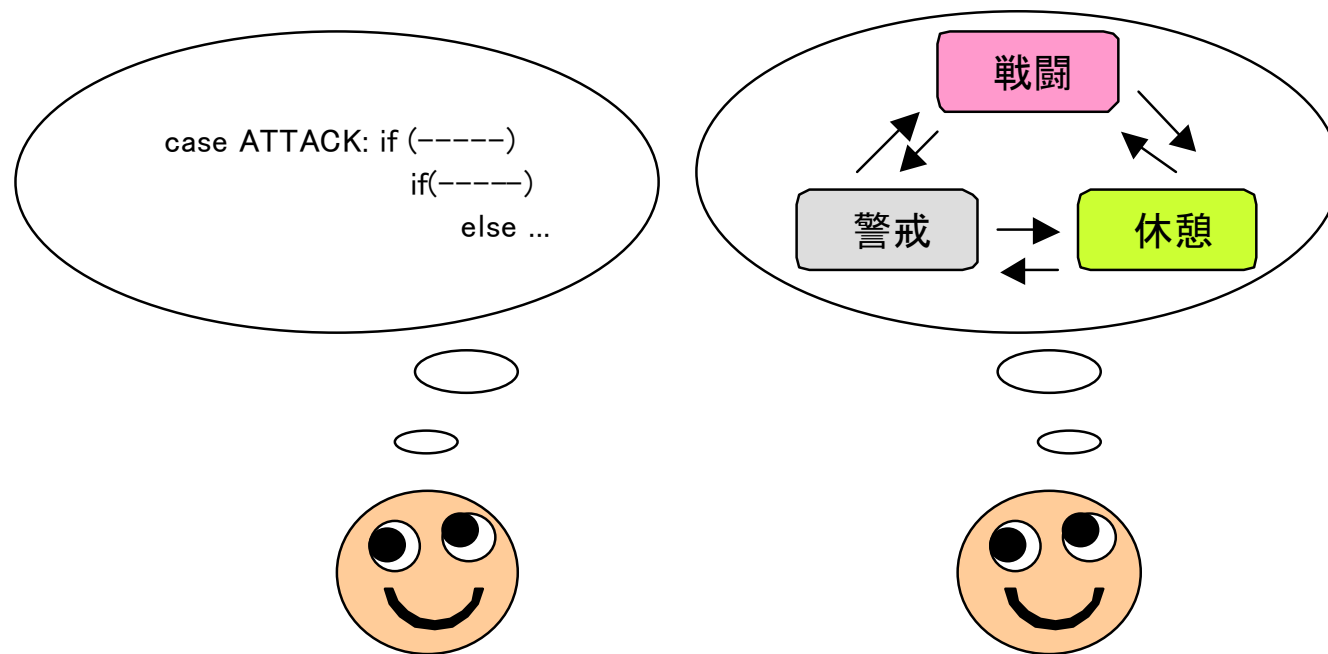
- ①AIの行動範囲をある限定した自由に移動できる空間に限定すること。
- ②ステージを単純化すること(例えば、空の上ならAIは動き放題である)。
- ③あらかじめ、地形に沿った運動をプログラミングしておく
(固定パスを与えておくなど)。

ゲームにおけるエージェントの歴史



第2期 ①AIの構造化とロジック実装

複雑な思考による行動

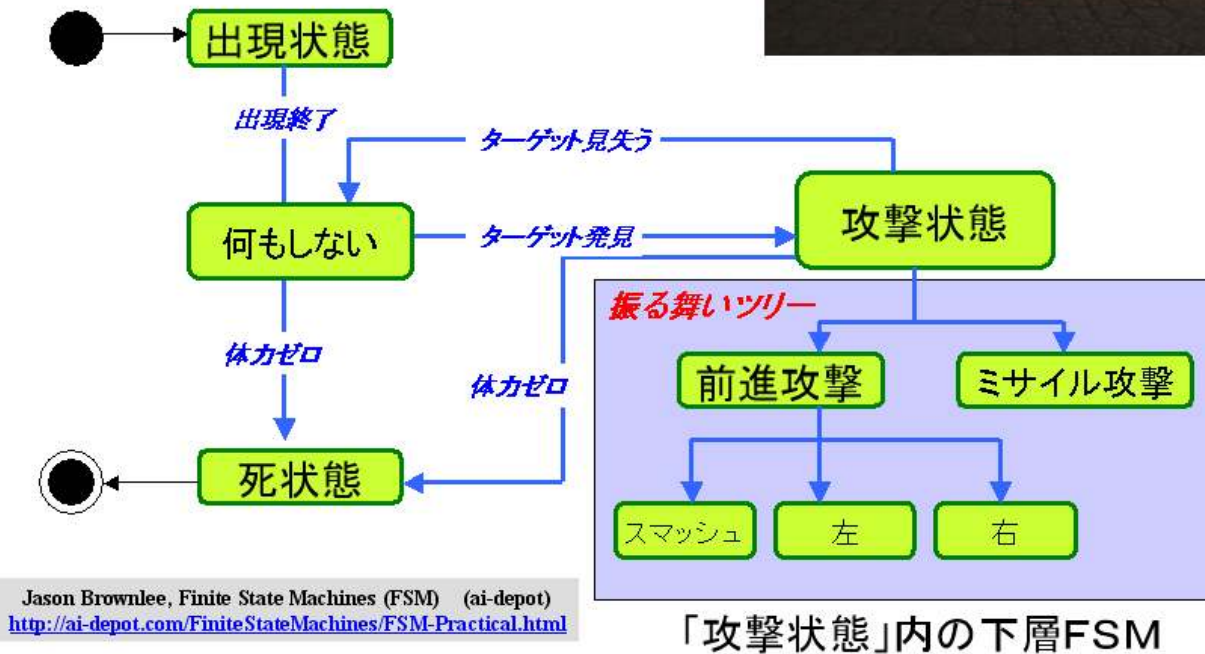


複雑な環境下(処理すべき情報が多い)で、
キャラクターを制御する仕組みを入れる。

(例) Quake HFSM

Quake HFSM

モンスターのFSM

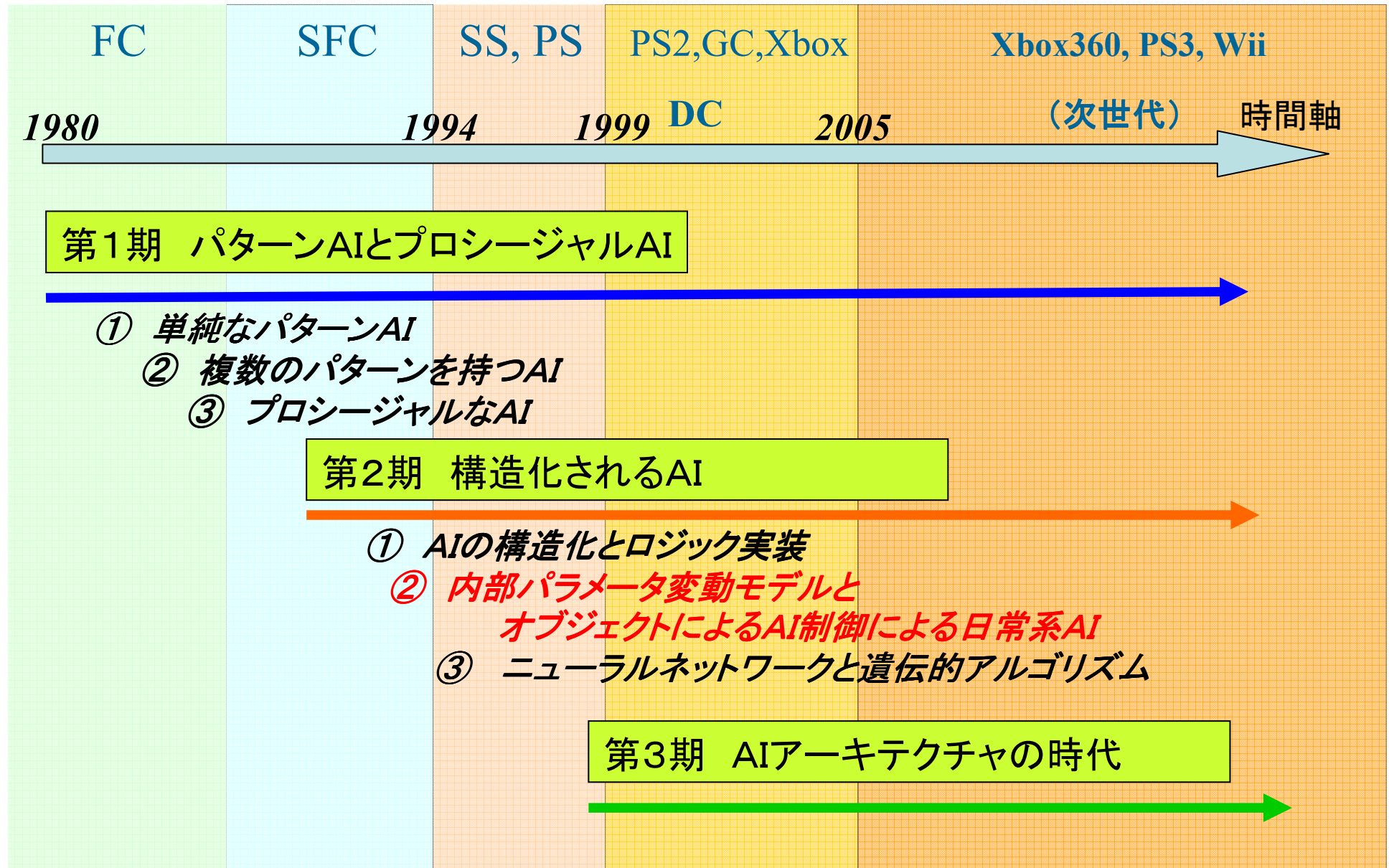


<http://ai-depot.com/FiniteStateMachines/FSM-Practical.html>

状態遷移図を用いる

Quake

ゲームにおけるエージェントの歴史



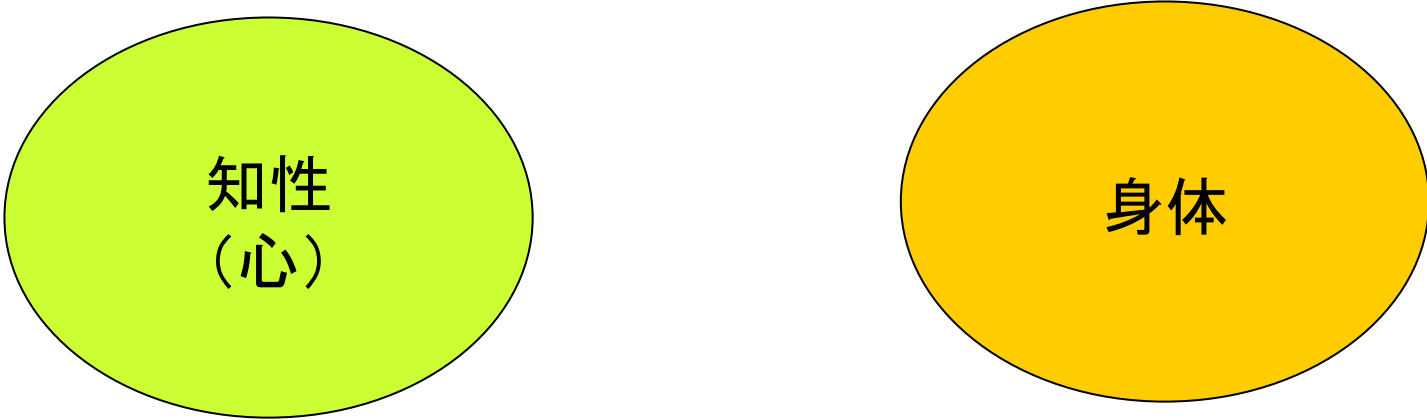
第2期 ②内部パラメータ変動モデルとオブジェクトによる AI制御による日常系AI



戦闘のAIから日常のAIへ

戦闘のAI(極限状態) < 日常のAI(煩雑)

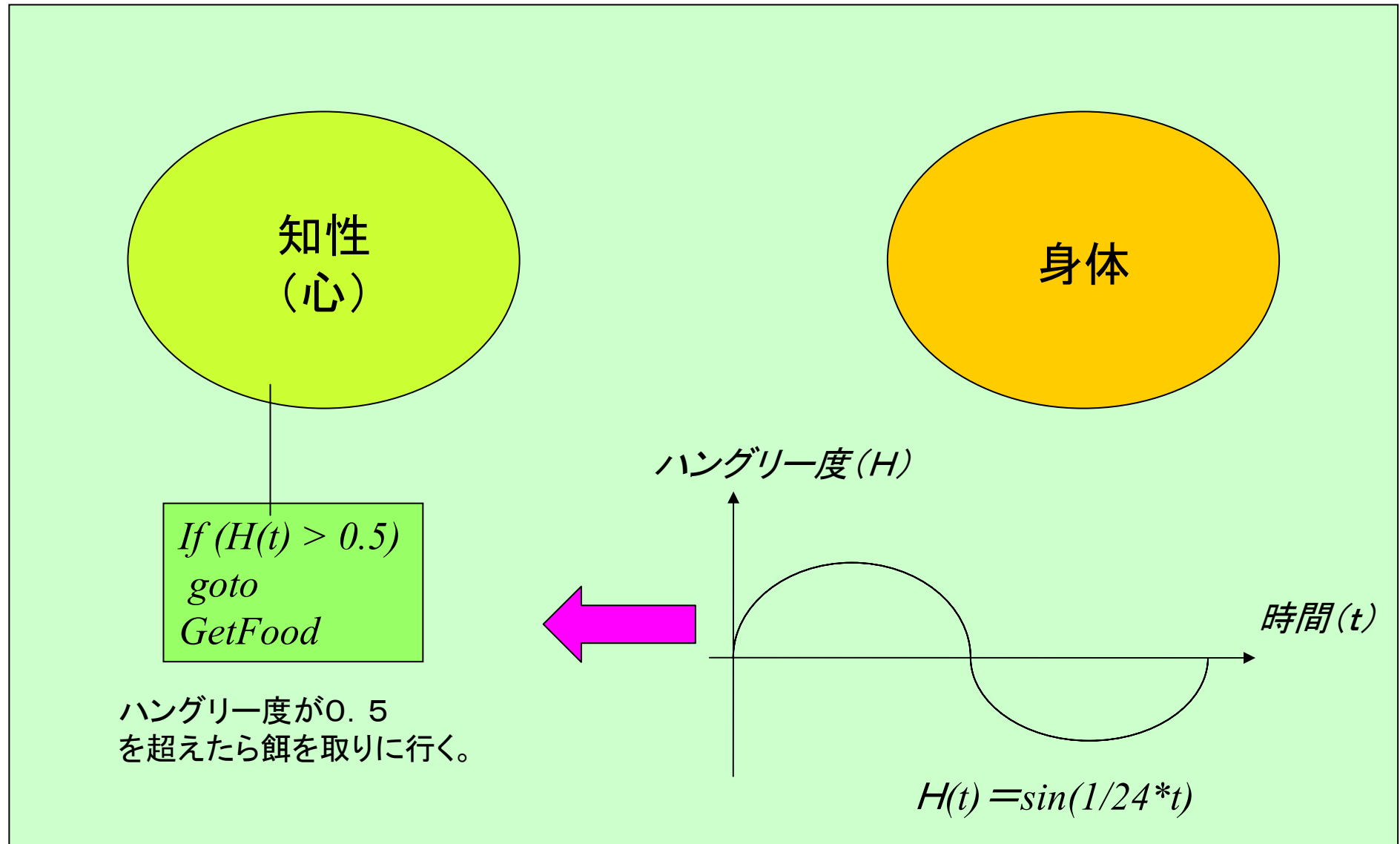
自律型AIの作り方



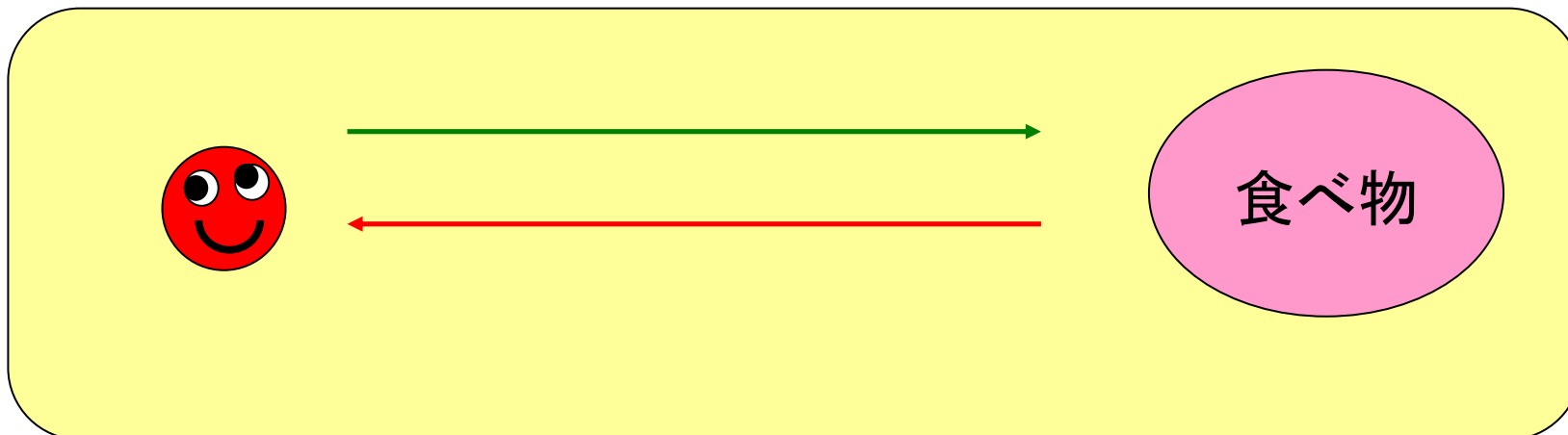
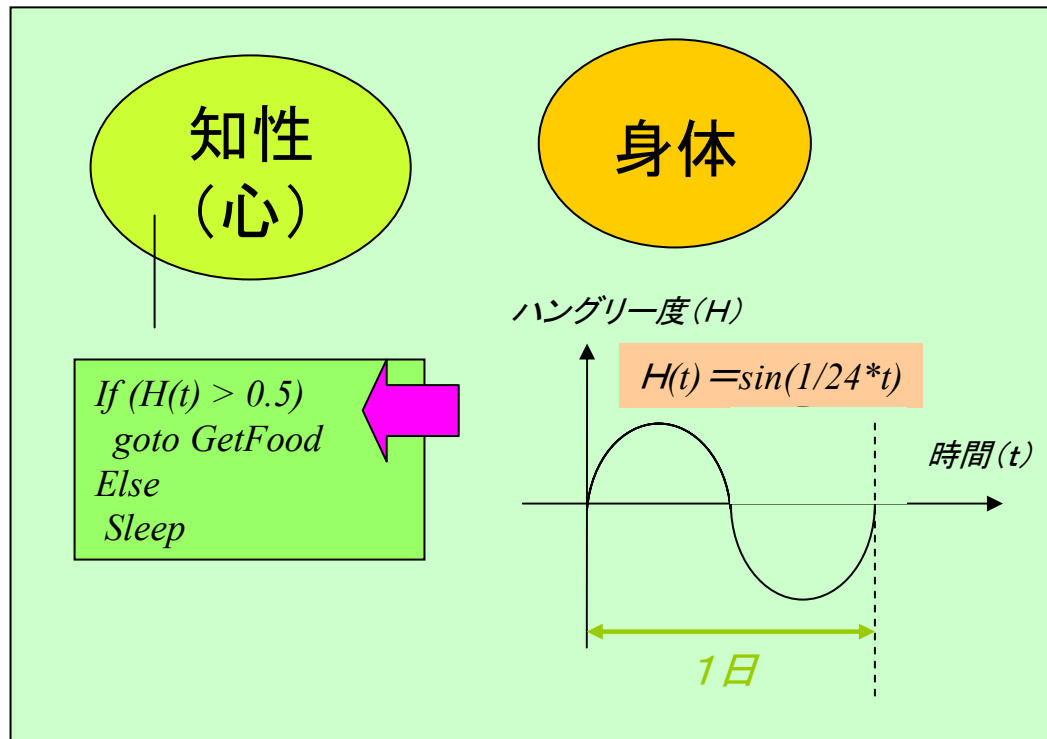
知性
(心)

身体

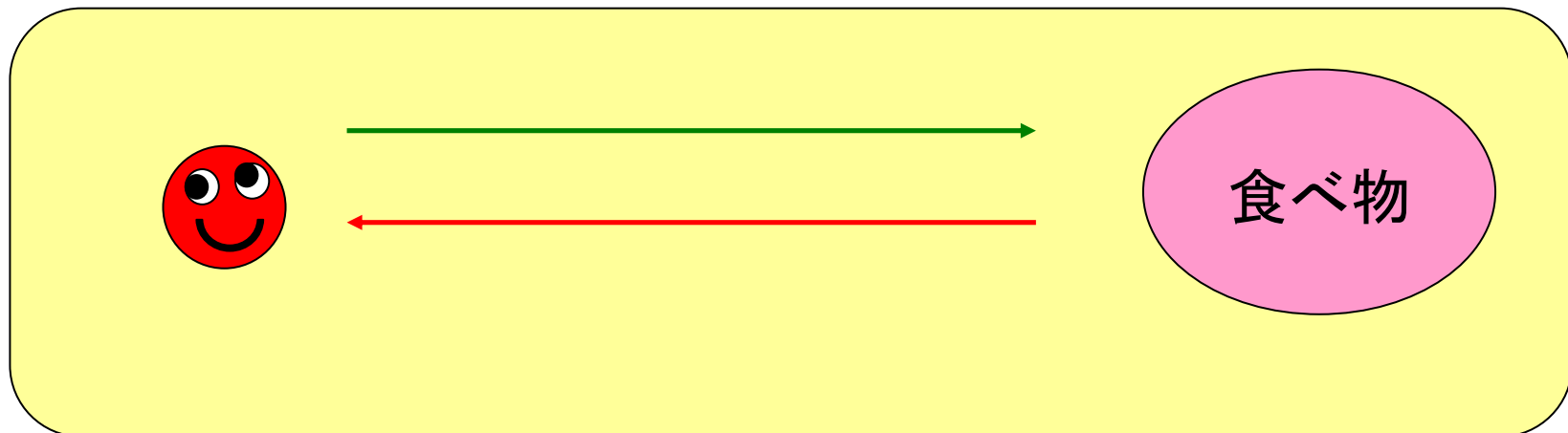
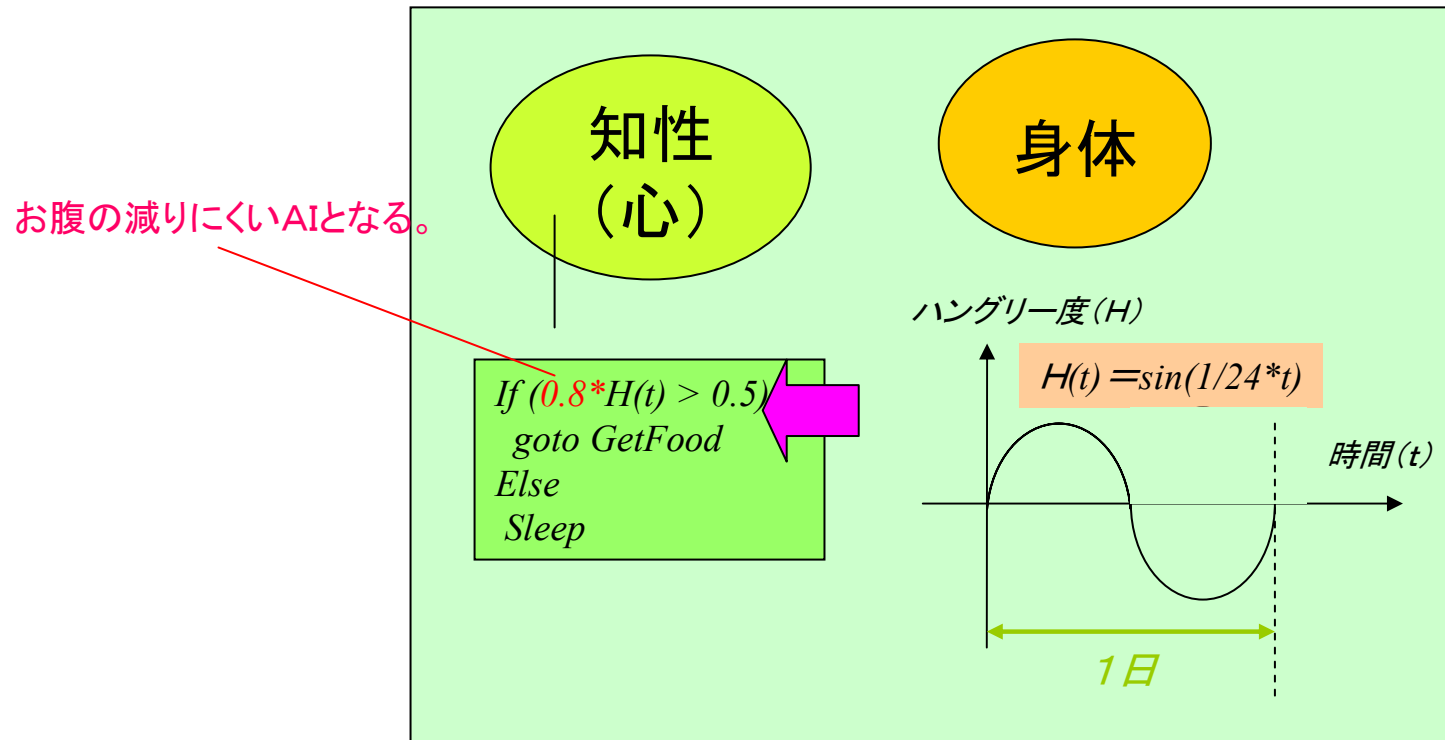
自律型AIの作り方



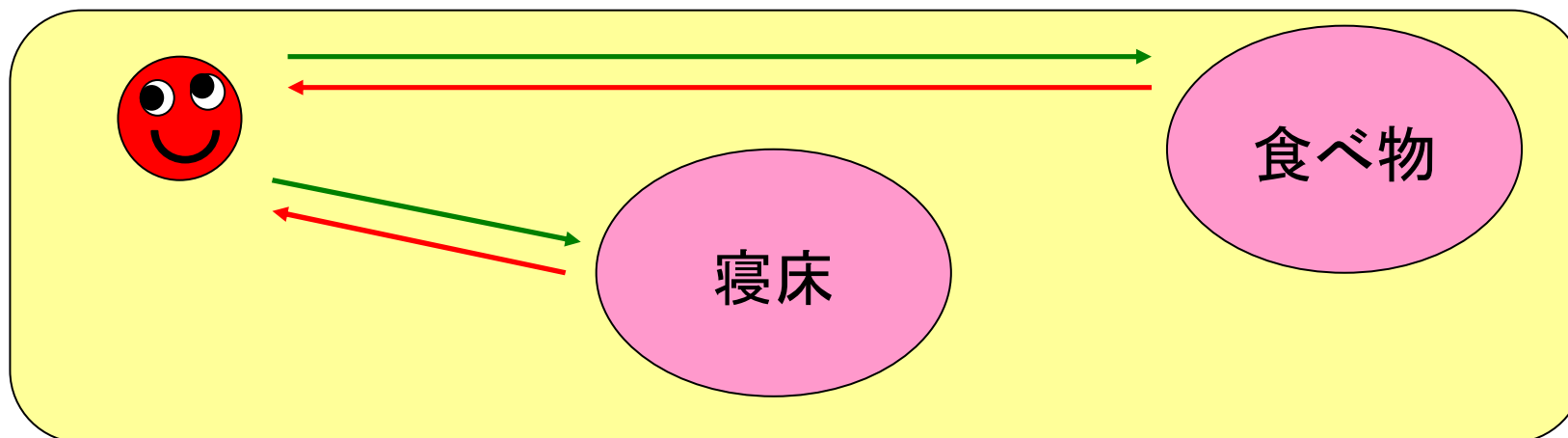
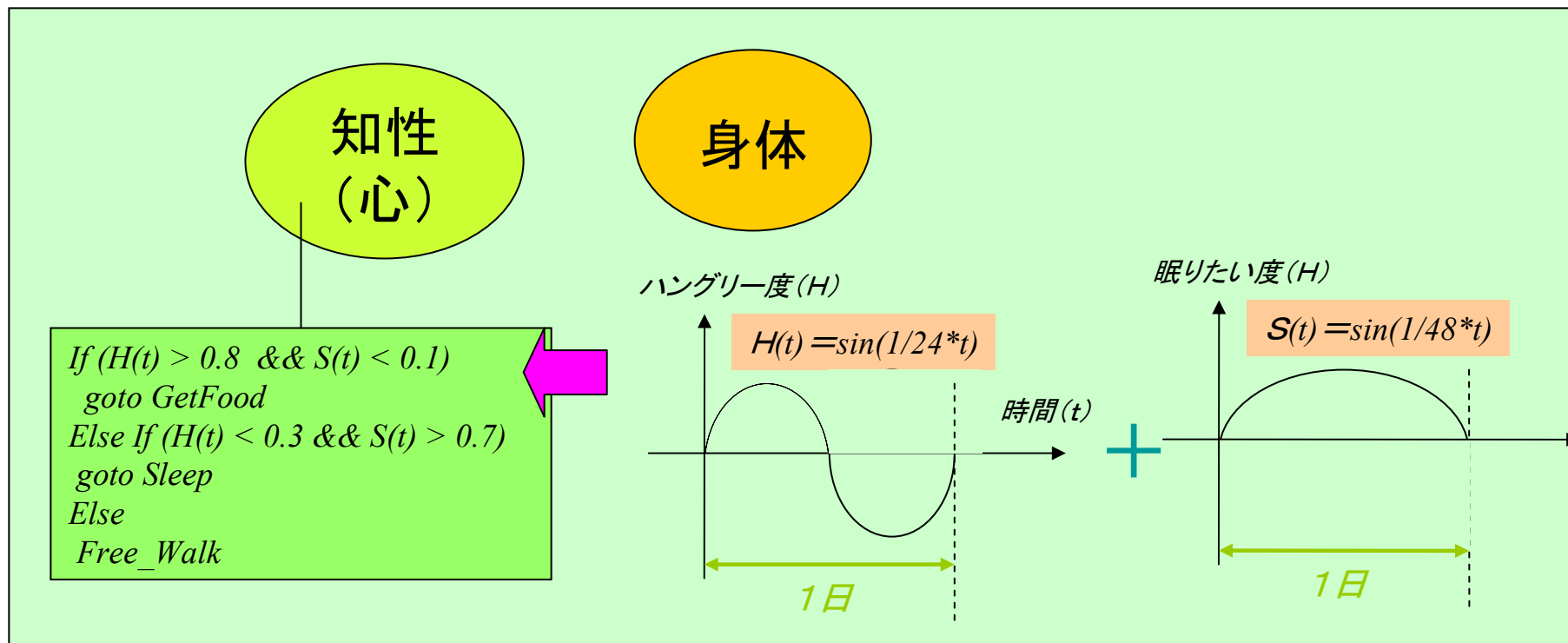
自律型AIの作り方



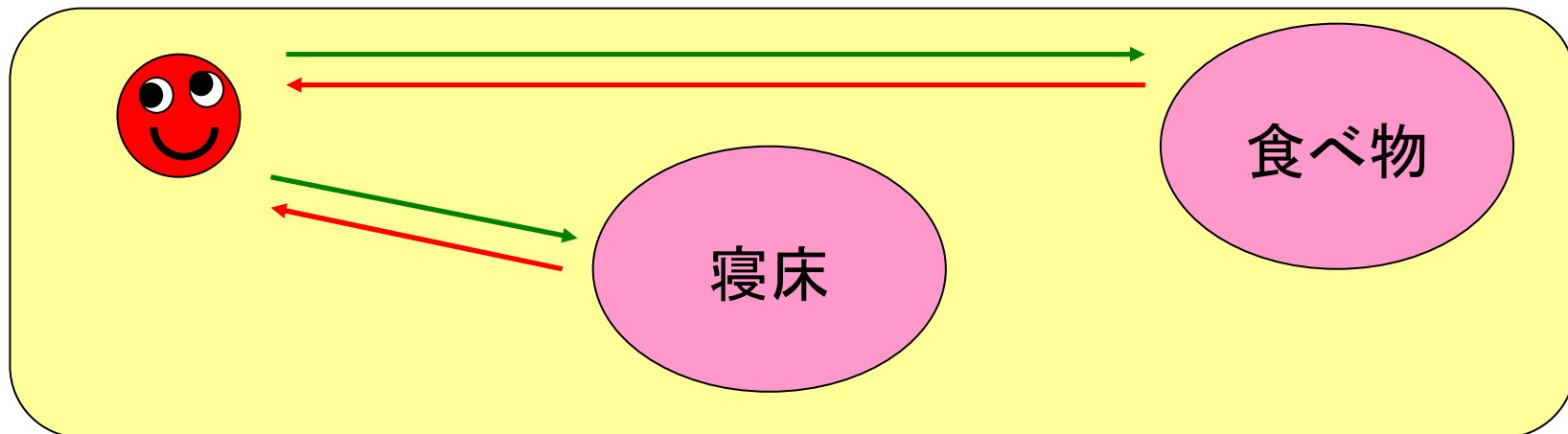
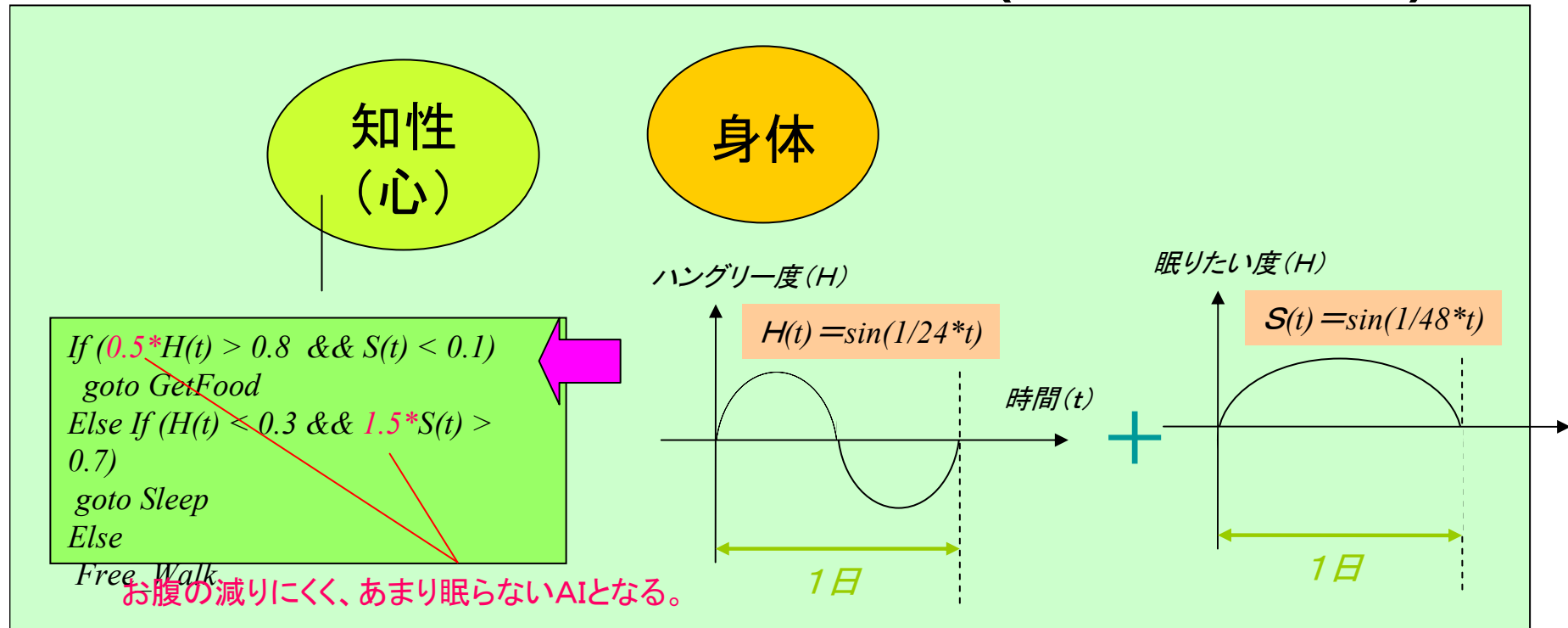
自律型AIの作り方(個性付け)



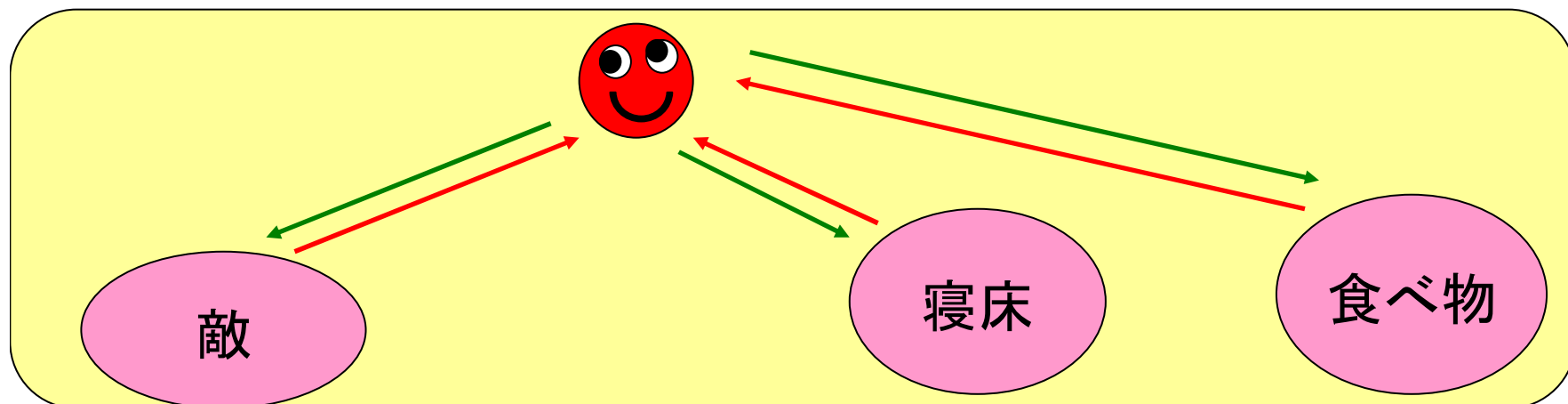
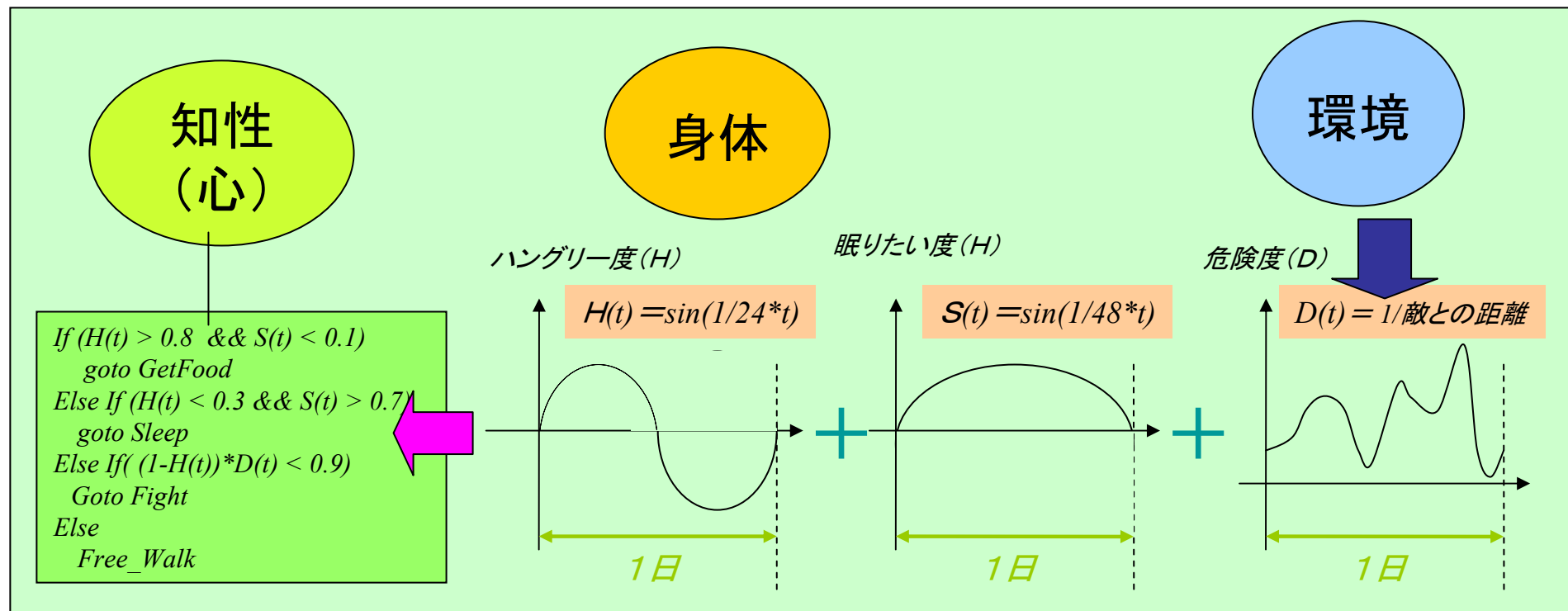
自律型AIの作り方



自律型AIの作り方(個性づけ)



自律型AIの作り方



Breaking
the Cookie-Cutter:
Modeling Individual
Personality,
Mood, and Emotion
in Characters

こういったデモに3Dモデルを被せると...

② *The Sims 3* (Maxis, EA)



ムービー



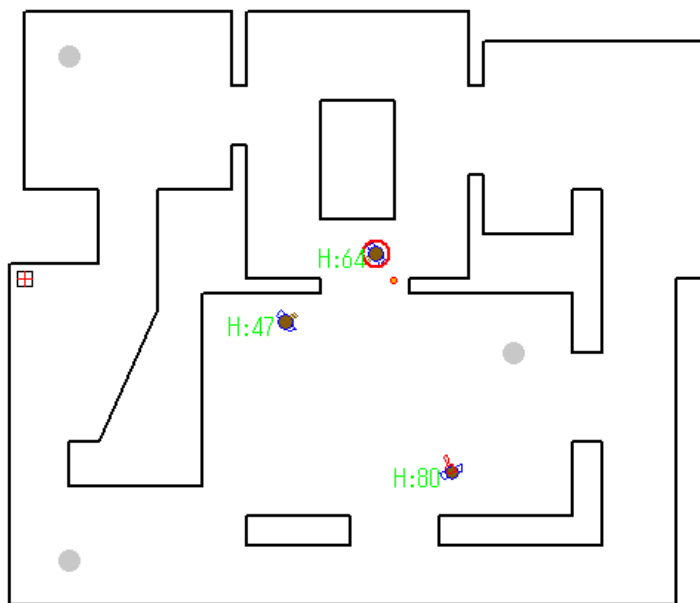
Trait ... 特性。性格パラメーター
Wish ... 将来の目標

Sims Getting Smarter: AI in The Sims 3(IGN)
<http://pc.ign.com/articles/961/961065p1.html>

GDC09 資料 <http://cmpmedia.vo.llnwd.net/o1/vault/gdc09/slides/Combined.ppt>



Mat Bucklandの自律型エージェントのデモ



デモ



目標を持ち、目標ごとに内部状態に応じて変動する関数を持つ

- ① 攻撃する
- ② 移動する
- ③ 回復アイテムを取る
- ④ 武器を取る

...

意思決定: 全てのゴールのうち、最大値を持つゴールを選択して行動する。

(例) ③

$$H(t) = 0.8 * ((1 - \text{現在のHP}) / \text{回復アイテムまでの距離})$$



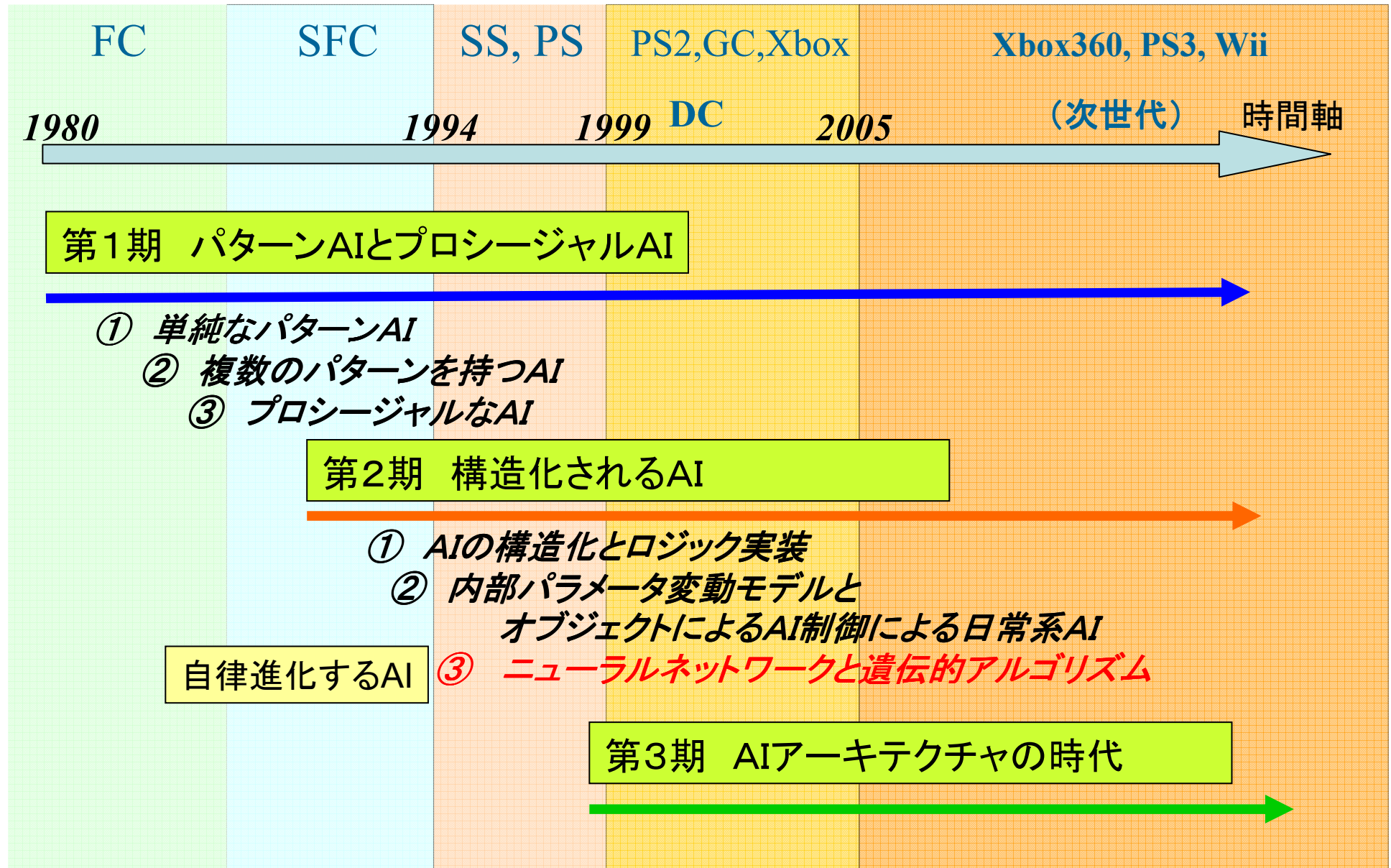
オライリー・ジャパン

「実例で学ぶゲームAIプログラミング」

(Mat Buckland 著、松田晃一 訳)

<http://www.oreilly.co.jp/books/9784873113395/>

ゲームにおけるエージェントの歴史



(例)決して多くない

ニューラルネットワーク

1996年 BATTLECRUISER: 3000AD (3000AD)

1997年 がんばれ森川君2号 (muumuu)

2000年 Colin McRae Rally 2.0 (Codemasters)

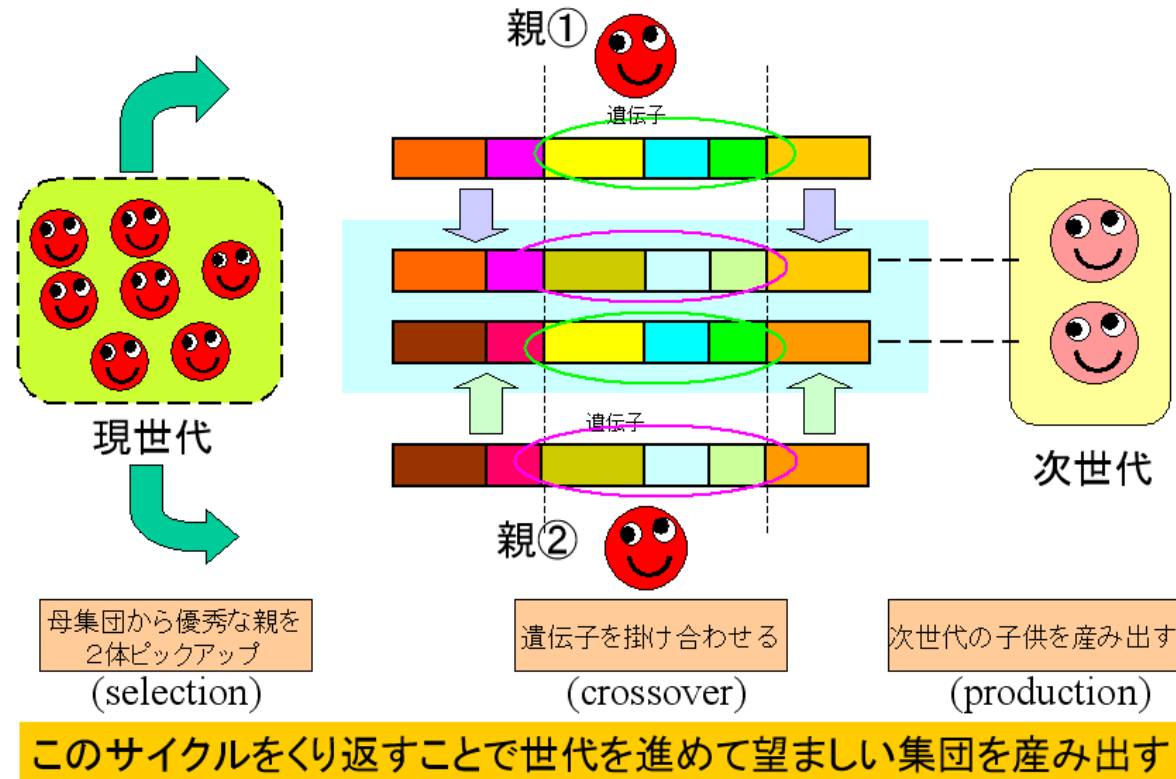
2001年 Black & White (Lionhead Studio)

遺伝的アルゴリズム

1998年 アストロノーカ (muumuu)

第2期③ニューラルネットワークと遺伝的アルゴリズム

遺伝的アルゴリズムの仕組み



遺伝的アルゴリズムとニューラルネットワークが
ゲームAIの視野の中に入って来た。

(例) アストロノーカ



MuuMuu, プレイステーション用ソフト「アストロノーカ」(Enix, 1998)

<http://www.muumu.com/games/astro/>

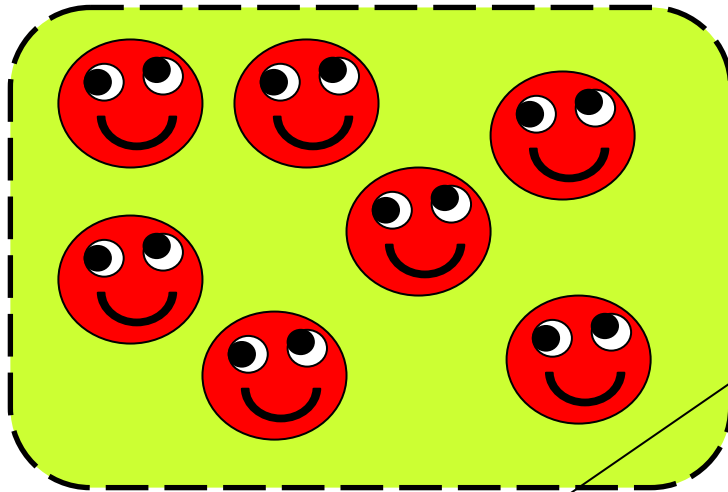
参考文献

森川幸人,
「テレビゲームへの人工知能技術の利用」,
人工知能学会誌vol.14 No.2 1999-3
<http://www.1101.com/morikawa/1999-04-10.html>

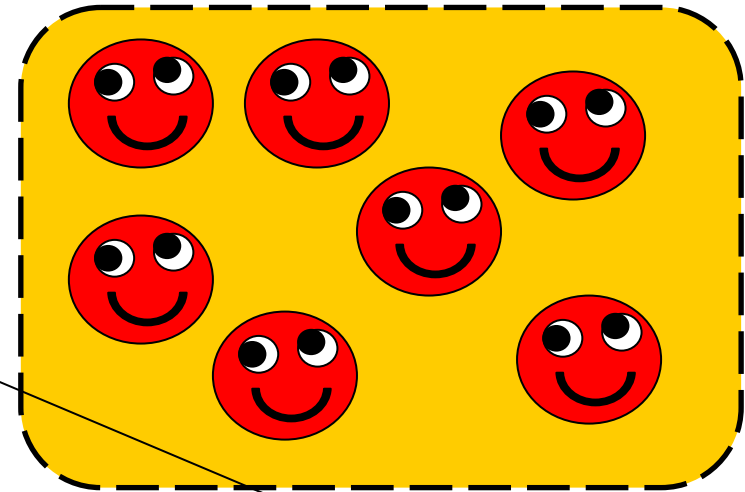
遺伝的アルゴリズム

集団を一定の方向に進化させる方法

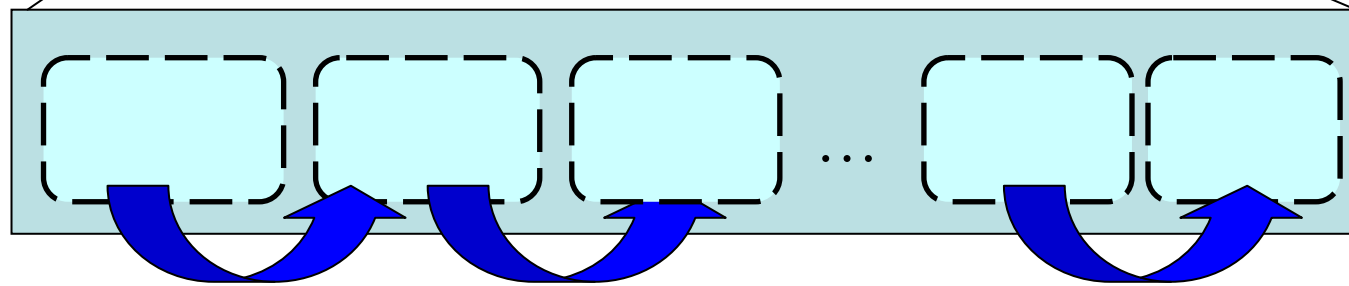
最初の世代



新世代(100~世代後)

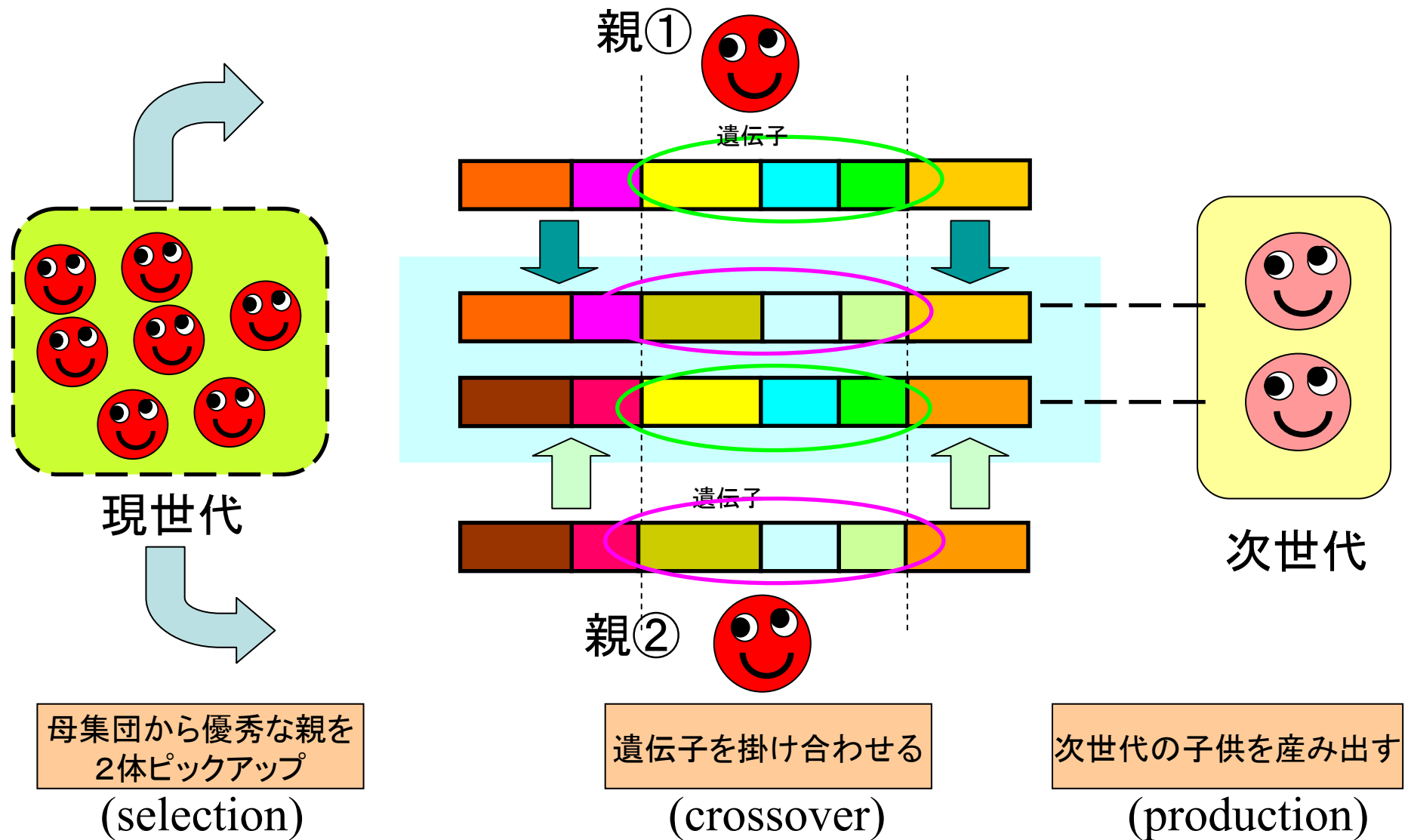


世代を経て進化させる



一つの世代が次の世代を交配によって産み出す

遺伝的アルゴリズムの仕組み

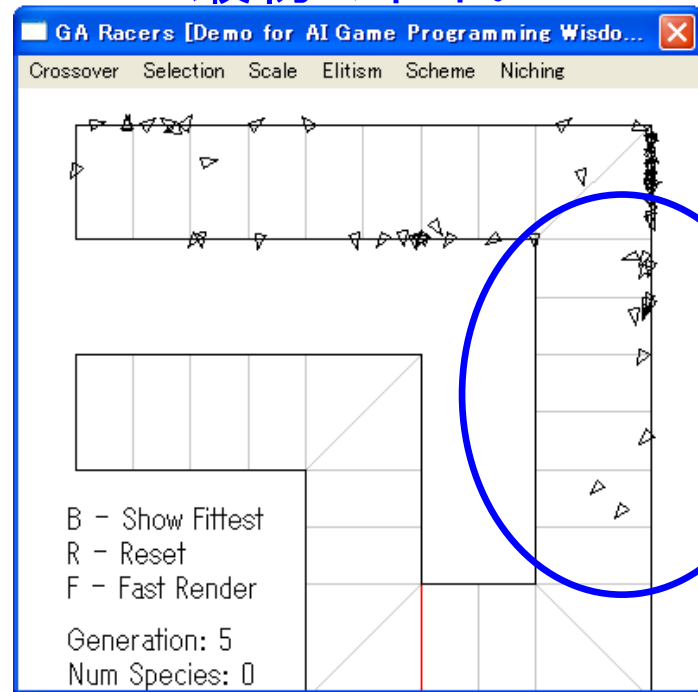


このサイクルをくり返すことで世代を進めて望ましい集団を産み出す

(例)① GA Racer

遺伝的アルゴリズムによって、遠くまで到達できるレーサーを作成する

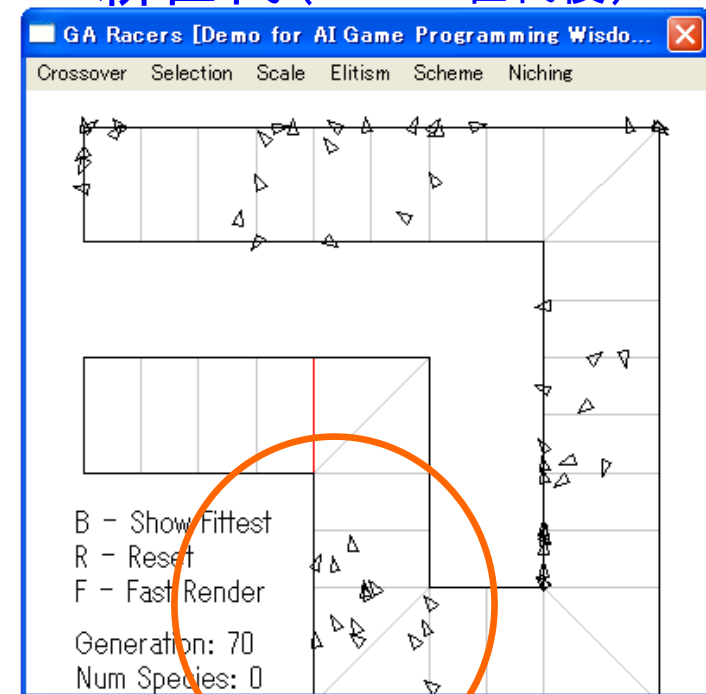
最初の世代



最初はここまでしか
たどり着けないけど...



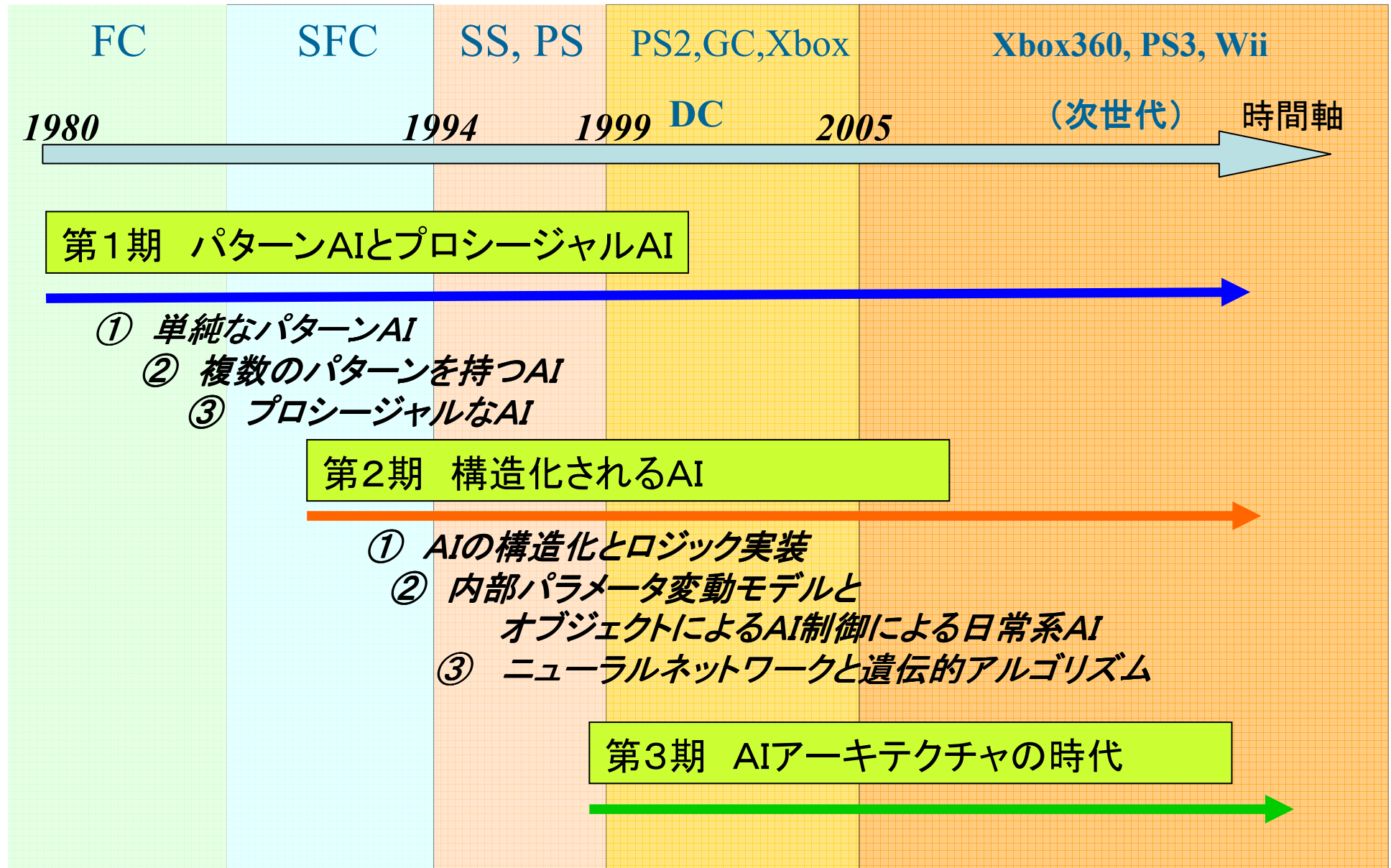
新世代(100~世代後)



だんだんと遠くまで、
たどりつけるようになる。

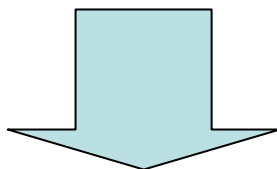
Mat Buckland, "Building Better Genetic Algorithm", 11.4., AI Game Programming Wisdom 2
(CD-ROMにソースコードと実行ファイルがあります)

ゲームにおけるエージェントの歴史



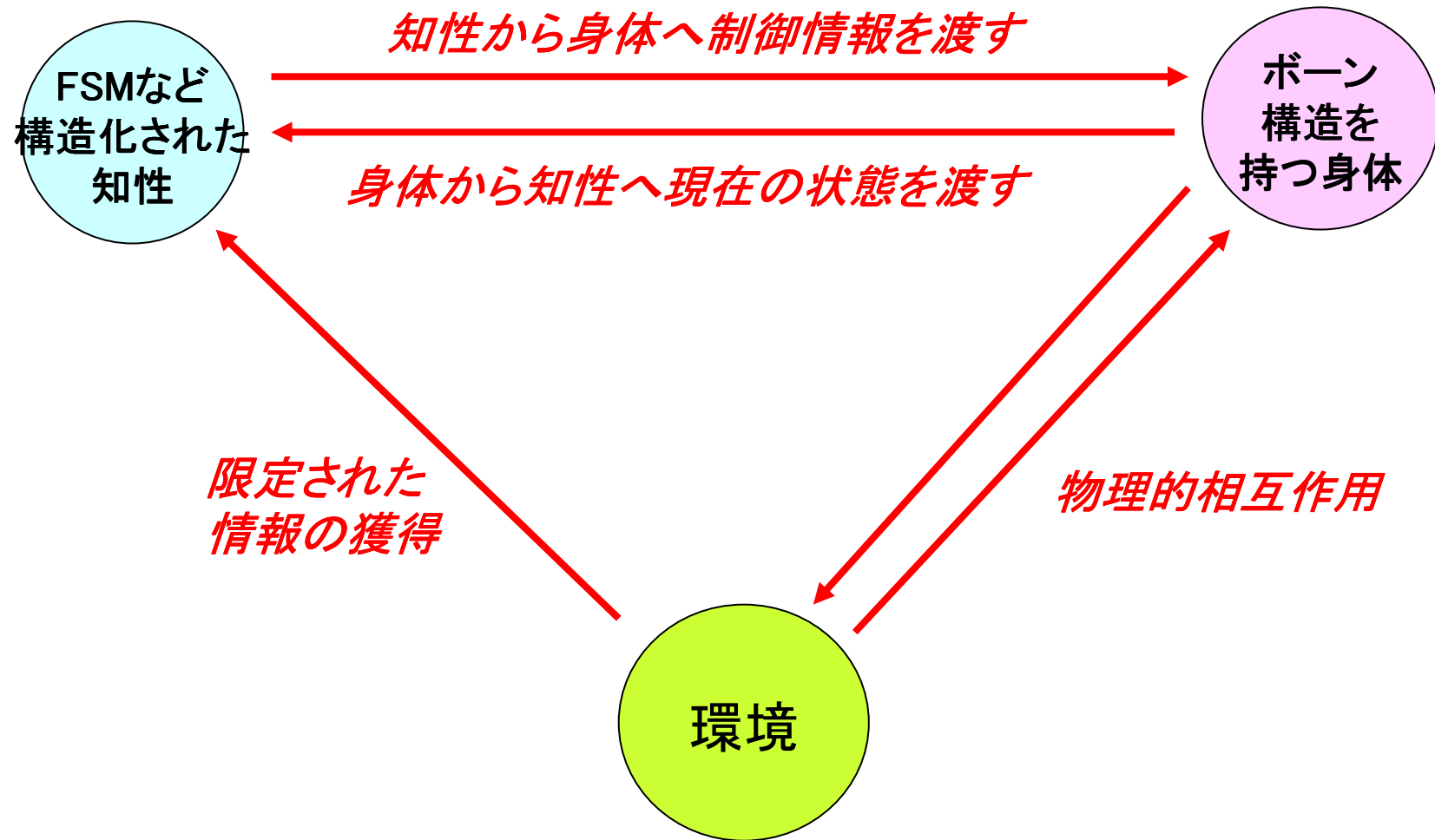
第3期 AIアーキテクチャの時代

個々のアルゴリズムや構造的なAI(第2期)

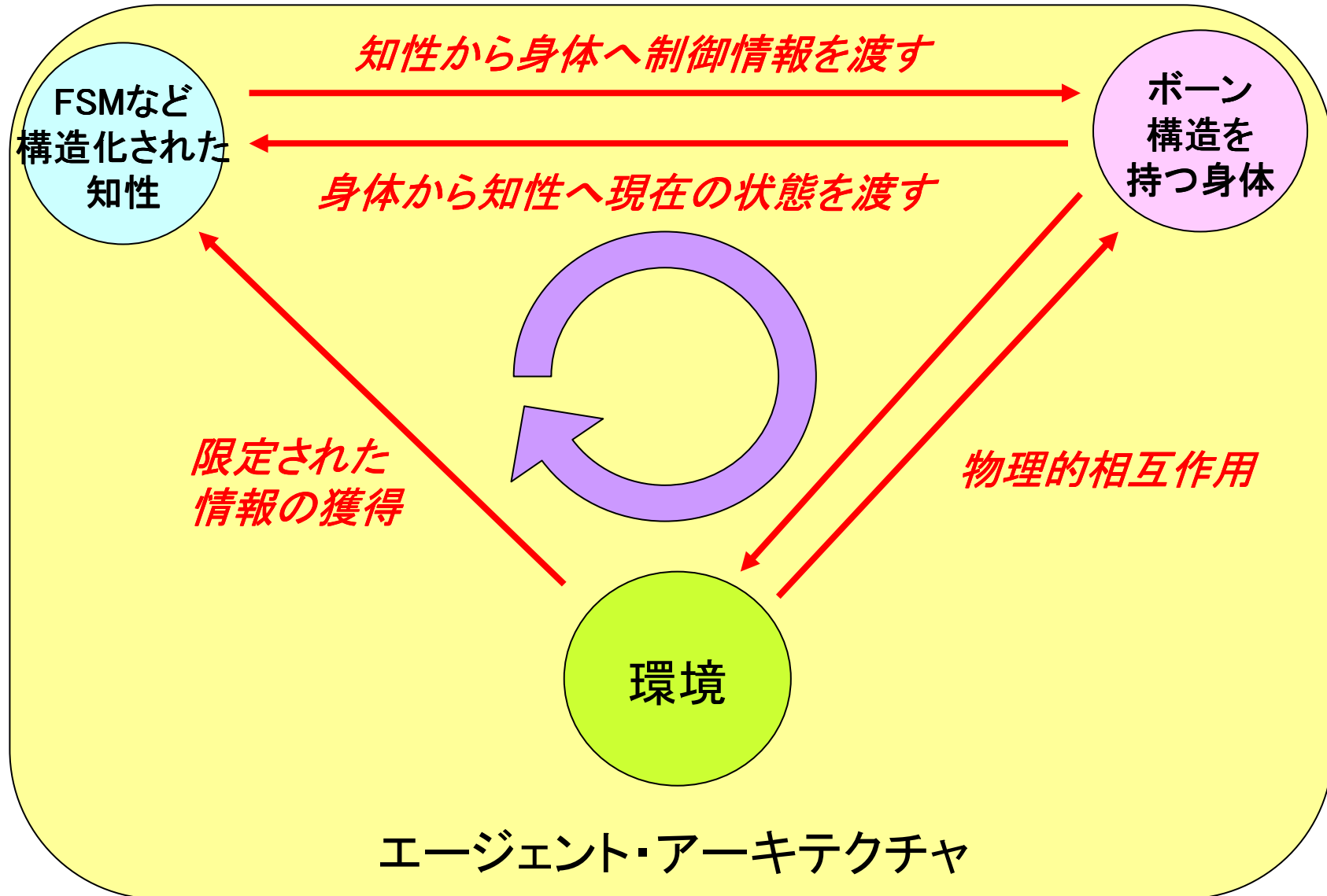


包括的なアーキテクチャへ
キャラクターAIのためのフレームワークを構築する。

「知性 - 環境 - 身体」相関図



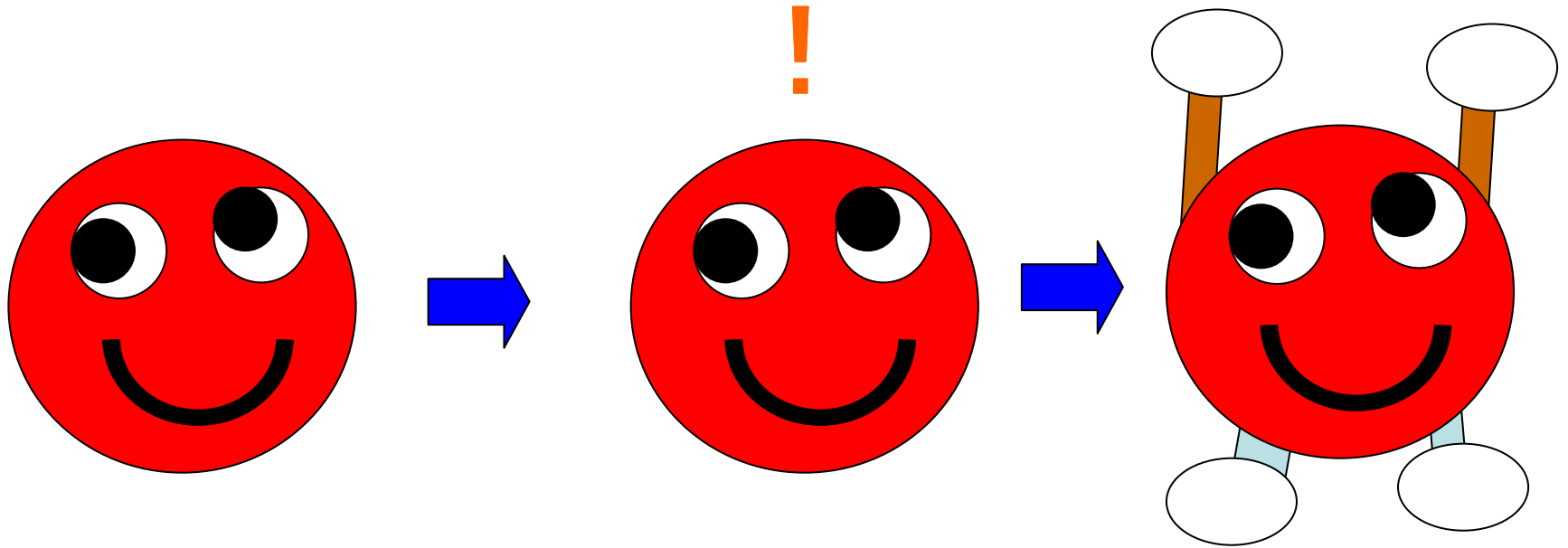
「知性 - 環境 - 身体」相関図



全体を包括する共通の基盤システムを作りましょう！

エージェントとは？

- ① 環境に対して情報を集める感覚(センサー)を持つ
- ② 自ら判断する能力を持つ。
- ③ 環境に対して働きかけることができる能力を持つ。

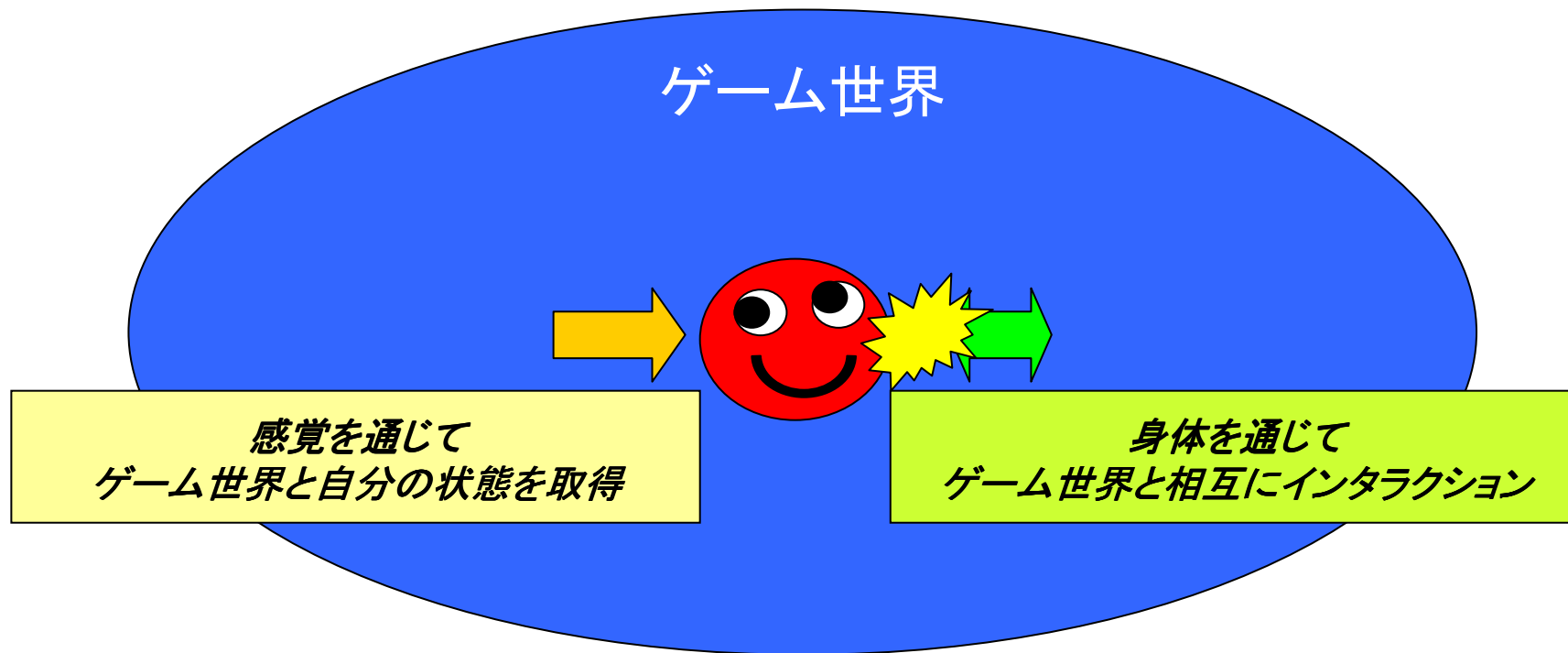


感覚を持ち

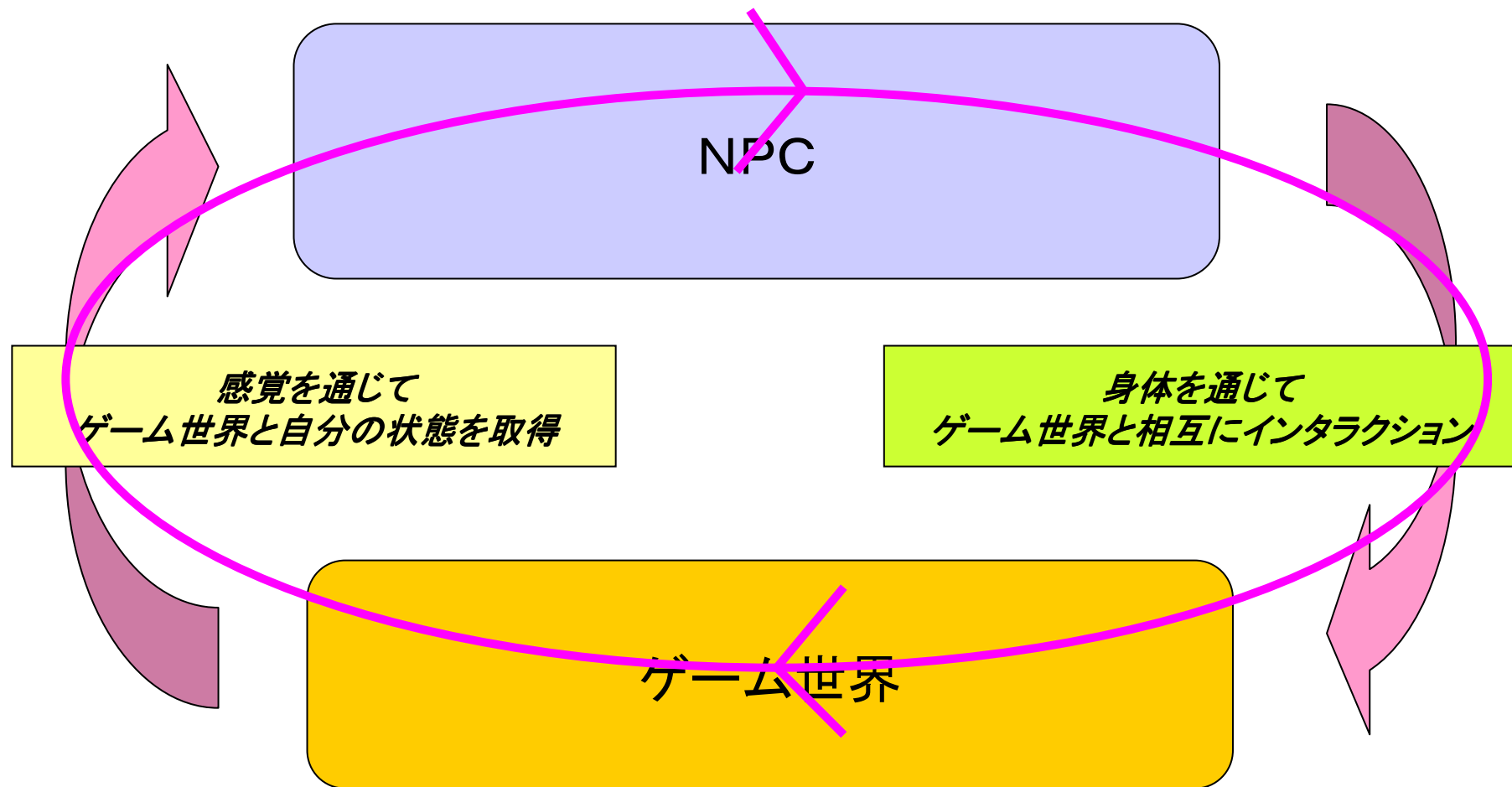
自ら判断して

世界に働きかける
能力を持つ

ゲームにおけるエージェント



エージェント・アーキテクチャーにおける情報の流れ

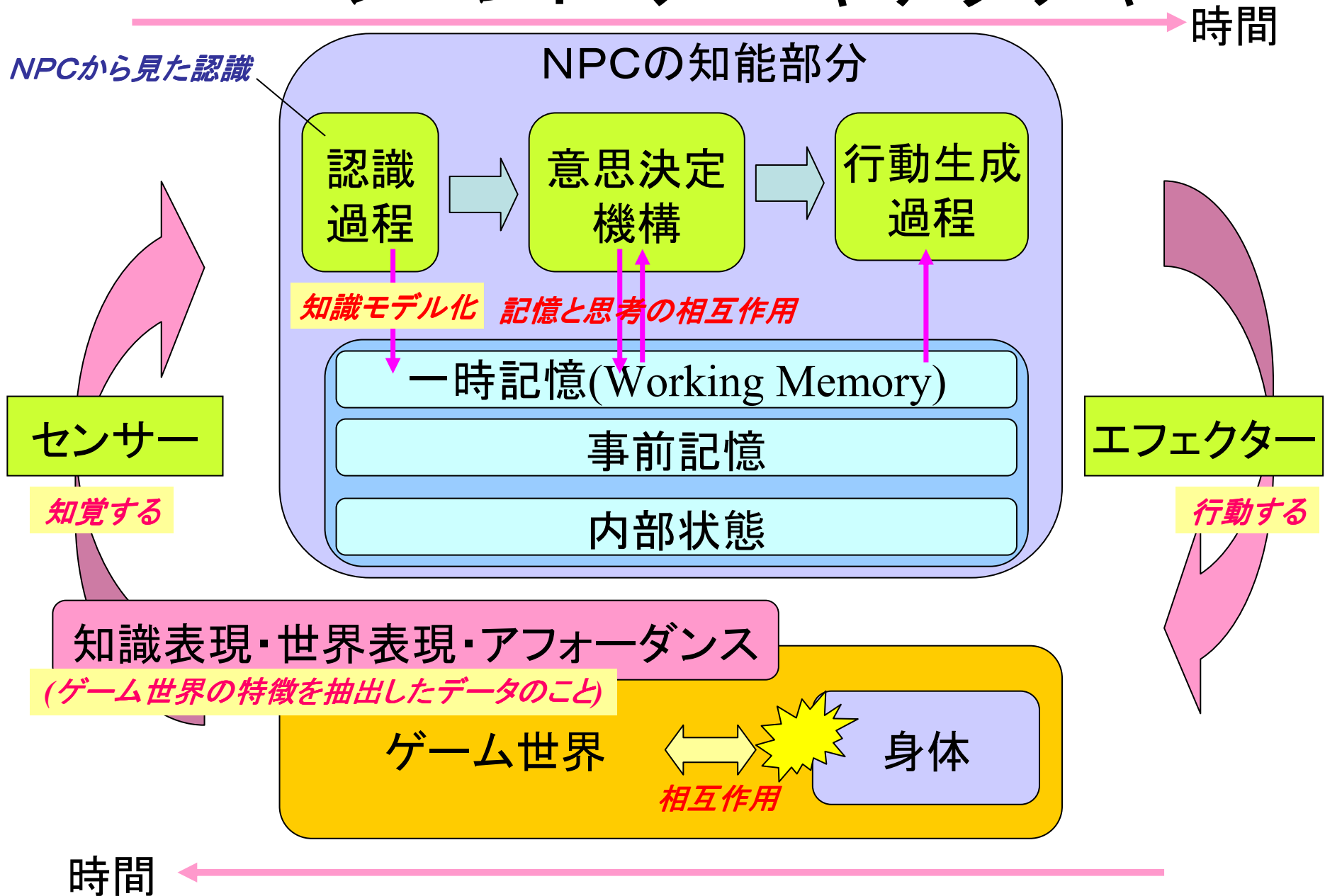


この情報の流れに仕掛けをしてみよう！

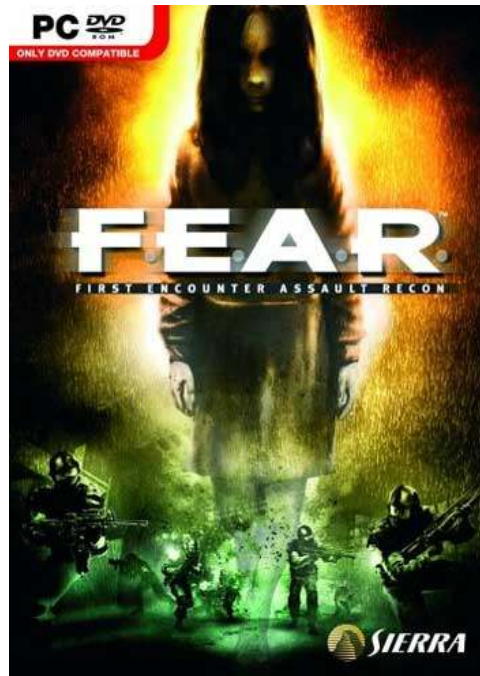
「人工知能＝からくり」

作り方を勉強しよう！

エージェント・アーキテクチャ



F.E.A.R.



内容:閉鎖空間の中のホラーFPS

開発元: Monolith Production

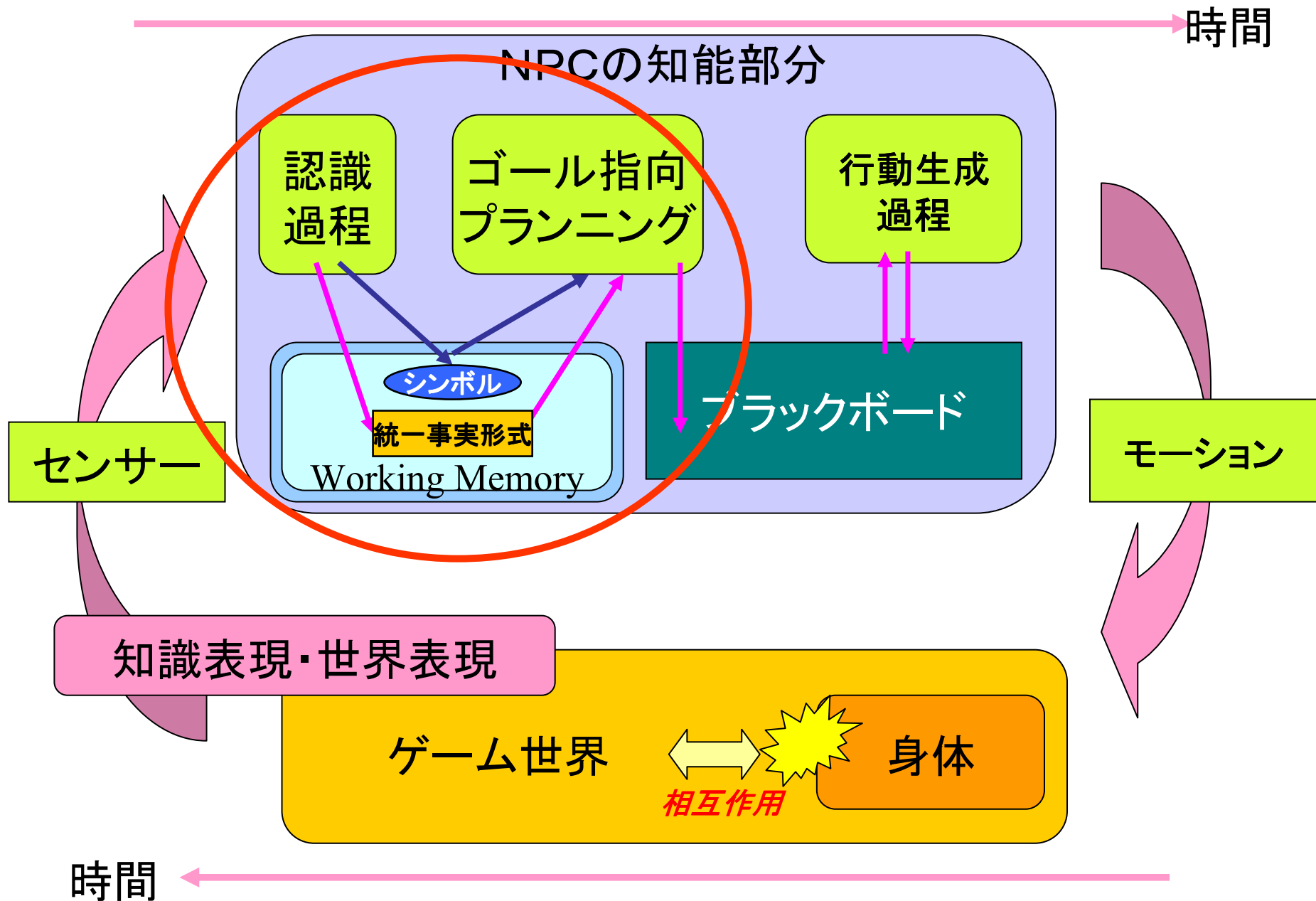
出版: SIERRA

Hardware: Windows, PS3

出版年: 2004年

FPSとホラーを、映画的な演出によって結びつけたエポックメイキングな名作。
長年発展させて来たAI技術の本領が発揮され、開発者、プレイヤーから高い支持を集める。

F.E.A.R NPCのAIのアーキテクチャー



プランニング



プランニング



F.E.A.R NPCのAIのアーキテクチャー

時間 →

NPCの知能部分

F.E.A.R .のポイント

C4 アーキテクチャーの基本の上にプランニング技術を組み込む
= 状況を認識して、目的を選択して計画を立てるAI

企画の発想

単一の行動でなく、一つなぎの行動のシーケンスを
指定できる時代になったのだ(これからのAI)

プログラマーの発想

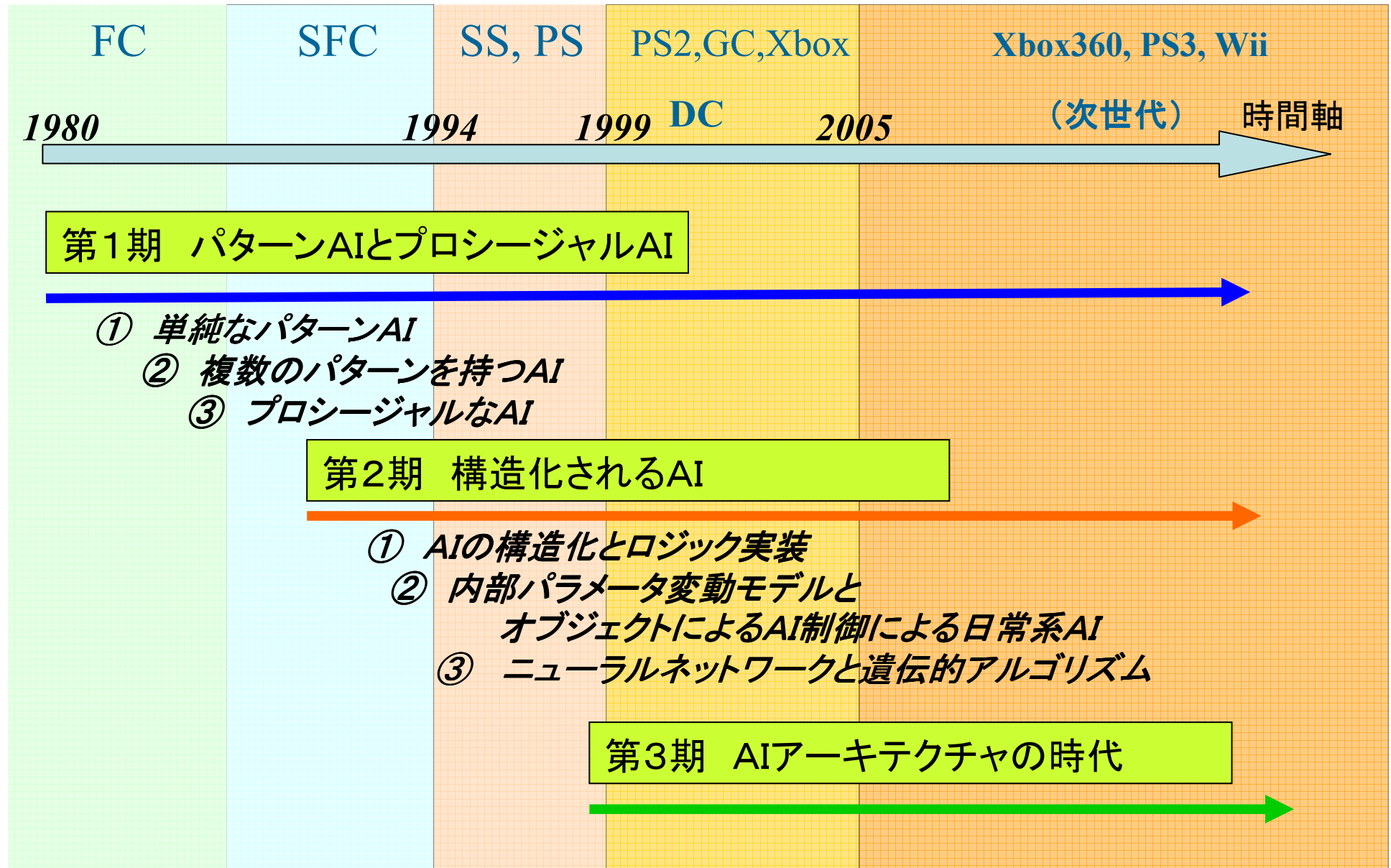
プランニングの実装技術を身につける(これからのゲームAIの最大の武器)。

ン

1851777

← 時間

ゲームにおけるエージェントの歴史



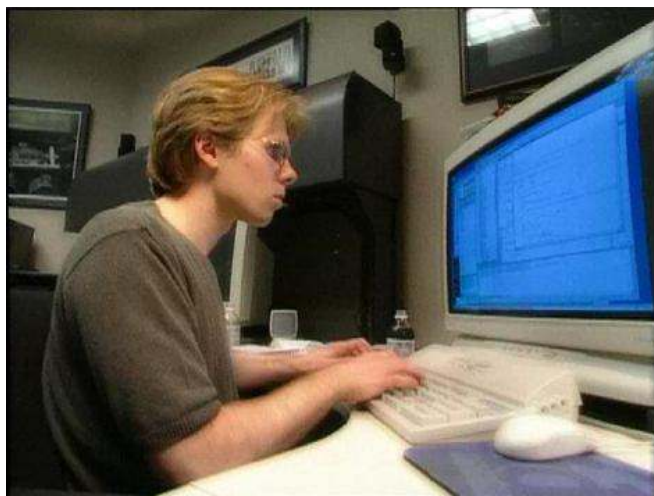
第1章 自律型エージェント (Autonomous Agent)

まとめと考察

プログラマーとエージェント

ハードコーディング型実装

プログラマーがステージごとに考えた思考を実装する



プログラミング
実装

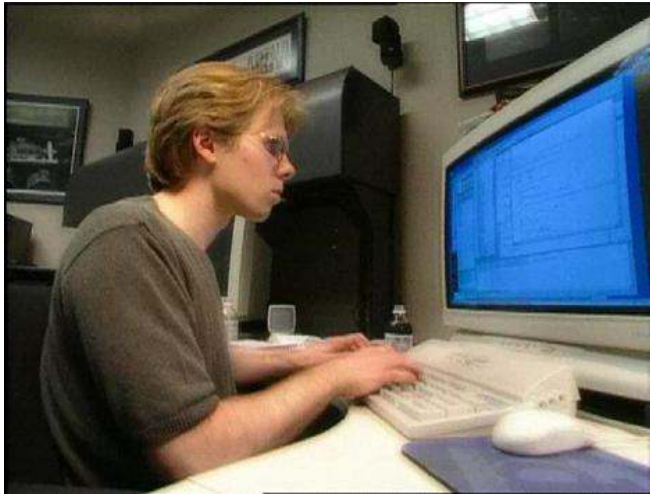


プログラマーが答えを解いて
それをAIに実装する

プログラマーとエージェント

自律型エージェント実装

どんなステージでもエージェント自身が考えて実装する



プログラミング
実装

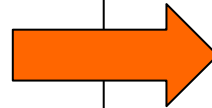


プログラマーはAI自身が問題を
解くように実装する

AI自身が答えを解く

現代のエージェント技術の流れ

ハードコーディング型実装



自律型エージェント実装

デジタルゲームのエージェントは、
「プログラマーがステージについて考えたことを実装する」方式から、
「エージェント自身に考えさせる」という方式に以降しつつある。

自律型エージェント (autonomous agent)

本授業の主旨

4つの言葉を覚えて帰りましょう

- (1) エージェント (Agent)
- (2) プロシージャル (Procedural)
- (3) メタAI (Meta-AI)
- (4) マルチエージェント (Multi-agent)

コンテンツ

第1部 ゲームとは何か？

第2部 デジタルゲームにおける人工知能技術

4つの言葉を覚えて帰りましょう

第一章 自律型エージェント (Autonomous Agent)

第二章 プロシージャル (Procedural)

第三章 メタAI (Meta-AI)

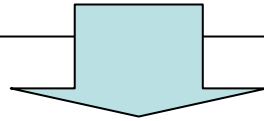
第四章 マルチエージェント (Multi-agent)

質疑応答

第2章 プロシージャル (Procedural)

プロシージャルとは？

プロシージャル(Procedural)
= 計算による、連続した操作による



デジタルゲームでは？

ゲーム空間、デジタル空間において、
自身の連続操作によって自律的な力を持つファクター

(例) 自動生成、自律的なAI

プロシージャル・コンテンツ・ジェネレーション
(Procedural Contents Generation = PCG)

プロシージャル制御
(Procedural AI)

プログラマーとゲーム世界

ハードコーディング型実装

プログラマーがステージの動作を定義する



プログラミング
実装

Game
WORLD

プログラマーが世界や
オブジェクトの動作を
全て定義する



プログラマーとゲーム世界

プロシージャルな世界(自律的に変化する世界)

ゲーム世界が自身が固有の法則をもって自律的に変化する



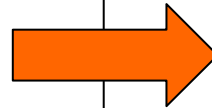
プログラミング
実装

Game
WORLD



現代のプロシージャルの流れ

ハードコーディング型実装



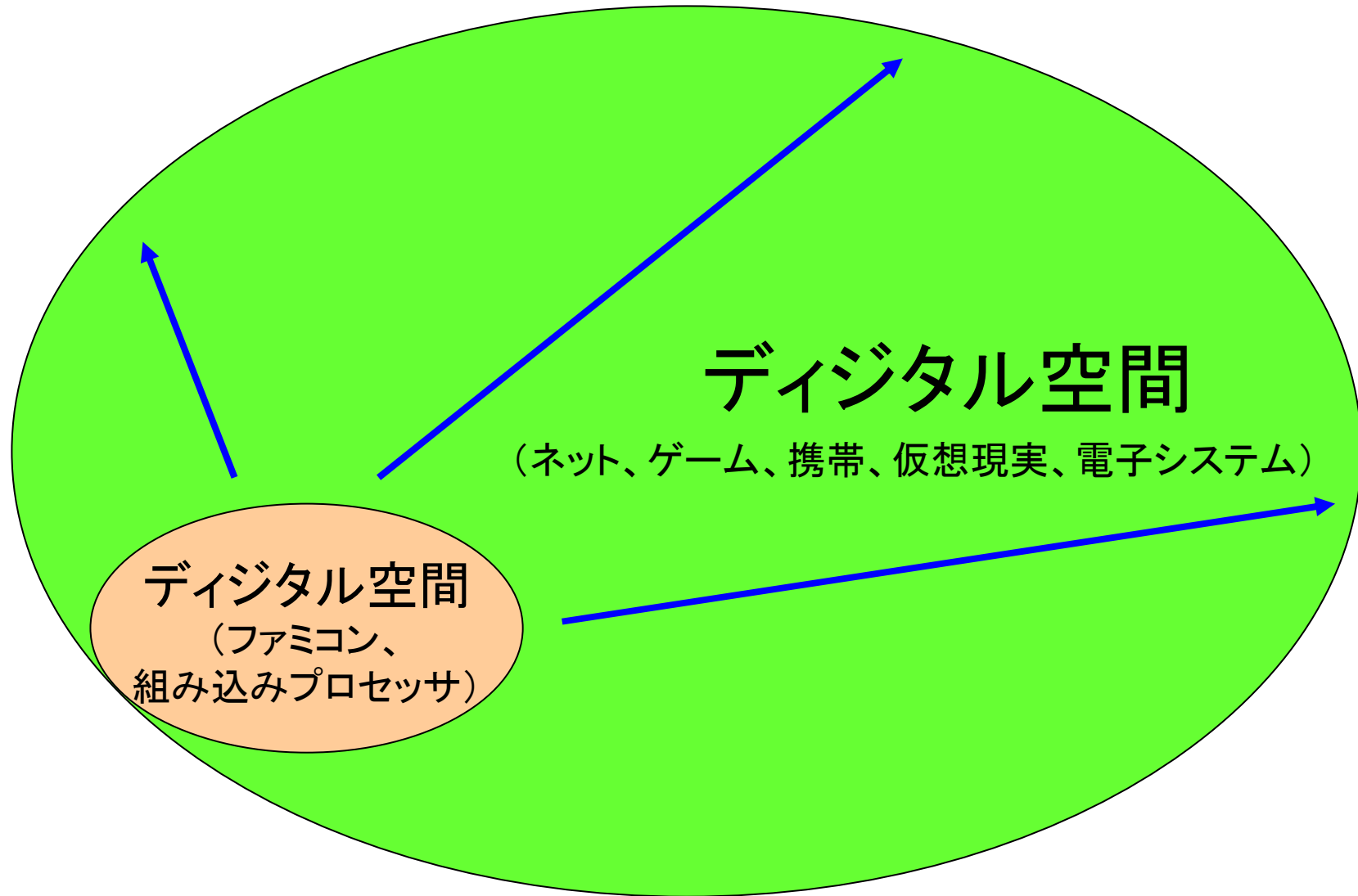
プロシージャル実装

デジタルゲームの世界は、
「プログラマーがステージのオブジェクトごとに法則を実装」方式から、
「普遍的な変化・発展の法則を世界に実装する」
という方式に以降しつつある

。

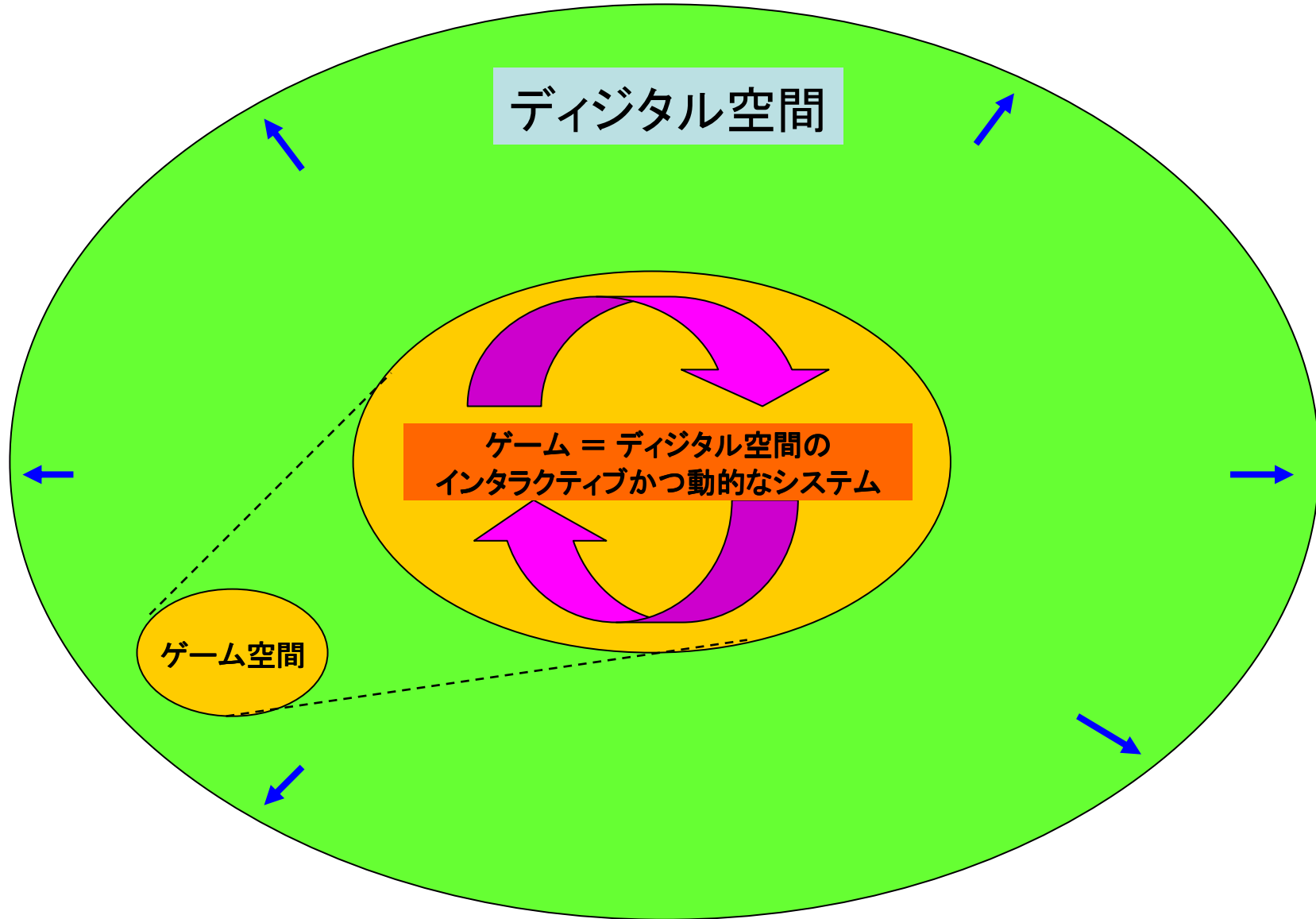
自律変化・発展する世界 (Procedural World)

デジタル空間とゲーム



この30年で我々が触れることが出来るデジタル空間は格段に増大した

デジタル空間とゲーム



ゲーム空間も格段に増大した

デジタル空間とゲーム

ツール・プログラマー



デジタル空間

ツール・プログラム

3Dツール

2Dツール

スクリプト言語

サウンドツール

エクセルなど

3Dモデル

2Dデータ

スクリプト

サウンド

ゲーム設定
データ

ゲーム
プログラム

ツール・チェイン

(ツール間の連携)

コンテンツ・データ

(データ間の連携)

コンテンツ・パイプライン

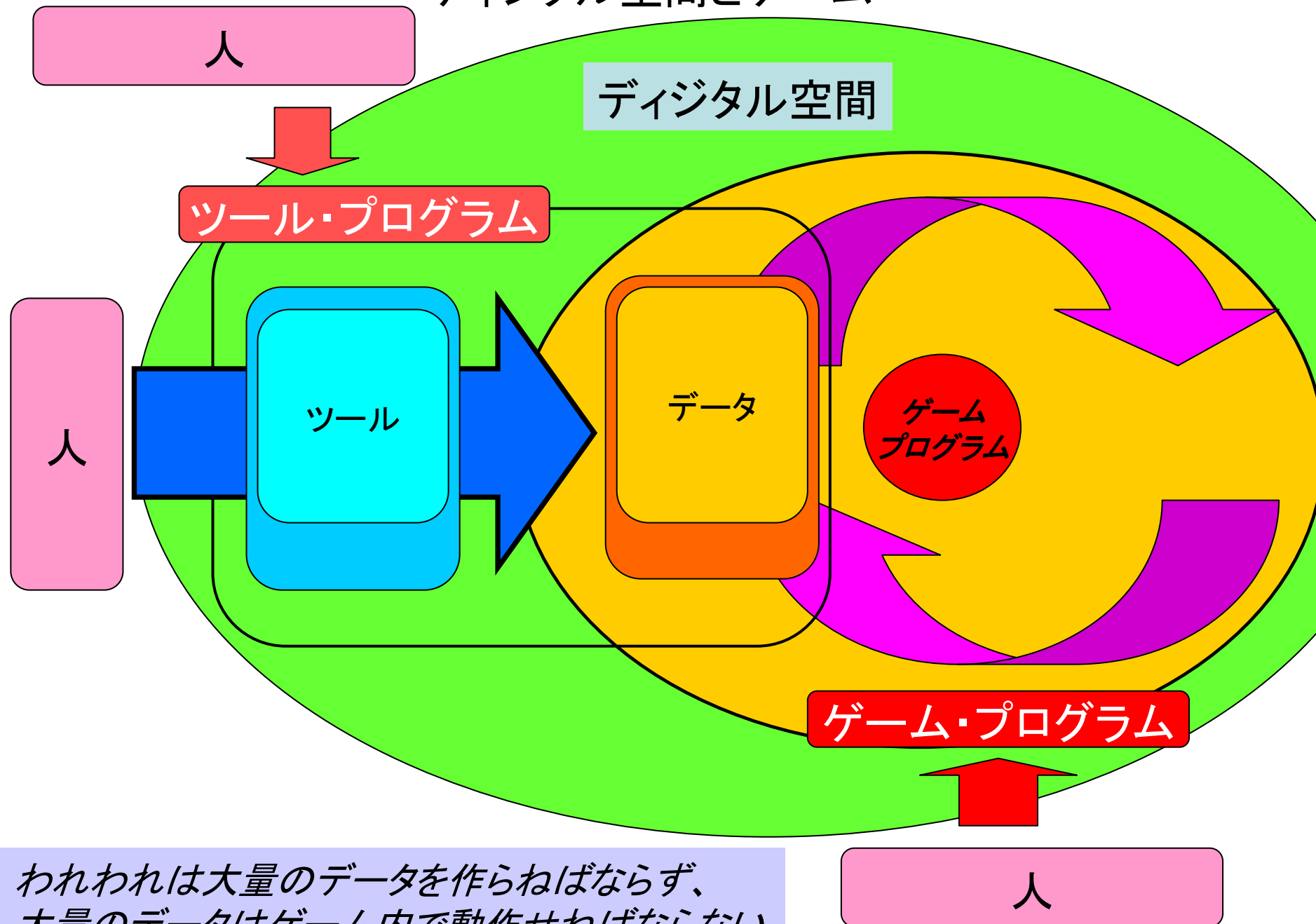
ゲーム・プログラム

開発者

ゲーム・プログラマー

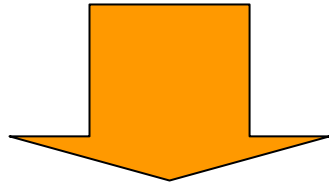


デジタル空間とゲーム



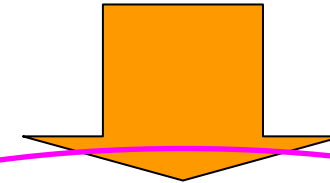
拡大するゲーム開発に対してゲーム開発者は何が欲しいか？

- ①ゲームデータを生成・発展・消滅させる力
- ②ゲームを動作させる力



- ①人の手でデータを生成する
- ②人の手で動作を定義する

マンパワー



- ①計算によってデータを生成する
- ②計算によって動作を定義する

プロシージャル

どちらかを選ぶかはゲームの主に規模による = 次世代機からは需要が急増

導入コスト: 人件費(コンスタント)

問題点 : 開発の規模に比例して
人手が必要
巨額の開発費があれば...

導入コスト: ソフトウェア・システムを
整備するまでの投資(2~3年)

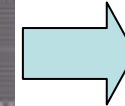
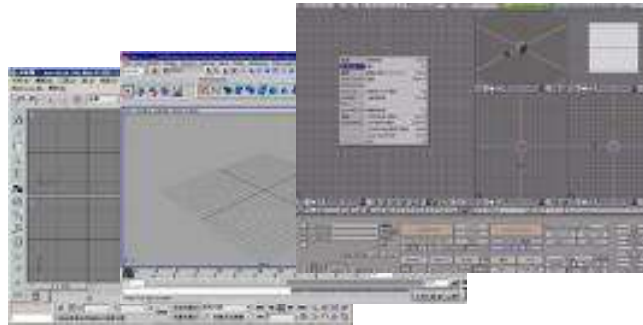
問題点 : 高度な技術レベル(簡単なものもある)
技術を使いこなすノウハウの必要性
中途半端ではケガをする

プロシージャル・コンテンツ・ジェネレーション

デジタル空間で草原を作りなさい

①アプローチ1

人の手によって
モデリングを行う

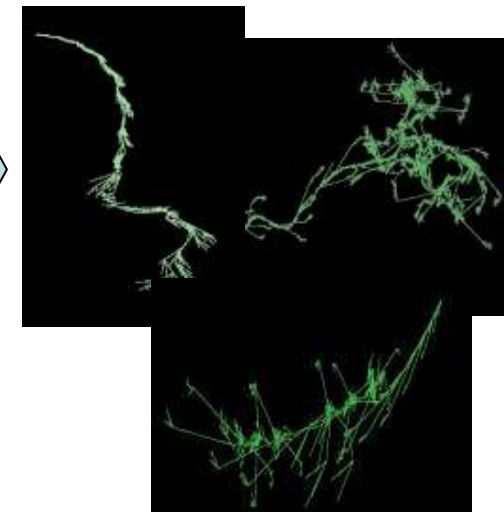
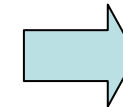
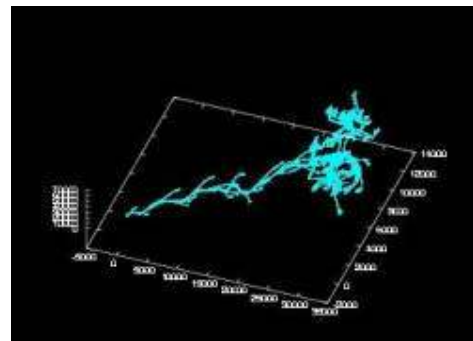


②アプローチ2

アルゴリズム

プログラムによって
モデルを自動生成する

プロシージャル



2003, Youichiro Miyake

```
void gen(){  
    if(unit_length > MAX_LENGTH) { add_stem(); return; }  
    add_rings(current_stem);  
  
    if(steps > 1000);return;  
    gen();  
}  
  
int make(){ gen(); return; }
```

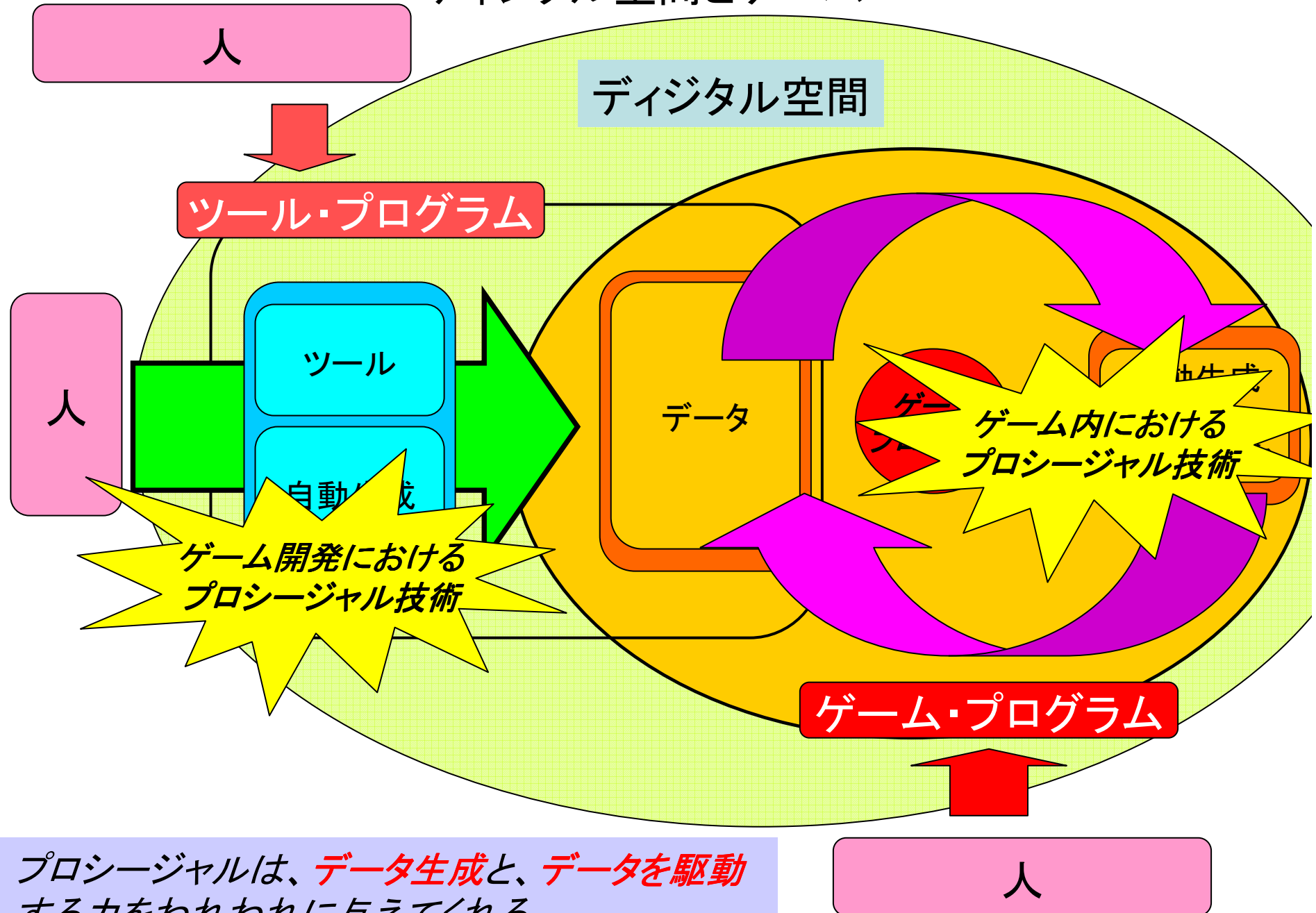
反復関数法

プロシージャル技術 → ゲームデザインへ

たくさんの演算によってモデルを生成する

コンテンツ自動生成 (Procedural Contents
Generation)

デジタル空間とゲーム



プロシージャルは、**データ生成**と、**データを駆動**する力をわれわれに与えてくれる。

プロシージャルの歴史: ダンジョン自動生成

1980

Rogue Clone
ローグライクゲーム

Rogue



1980

NetHack



1987

ティル・ナ・ノーグ

システムソフト



1987~

不思議のダンジョン

チュンソフト



1993~

Diablo

Blizzard



1996

Hellgate: London

flagshipstudio



2006

ブルードラゴン

Microsoft



2007

CODÉD ARMS

KONAMI



2005

TOEO

namco



2006

マビノギ

nexon



2004

FF11

Square Enix

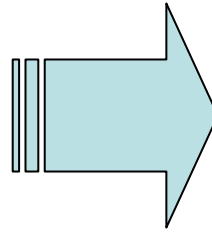


2007

MMOにおける応用

自動生成ダンジョンの面白さ

固定したダンジョン



自動生成ダンジョン

特徴

企画(米ではレベルデザイナー)
による入念な設計と仕掛け
何度プレイしても同じ

特徴

入るたびに変わるダンジョン
生成されるダンジョンのクオリティー
覚えゲーにならないですむ

ダンジョン生成法

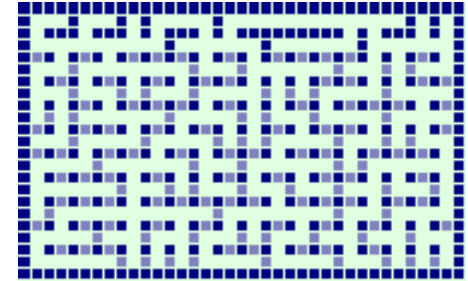
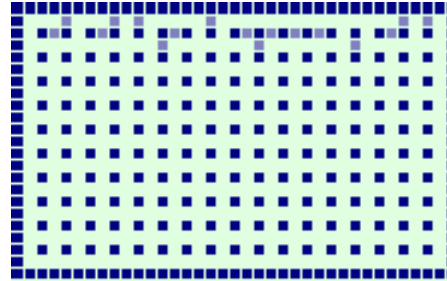
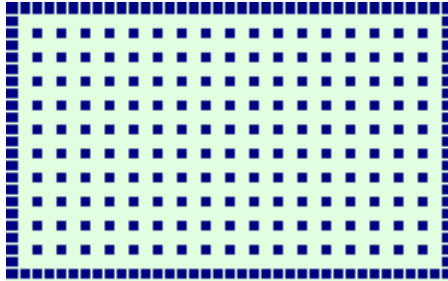
Rogue Clone
ローグライクゲーム

Rogue

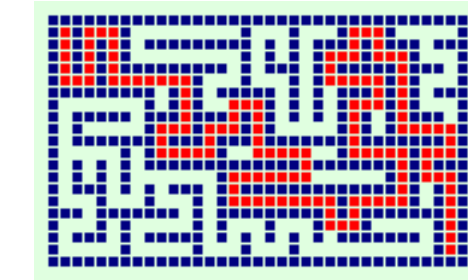
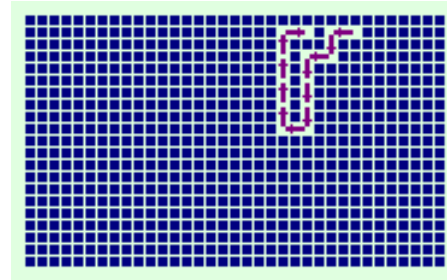
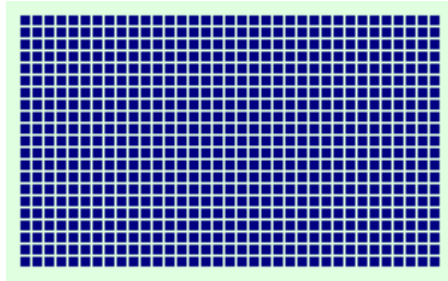
NetHack

迷路自動生成法

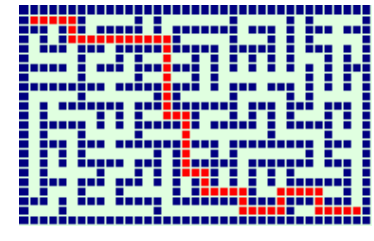
棒倒し法



穴掘り法



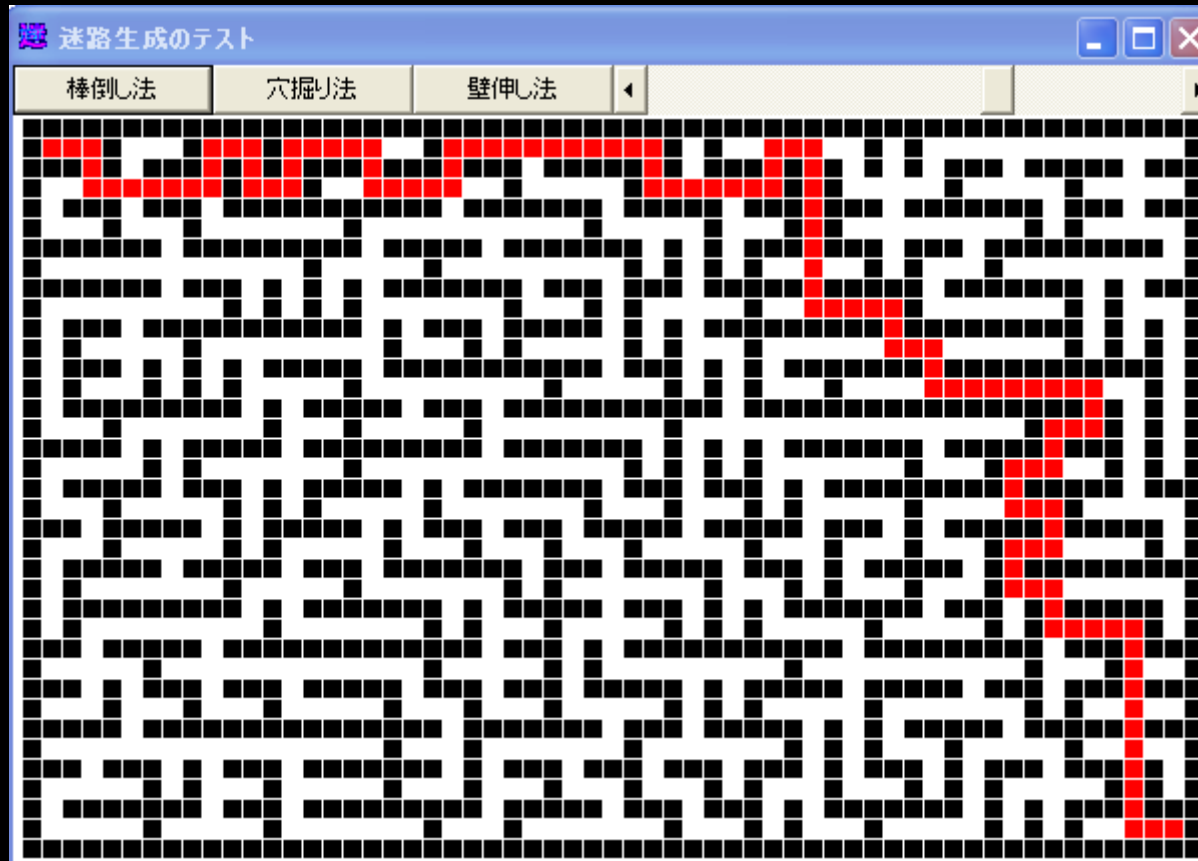
壁延ばし法



その他、いろいろな方がいろいろなところで独自のアルゴリズムを開発

[Ishida So,「迷路のプログラム」, 2005](#)

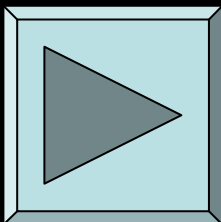
デモ



iMaze5

自動迷路生成

Ishida So, 「迷路のプログラム」, 2005



Rogue Clone
ローグライクゲーム

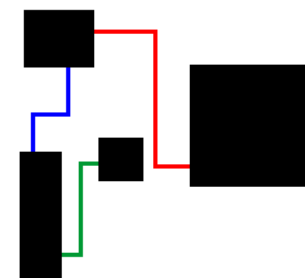
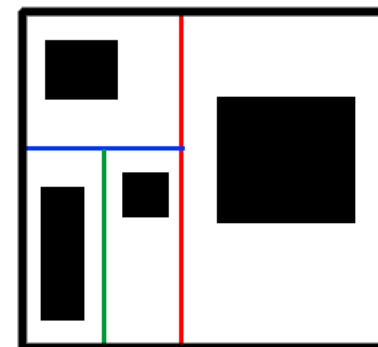
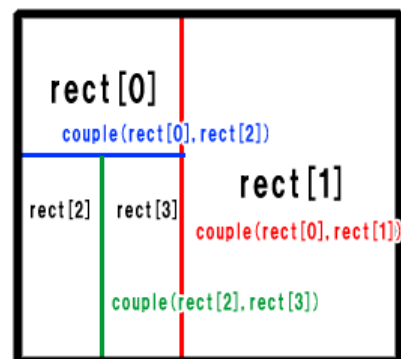
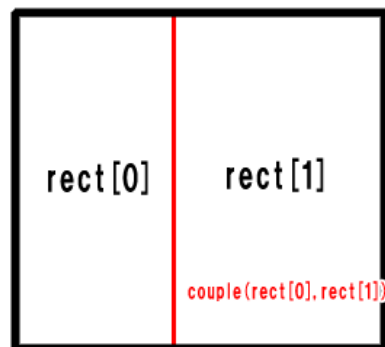
Rogue

NetHack

Racanhack

ダンジョン生成法

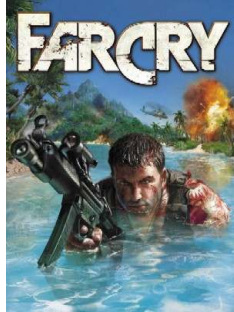
ダンジョン自動生成法



[源馬照明, 「Racanhack コード解説」, 2005](#)

プロシージャルなゲームエンジン

Far Cry(Crytek & Ubisoft)



CryEngine 1.0



Far Cry Instincts

Crysis(Crytek & EA)



CryEngine 2.0

Far Cry 2(Ubisoft Montreal & Ubisoft)



Dunia Engine



Frostbite Engine

次世代Battlefield用エンジン

Practical Example: Mountains Generation and Realistic Snow Accumulation



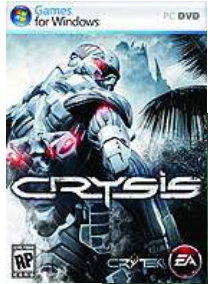
次世代Battlefieldシリーズ用エンジンFrostbite (4gamers)

<http://www.4gamer.net/news/history/2007.03/20070308211803detail.html>

CryEngine 2.0

Crysis における植物アニメーション自動生成

Procedural Vertex Animation



http://www.crytek.com/fileadmin/user_upload/cryengine2/CryENGINE2Features.pdf

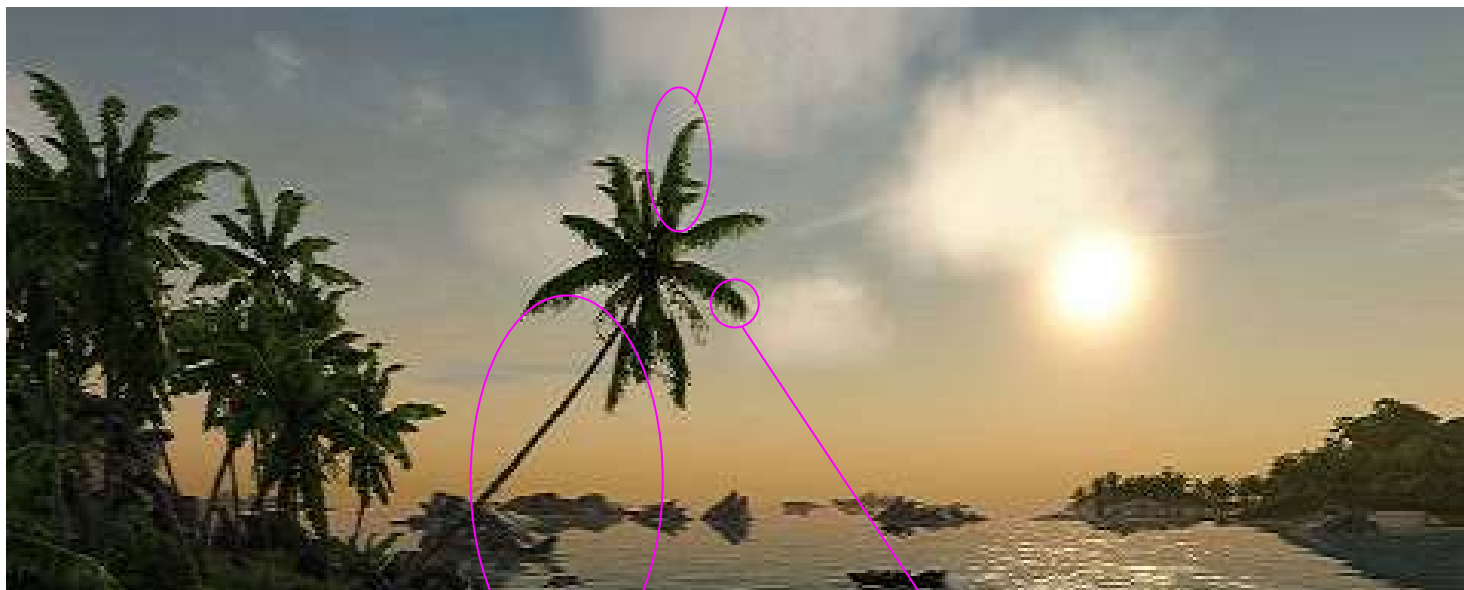
[Tiago Sousa, "Chapter 16: Vegetation Procedural Animation and Shading in Crysis", GPU Gems 3, 2007](#)

Finding Next Gen CryEngine2,

[http://ati.amd.com/developer/gdc/2007/Mittring-Finding_NextGen_CryEngine2\(Siggraph07\).pdf](http://ati.amd.com/developer/gdc/2007/Mittring-Finding_NextGen_CryEngine2(Siggraph07).pdf)

風に揺れる森 in Crysis

葉全体(leaves)

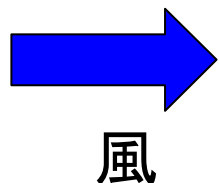


Finding Next Gen CryEngine2, [http://ati.amd.com/developer/gdc/2007/Mittring-Finding_NextGen_CryEngine2\(Siggraph07\).pdf](http://ati.amd.com/developer/gdc/2007/Mittring-Finding_NextGen_CryEngine2(Siggraph07).pdf)

全体(entire)

葉(leaf)

風と葉の硬さからモデルの曲がり具合を計算する



風



www.two-kings.de/.../dxgraphics10.html

×

各エッジ(辺)
に埋め込まれる
硬さ情報を
埋め込む



葉全体の硬さ

一枚一枚としての硬さ

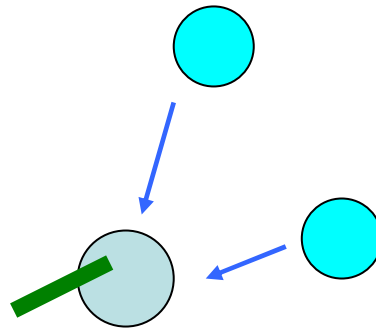
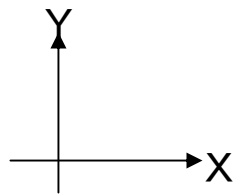
上の二つを足した総合のかたさ

α

Precomputed ambient occlusion

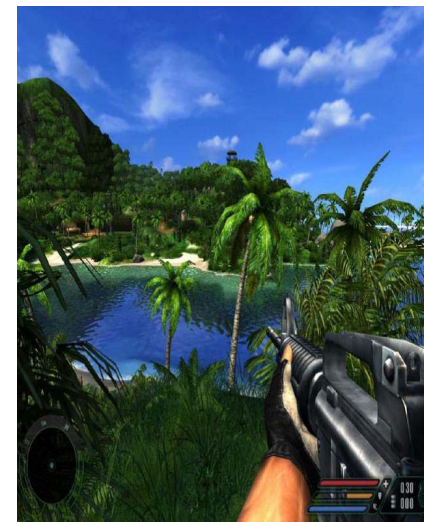
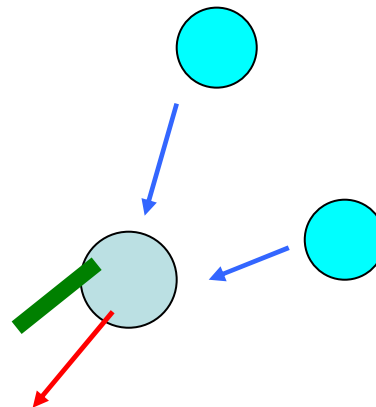
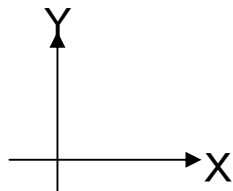
風に揺れる森 in Crysis

- ① その樹木に対する幾つかの風源の効果を足し合わせる



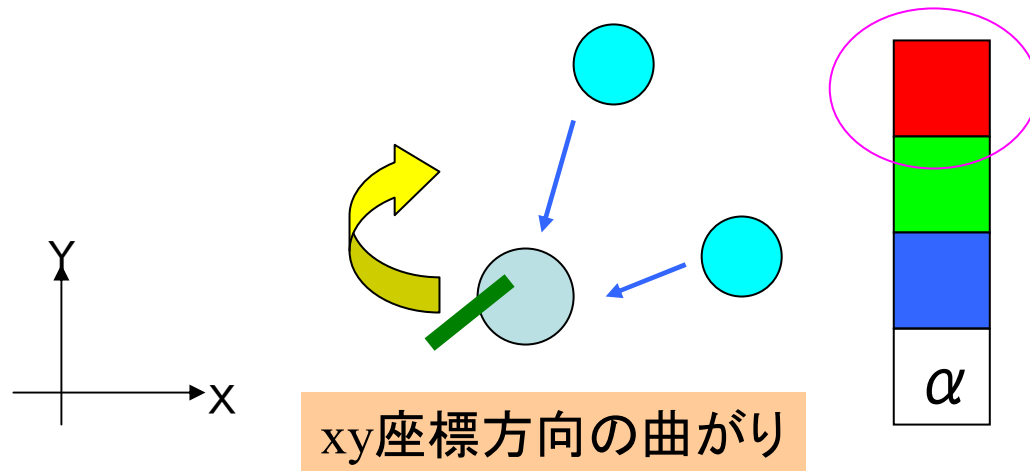
http://www.geimeris.com/wp-content/images/crysis_later.jpg

- ② 世界座標(xy)上の風ベクトルから、全体の曲げ方を決める。



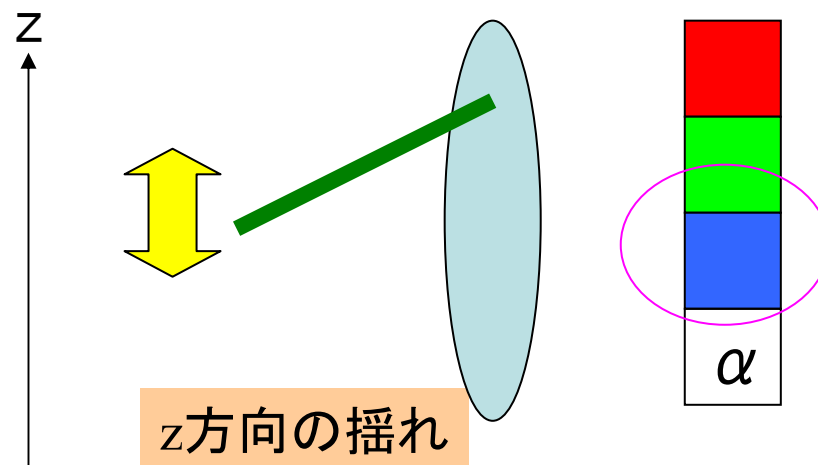
風に揺れる森 in Crysis

- ③ 葉のまとまりに対する曲がりを、風の強さだけから計算する



<http://www.crytek.com/technology/cryengine-2/specifications/>

- ④ 一枚の葉の揺れ(z-方向)を、風の強さだけから計算する



デモ



Procedural Vegetation Animation in Crysis

tiago_gpuGems3_1280_mpeg

[Tiago Sousa, "Chapter 16: Vegetation Procedural Animation and Shading in Crysis", GPU Gems 3, 2007](#)

References

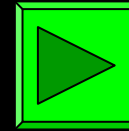
- (1) SpeedTree
<http://www.speedtree.com/>
- (2) Procedural vertex-animation(OceanWater)
http://b0rken.dk/hornet/shaders/proc_water.html
- (3) [Tiago Sousa, "Chapter 16: Vegetation Procedural Animation and Shading in Crysis", GPU Gems 3, 2007](#)
- (4) natural motion, <http://www.naturalmotion.com>
- (5) E3 2006: Indiana Jones Eyes-on,
<http://ps3.ign.com/articles/705/705489p1.html>
- (6) Finding Next Gen CryEngine2,
[http://ati.amd.com/developer/gdc/2007/Mittring-Finding_NextGen_CryEngine2\(Siggraph07\).pdf](http://ati.amd.com/developer/gdc/2007/Mittring-Finding_NextGen_CryEngine2(Siggraph07).pdf)

Figures on the pages are from these references.

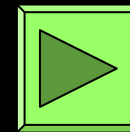
DUNIA ENGINE 2.0

DUNIA Engine デモ

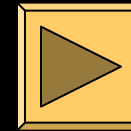
(I) Growing Vegetation



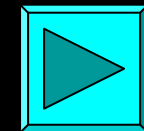
(II) Tree Generation



(III) Day and Night cycle



(IV) Realistic Weather System



[参考]データ自動生成技術を大胆に導入したFPS, 「Far Cry 2」の光と(4gamers)

<http://www.4gamer.net/games/047/G004713/20080222019/>

プロシージャルなゲームエンジン②



Frostbite Engine

① オブジェクト生成

(We can generate objects with procedural techniques

- Then use rules to deform / destroy / modify / move them
- Better interactivity)

② Semi-procedural surface shader

③ Procedural shader

④ Procedurally distributed on the fly

Practical Example: Mountains Generation and Realistic Snow Accumulation



GDC 2007 Frostbite “Rendering Architecture and Real-time Procedural Shading & Texturing Techniques”

[http://developer.amd.com/assets/Andersson-Tatarchuk-FrostbiteRenderingArchitecture\(GDC07_AMD_Session\).pdf](http://developer.amd.com/assets/Andersson-Tatarchuk-FrostbiteRenderingArchitecture(GDC07_AMD_Session).pdf)

GDC 2007 “The Importance of Being Noisy: Fast, High Quality Noise”, N. Tatarchuk

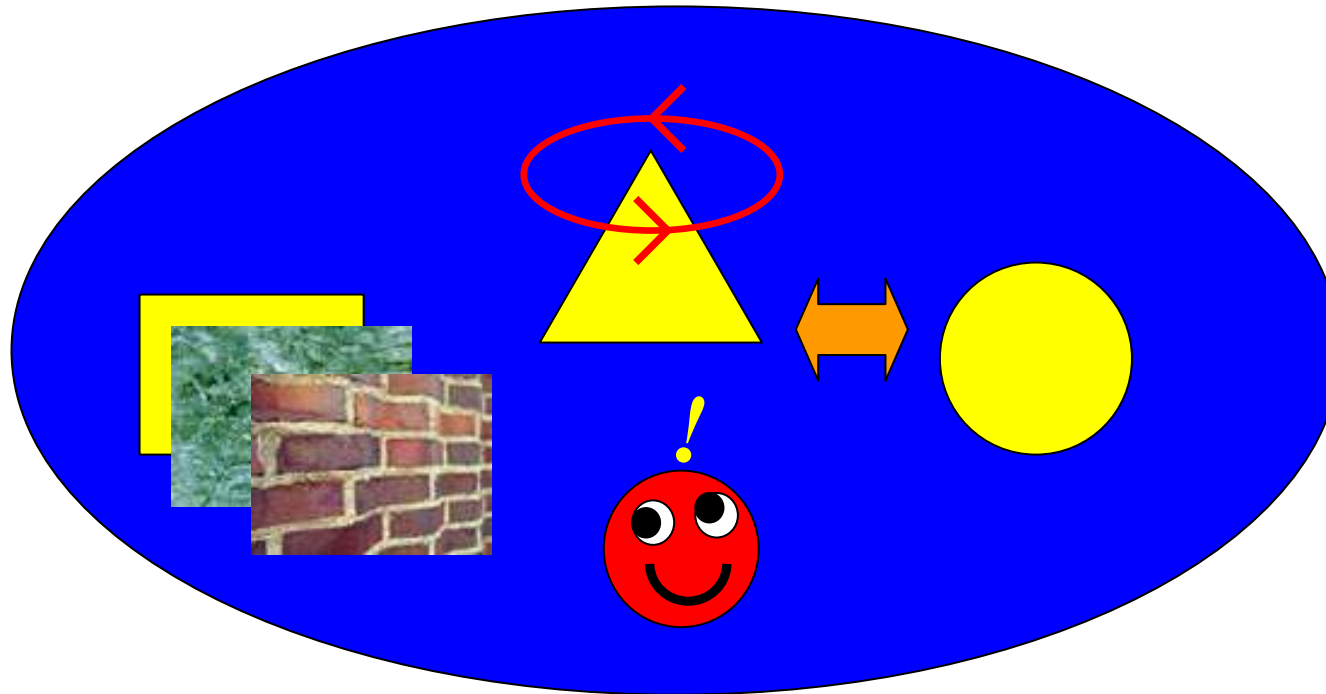
[http://developer.amd.com/Assets/Tatarchuk-Noise\(GDC07-D3D_Day\).pdf](http://developer.amd.com/Assets/Tatarchuk-Noise(GDC07-D3D_Day).pdf)

SIGGRAPH 2007 Johan Andersson “Terrain Rendering in Frostbite using Procedural Shader Splatting”

[http://ati.amd.com/developer/gdc/2007/Andersson-TerrainRendering\(Siggraph07\).pdf](http://ati.amd.com/developer/gdc/2007/Andersson-TerrainRendering(Siggraph07).pdf)

プロシージャル – 固定された世界から変化する世界へ

デジタル・ワールド



固定モデル 固定テクスチャー 固定アニメーション 固定AI 固定インタラクション

プロシージャル・コンテンツ・ジェネレーション

テクスチャー生成

Procedural Texture

モデル自動生成

Deformable Model

アニメーション生成

Animation Synthesis

即応インタラクション

物理

認識・判断するAI

Procedural AI

街生成

City Generation

会話・ストーリー生成

Story Generation

人工市場

Artificial Societies

究極地球シミュレーター

(世界シミュレーターという意味で)

Earth Simulator



プロシージャルの歴史: CG

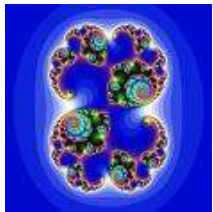
1980

1990

2000

フラクタル
幾何学

1975



フラクタル・イメージ
SIGGRAPH 1987

フラクタル幾何学を基礎とした
プロシージャルなモデル生成



1988

地形自動生成

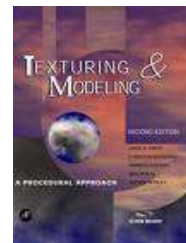
植物自動生成

雲自動生成

3Dモデル生成

テクスチャ自動生成

2Dモデル生成



1994

Terragen

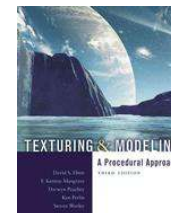
SpeedTree

natFX MAX

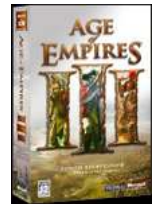
パリン・ノイズ

1997, 2002

Pro FX



2003

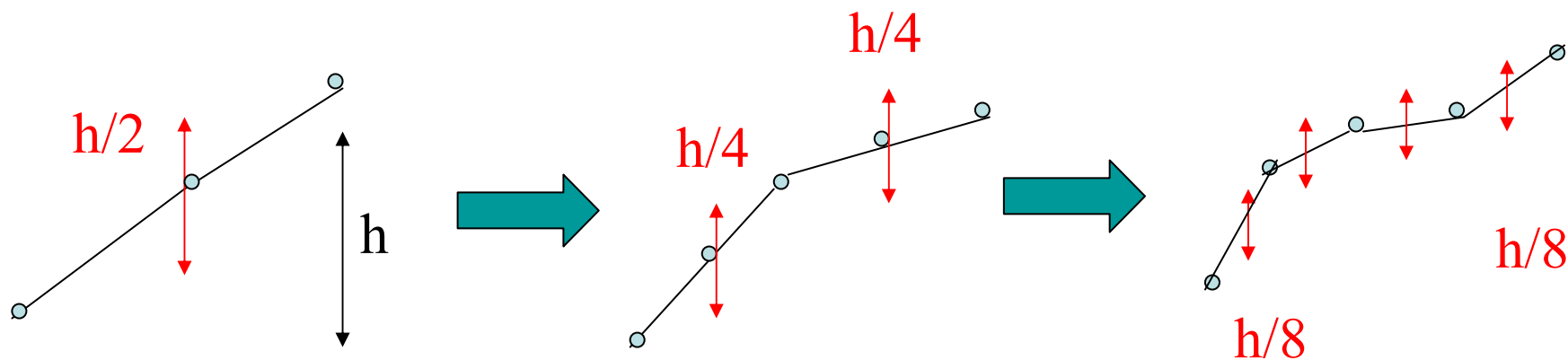


②グラフィックス自動生成

- I. 地形自動生成
- II. 植物自動生成
- III. 雲自動生成
- IV. パスデータ自動生成

地形自動生成

中点変位法



スケーリングに応じて、振れ幅を変えていく



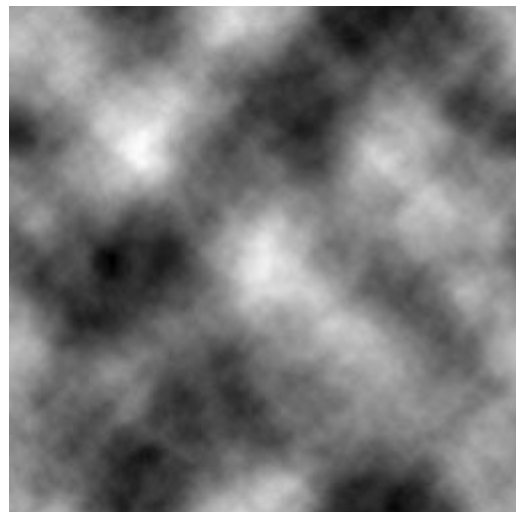
フラクタルイメージ理論とプログラミング (ハードカバー)
 ハインツ・オットー パイトゲン(編集), ディートマー ザウペ(編集), 山口 昌哉(翻訳)
 シュプリンガー・フェアラーク東京 (1990/08)



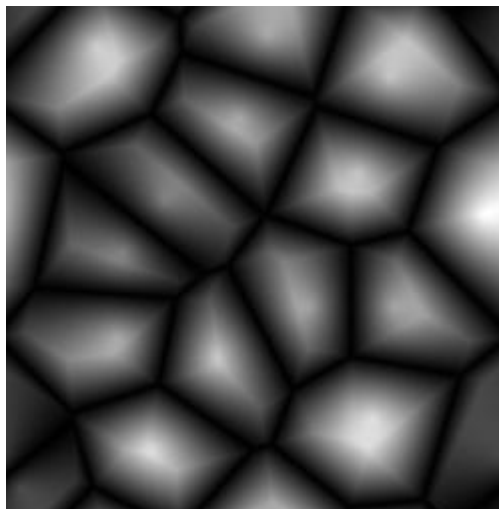
中点変位法によるフラクタル曲線の描画,
<http://nis-lab.is.s.u-tokyo.ac.jp/~hasimoto/applet/CG/FractalLine.html>

地形自動生成

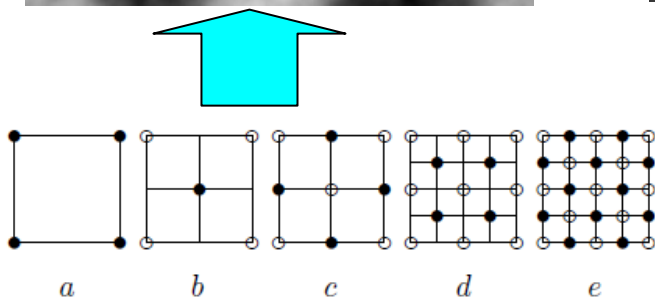
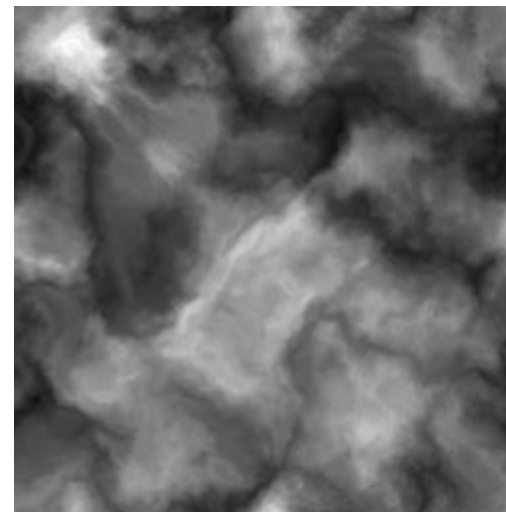
ノイズ法(濃い=低い、白い=高い)



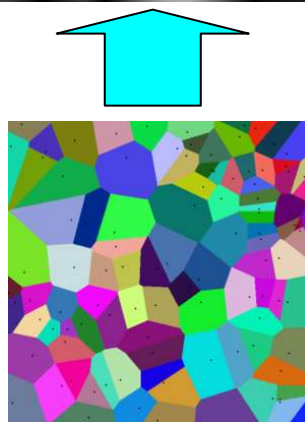
+



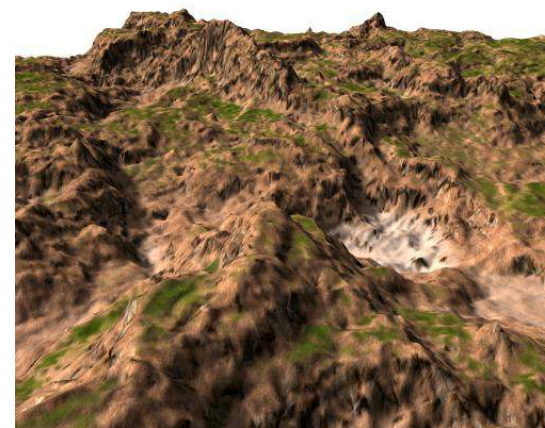
=



2次元中点変位法

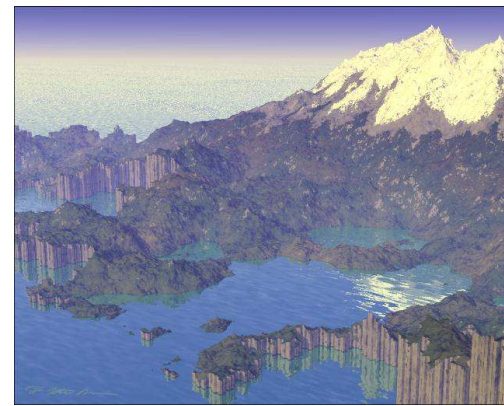


ボロノイ図



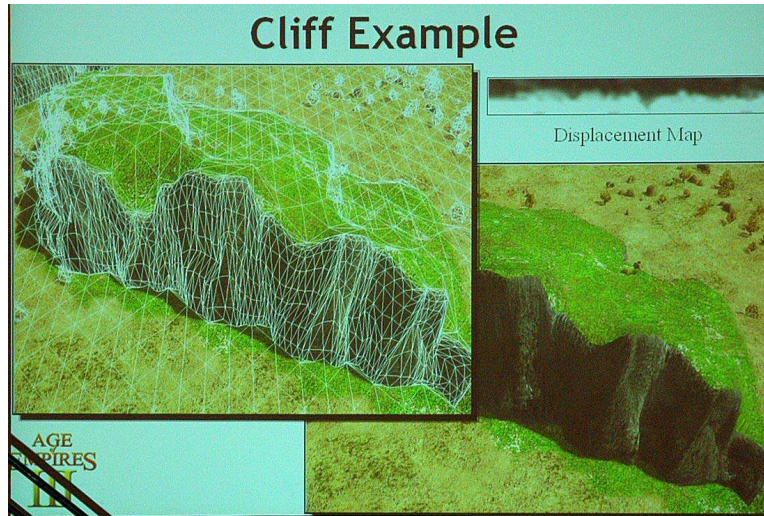
Jacob Olsen, Realtime Procedural Terrain Generation
http://oddlabs.com/download/terrain_generation.pdf

Ken Musgrave



<http://www.kenmusgrave.com/>

Age of Empires III における地形自動生成



西川善司,「3DゲームファンのためのAGE OF EMPIRESエンジン講座(後編)こだわりの影生成と算術合成されるディテール、次回作はXbox2?」, GAME Watch, 2005

References

(1) Jacob Olsen, Realtime Procedural Terrain Generation
http://oddlabs.com/download/terrain_generation.pdf

(2) Ken Musgrave
<http://www.kenmusgrave.com>

(3) Terragen (Planetside Software)
<http://www.planetside.co.uk/terrigen>

(4) Introversion Software, "Procedural Content Generation",
GameCareerGuide.com, 2007

http://www.gamecareerguide.com/features/336/procedural_content_.php

(5) 西川善司, 「3DゲームファンのためのAGE OF EMPIRESエンジン講座(後編)こだわりの影生成と算術合成されるディテール、次回作はXbox2?」,
GAME Watch, 2005

[h](#) *Figures on the pages are from these references.* [m](#)

②グラフィックス自動生成

- I. 地形自動生成
- II. 植物自動生成
- III. 雲自動生成
- IV. パスデータ自動生成

植物自動生成

研究分野としては
人工生命、CGにまたがる

Since 1968 A. Lindenmayer

L-system

文法規則

構成要素

F, +, -, [,]

規則

F $\rightarrow$ F[-F]F[+F][F]

0世代

1世代

2世代

F $\rightarrow$ F[-F]F[+F][F] $\rightarrow$ F[-F]F[+F][F][- F[-F]F[+F][F]] F[-F]F[+F][F][+ F[-F]F[+F][F]][F[-F]F[+F][F]]

解釈

F - まっすぐ進む

+ 30度左回転

- 30度右回転

[]内は個別のスタック



植物自動生成

Since 1968 A. Lindenmayer

L-system

文法規則

構成要素
規則

F, +, -, [,]

F $\rightarrow$ F[-F]F[+F][F]

0世代

1世代

2世代

3世代

4世代

5世代

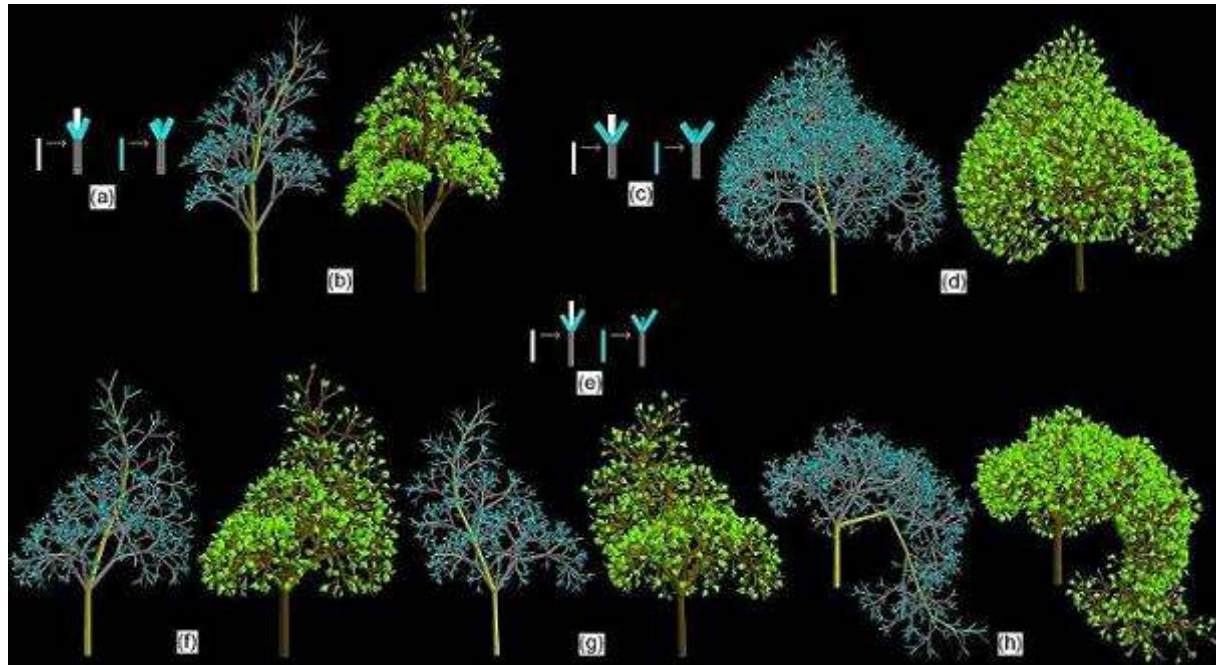
F $\rightarrow$

F[-F]F[+F][F]



The Sketch L-System: Global Control of Tree Modeling Using Free-form Strokes

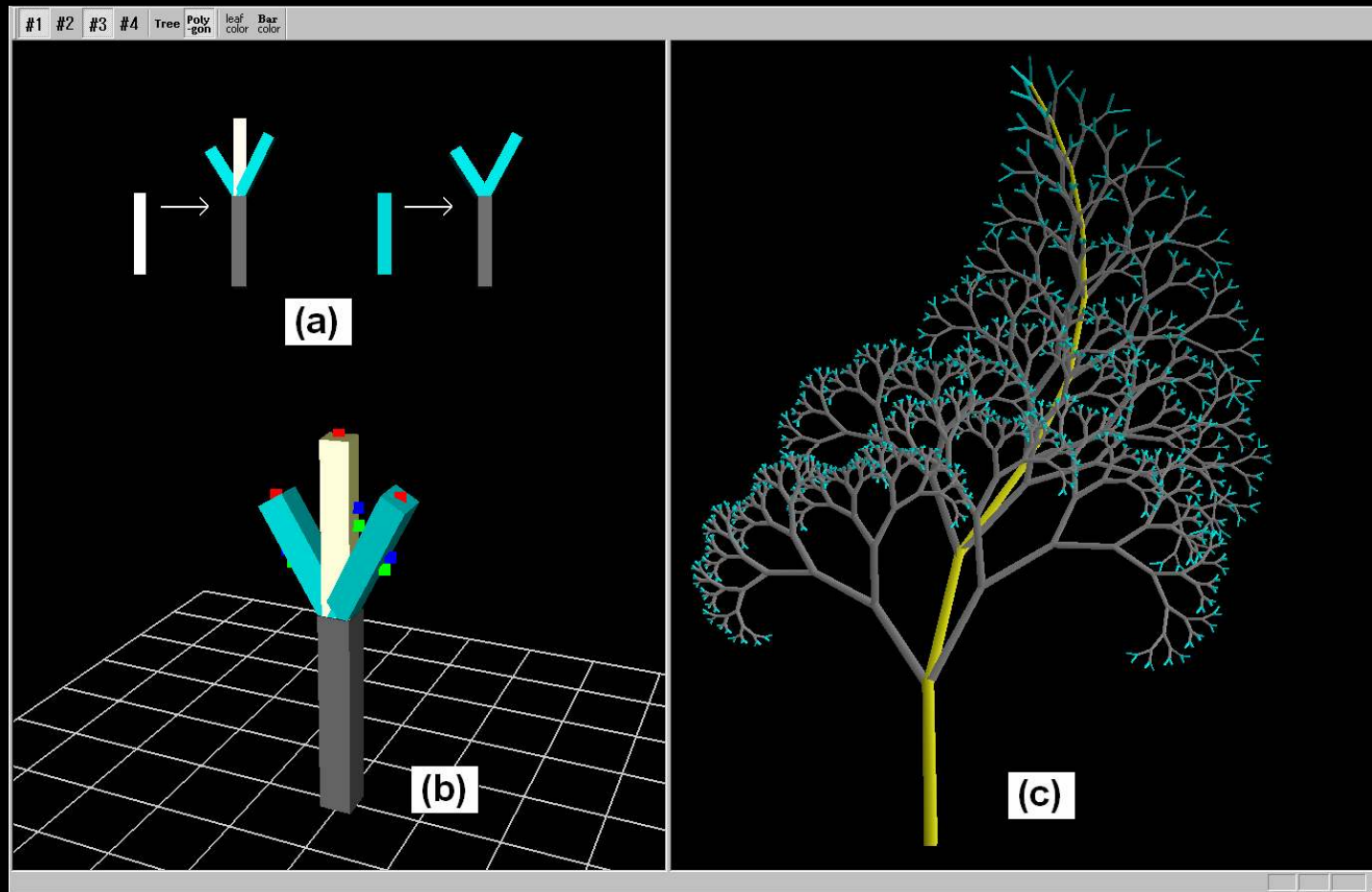
[Takashi Ijiri](#), [Shigeru Owada](#), [Takeo Igarashi](#).



L-system を用いて簡単な操作で木のモデルを作成するツール

The Sketch L-System:
Global Control of Tree Modeling Using Free-form Strokes
<http://www-ui.is.s.u-tokyo.ac.jp/~ijiri/SketchLSystem/index.html>

デモ



SG06_SketchLSystem

The Sketch L-system

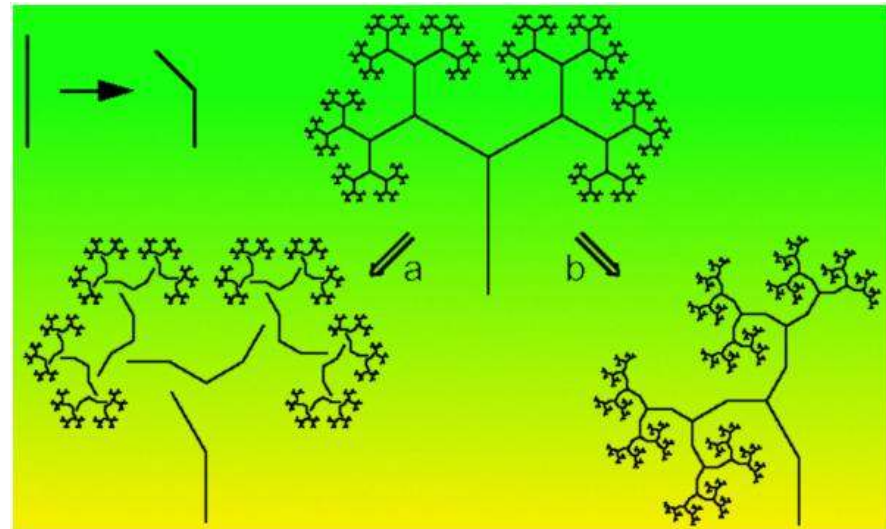
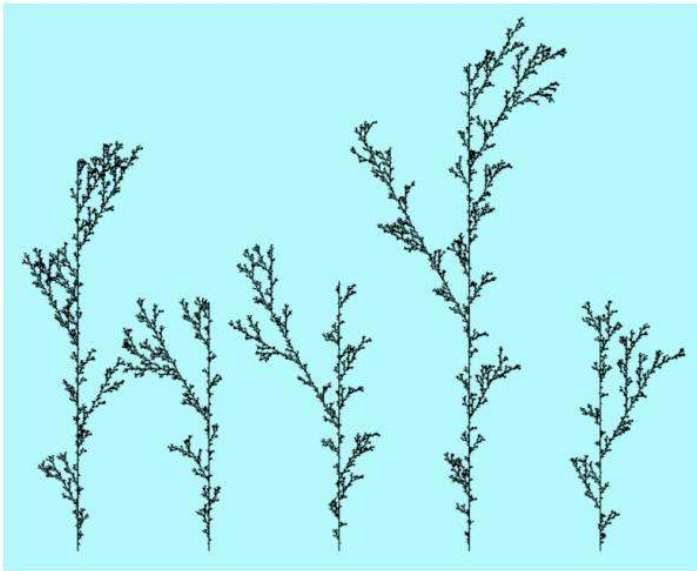
Takashi Ijiri Shigeru Owada Takeo Igarashi

The Sketch L-System: Global Control of Tree Modeling Using Free-form Strokes

<http://www-ui.is.s.u-tokyo.ac.jp/~ijiri/SketchLSystem/index.html>

発展: 確率的 L-system

構成要素	F, +, -, [,]
規則	$F \Rightarrow F[+F]F[-F]F$ 0.33
	$F \Rightarrow F[+F]F$ 0.33
	$F \Rightarrow F[-F]$ 0.33



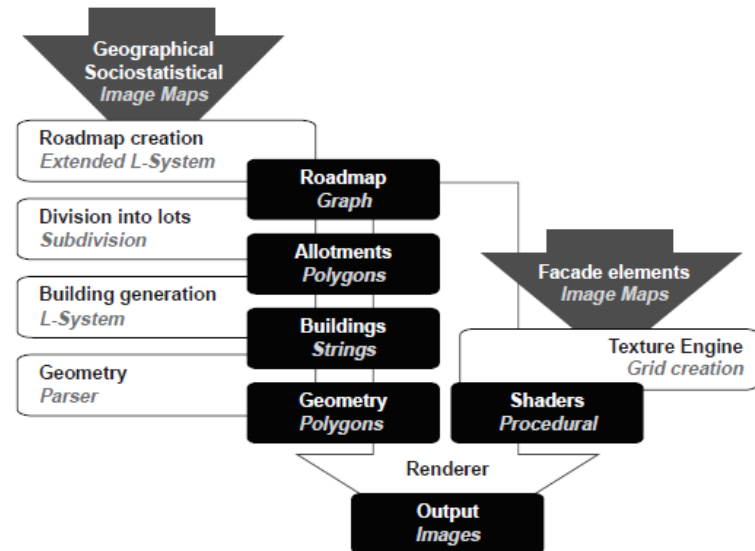
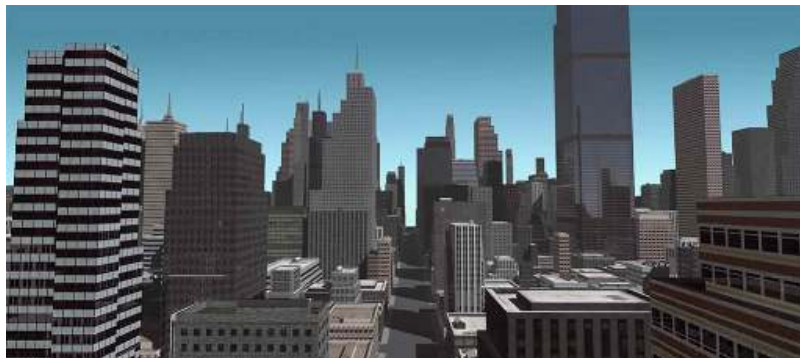
規則を変えればいろいろな模様が自動的に生成される

植物以外に使えないだろうか？ じっと見る

L-system による街の自動生成

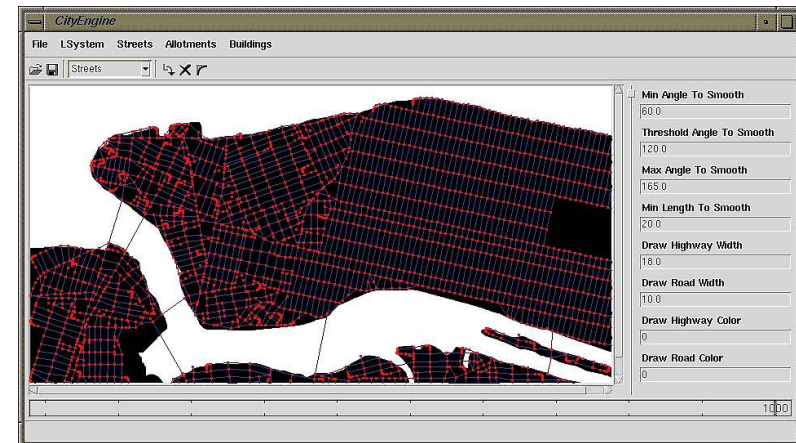
City Engine(central pictures)

<http://www.centralpictures.com/ce/>



Yoav I H Parish, Pascal Müller

<http://www.centralpictures.com/ce/tp/paper.pdf>



George Kelly, Hugh McCabe,

A Survey of Procedural Techniques for City Generation

<http://www.gamesitb.com/SurveyProcedural.pdf>

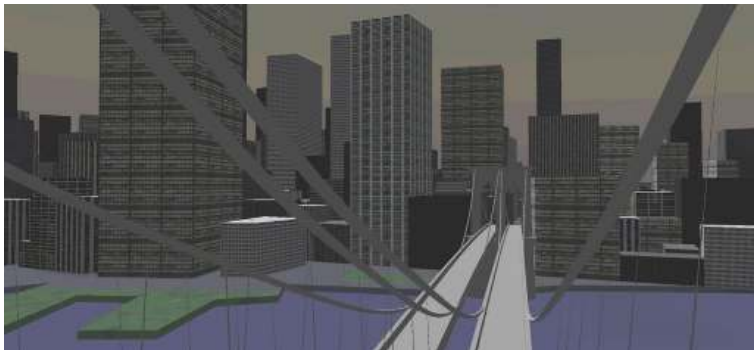
L-system による街の自動生成

City Engine(central pictures)



Yoav I H Parish, Pascal Müller

<http://www.centralpictures.com/ce/tp/paper.pdf>



References

- (1) **Simulating plant growth** by *Marco Grubert*
<http://www.acm.org/crossroads/xrds8-2/plantsim.html>
- (2) L-System Wiki
<http://en.wikipedia.org/wiki/L-system>
- (3) **The Sketch L-System:
Global Control of Tree Modeling Using Free-form Strokes**
<http://www-ui.is.s.utokyo.ac.jp/~ijiri/SketchLSystem/index.html>
- (4) Yoav I H Parish, Pascal Müller
<http://www.centralpictures.com/ce/tp/paper.pdf>
- (5) City Engine(central pictures) <http://www.centralpictures.com/ce/>
- (6) George Kelly, Hugh McCabe,
A Survey of Procedural Techniques for City Generation
<http://www.gamesitb.com/SurveyProcedural.pdf>

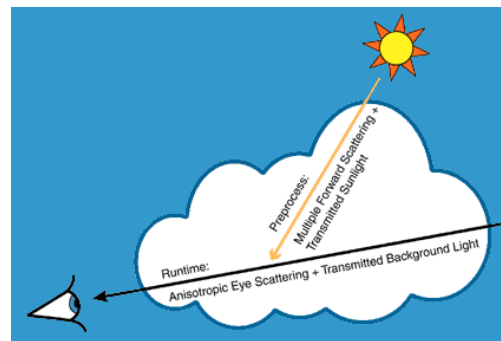
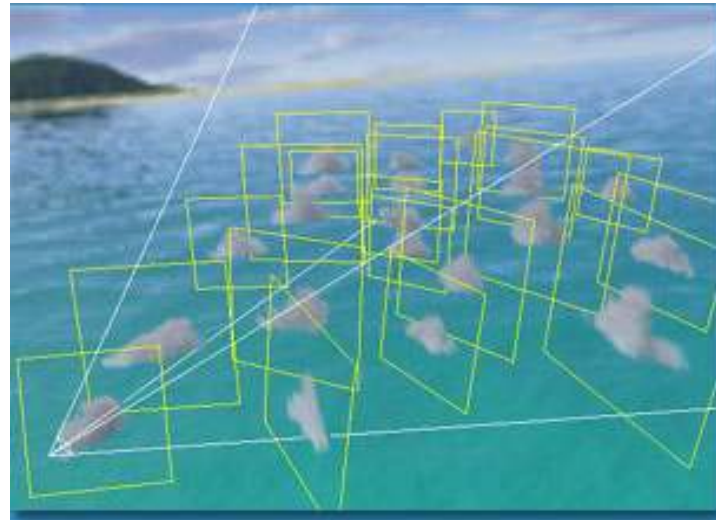
Figures on following pages are from these references.

②グラフィックス自動生成

- I. 地形自動生成
- II. 植物自動生成
- III. 雲自動生成
- IV. パスデータ自動生成



Microsoft Flight Simulator 2004

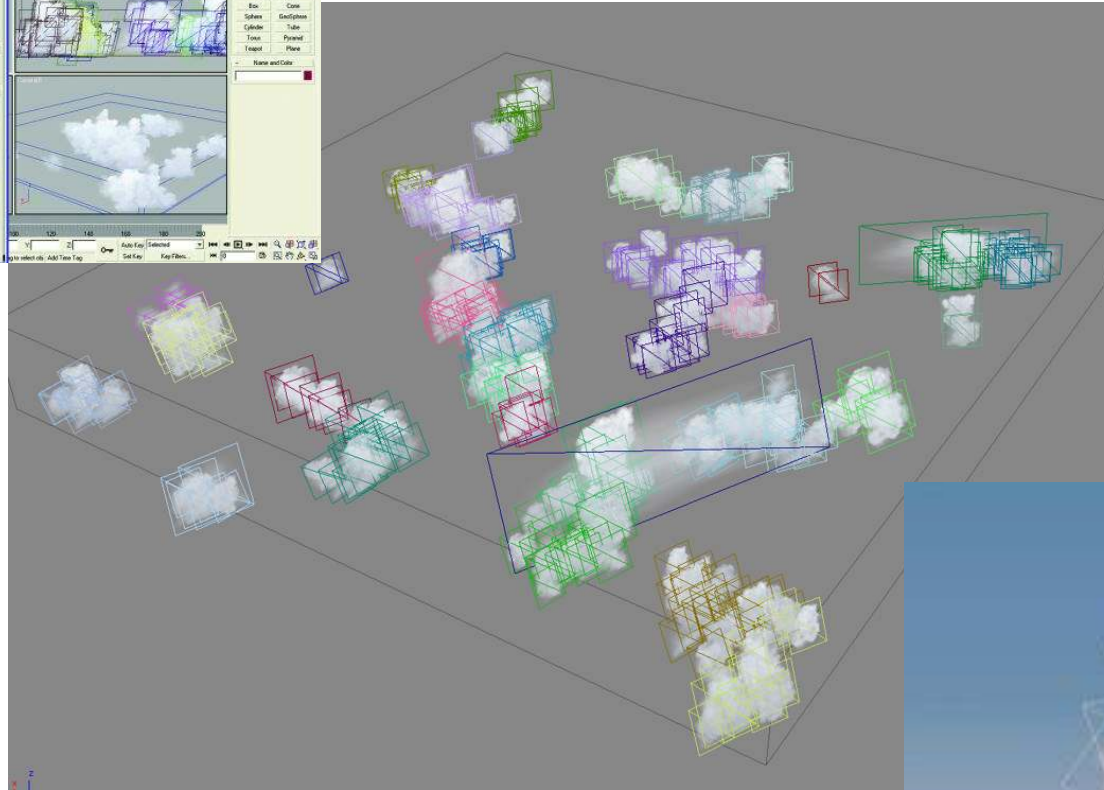
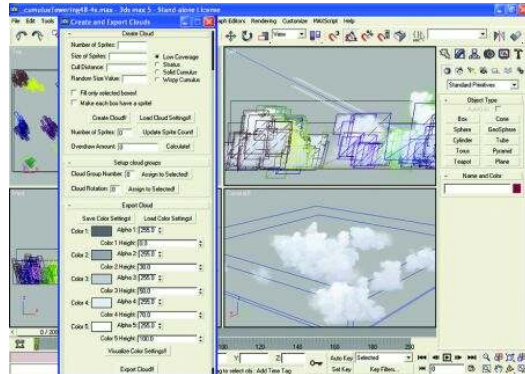


ビルボードからの様々なレンダリング効果を重ねてリアリティーのある空を再現
(Skywork、GDCで発表)

Mark.J.Harris, REALISTIC CLOUD ILLUMINATION

<http://www.markmark.net/clouds/index.htm>

Microsoft Flight Simulator 2004

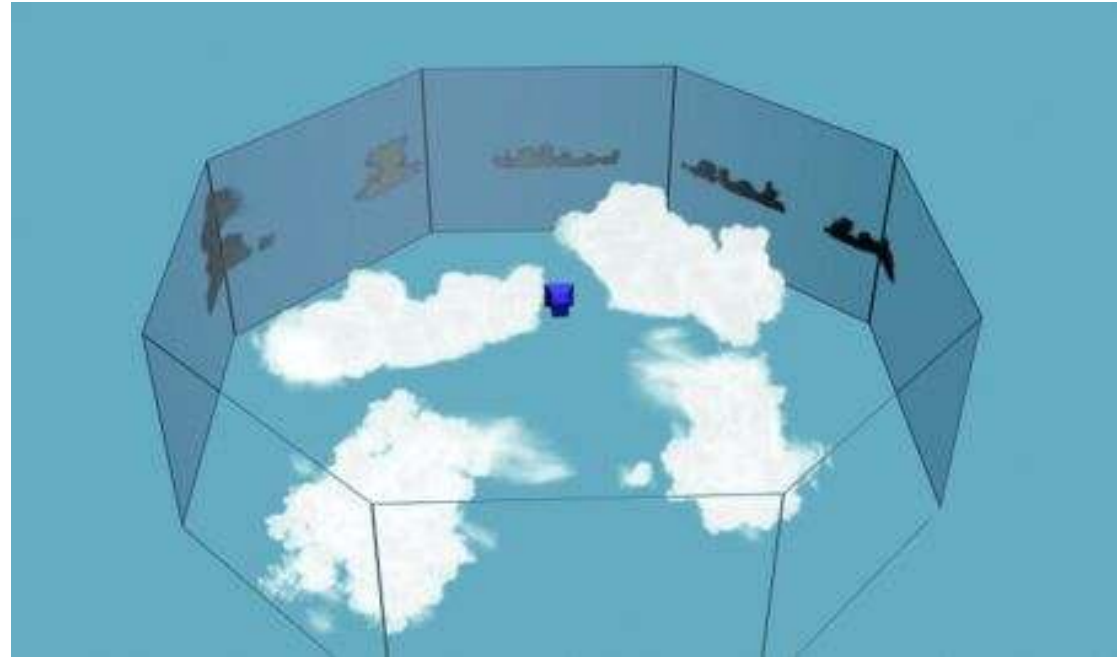
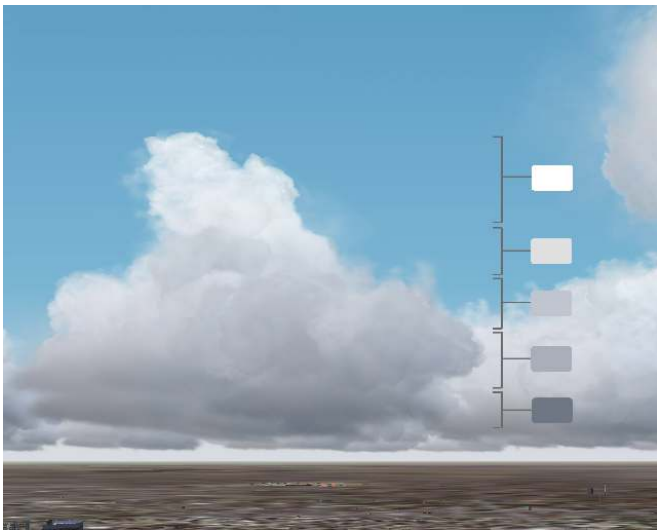
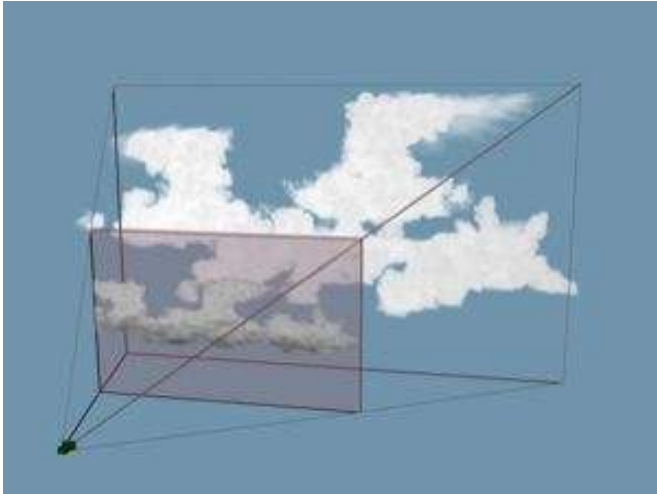


Niniane Wang ,Realistic and Fast Cloud Rendering

<http://jgt.akpeters.com/papers/Wang04/>

Mark Harris の方法を発展させる

Microsoft Flight Simulator 2004

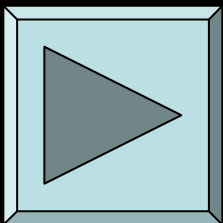


Niniane Wang ,Realistic and Fast Cloud Rendering

<http://jgt.akpeters.com/papers/Wang04/>

8つのリング状の背景とビルボードによる写し込み

デモ



Niniane Wang ,Realistic and Fast Cloud Rendering

<http://jgt.akpeters.com/papers/Wang04/>

Microsoft Flight Simulator 2004

References

(1) hugo.elias Cloud Cover

http://freespace.virgin.net/hugo.elias/models/m_clouds.htm

(2) Kim Pallister, Generating Procedural Clouds in real time on 3D HW

<http://www.intel.com/cd/ids/developer/asmo-na/eng/segments/games/20534.htm?page=11>

(3) David S. Ebert , Volumetric Procedural Implicit Functions

<http://www.csee.umbc.edu/~ebert/cloud>

(4) Mark.J.Harris, REALISTIC CLOUD ILLUMINATION

<http://www.markmark.net/clouds/index.htm>

(5) Niniane Wang , **Realistic and Fast Cloud Rendering**

<http://jgt.akpeters.com/papers/Wang04/>

(6) Perlin Noise

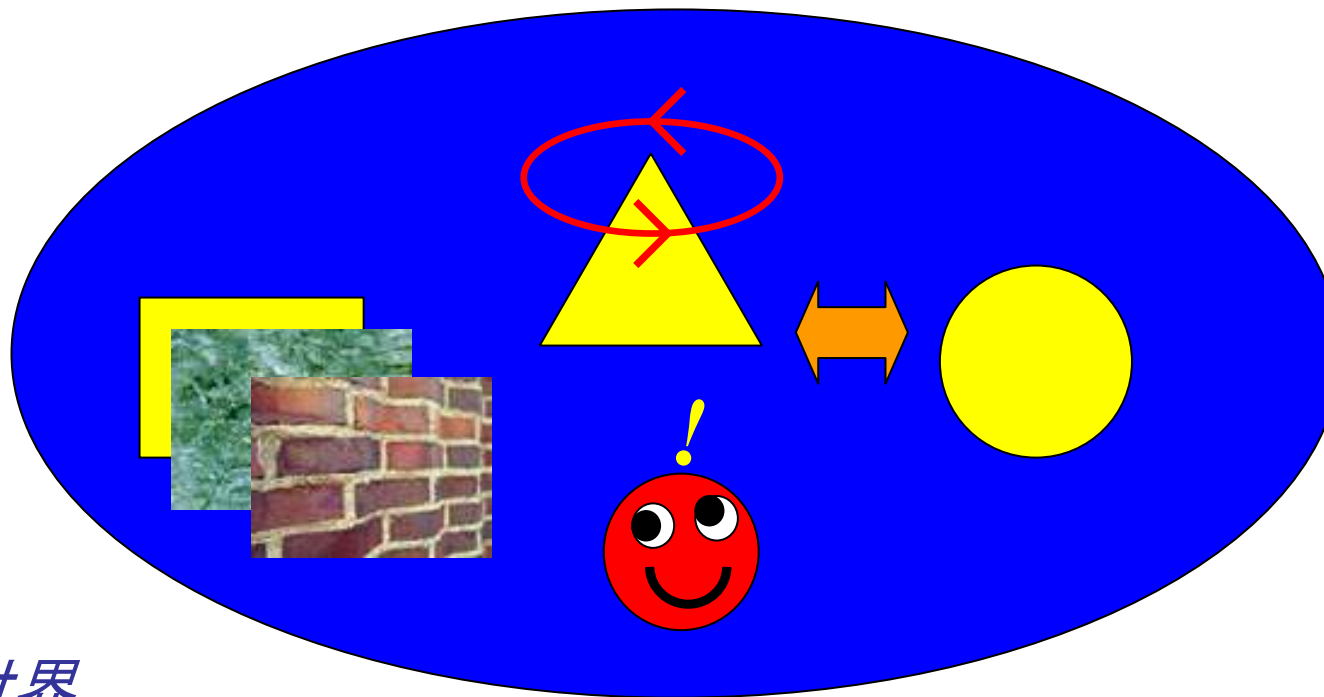
<http://m>

Figures on the pages are from these references.

in.ht

プロシージャル – 固定された世界から変化する世界へ

デジタル・ワールド



固定した世界

固定モデル 固定テクスチャ 固定アニメーション 固定AI 固定インタラクション



変形するモデル 生成するテクスチャ 生成するアニメーション 認識・判断するAI 即応インタラクション

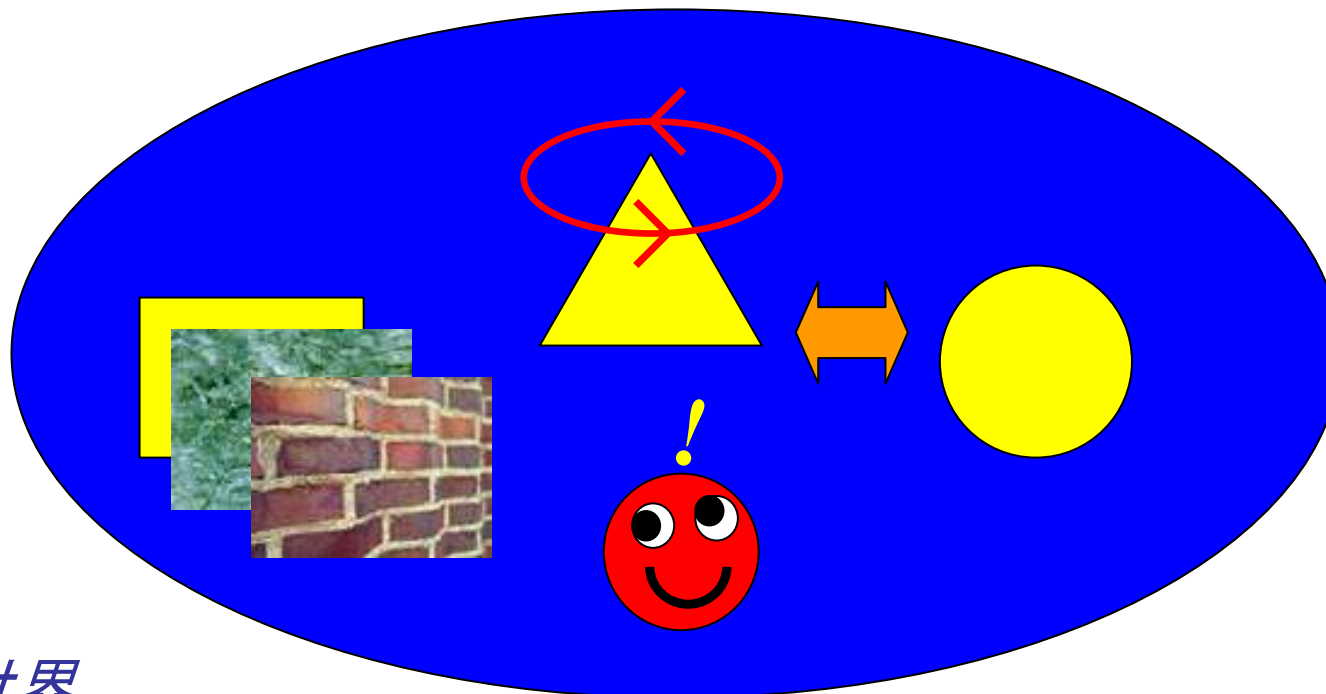
変化する世界

Procedural World

(物理)

プロシージャル – 固定された世界から変化する世界へ

デジタル・ワールド



固定した世界

固定モデル 固定テクスチャー 固定アニメーション 固定AI 固定インタラクション

変形するモデル 生成するテクスチャー アニメーション生成 認識・判断するAI 即応インタラクション

変化の世界

Procedural World (全てが計算される世界)

(物理)

プロシージャルとは？

プロシージャル(Procedural)
= 計算による、連続した操作による



デジタルゲームでは？

ゲーム空間、デジタル空間において、
自身の連続操作によって自律的な力を持つファクター

プロシージャル・コンテンツ・ジェネレーション
(Procedural Contents Generation = PCG)

プロシージャル制御
(Procedural AI など)



(質はともかく) プロシージャル技術は、デジタル空間で
コンテンツを自動生成、自動制御する力を与えてくれる

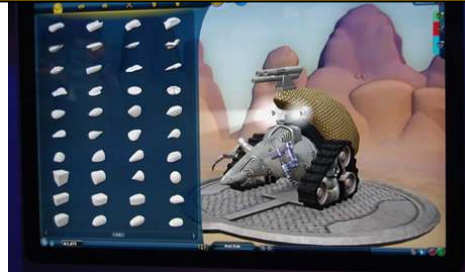
Sporeにおけるプロシージャル・ユーザー・アクション

簡単な操作で3Dクリーチャー・モデルからアニメーションまでを作成する

ユーザー作成コンテンツ (User Generated Contents, **UGC**)



キャラクターを作成するエディター



車を作成するエディター



都市(要塞)を作成するエディター

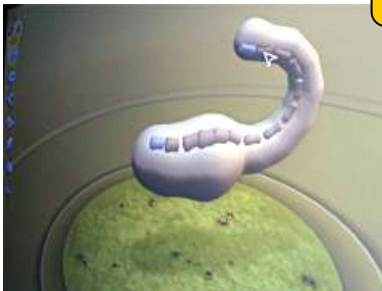


宇宙船を作成するエディター

生物、建物、惑星、主要なものは全てユーザーが生成可能

シミュレーション
(Sim Everything !)

単細胞から宇宙までの進化シミュレーションゲーム



単細胞フェーズ



部族フェーズ



文明化フェーズ



ギャラクティックフェーズ

<http://www.4gamer.net/news/history/2006.05/20060511195155detail.html>

<http://www.gamespark.jp/modules/news/index.php?p=6614>

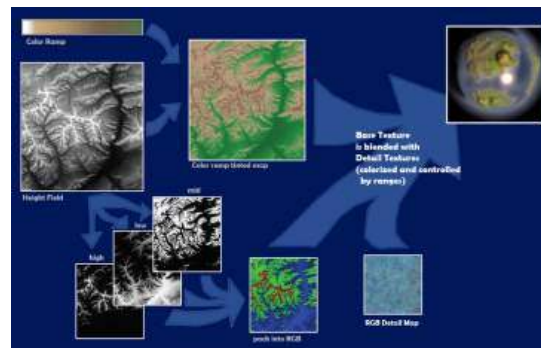
<http://www.watch.impress.co.jp/game/docs/20080814/spore1.htm>

「Spore」はプロシージャル技術の結晶

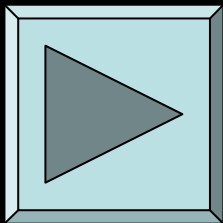
グラフィック、オブジェクト、サウンド、あらゆる分野においてプロシージャル技術を応用する次世代のゲーム



- ① テクスチャ (SIGGRAPH2007)
- ② オブジェクト配置 (SIGGRAPH2007)
- ③ モデル生成 (SIGGRAPH2007)
- ④ モデル変形 (SIGGRAPH2007)
- ⑤ アニメーション生成 (SIGGRAPH2008)
- ⑥ 音楽プロシージャル (GDC2008初公開)



Procedural Brushing Demo

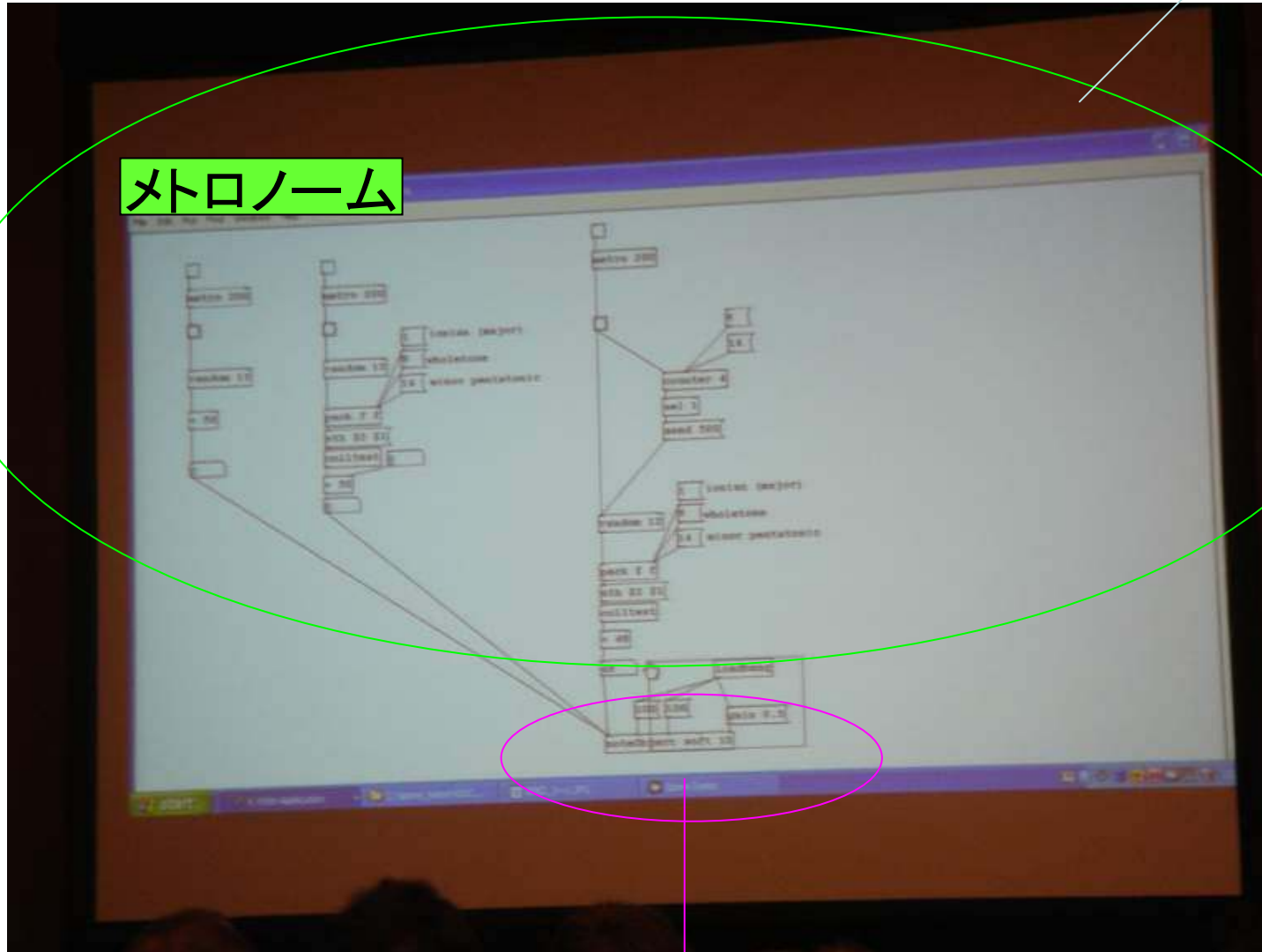


3rd spore demonstration at SIGGRAPH 07 (YouTube)

Pd とは？

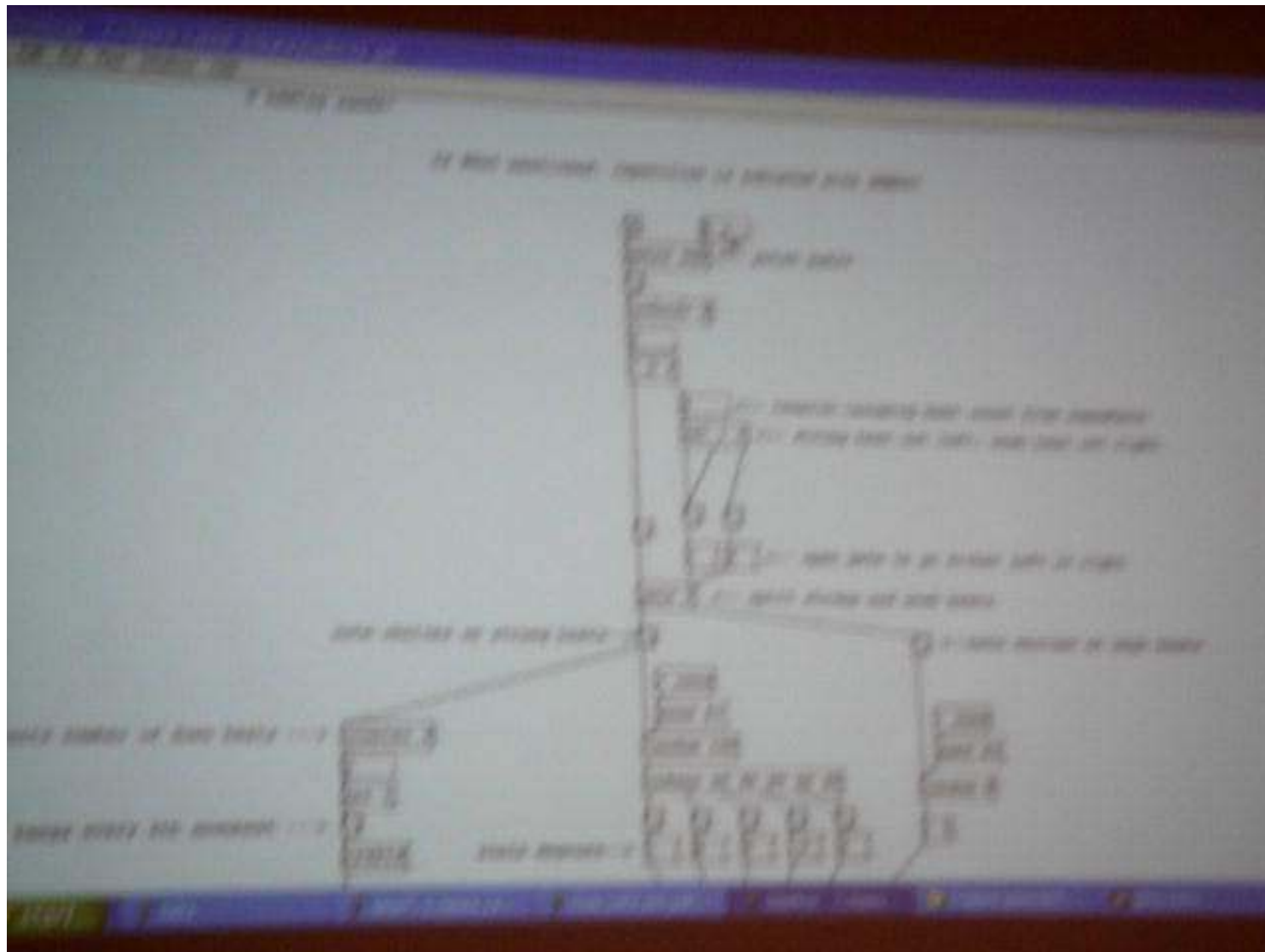
演奏者にあたるAI

メトロノーム



音を鳴らすところ

EAPd



伝染病



Plague Outbreak – World of Warcraft – September 2005

<http://www.mmocrunch.com/2007/12/04/top-5-most-memorable-events-in-mmorpg-history/>

Spore におけるプロシージャル技術

UGC (User Generated Contents, ユーザー作成コンテンツ) に
PCG (Procedural Contents Generation, 自動生成技術) を
応用することで、 ユーザーにコンテンツを産出す力を与え、
小さな労力で、大きなコンテンツを作り出す面白さを実現した。

References

- (1) カーネギー・メロン大学の [Andrew J. Willmot](http://www.cs.cmu.edu/~ajw/s2007/) 博士のHPページのSIIGRAPH2007のコーナーにまとめられています。
<http://www.cs.cmu.edu/~ajw/s2007/>
- (2) [E3 2006 #011] E3 2006 最大の話題作？ ウィル・ライトの「Spore」
<http://www.4gamer.net/news/history/2006.05/20060511195155detail.html>
- (3) Steve Capell et al., Interactive Skeleton-Driven Dynamic Deformations
<http://grail.cs.washington.edu/projects/deformation>
- (4) Development of Spore
(このサイトによると、デモシーナーたち、著名なフラクタル関係の技術者を Maxis は集めているようです)
http://en.wikipedia.org/wiki/Development_of_Spore
- (5) Spore の Procedural Music については、WEB に動画などがアップされているので、**Spore Procedural Music** などで検索してみよう！

Figures on the pages are from these references.

プログラマーとゲーム世界

ハードコーディング型実装

プログラマーがステージの動作を定義する



プログラミング
実装

Game
WORLD

プログラマーが世界や
オブジェクトの動作を
全て定義する



プログラマーとゲーム世界

プロシージャルな世界(自律的に変化する世界)

ゲーム世界が自身が固有の法則をもって自律的に変化する



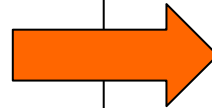
プログラミング
実装

Game
WORLD



現代のプロシージャルの流れ

ハードコーディング型実装



プロシージャル実装

デジタルゲームの世界は、
「プログラマーがステージのオブジェクトごとに法則を実装」方式から、
「普遍的な変化・発展の法則を世界に実装する」
という方式に以降しつつある

。

自律変化・発展する世界 (Procedural World)

本授業の主旨

4つの言葉を覚えて帰りましょう

- (1) エージェント (Agent)
- (2) プロシージャル (Procedural)
- (3) メタAI (Meta-AI)
- (4) マルチエージェント (Multi-agent)

プログラマーとゲーム

ハードコーディング型実装

ゲームを作ってリリース。ゲーム性は固定。



プログラミング
実装

GAME



プログラマーとゲーム

動的に変化するゲーム

リアルタイムにユーザーの特性やゲーム状態を解析して
自分自身(ゲーム性)を変化させるゲーム



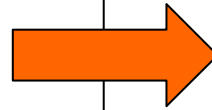
プログラミング
実装

GAME



メタAIの出現

万人向けに
完成したバランスのゲーム



ユーザーに応じた
ゲームがその場で生成

デジタルゲームの世界は、
「誰にでも楽しめるバランスを取れるゲーム」に対して、
「ユーザー特性やスキルに応じたゲームをリアルタイム制御で生成」
するメタAI技術が台頭しつつある。

ゲーム性をリアルタイムにコントロールするメタAI (Meta AI)

デジタルゲームの簡易モデル

基本的にこの4つの要素の
何れにも、AI(知的機能)を
組み込むことが出来る。

開発者

ゲームの意図

ゲームシステム
(ゲーム・ロジック)

ゲーム・デザイン

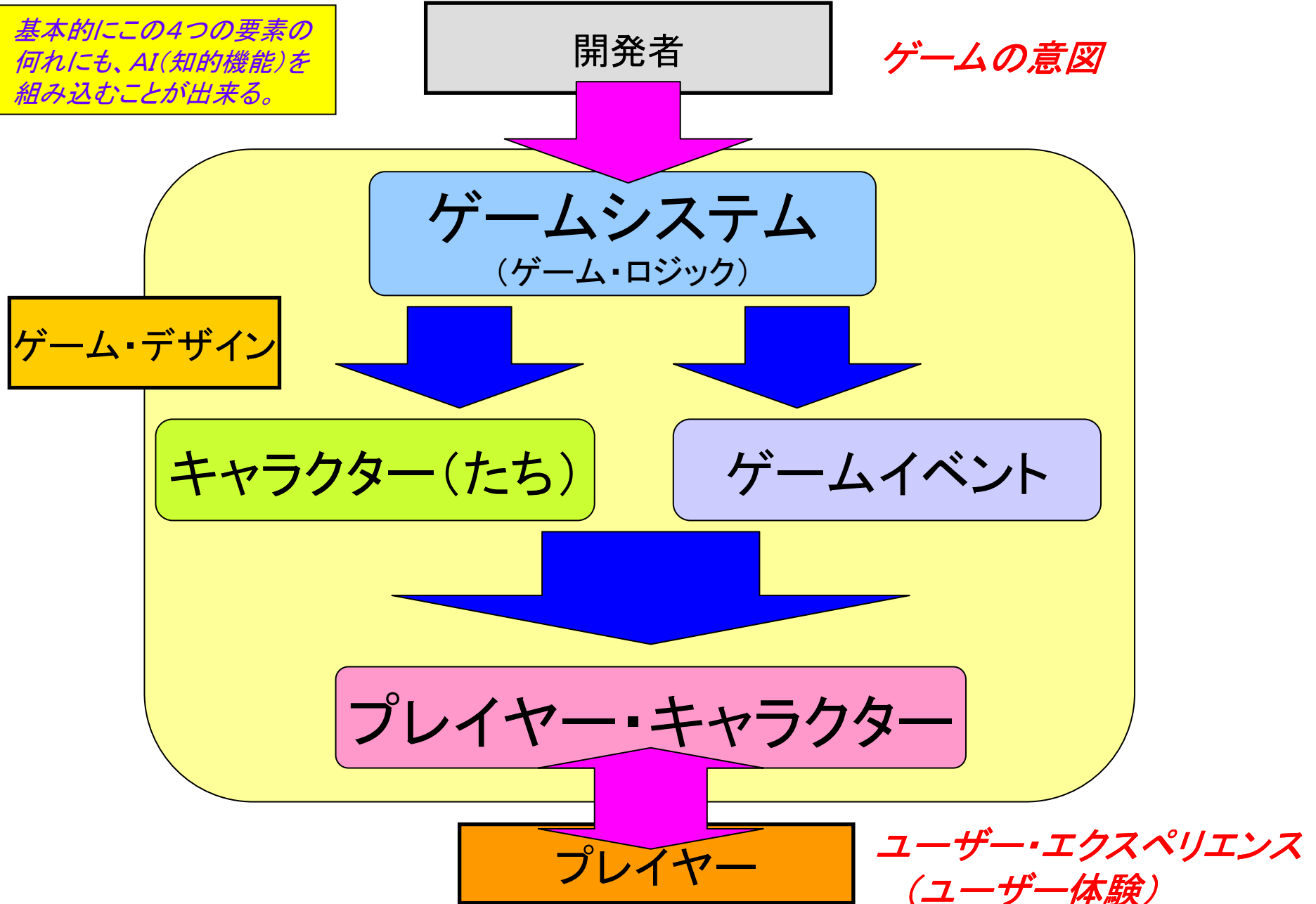
キャラクター(たち)

ゲームイベント

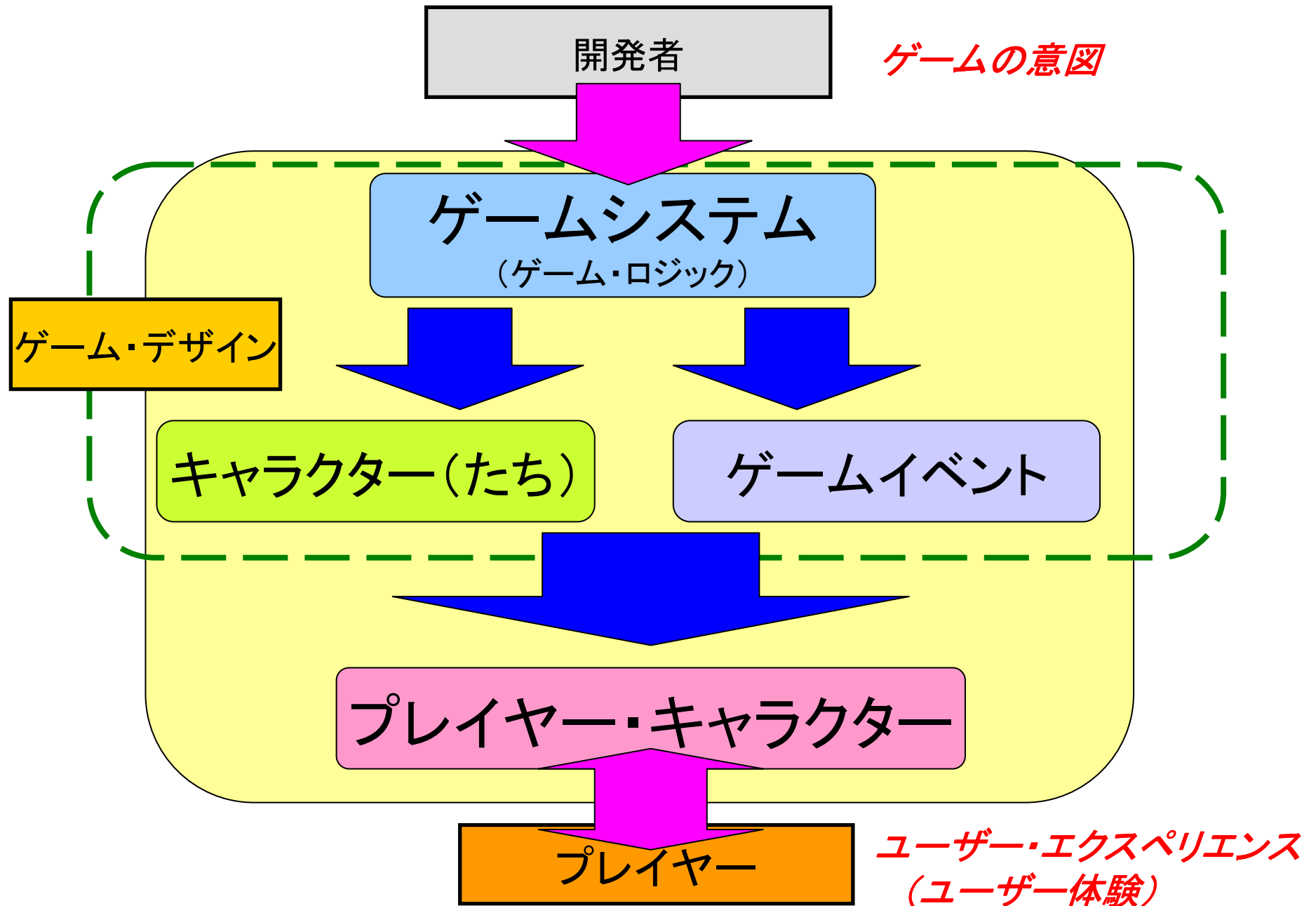
プレイヤー・キャラクター

プレイヤー

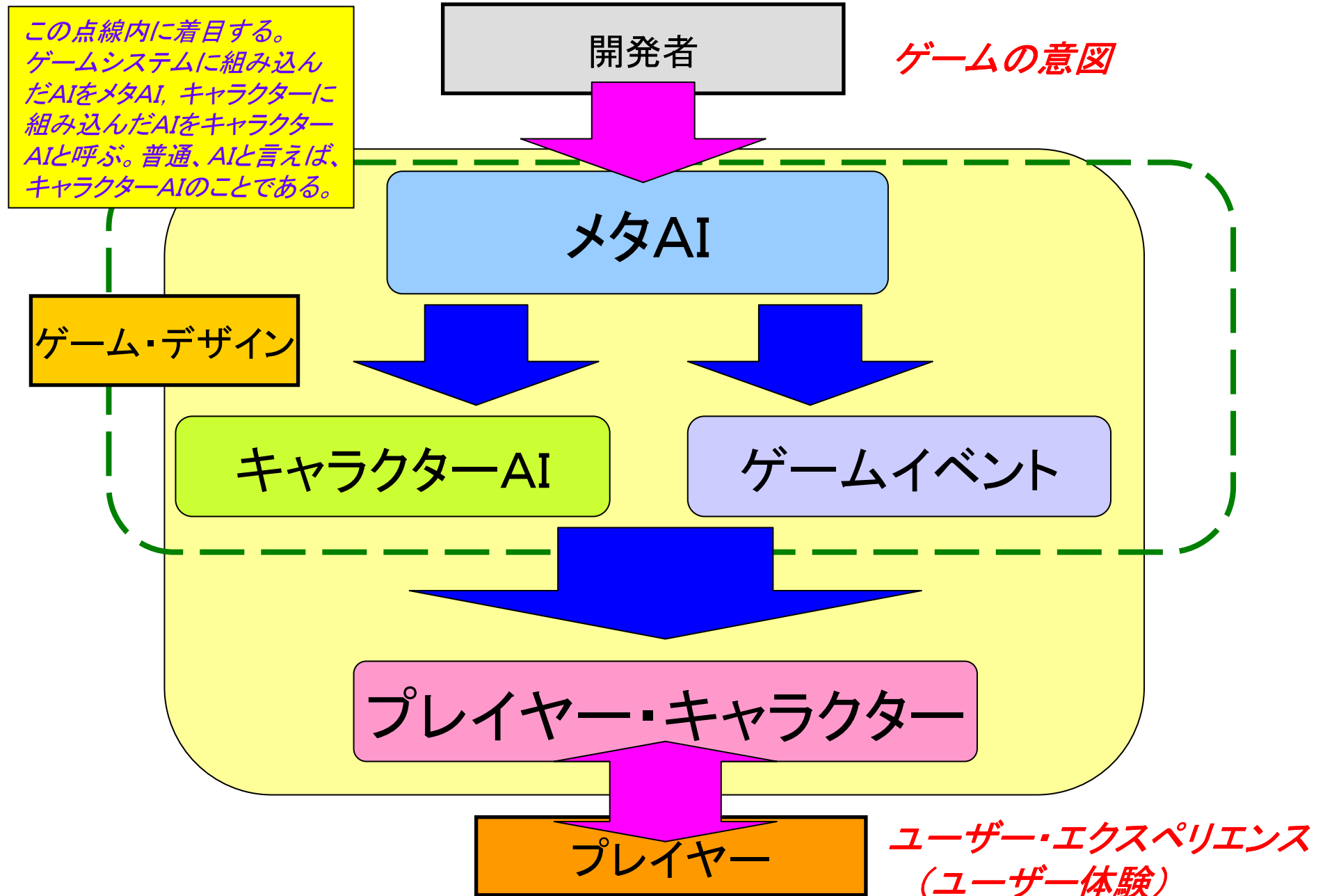
ユーザー・エクスペリエンス
(ユーザー体験)



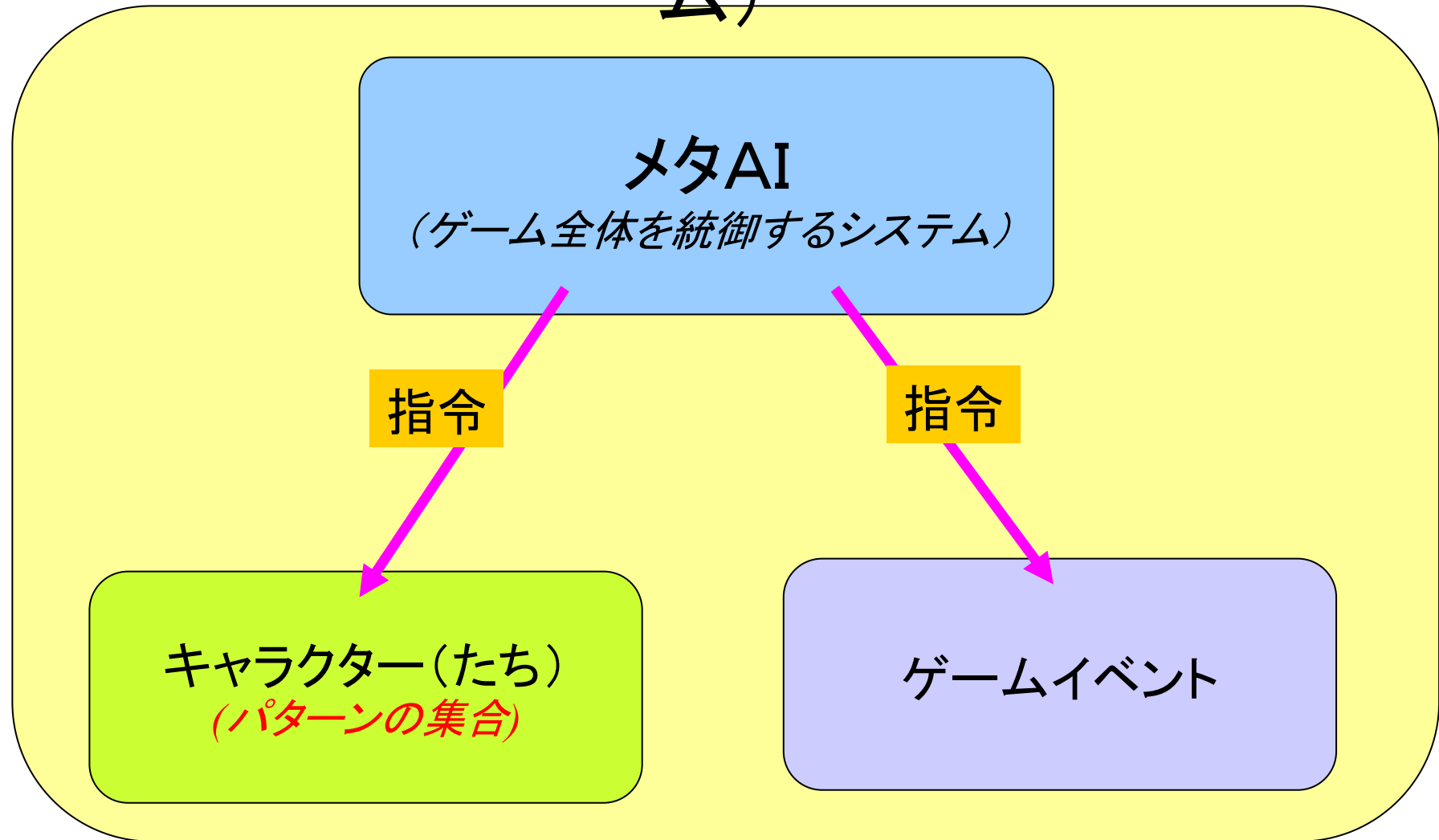
デジタルゲームの簡易モデル



デジタルゲームの簡易モデル

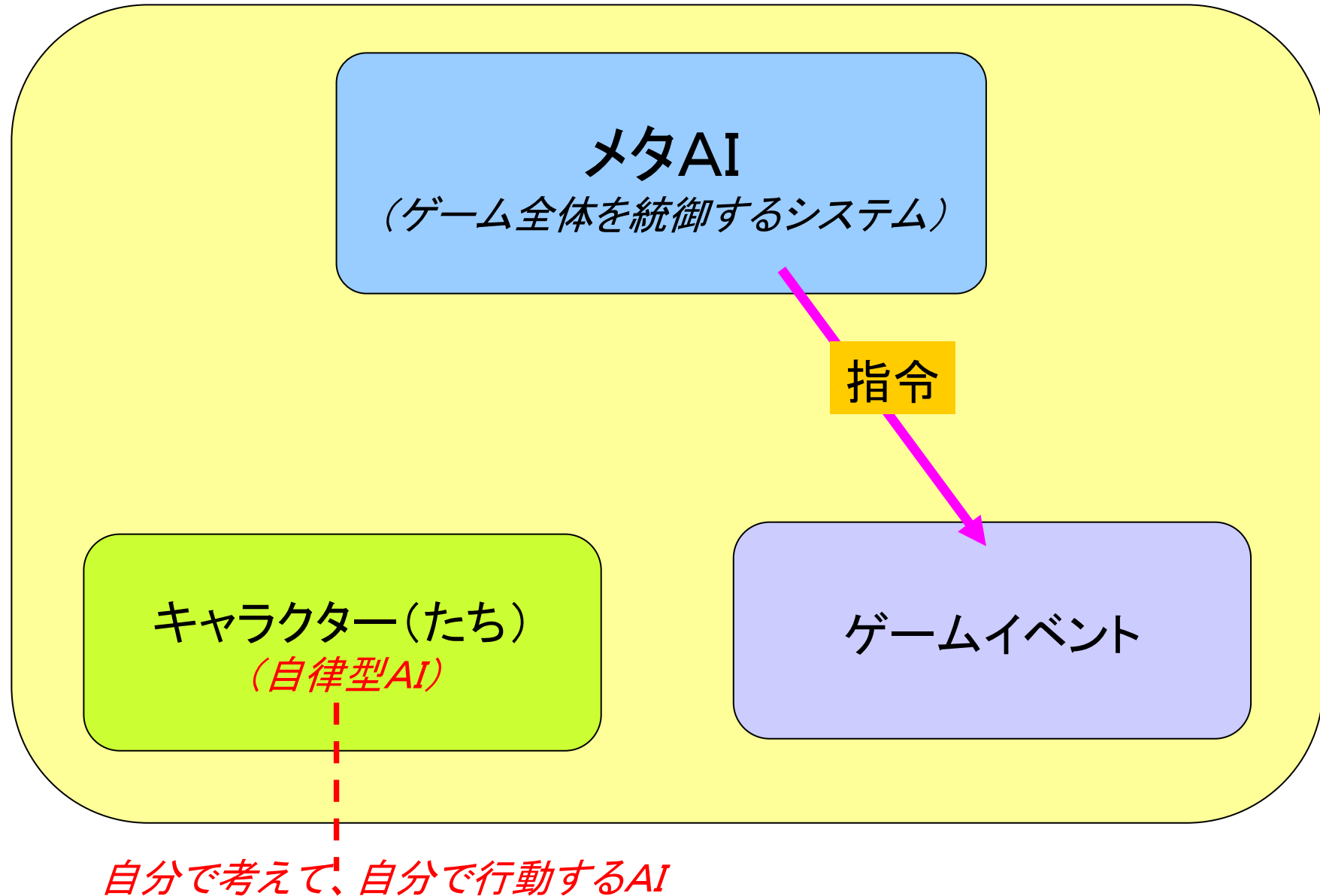


かつてのゲームAIの全体像(フルフレームム)

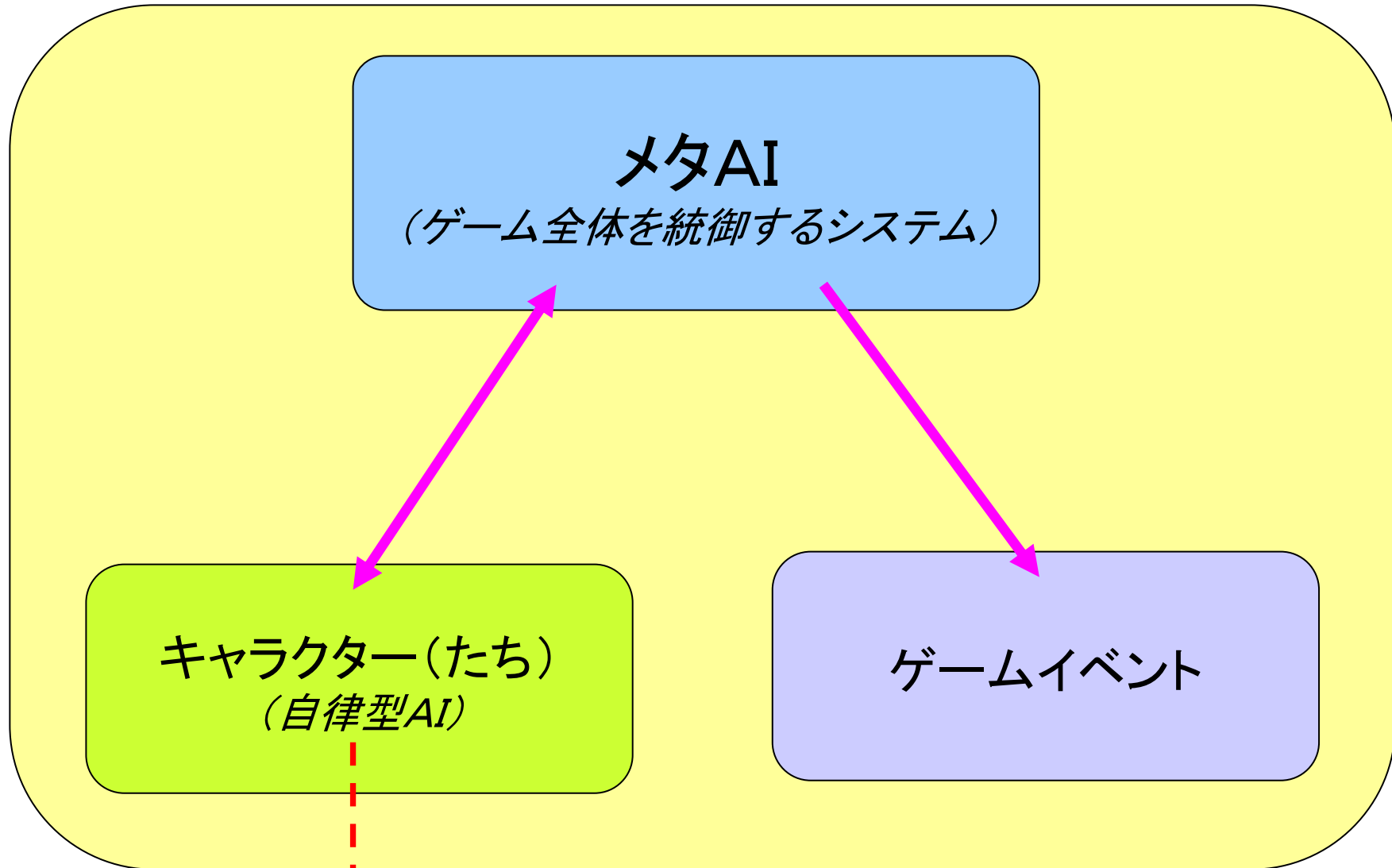


ゲームシステムはゲームステージの一部として、
キャラクターが制御される(お化け屋敷AI)

最近のゲームAIの全体像(フルフレーム)



最新のゲームAIの全体像(フルフレーム)



自分で考えて、自分で行動するAI

Left 4 Dead



内容: 4人チームでゾンビと戦うオンラインゲーム

開発元: Valve Software

出版: Valve Software

Hardware: PC, Xbox360

出版年: 2008年

オンラインで4人でチームを組んでゾンビと戦う。
AI Director によって、毎回、敵の出方が変化する。



Michael Booth, "The AI Systems of Left 4 Dead,"
Artificial Intelligence and Interactive Digital Entertainment Conference at Stanford.
<http://www.valvesoftware.com/publications.html>

From
COUNTER-STRIKE
to LEFT 4 DEAD:
Creating Replayable
Cooperative Experiences

Counter Strike から L4D へ: くり返し遊べる協力ゲーム体験

構築された非予測性

パーティの感情:
総合的な緊張感



モンスターの
配布

プレイヤーのインテンシティー(緊張感)を計算→ リラックスした後にモンスターを放つ！

Michael Booth, "The AI Systems of Left 4 Dead,"
Artificial Intelligence and Interactive Digital Entertainment Conference at Stanford.
<http://www.valvesoftware.com/publications.html>

ナビゲーションメッシュによる オフライン・リアルタイム空間情報解析

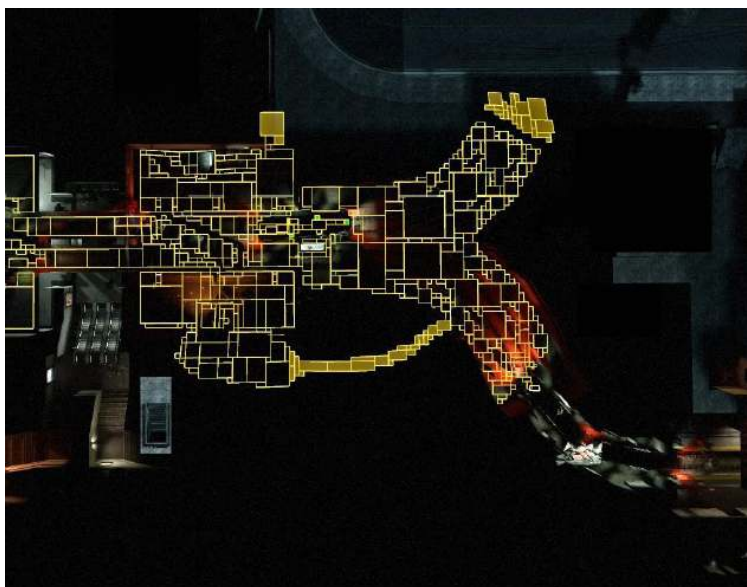
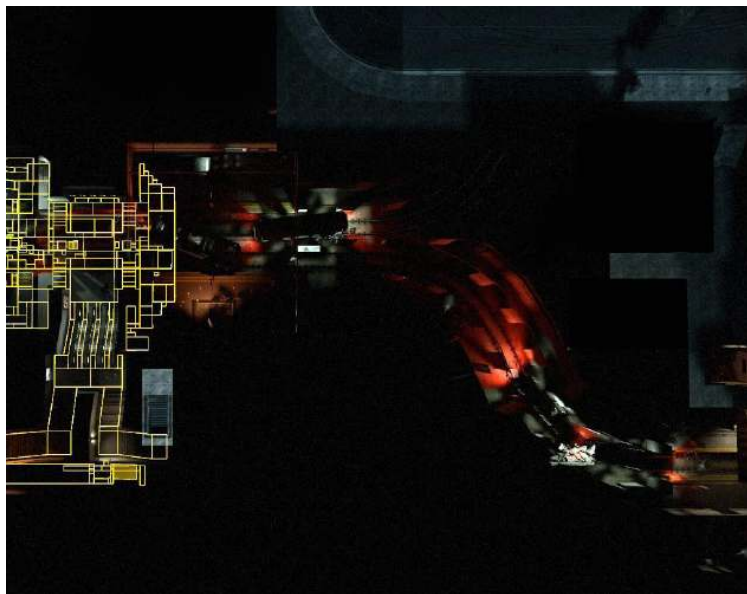
➤ The Navigation Mesh

- Represents “walkable space”
- Originally created for Counter-Strike Bot pathfinding
- Useful for general spatial reasoning and spatially localized information
 - Has area A ever been seen by actor B?
 - Is area X potentially visible by area Y?
 - Where is a spot near the Survivors, not visible to any of them?
 - How far have we traveled to reach this area?



その領域をプレイヤーは見たか？あるエリアは他のエリアから見えるか？ など。

ナビゲーションメッシュによる オフライン・リアルタイム空間情報解析



ナビゲーションメッシュによる
アクティブエリアセット(プレイヤーがいて
AIディレクターが制御対象とする領域)
の追跡

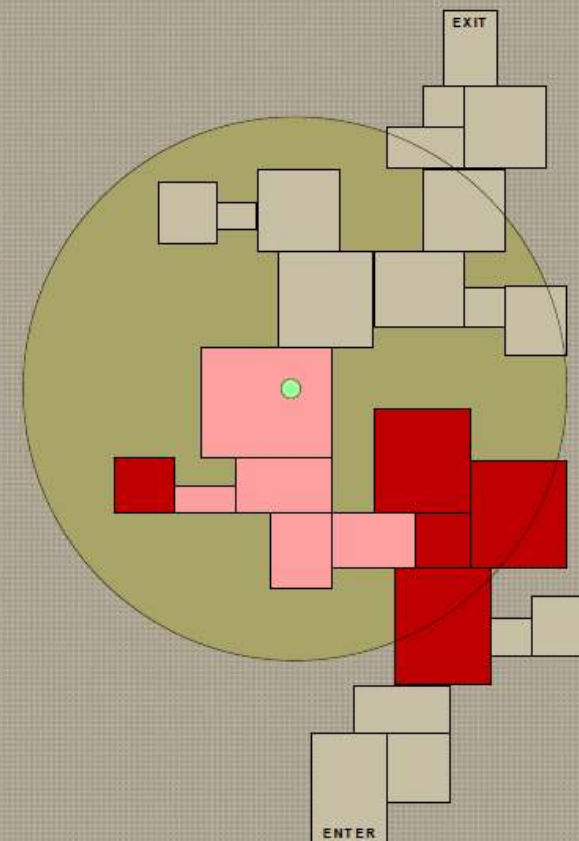
Michael Booth, "The AI Systems of Left 4 Dead," Artificial Intelligence and Interactive Digital Entertainment Conference at Stanford.
<http://www.valvesoftware.com/publications.html>



Goal: Promote Replayability Procedurally Populated Environments

➤ Where to spawn

- Behind Survivors
 - Only select valid areas in the AAS that are at or behind the Survivor team's "flow" distance
 - 75% of Mobs come from behind, since wanderers and Special/Boss Infected are usually engaged ahead of the team



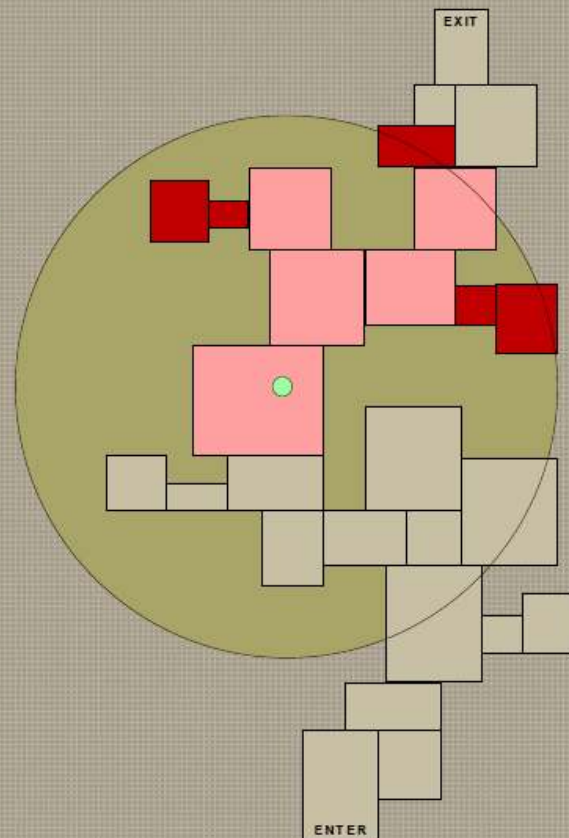
Michael Booth, "The AI Systems of Left 4 Dead,"
Artificial Intelligence and Interactive Digital Entertainment Conference at Stanford.
<http://www.valvesoftware.com/publications.html>



Goal: Promote Replayability Procedurally Populated Environments

➤ Where to spawn

- Ahead of Survivors
 - Only select valid areas in the AAS that are at or greater than the Survivor team's "flow" distance



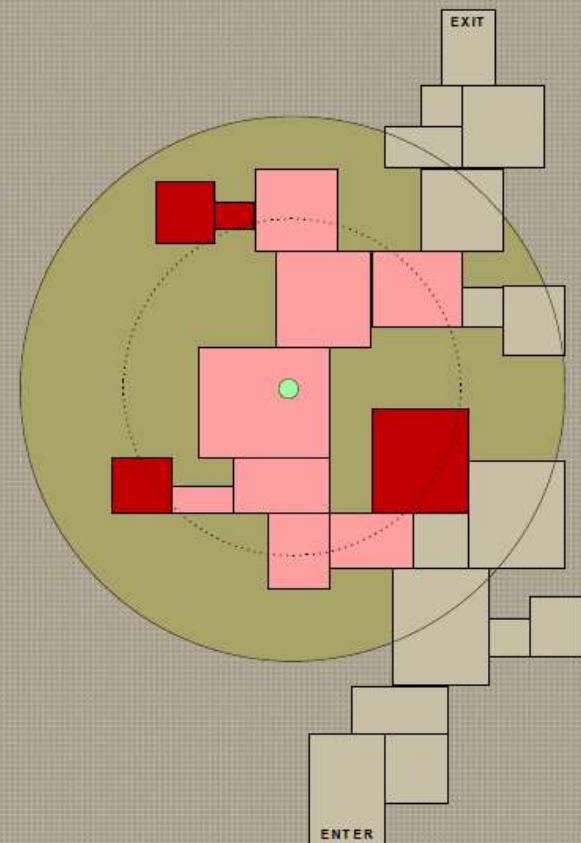
Michael Booth, "The AI Systems of Left 4 Dead,"
Artificial Intelligence and Interactive Digital Entertainment Conference at Stanford.
<http://www.valvesoftware.com/publications.html>



Goal: Promote Replayability Procedurally Populated Environments

➤ Where to spawn

- Near Boomer Vomit Victim
 - Only select valid areas in the AAS that are near the Boomer Vomit Victim's "flow" distance

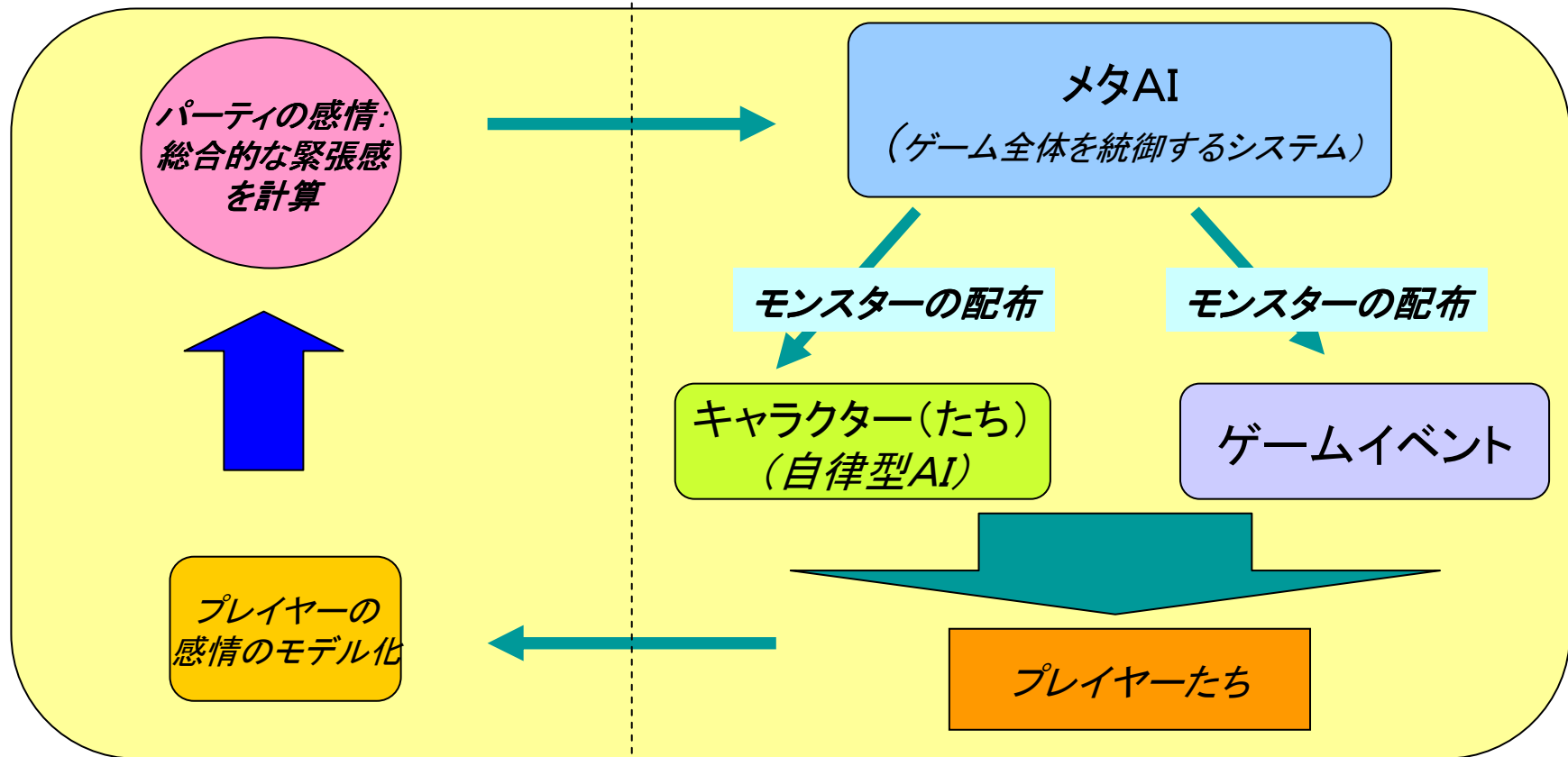


Michael Booth, "The AI Systems of Left 4 Dead,"
Artificial Intelligence and Interactive Digital Entertainment Conference at Stanford.
<http://www.valvesoftware.com/publications.html>

From
COUNTER-STRIKE
to LEFT 4 DEAD:
Creating Replayable
Cooperative Experiences

Counter Strike から L4D へ: くり返し遊べる協力ゲーム体験

構築された非予測性

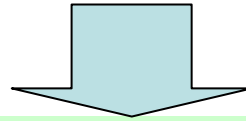


From
COUNTER-STRIKE
to LEFT 4 DEAD:
Creating Replayable
Cooperative Experiences

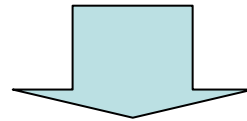
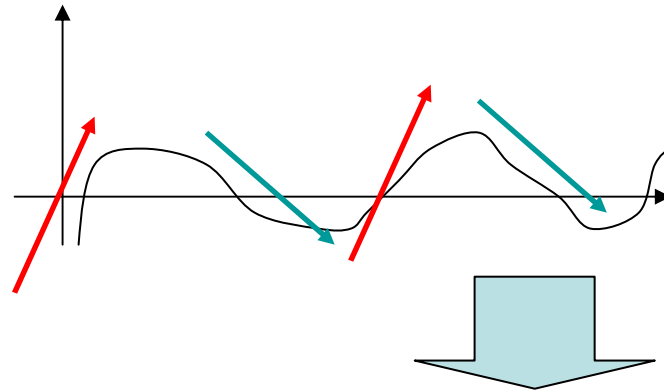
Counter Strike から L4D へ: くり返し遊べる協力ゲーム体験

構築された非予測性

- ① カウンターストライクは今なおロングセラー →何故？
→分析→オンラインゲームでの緊張感と緩和の緩急



- ②プレイヤーに緊張の波を作りたい→敵の出現でコントロール



- ③ゾンビ恐怖ゲームが最適！

参考文献

- (1) 良質なゲームを生み出すプレイテストのアプローチ Valveの実験心理学博士が明かす、プレイテスト手法のあれこれ(GameWatch)
http://game.watch.impress.co.jp/docs/news/20090329_80132.html
- (2) Valve語る、「Counter-Strike」から「Left 4 Dead」へ
協力プレイ、リプレイ性、AIディレクターの秘密(GameWatch)
http://game.watch.impress.co.jp/docs/news/20090327_80051.html
- (3) Valve Software、「Physical Gameplay in Half-Life2」 (GameWatch)
物理エンジンが生むリアリティーとゲームとしての面白さのバランス
<http://game.watch.impress.co.jp/docs/20060325/valve.htm>
(実験段階)
- (4) Left 4 Dead Interview (Eurogamer)
<http://www.eurogamer.net/articles/left-4-dead-interview>
- (5) なんのためにボスキャラがいるのか？ (4gamers)
「Left 4 Dead」に見る、プレイヤーをCo-opに誘う仕掛け
<http://www.4gamer.net/games/035/G003579/20090327043/>

参考文献

昨年までのValve の発表資料です。

- (1) How To Go From PC to Console Development (GDC 2008)
http://www.valvesoftware.com/publications/2008/GDC2008_CrossPlatformDevelopment.pdf
- (2) GRID Broadband Content Delivery Platform (GDC 2008)
http://www.valvesoftware.com/publications/2008/GDC2008_StylizationWithAPurpose_TF2.pdf
- (3) Source Engine 解説
<http://source.valvesoftware.com/SourceBrochure.pdf>

メタAIの歴史①

岩谷徹氏:

ファンと一般のユーザーを満足させる方法のひとつは
人工知能AIのような考え方です。

プレーヤースキルをプログラム側から判断して、
難易度を調整していくというものです。
これを私はセルフゲームコントロールシステムと呼んで
10年以上前から開発に使っています。

— International Game Designers Panel —

http://game.watch.impress.co.jp/docs/20050312/gdc_int.htm

<http://spitfire.client.jp/shooting/xvious2.html>

メタAIの歴史①

遠藤雅伸氏 あと面白い機能なんですけれど、ゼビウスには非常に簡単なAIが組み込まれています。

「プレイヤーがどれくらいの腕か」というのを判断して、出てくる敵が強くなるんです。

強いと思った相手には強い敵が出てきて、弱いと思った相手には弱い敵が出てきます。そういったプログラムが組み込まれています。ゲームの難易度というのは「初心者には難しくて、上級者には簡単だ」ということが、ひとつの難易度で(調整を)やっていくと起きてしまうので、その辺を何とか改善したいな、ということでそういったことを始めてみたのですけれど、お陰で割合にあまり上手くない人でも比較的長くプレイできる、うまい人でも最後のほうに行くまで結構ドラマチックに楽しめる、そういった感じになっています。

— ゼビウスセミナー

メタAIの歴史②

「エージェント・アーキテクチャーから作るキャラクターAI」
(CEDEC2007、三宅)

<http://www.igda.jp/modules/mydownloads/visit.php?cid=2&lid=78>

...その他、多数の使い方がある。

Left 4 Dead のメタAIの新しい 点

Left 4 Dead は、メタAIを、調整と言う意味ではなく、

- ① ゲームの面白さを作るために、積極的に用いた
- ② ユーザーをモデル化してプロシージャルにユーザーの状態を解析して対応した。

という2点で、これまでのメタAIの使い方とは違う。

メタAIを使う場合の注意

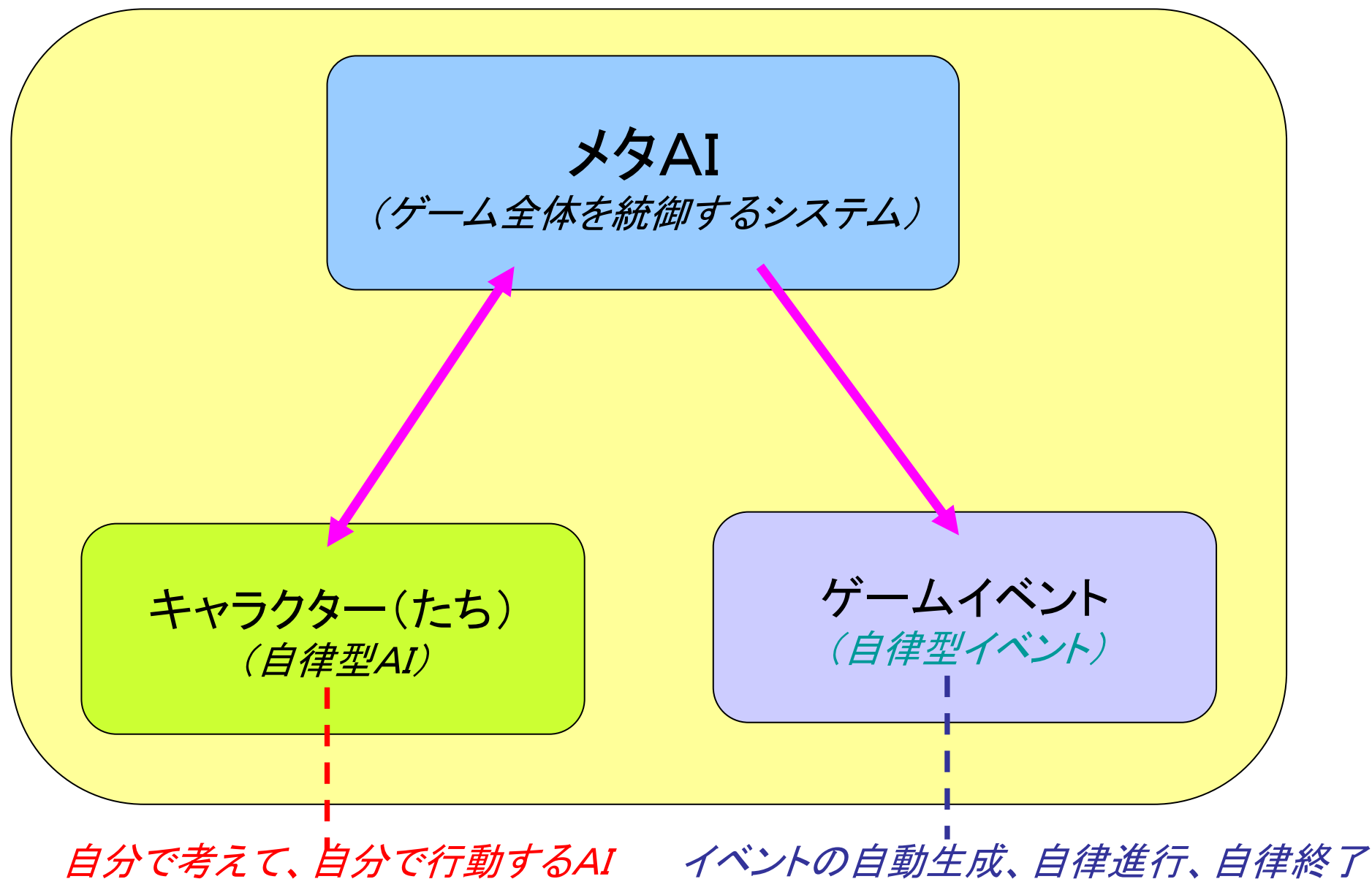
個々のAIはローカルな変化に対応し、
メタAIは、グローバルな状況と時間的变化を解析する。

メタAIは、個体のAIがある程度できた後でこそ、
最大の威力を発揮する。

個体のAIの不備は、グローバルなAIで埋めようとしても
埋まらない。そして、ユーザーのファースト・インプレッションは、
個々のAIの振る舞い、ゲーム全体としては、メタAIで決まる。

メタAIを使う場合は、
まず、個々のAIの知性をある程度完成させることが必要。

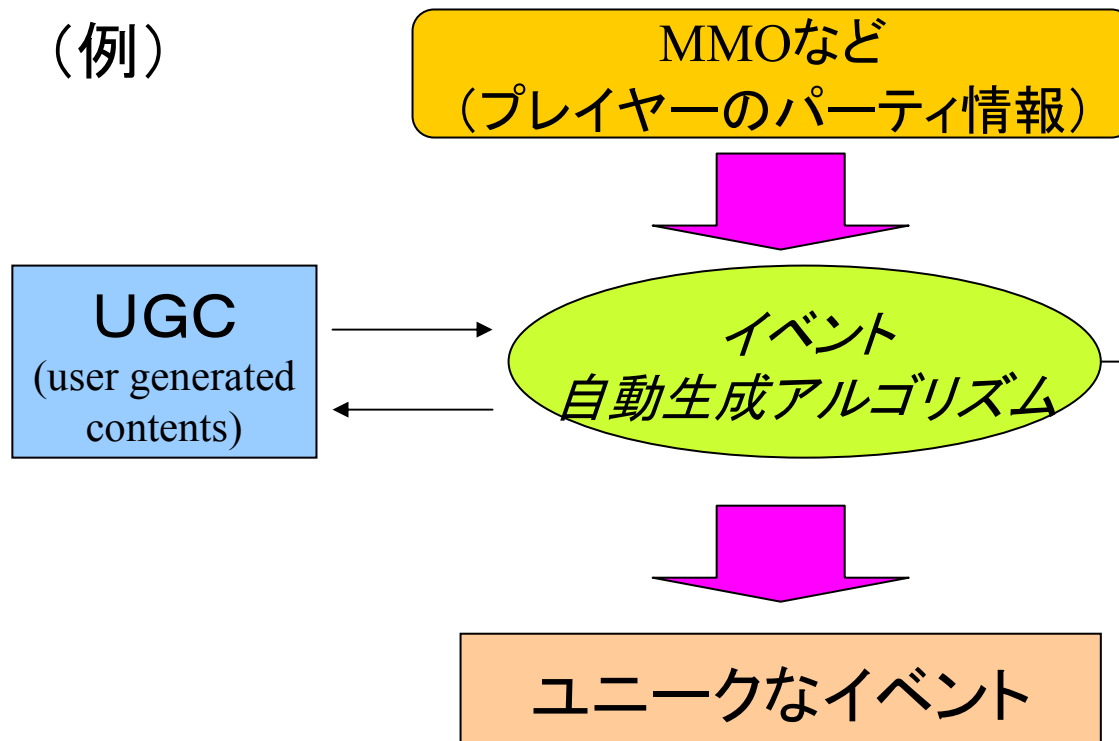
これからのゲームAIの全体像(フルフレーム)



イベントの自動生成はあり得るか？

- アルゴリズム(プランニング、イベントグラフなど)
- パス検索技術によるNPCの非局在化
(これまでのイベントのように場所を限定する必要がない)

(例)



テクニック

①ある程度、内容が決まったイベントで、トリガー条件が満たされれば、発生する。
(例)ユーザーが作る家の密度が増すと、ドラゴンが出て来て焼き払おうとする。

②プレイヤーに善悪度や評判度というパラメーターを振っておき、その数値によって街の人々の対応や、イベント発生率、商店での価格が変化する。
(例) Fable (Lionhead Studio)

③シナリオ生成: あらかじめ決められたシナリオの役割に、プレイヤーやAIを当てはめる。
(例)パーティで一番強いプレイヤーの一人が他のプレイヤーから魔物に見えてしまうイベントなど。MMOの中で、「宿敵」を自動選別して、マッチングさせる。

④プランニング: イベントの単位を、(前提条件、イベント、結果)という形で複数用意し、それを連鎖アルゴリズムで、条件に応じてつないでいく。

(例) Façade (Michal Metas, CMU, Oz Project)

参考文献

- (1) FABLE (XBOX, Lionhead Studio, published by Microsoft)
<http://www.xbox.com/ja-JP/games/f/fable/>
- (2) Façade <http://www.interactivestory.net/>
- (3) Oz Project
<http://www.cs.cmu.edu/afs/cs.cmu.edu/project/oz/web/papers.html>

プログラマーとゲーム

ハードコーディング型実装

ゲームを作ってリリース。ゲーム性は固定。



プログラミング
実装

GAME



プログラマーとゲーム

動的に変化するゲーム

リアルタイムにユーザーの特性やゲーム状態を解析して
自分自身(ゲーム性)を変化させるゲーム



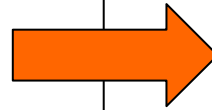
プログラミング
実装

GAME



メタAIの出現

万人向けに
完成したバランスのゲーム



ユーザーに応じた
ゲームがその場で生成

デジタルゲームの世界は、
「誰にでも楽しめるバランスを取れるゲーム」に対して、
「ユーザー特性やスキルに応じたゲームをリアルタイム制御で生成」
するメタAI技術が台頭しつつある。

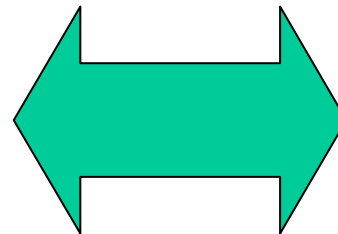
ゲーム性をリアルタイムにコントロールするメタAI (Meta AI)

本授業の主旨

4つの言葉を覚えて帰りましょう

- (1) エージェント (Agent)
- (2) プロシージャル (Procedural)
- (3) メタAI (Meta-AI)
- (4) マルチエージェント (Multi-agent)

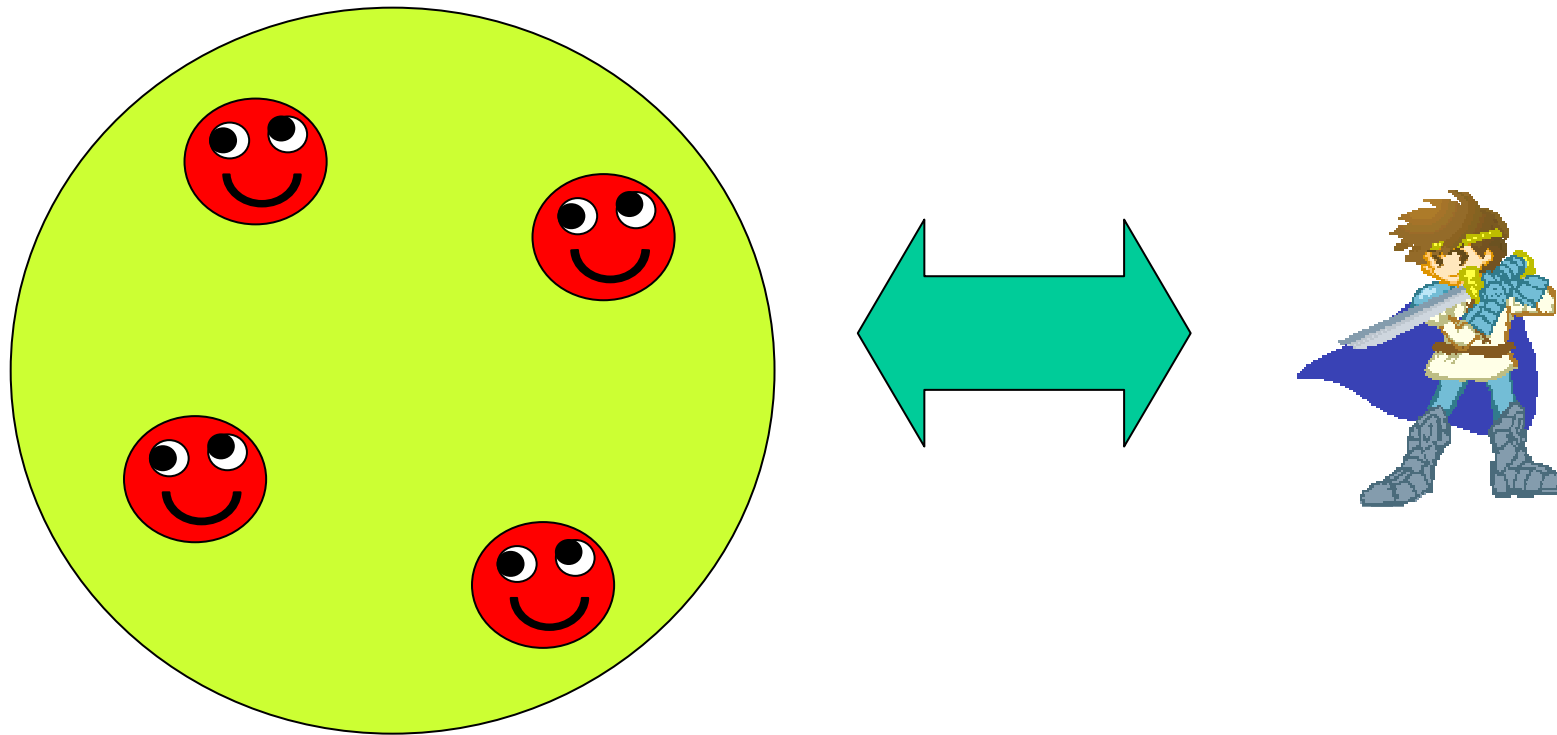
ゲーム体験 (AI vs プレイヤー)



それなりに賢くする(通常のゲーム)
一体(巨大)に特化して賢くする
ボス戦などで特別なアルゴリズム

例) ワンダと巨像

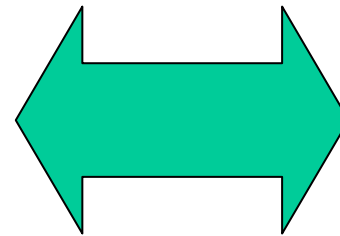
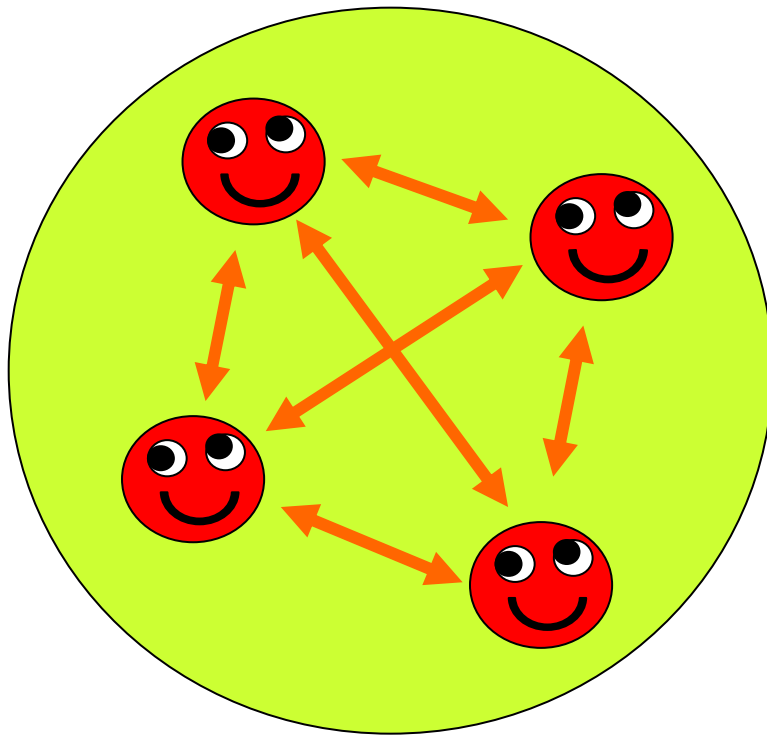
ゲーム体験 (AIたち vs プレイヤー)



複数の敵が出るが連携しない

例) JRPGの敵たち

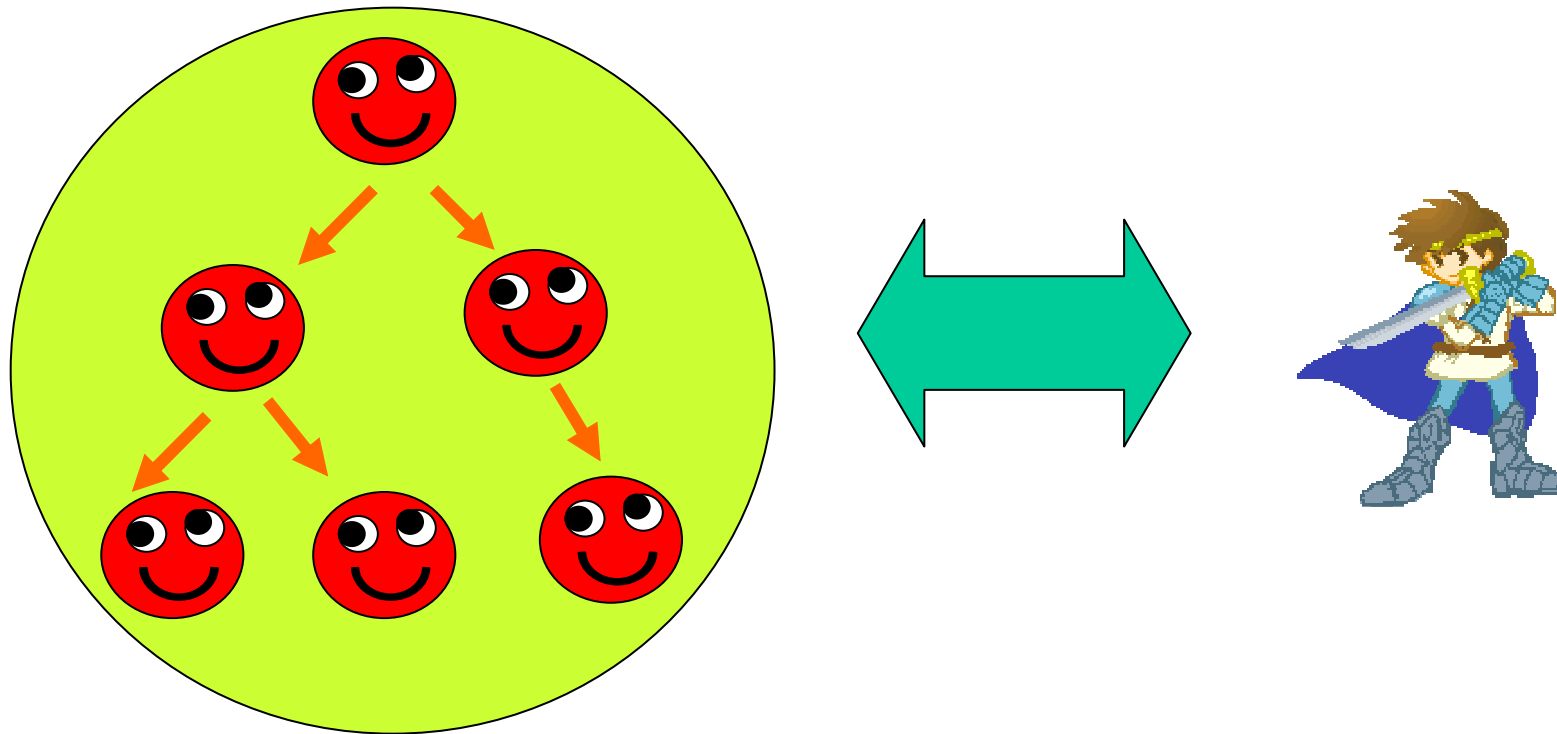
ゲーム体験 (マルチエージェント vs プレイヤー)



相互に連携する(マルチエージェント)

例) クロムハウズ(第3回資料、CEDEC2007資料参照)

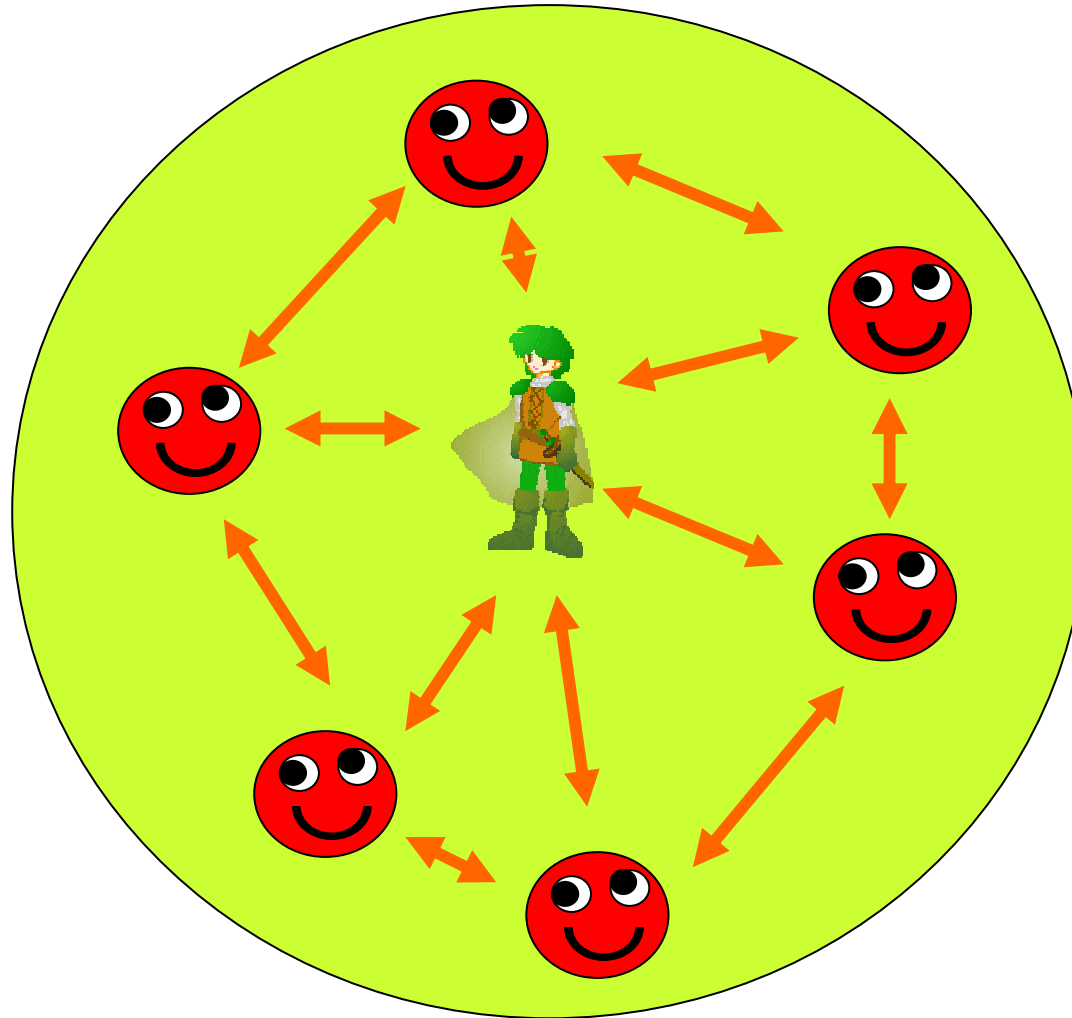
ゲーム体験 (AIチーム vs プレイヤー)



組織的構造を入れてチームとして行動する

例) Killzone2 (GDC10 報告会資料参考) など FPS

ゲーム体験(プレイヤー in AI社会)



AIたちの作る社会の中で社会的な活動する

例) The Sims (GDC10 報告会資料参考) など

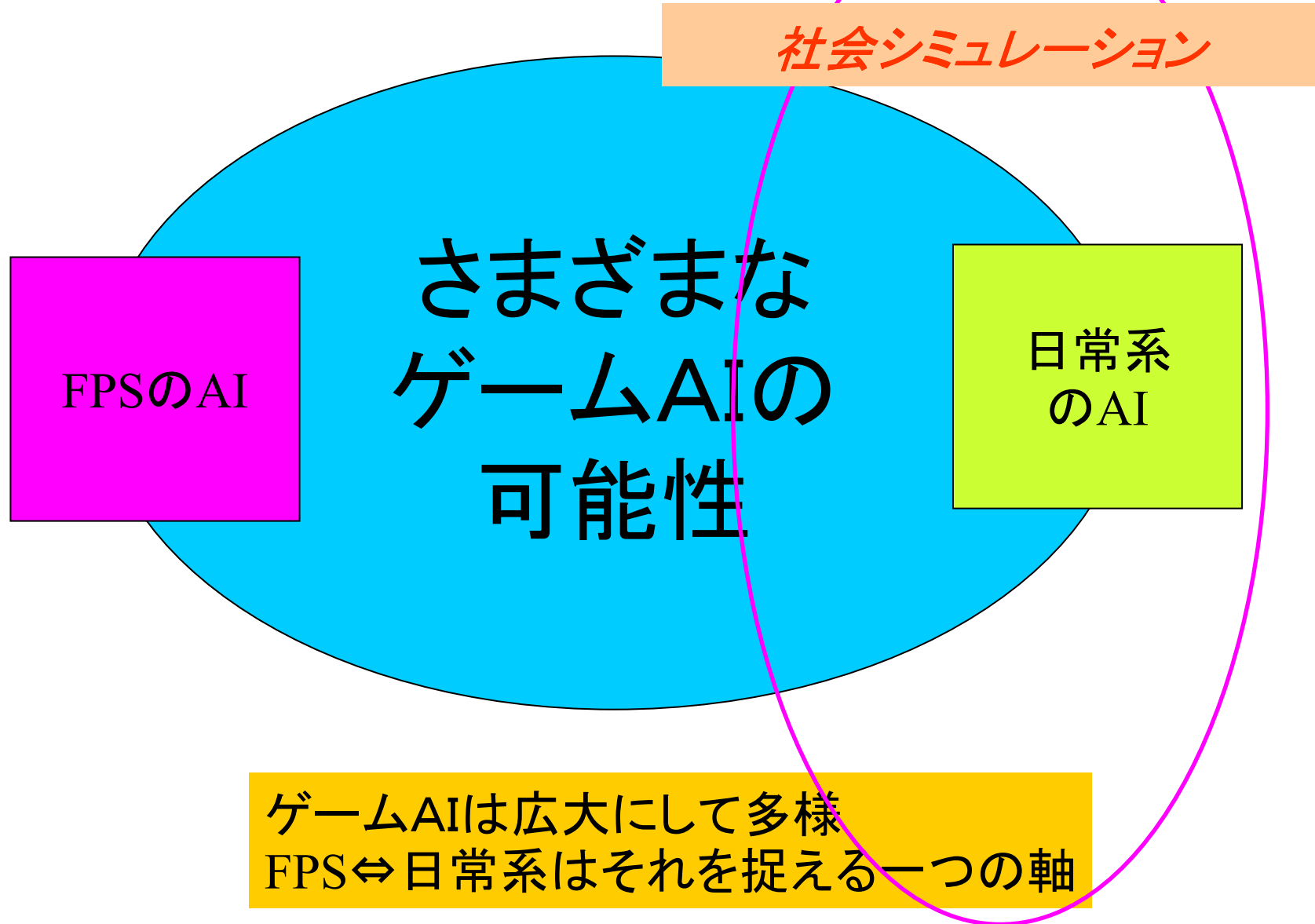
イラスト: アンの小箱 <http://www.anne-box.com/>

FPS(戦闘) ⇔ 社会系のAIの比較

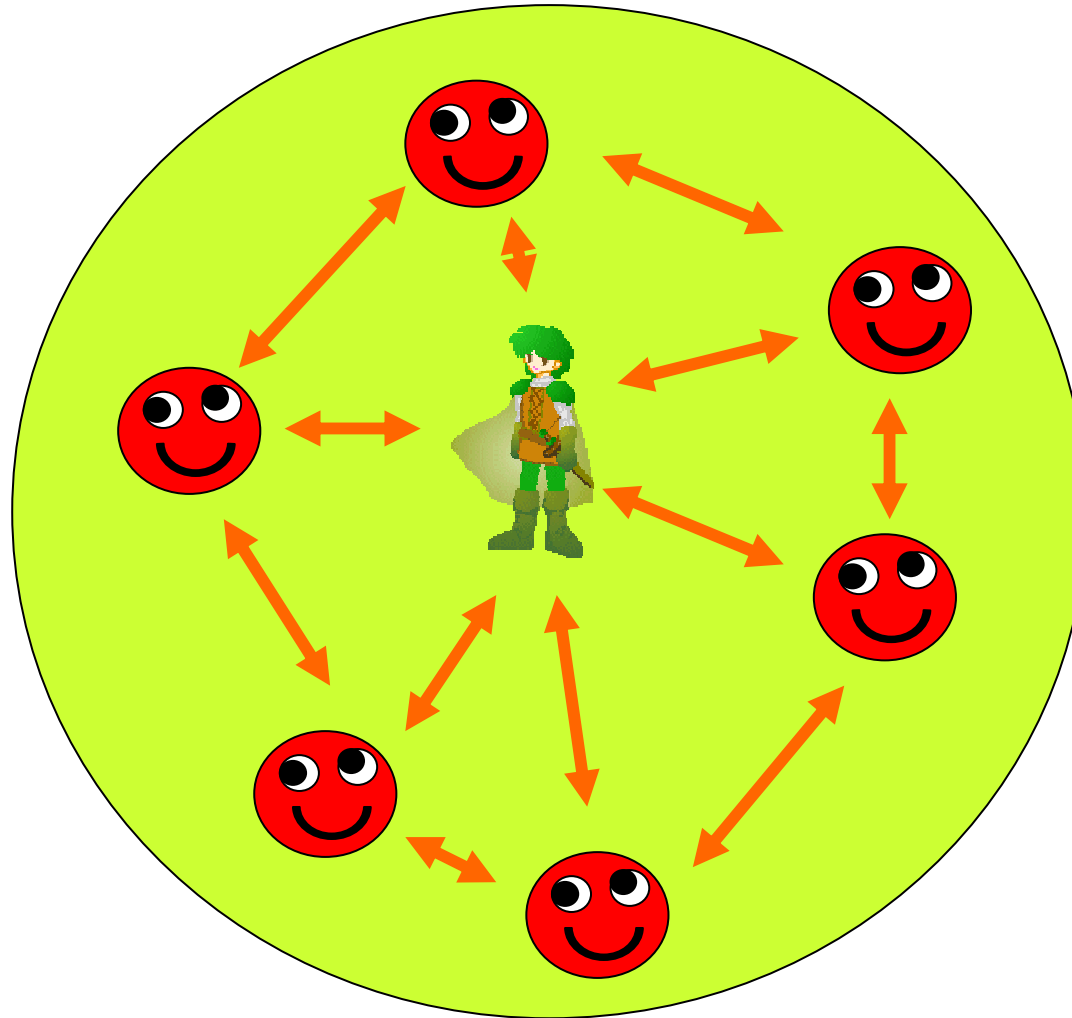
	FPS	社会系
内面	シンプル	複雑
思考	複雑	シンプル
周囲の環境の見える方	視線・射線に対する障壁	多様な行為のアフォーダンス
産み出したいもの	戦場	社会シミュレーション
組織・人間関係	階層的・厳密	平坦的・緩い繋がり
スコープ時間	短期～長期	短期～長期
重要なもの	敵味方の生死	自分の幸福度
インタラクション	暴力・治癒	多様
学術との関係	アルゴリズム	哲学、認知科学、心理学など

キャラクターAIの技術はこの両極の間にある

FPS(戦闘) ⇔ 日常系のAIの比較



ゲーム体験(プレイヤー in AI社会)



AIたちの作る社会の中で社会的な活動する

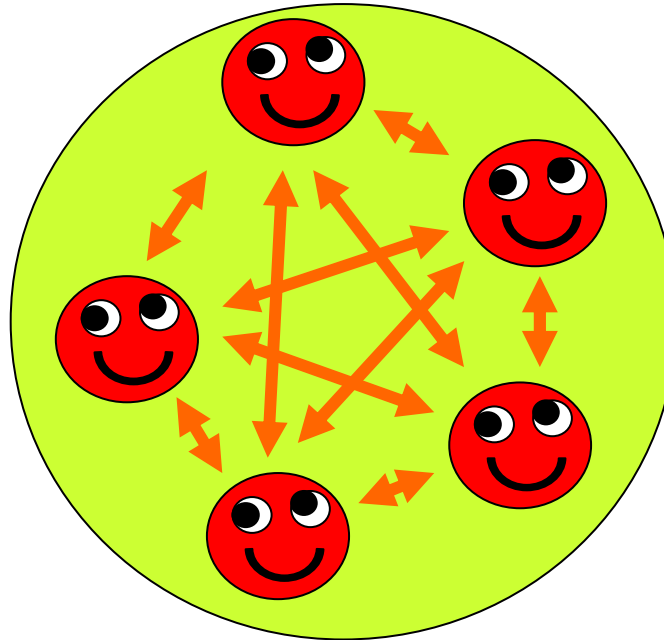
例) The Sims (GDC10 報告会資料参考) など

イラスト: アンの小箱 <http://www.anne-box.com/>

ゲーム体験(プレイヤー to AI社会)



干渉



AIたちの作る社会に対して干渉する

例) くりきん、牧場系ゲーム、The Sims (所謂GODゲーム)

イラスト: アンの小箱 <http://www.anne-box.com/>

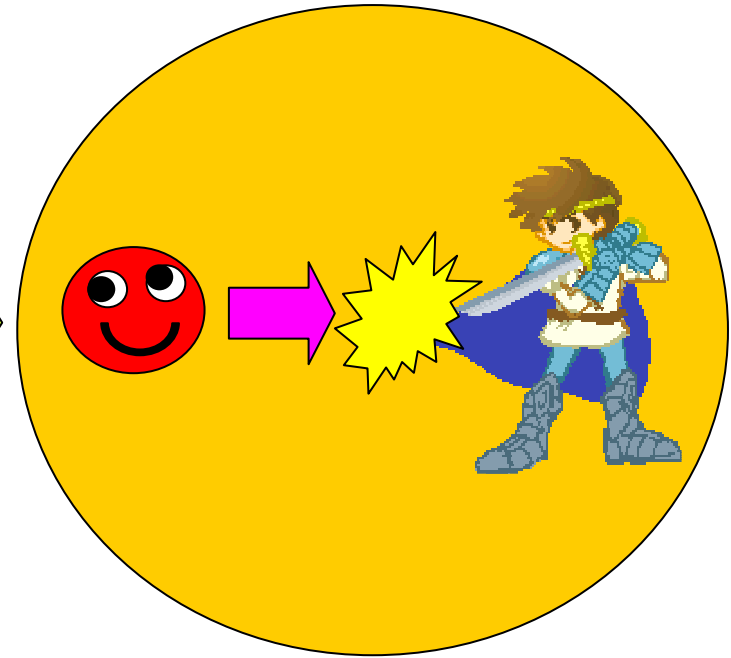
プログラマーとユーザー

ハードコーディング型実装

エージェントを作ってユーザーにぶつける。



プログラミング
実装



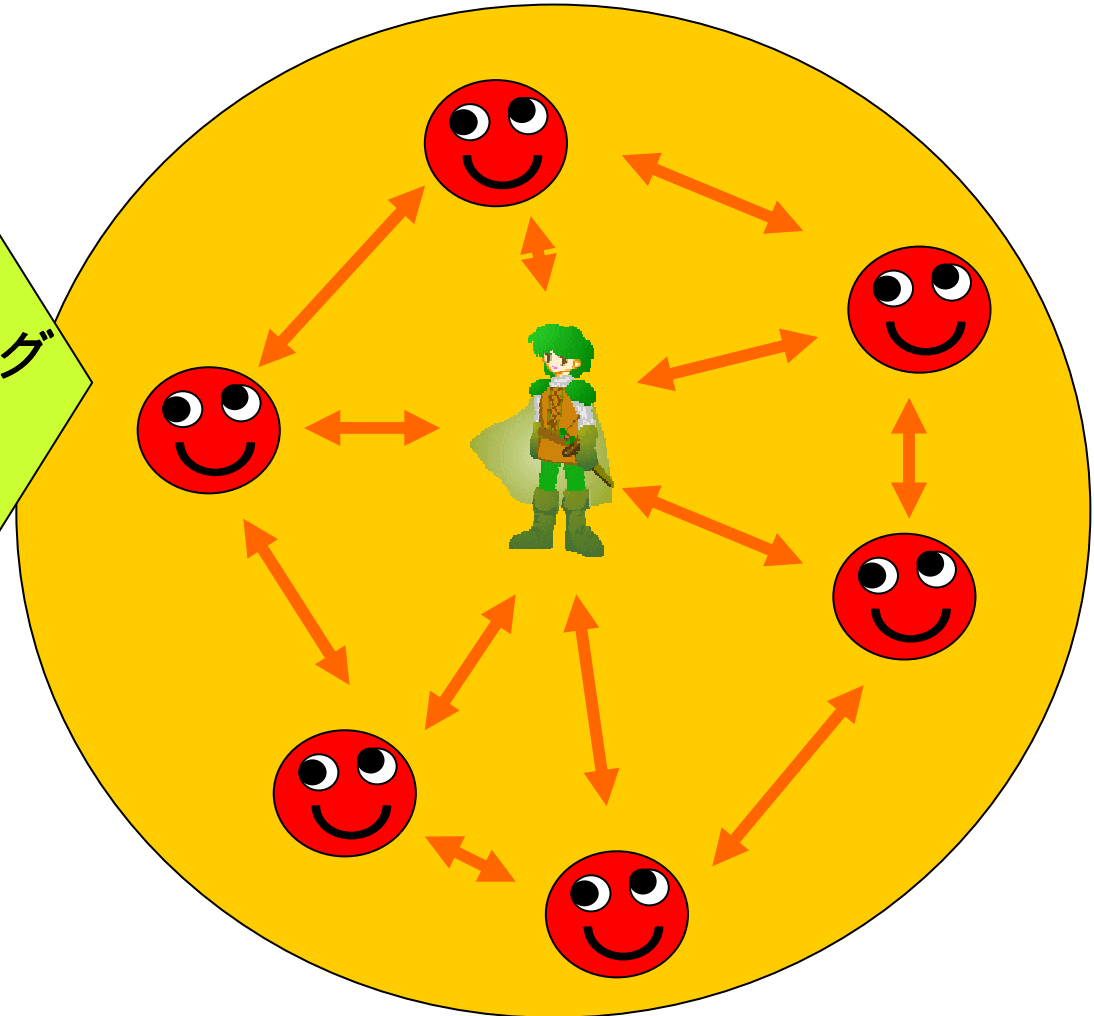
プログラマーとユーザー

エージェントたちがおもてなしするゲーム

マルチエージェントシステムを作ってユーザーに参加して貰う

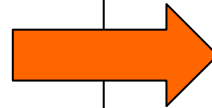


プログラミング
実装



マルチエージェント・システムを利用しよう (未開拓)

エージェントをユーザーに
ぶつける



マルチエージェント・システム
にユーザーが参加する

デジタルゲームの世界は、
「ユーザーとエージェントの対立構造」から、
「エージェントの協調によるマルチエージェント・システムを
どうゲームの中に利用して行くか」
が将来的な課題となるだろう。

エージェントを協調させてゲームの中に社会を作る
マルチエージェント・システム(Multi-agent system)

社会的デジタル空間

世界を俯瞰的に観察する
(シミュレーション)

社会
シミュレーション
(プレイヤーが
干渉しない。
プレイヤーという
概念はない)

世界を内側から体験する
(ゲーム)

俯瞰型
シミュレーション
ゲーム
(プレイヤーが
神的な立場から干渉
する)

体験型
ゲーム
(プレイヤーが
エージェントの
一人として活動する)

社会的なゲームに必要なこと

技術

システム全体を設計してシミュレーションできる能力

攻撃型ゲーム⇔社会系ゲームの面白さ

攻撃型ゲーム

- 敵を攻撃して殺す(FPSなど)
- 敵の弱点を探して急所を打つ
- 力で押す
- 相手の見えないところから攻撃
- 装備を充実させる
- 煙幕などで敵を混乱させる
- 修行して力を付ける(レベル上げ)
- チームを組んで圧倒する
- 作戦を立てる
- 戦力を集中する
- 敵の背後を取る
- 銃を撃ちまくる
- 敵の行動を読む
- トラップを仕掛ける
- 味方を犠牲にして自分が勝つ
- 相手を偽情報でだます
- 崖から突き落とす
- 遠くから狙撃する
- ...

社会系ゲーム



課題: 社会系ゲームの面白さを
考えて列挙してみよう!

攻撃型ゲーム⇔社会系ゲームの面白さ

攻撃型ゲーム

- 敵を攻撃して殺す(FPSなど)
- 敵の弱点を探して急所を打つ
- 力で押す
- 相手の見えないところから攻撃
- 装備を充実させる
- 煙幕などで敵を混乱させる
- 修行して力を付ける(レベル上げ)
- チームを組んで圧倒する
- 作戦を立てる
- 戦力を集中する
- 敵の背後を取る
- 銃を撃ちまくる
- 敵の行動を読む
- トラップを仕掛ける
- 味方を犠牲にして自分が勝つ
- 相手を偽情報でだます
- 崖から突き落とす
- 遠くから狙撃する
- ...

社会系ゲーム

- 喋る
- 経済活動をする
- 社会的活動をする
- 日常生活を行う
- 衣食住を充実させる
- 物を交換する
- 知識や技術を見に付ける
- コミュニティーを作る
- 計画を立てる
- 一つのことに取り組む
- こっそり人の役に経つ
- 演説する
- 社会の情勢、経済の動向を読む
- 街を整備する
- 人のために何かする
- 正確な情報を伝える／偽情報を流す
- 崖から落ちそうな人を助ける
- 近くでお年寄りをサポートする
- ...

攻撃型ゲーム⇔社会系ゲームの面白さ

攻撃型ゲーム

- 敵を攻撃して殺す(FPSなど)
- 敵の弱点を探して急所を打つ
- 力で押す
- 相手の見えないところから攻撃
- 装備を充実させる
- 煙幕などで敵を混乱させる
- 修行して力を付ける(レベル上げ)
- チームを組んで圧倒する
- 作戦を立てる
- 戦力を集中する
- 敵の背後を取る
- 銃を撃ちまくる
- 敵の行動を読む
- トラップを仕掛ける
- 味方を犠牲にして自分が勝つ
- 相手を偽情報でだます
- 崖から突き落とす
- 遠くから狙撃する
- ...

社会系ゲーム

- 喋る
- 経済活動をする
- 社会的活動をする
- 日常生活を行う
- 衣食住を充実させる
- 人の役に立つ
- 知識や技術を見に付ける
- コミュニティを築く
- 計画を立てる
- 一つのことに取り組む
- こつこつと努力を続ける
- 演説する
- 社会の情勢、経済の動向を読む
- 街を整備する
- 人のために何かする
- 正確な情報を伝える／偽情報を流す
- 崖から落ちそうな人を助ける
- 近くでお年寄りをサポートする
- ...

しかし...
こういった楽しみが
デジタルゲームで
可能なのだろうか？

社会的なゲームに必要なこと

技術

システム全体を設計してシミュレーションできる能力

企画

社会的な喜びをゲーム内で再現する能力

社会的デジタル空間

世界を俯瞰的に観察する
(シミュレーション)

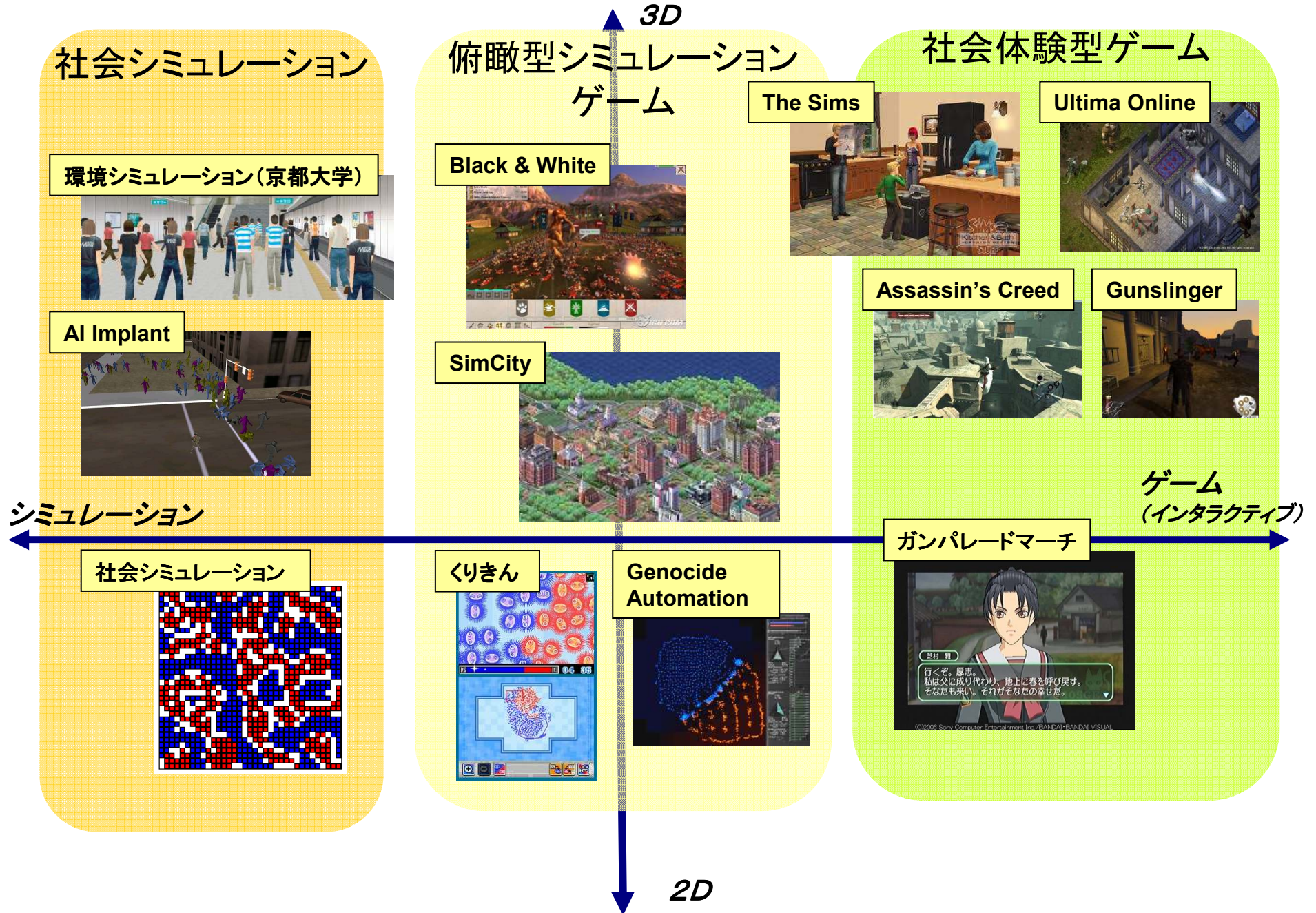
社会
シミュレーション
(プレイヤーが
干渉しない。
プレイヤーという
概念はない)

世界を内側から体験する
(ゲーム)

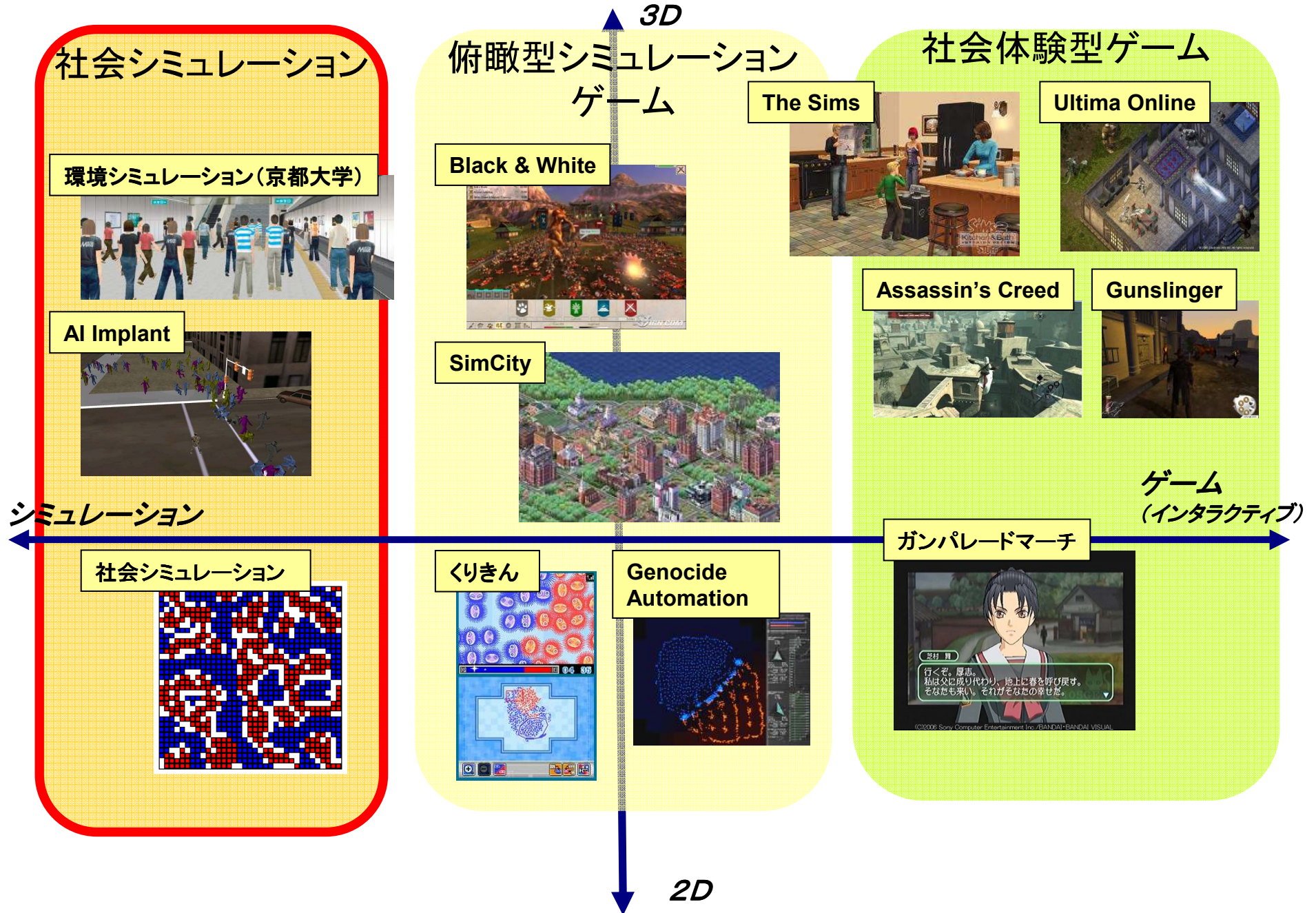
俯瞰型
シミュレーション
ゲーム
(プレイヤーが
神的な立場から干渉
する)

体験型
ゲーム
(プレイヤーが
エージェントの
一人として活動する)

社会的デジタル空間



社会的デジタル空間



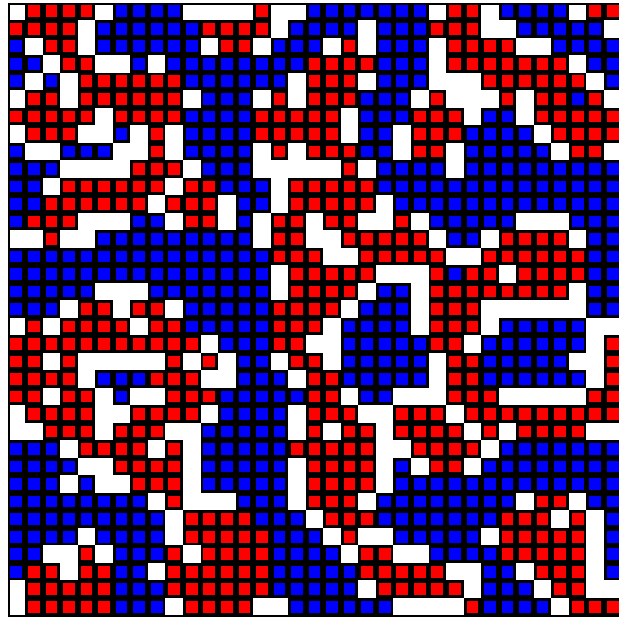
社会シミュレーション

完全なシミュレーション

地域社会の住み分け

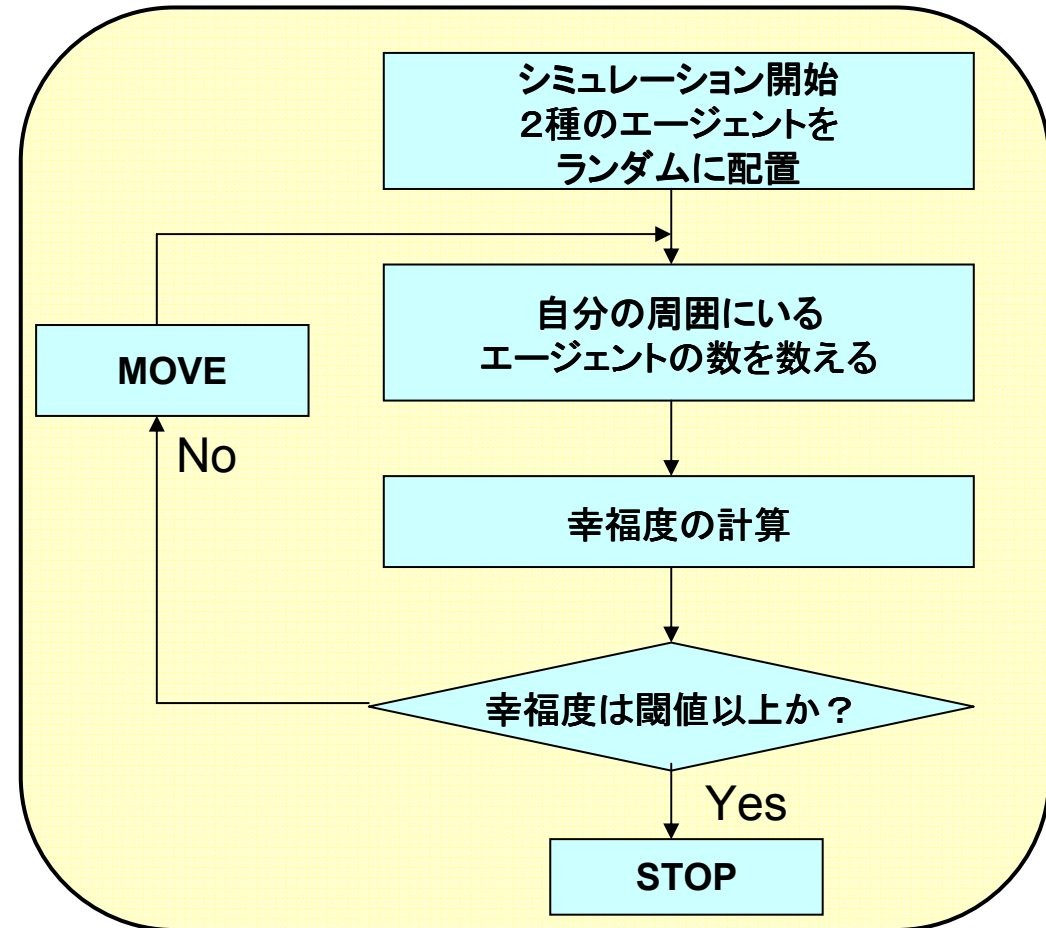
シェリング「分居」モデル

2種類のエージェントが周囲に自分と同じ種類のエージェントが住まう環境を求めて行動する。



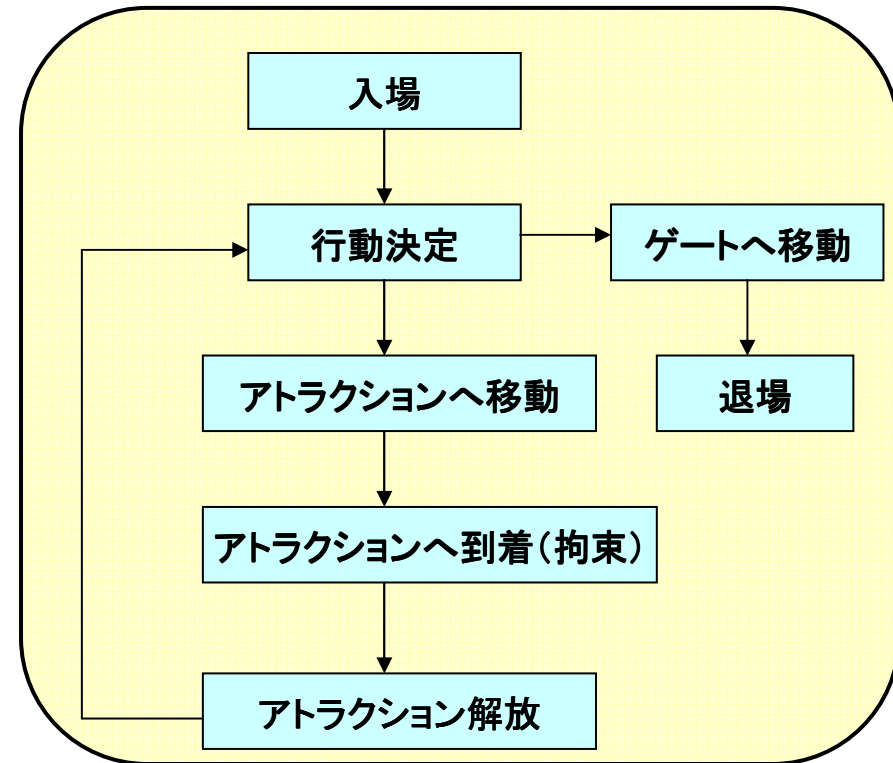
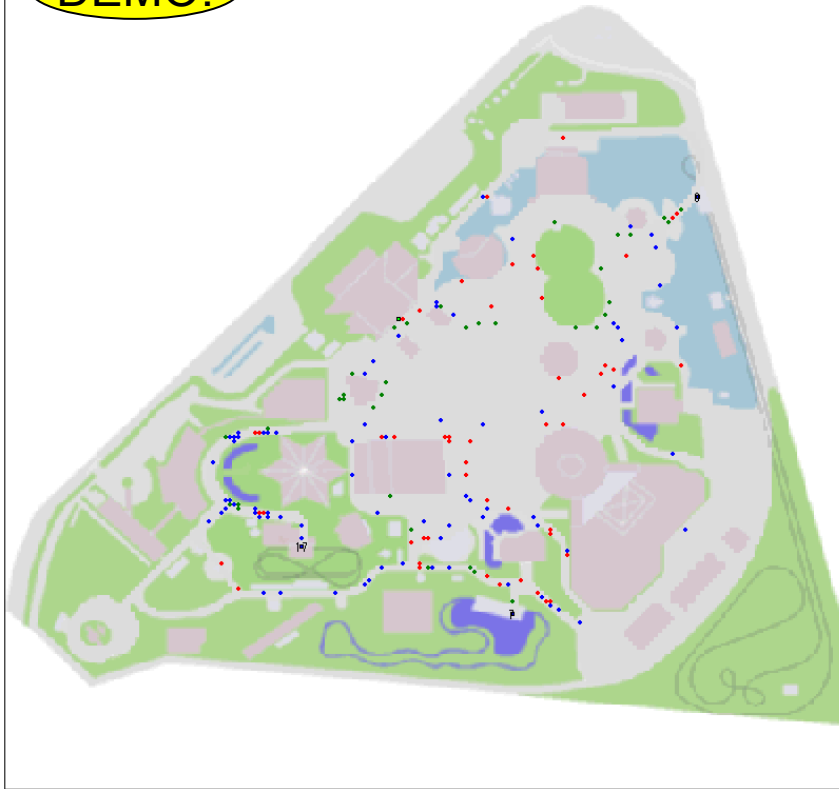
DEMO!

幸福度 = $\frac{\text{近傍にいる同類エージェント数}}{\text{近傍にいる総エージェント数}}$



テーマパーク・シミュレーション

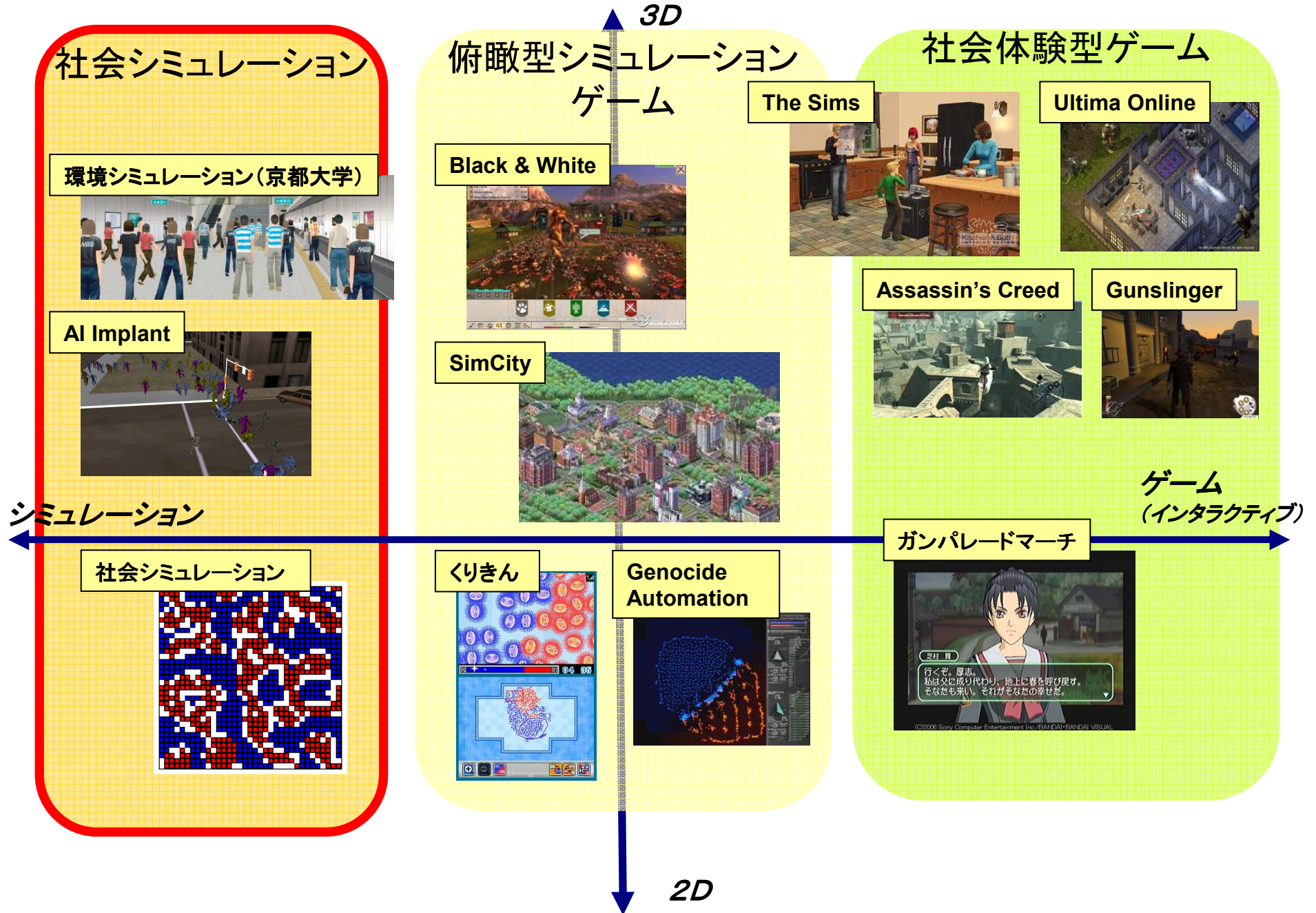
DEMO!



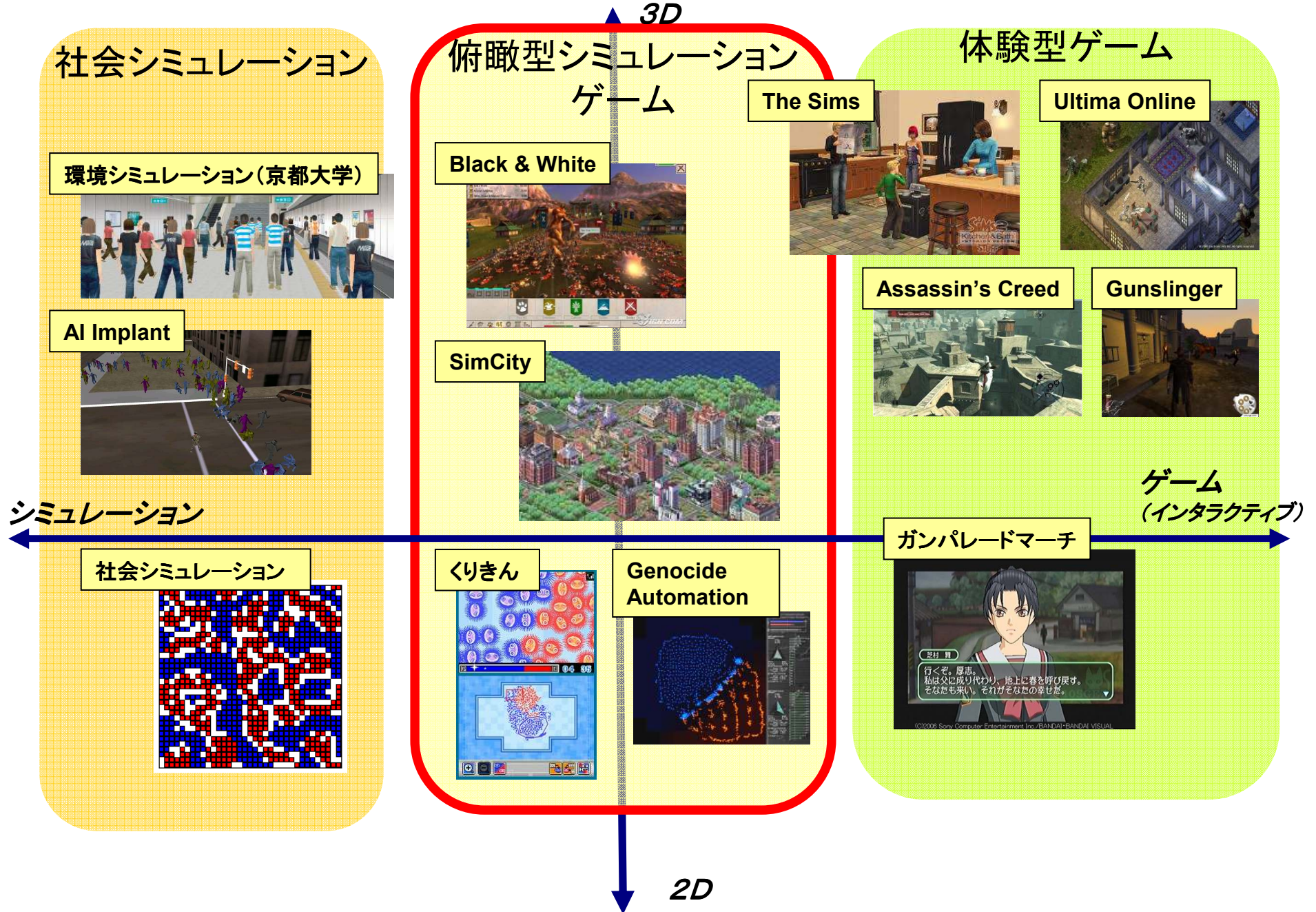
アトラクション効用(アトラクションを選ぶ評価式)
=アトラクションの選好度
- $\alpha \times$ アトラクションまでの直線距離
- $\beta \times$ アトラクションの混雑状況
(α 、 β はシミュレーション定数)

エージェントが持つパラメーター
(1) 各アトラクションへの選好度
(2) すでに訪問したアトラクション
(3) 満足度((1)(2)より計算)
(4) 帰宅満足度(満足度がこの値より大きければ帰宅)

社会的デジタル空間



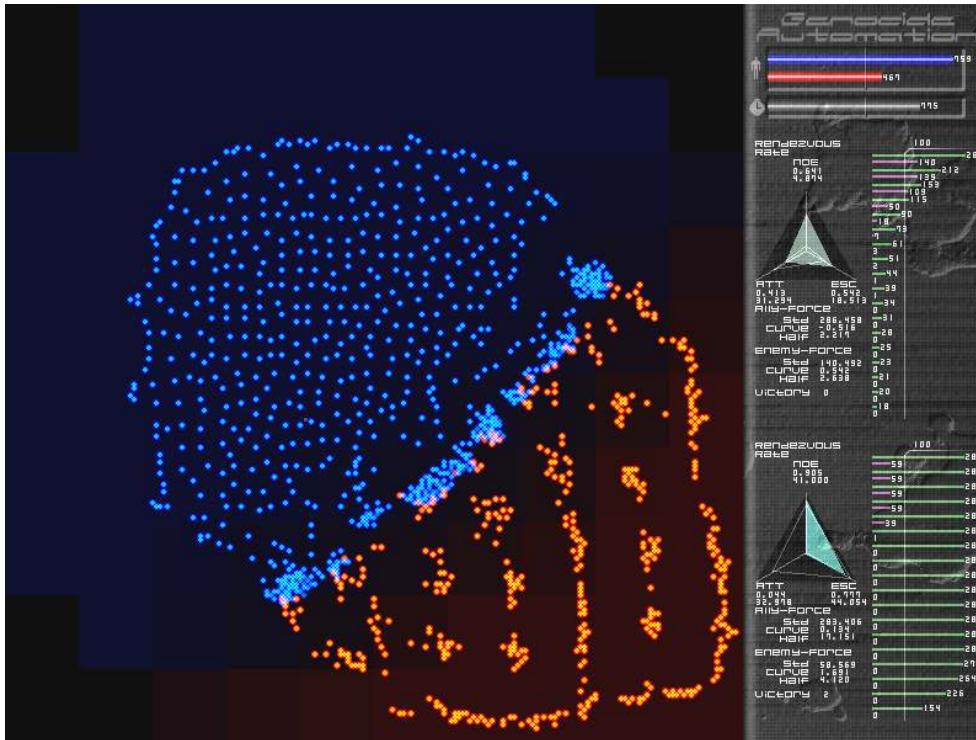
社会的デジタル空間



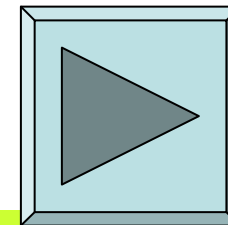
群知能乱戦シミュレータ「Genocide Automation」

Sense of Wonder Night 入賞作品、佐々木直哉、<http://www11.plala.or.jp/normal/>

DEMO!



- (1) ユニットに分かれている
- (2) ユニットは「資源」を消費。
資源が不足するとダメージ
- (3) エリアは敵、味方の数から占有率を計算。
- (4) ユニット毎に味方、敵戦力を計算。
優勢と判断すれば戦う、でなければ逃げる。
- (5) ユニットは合流閾値によって合流する。



[左クリック]:

- ・視界内に敵がいなければカーソル位置へ向かう。
- ・表示される矩形内にいて、敵が視界内にいる場合には、味方の戦力を5倍に見積もる(好戦的になる)

[右クリック]:

- ・視界内に敵がいなければカーソル位置から遠ざかる。
- ・表示される矩形内にいて、敵が視界内にいる場合には、敵の戦力を5倍に見積もる(消極的になる)

社会的デジタル空間

3D

社会シミュレーション

環境シミュレーション(京都大学)

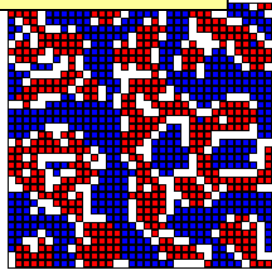


AI Implant



シミュレーション

社会シミュレーション

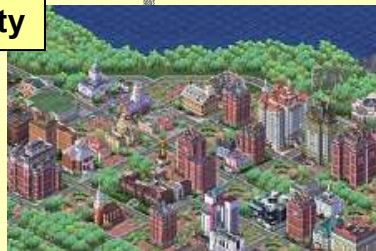


俯瞰型シミュレーション ゲーム

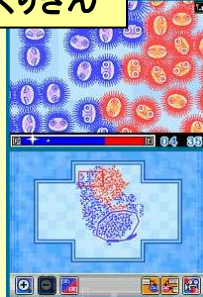
Black & White



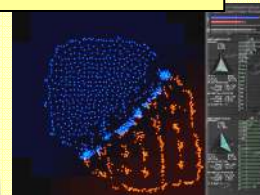
SimCity



くりきん



Genocide Automation



2D

社会体験型ゲーム

The Sims



Ultima Online



Assassin's Creed



Gunslinger

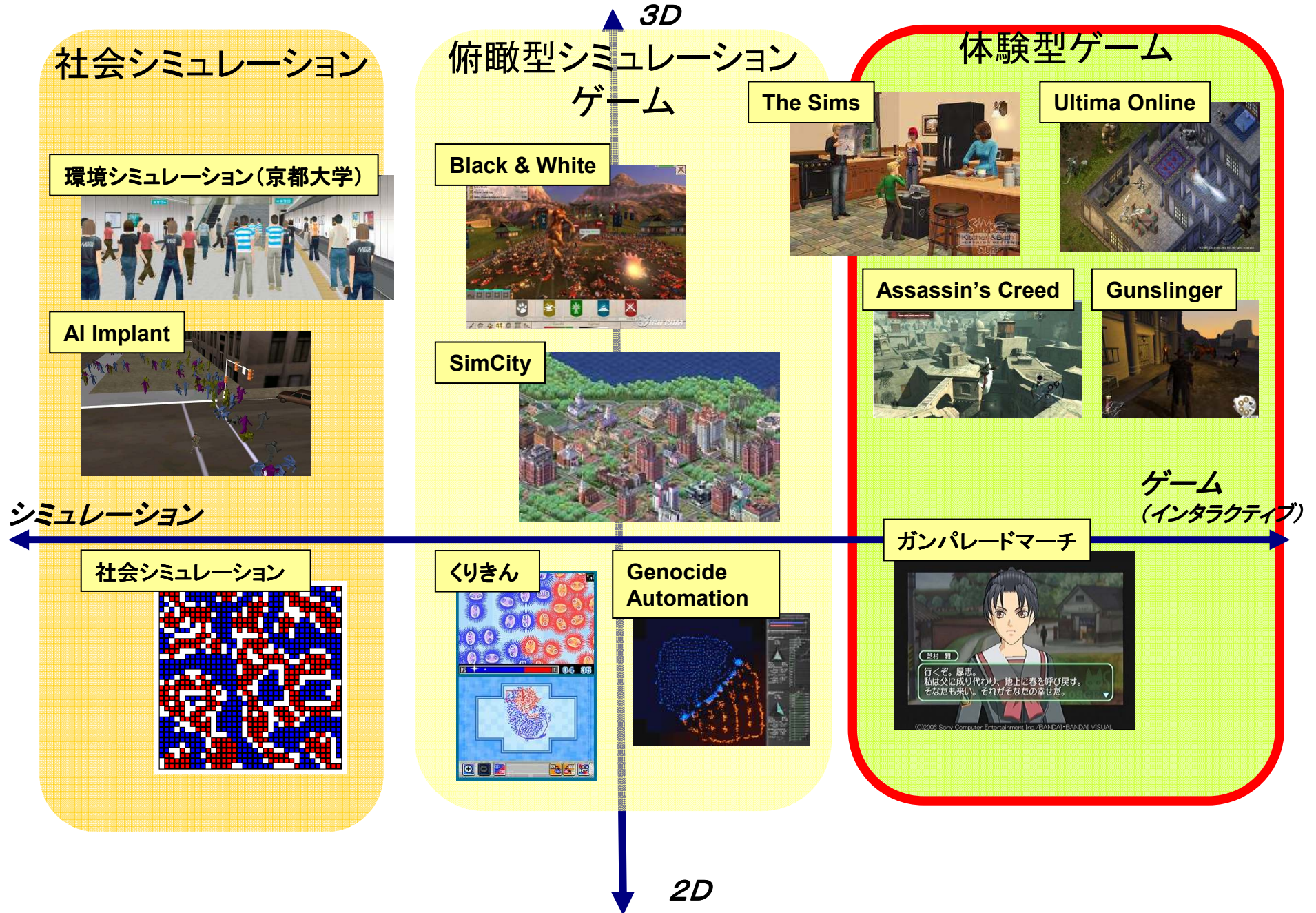


ゲーム
(インタラクティブ)

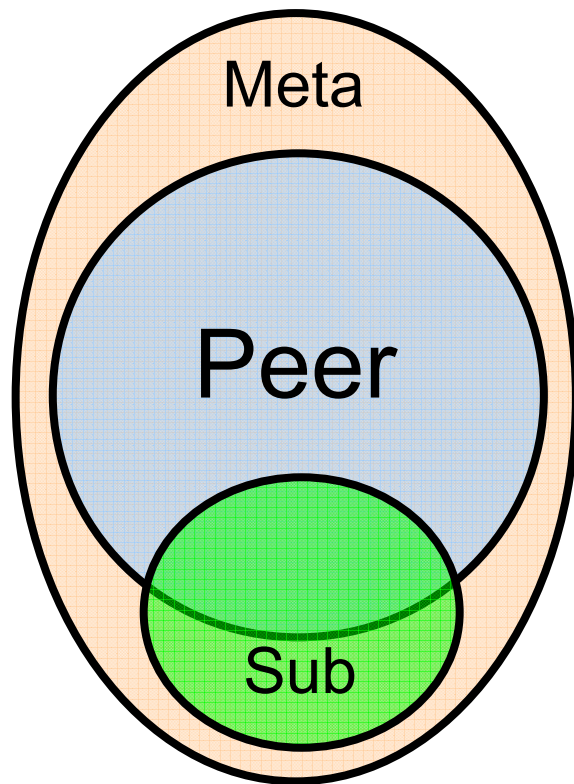
ガンパレードマーチ



社会的デジタル空間



The Sims シリーズのAIの作り方



Meta 
Peer 
Sub 

【原則】 周囲の対象に対する、あらゆる可能な行動から、ムード(幸せ) 係数を最大化する行動を選択する。

Sims (not under direct player control) choose what to do by selecting, from all of the possible behaviors in all of the objects, the behavior that maximizes their current happiness.

人をダイナミクス(力学系、動的な数値の仕組み)として動かす。
世界を動かす PeerAI(=キャラクターAI) を構築。

オブジェクトに仕込むデータ構造

Data (Class, State)

**Graphics (sprites, z-
Animations (skeletal)**

Sound Effects

Code (Edit)

- Main (object thread)
- External 1
- External 2
- External 3

パラメーター

グラフィックス
アニメーション

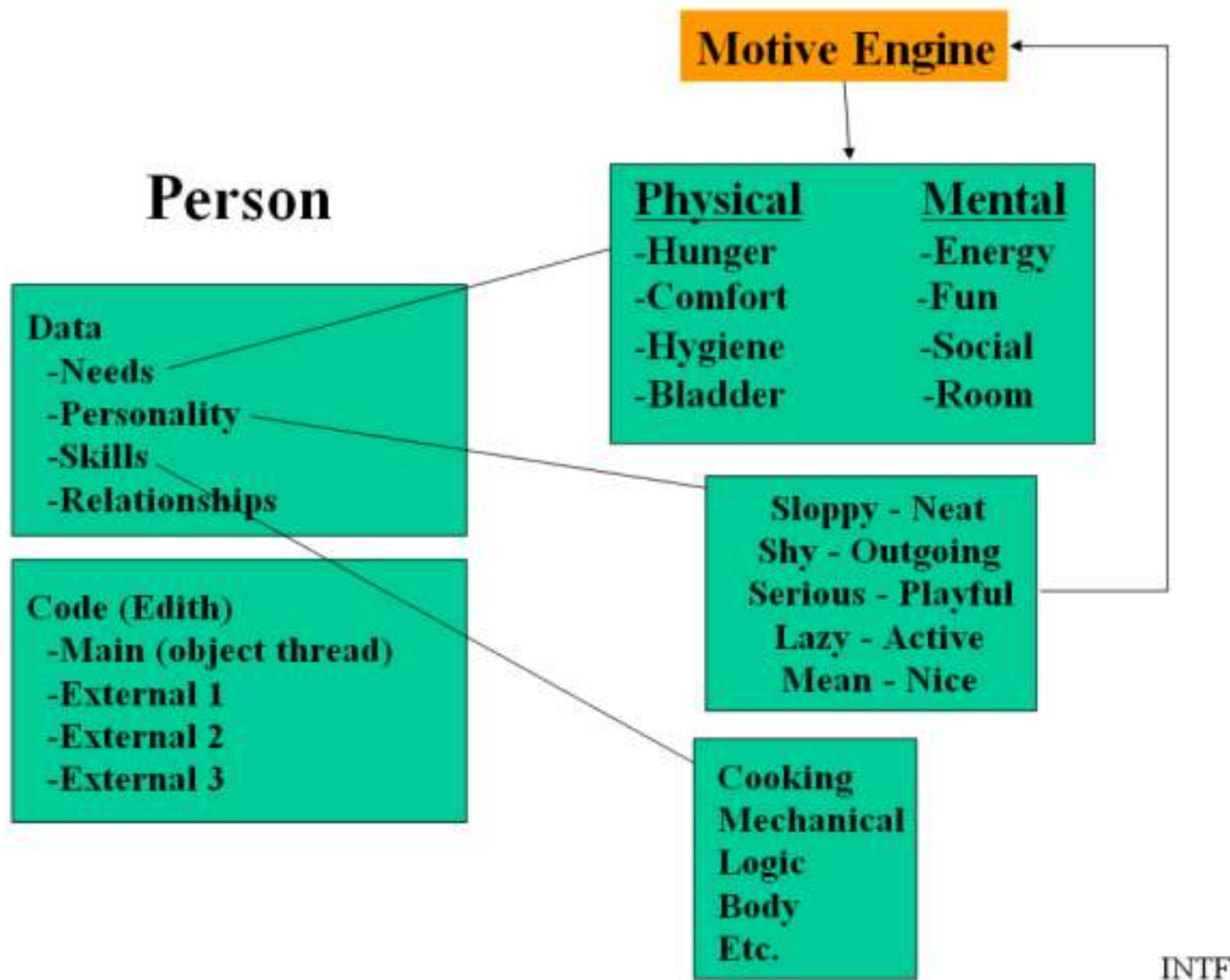
サウンド

メインスレッド

いろいろなインタラクションの仕方



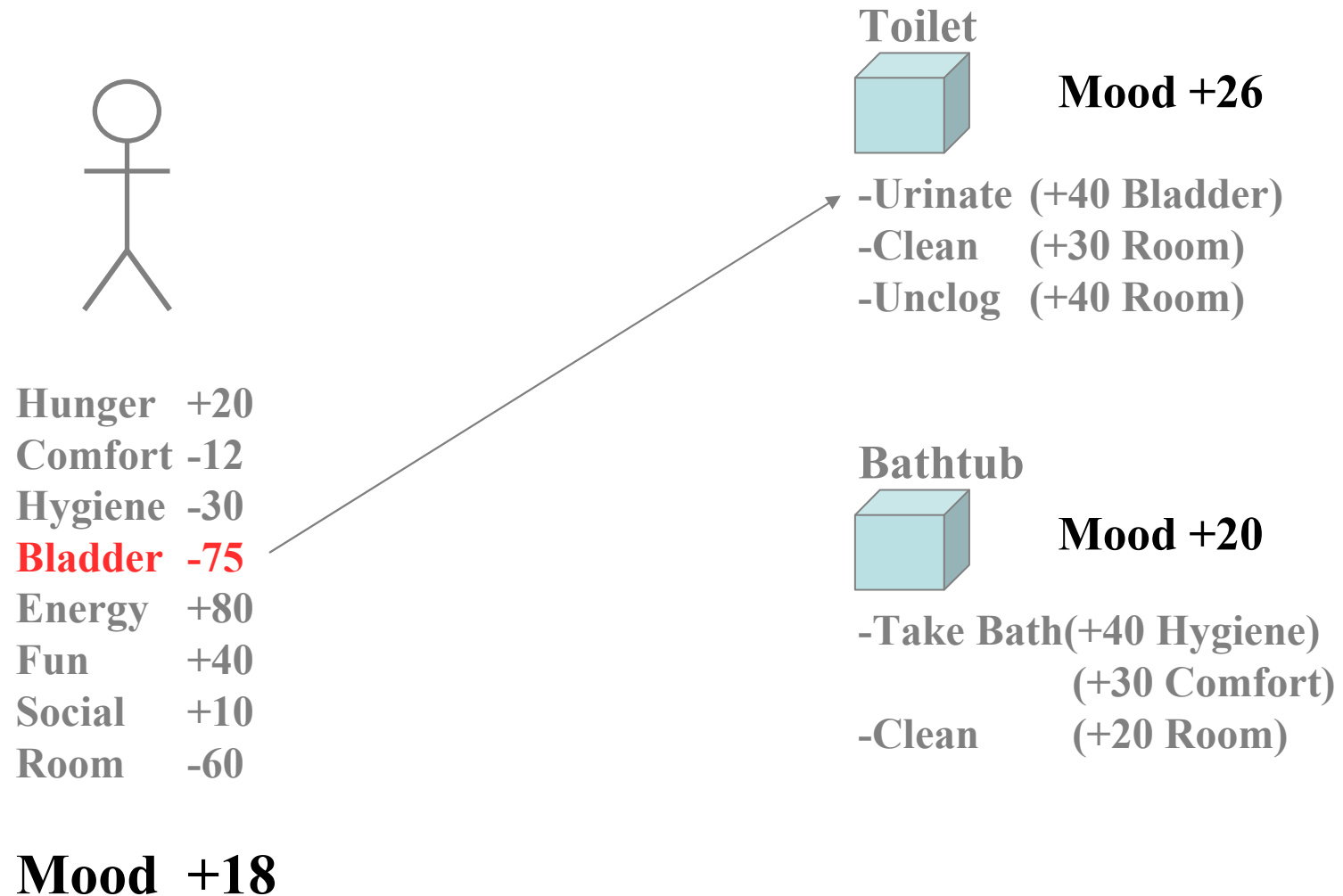
NPCに仕込むデータ構造



Ken Forbus, “Simulation and Modeling: Under the hood of The Sims” (NorthWestern大学、講義資料)

http://www.cs.northwestern.edu/%7Eforbus/c95-gd/lectures/The_Sims_Under_the_Hood_files/frame.htm

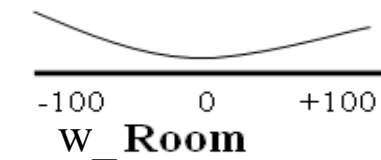
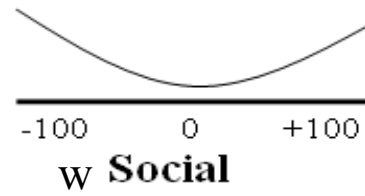
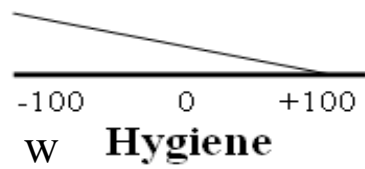
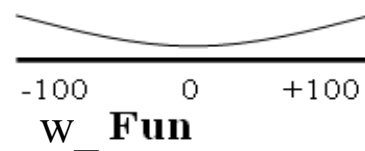
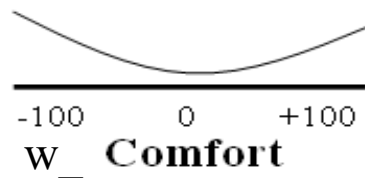
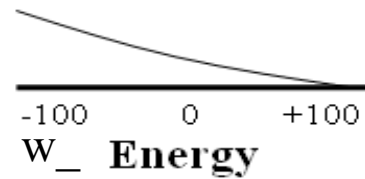
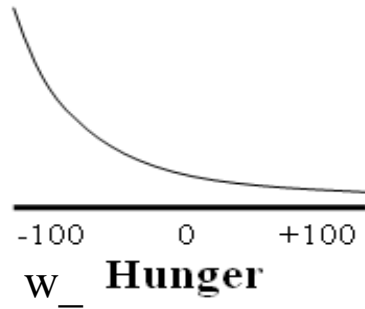
最適な行動を選択する



[原則] 周囲の対象に対する、あらゆる可能な行動から、**Happiness**（ここでは**Mood**）係数を最大化する行動を選択する。

Moodの計算のためのウェイト

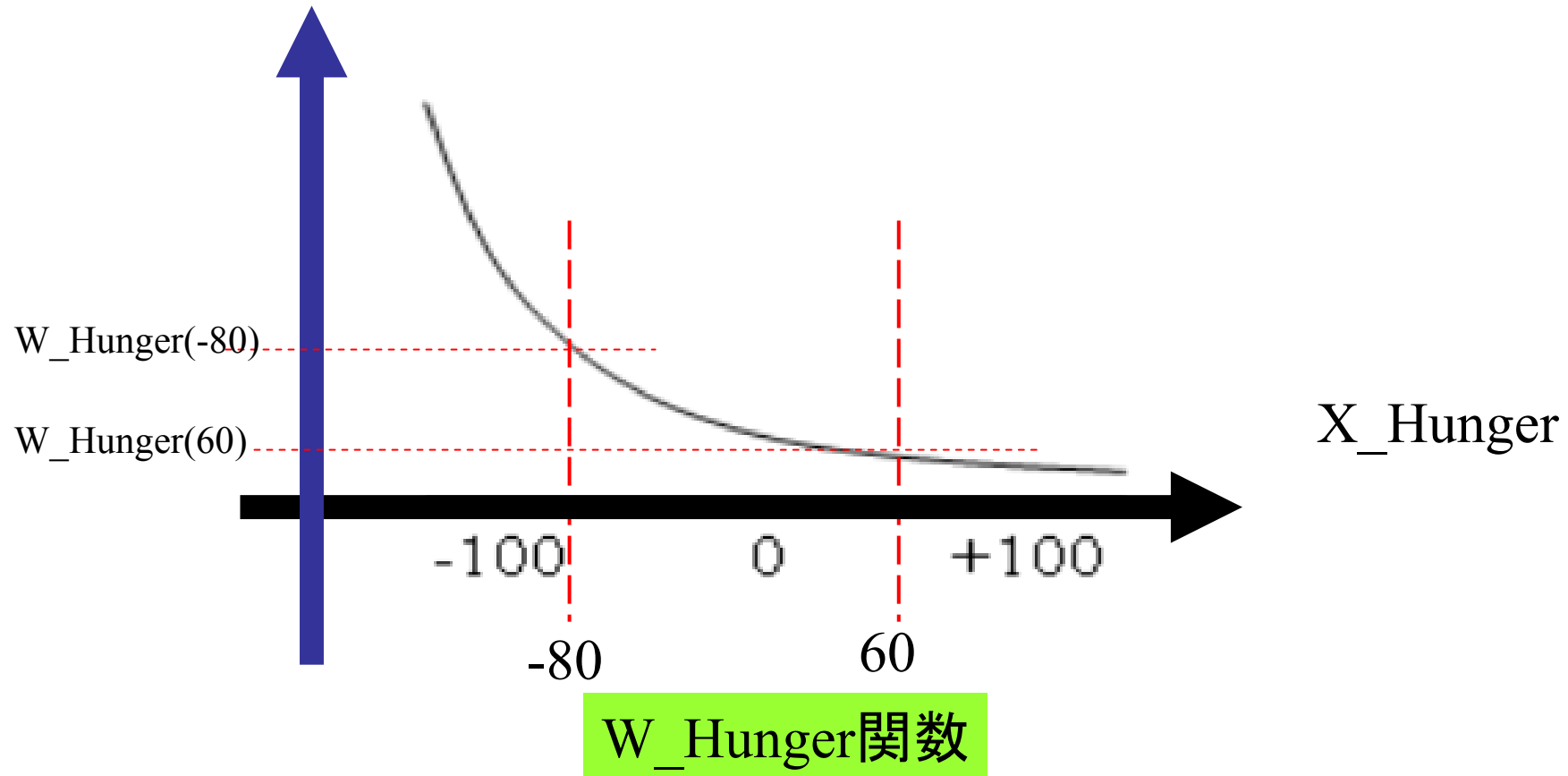
(ウェイト=Moodへどれくらい貢献するか)



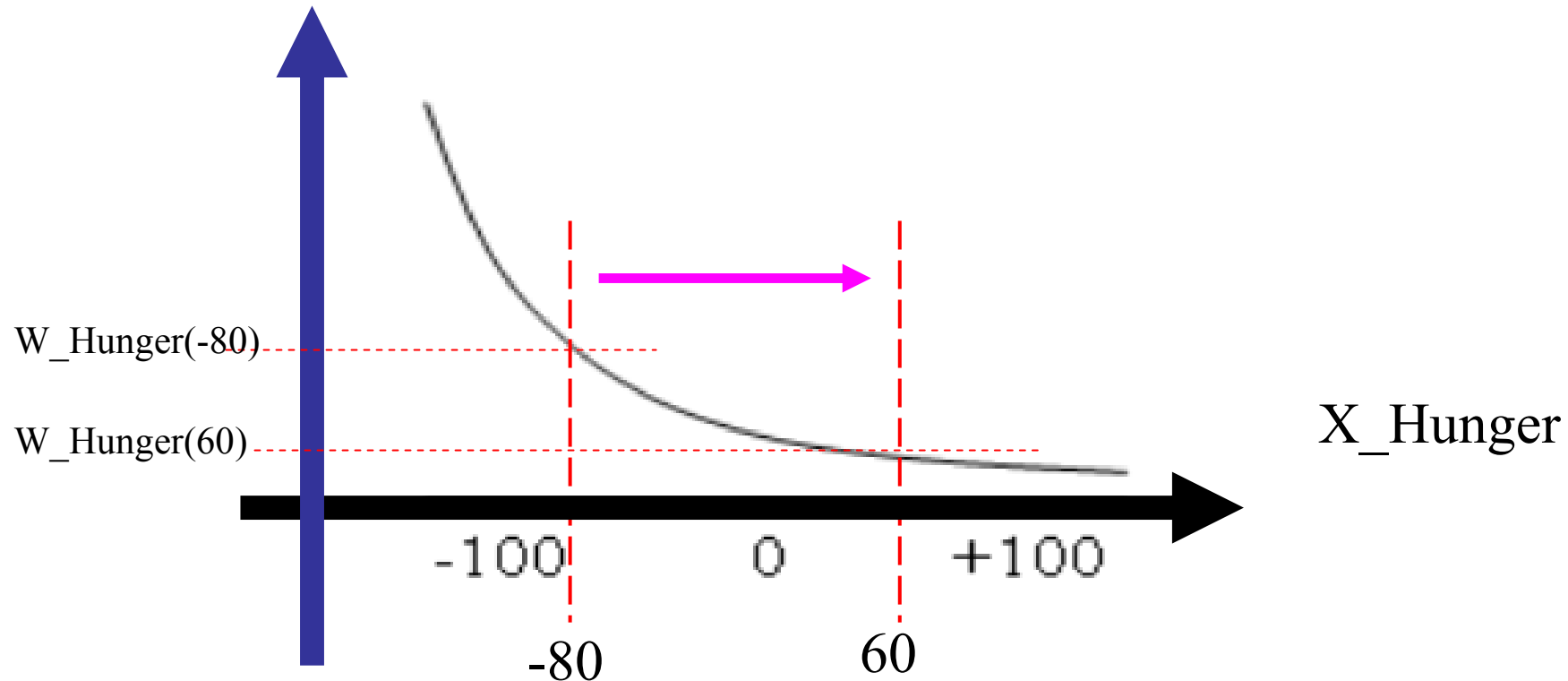
$$= \text{Mood} (-100..+100)$$

$$\text{Mood} = w_Hunger(X_Hunger) * X_Hunger + w_Energy(X_Energy) * X_Energy + \dots$$

計算の仕方



計算の仕方



W_Hunger関数

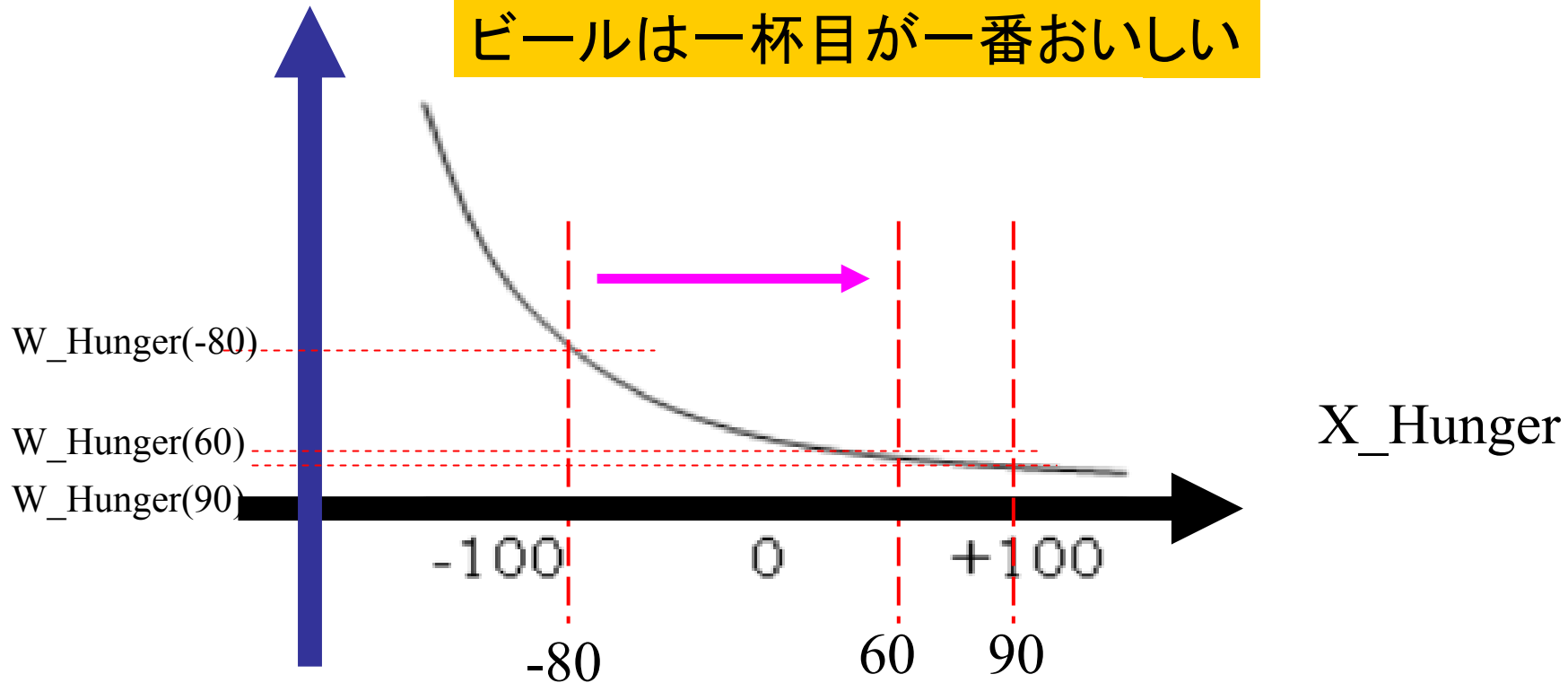
-80の時の食事欲求 = $W_Hunger(-80) * (-80)$

60の時の食事欲求 = $W_Hunger(60) * (60)$

Δ (変化分) = $W_Hunger(60) * (60) - W_Hunger(-80) * (-80)$

限界効用逓減の法則

ビールは一杯目が一番おいしい



$$\Delta (-80 \rightarrow 60) = W_Hunger(60) * (60) - W_Hunger(-80) * (-80)$$

$$\Delta (60 \rightarrow 90) = W_Hunger(90) * (80) - W_Hunger(90) * (60)$$

$\Delta (-80 \rightarrow 60)$ の方が $\Delta (60 \rightarrow 90)$ より圧倒的に大きい

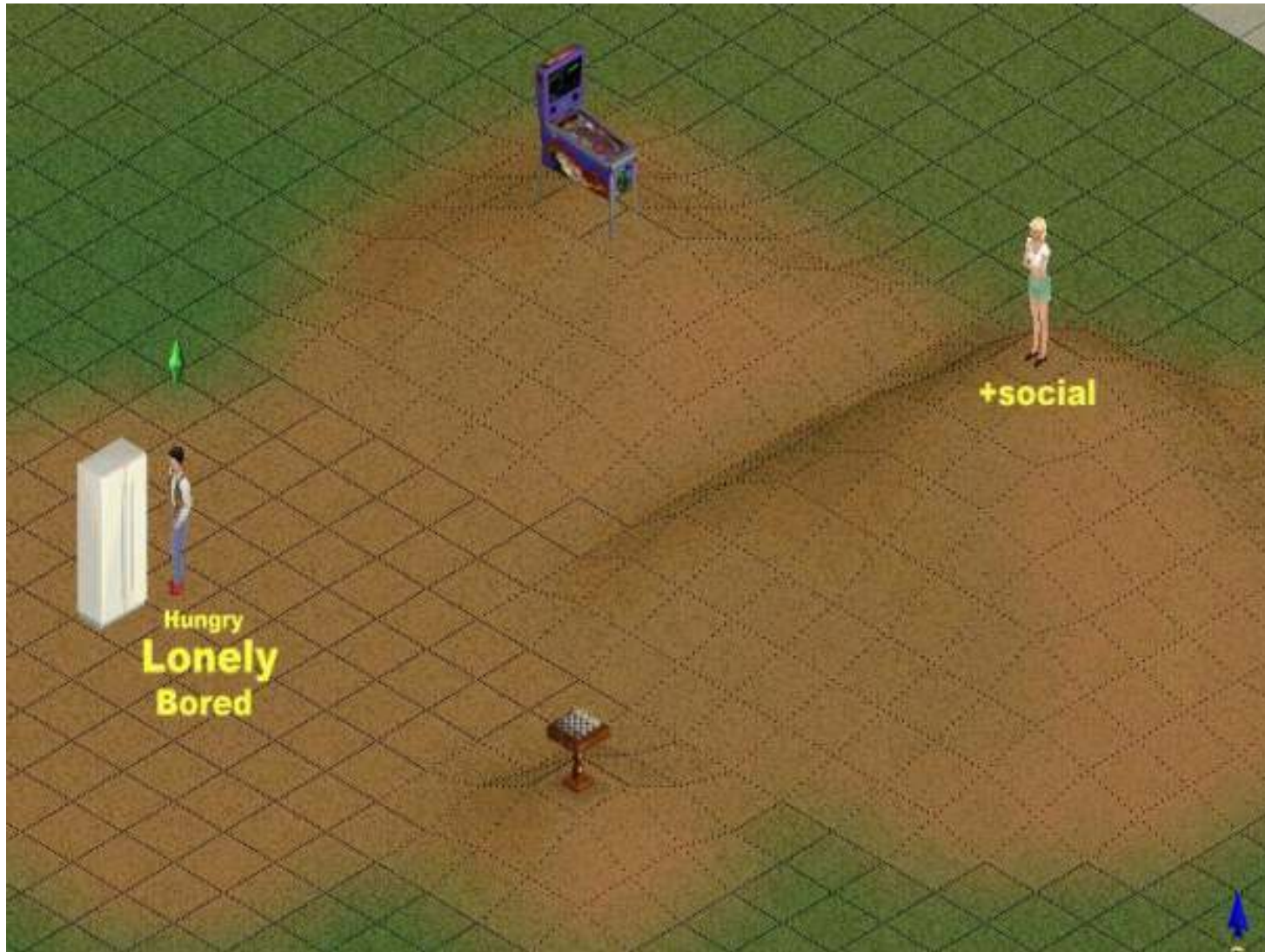
ある程度満たされたものを満たすより、
満たされないものをある程度満たす方が大きな満足をもたらす

Mood を最大化



冷蔵庫が最も総合的にMoodを上昇させるから

Mood を最大化



冷蔵庫へ行きます。

Mood を最大化

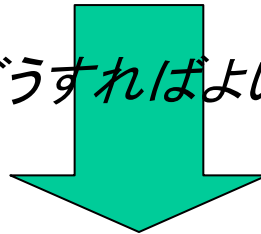


お腹が膨れたので、ちょっと退屈だから、女の子と話します。

The Sims の最初の目標

エージェントが社会生活を営みながら、
インタラクションする社会を作りたい。

どうすればよいか？

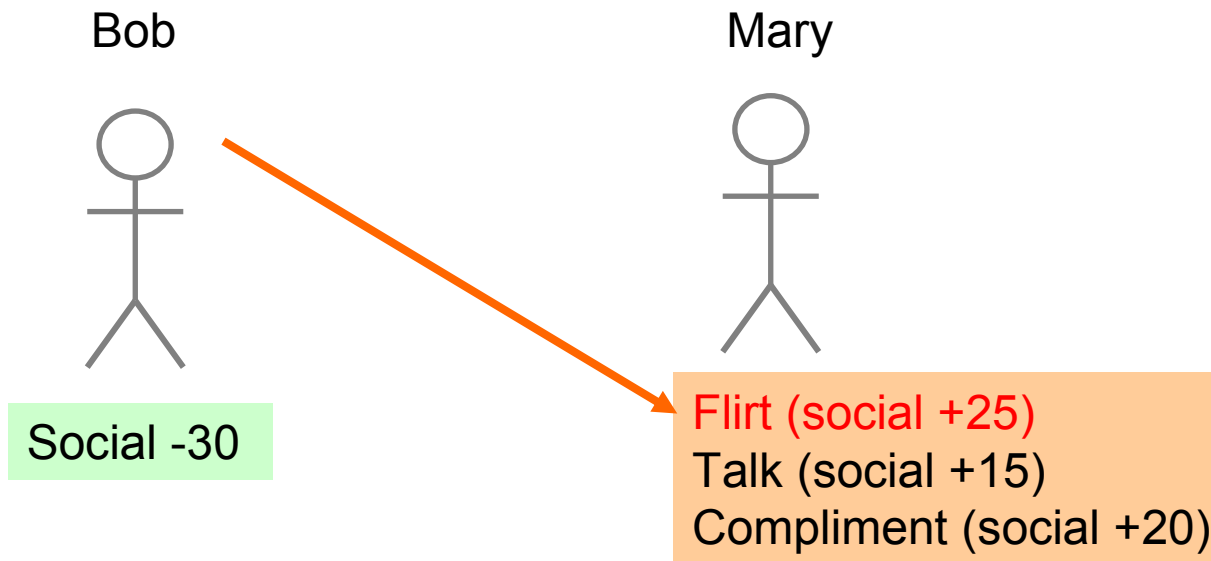


(1) 自律的に生活を営む。

(2) エージェント同士の
社会的にインタラクション

(3) Edith による
各行動のつくり込み

(2) NPC-NPC インタラクション



NPC 同士も、相手が自分のパラメーターをどう変化させるかを計算して決定する。上記では、Bob は、社交パラメータが低下しており、Flirt (いちゃつく)ことで、それが回復するので、Flirt を選択する。

NPC-NPC インタラクション

Bob



Social -30

Mary

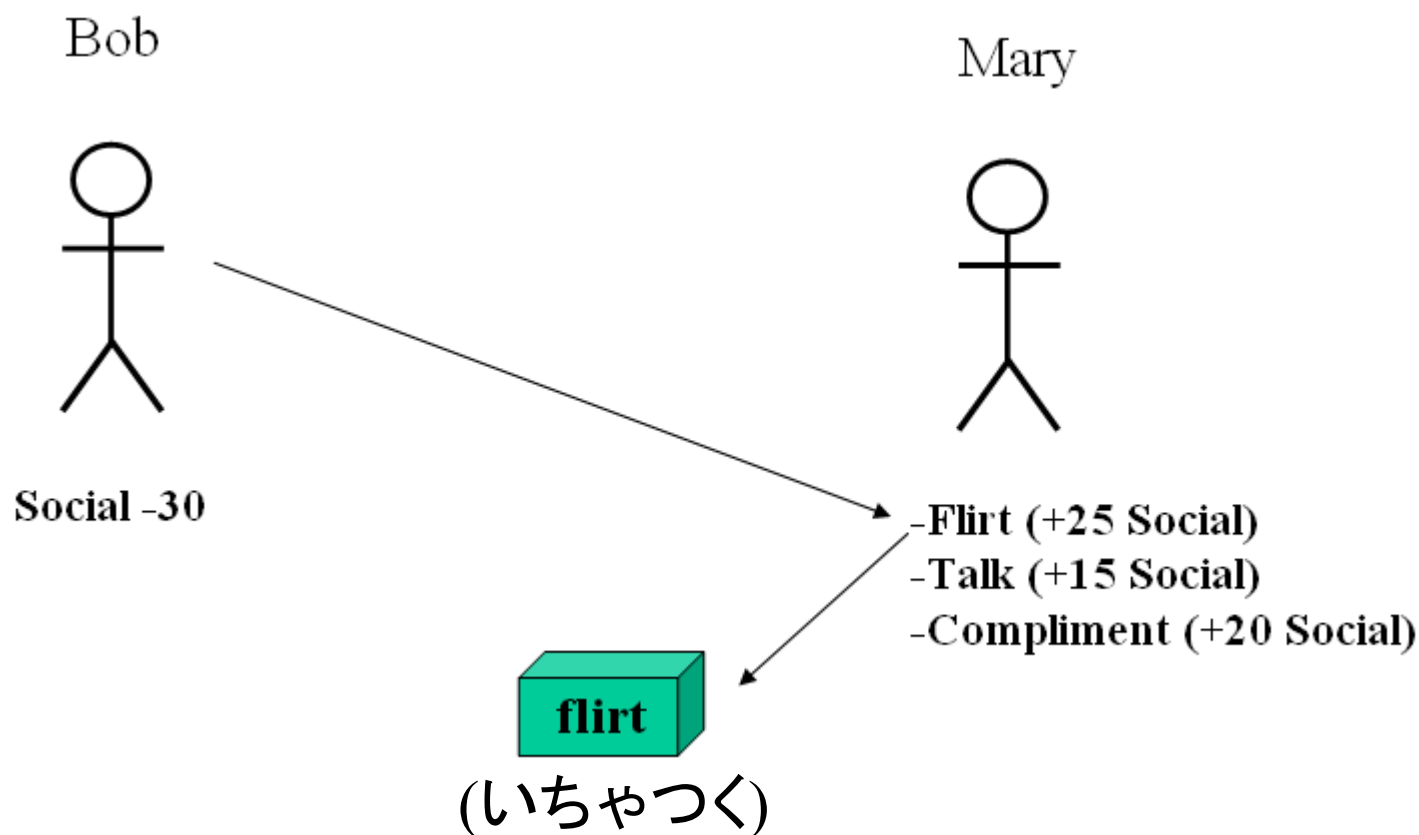


- Flirt (+25 Social)
- Talk (+15 Social)
- Compliment (+20 Social)

Kenneth D. Forbus “Some notes on programming objects in. The Sims”

http://www.qrg.northwestern.edu/papers/Files/Programming_Objects_in_The_Sims.pdf

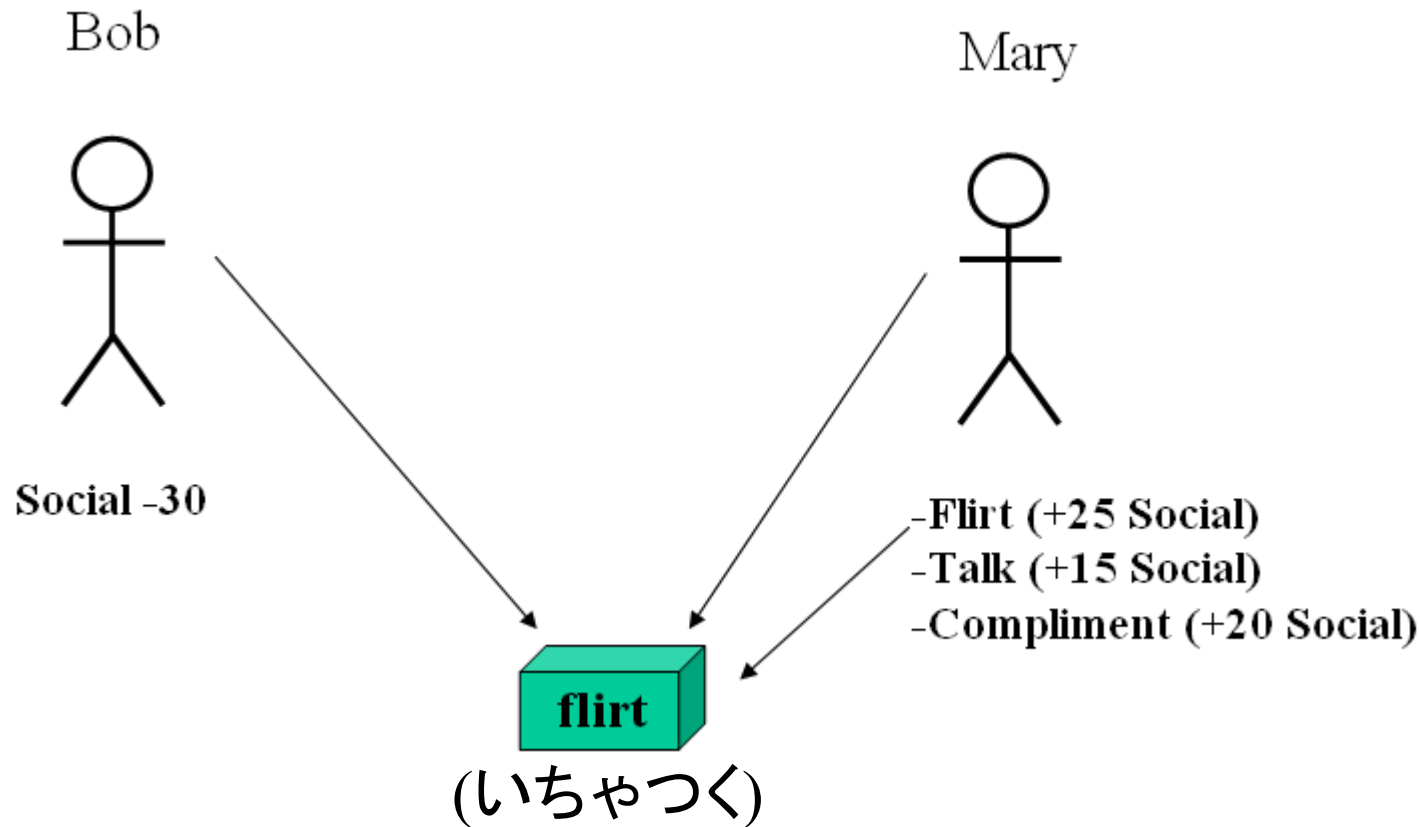
NPC-NPC インタラクション



Kenneth D. Forbus “Some notes on programming objects in. The Sims”

http://www.qrg.northwestern.edu/papers/Files/Programming_Objects_in_The_Sims.pdf

NPC-NPC インタラクション



Kenneth D. Forbus “Some notes on programming objects in. The Sims”

http://www.qrg.northwestern.edu/papers/Files/Programming_Objects_in_The_Sims.pdf



社会的デジタル空間

世界を俯瞰的に観察する
(シミュレーション)

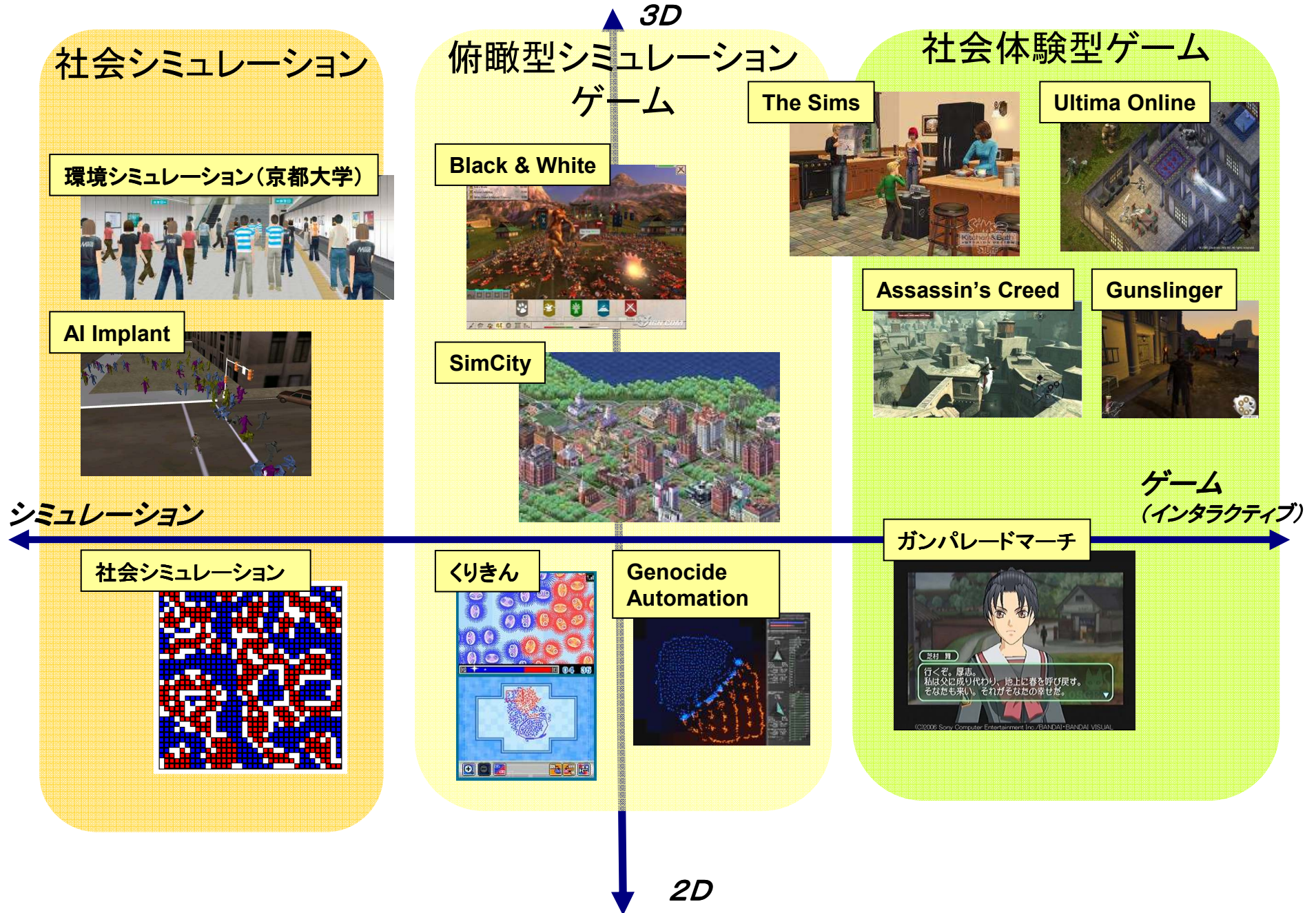
社会
シミュレーション
(プレイヤーが
干渉しない。
プレイヤーという
概念はない)

世界を内側から体験する
(ゲーム)

俯瞰型
シミュレーション
ゲーム
(プレイヤーが
神的な立場から干渉
する)

体験型
ゲーム
(プレイヤーが
エージェントの
一人として活動する)

社会的デジタル空間



社会的デジタル空間

世界を俯瞰的に観察する
(シミュレーション)

社会
シミュレーション

デジタル世界に
社会を実現する
基本技術

世界を内側から体験する
(ゲーム)

俯瞰型
シミュレーション
ゲーム

+インタラクション

体験型
ゲーム

+実現した
社会システムに
一人称として
入り込む

社会的デジタル空間

世界を俯瞰的に観察する
(シミュレーション)

世界を内側から体験する
(ゲーム)

社会
シミュレーション

デジタル世界に
社会を実現する
基本技術

俯瞰型
シミュレーション
ゲーム

+ インタラクション

体験型
ゲーム

+ 実現した
社会システムに
一人称として
入り込む

ゲーム体験として豊かにするには、全体のシステムを豊かにする視点と、
一人称視点で体験をリッチにして行く視点、二つの視点が必要

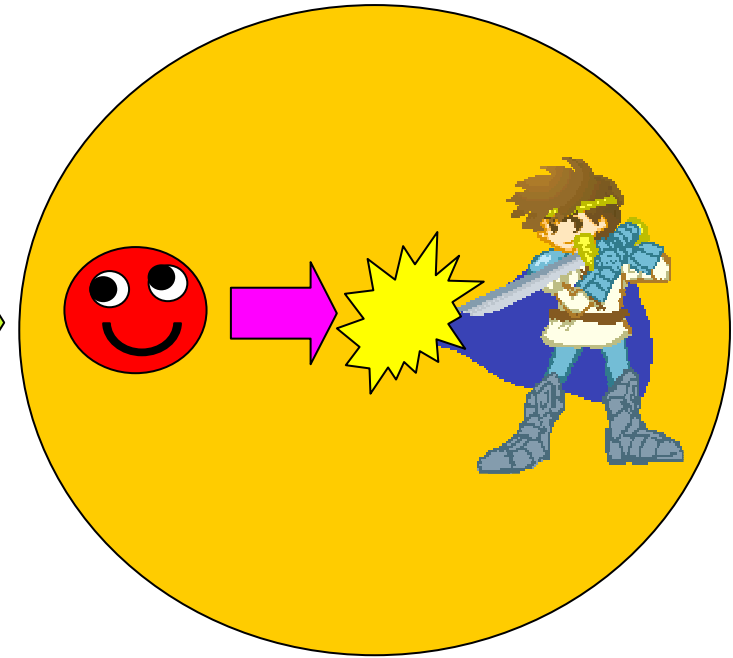
プログラマーとユーザー

ハードコーディング型実装

エージェントを作ってユーザーにぶつける。



プログラミング
実装



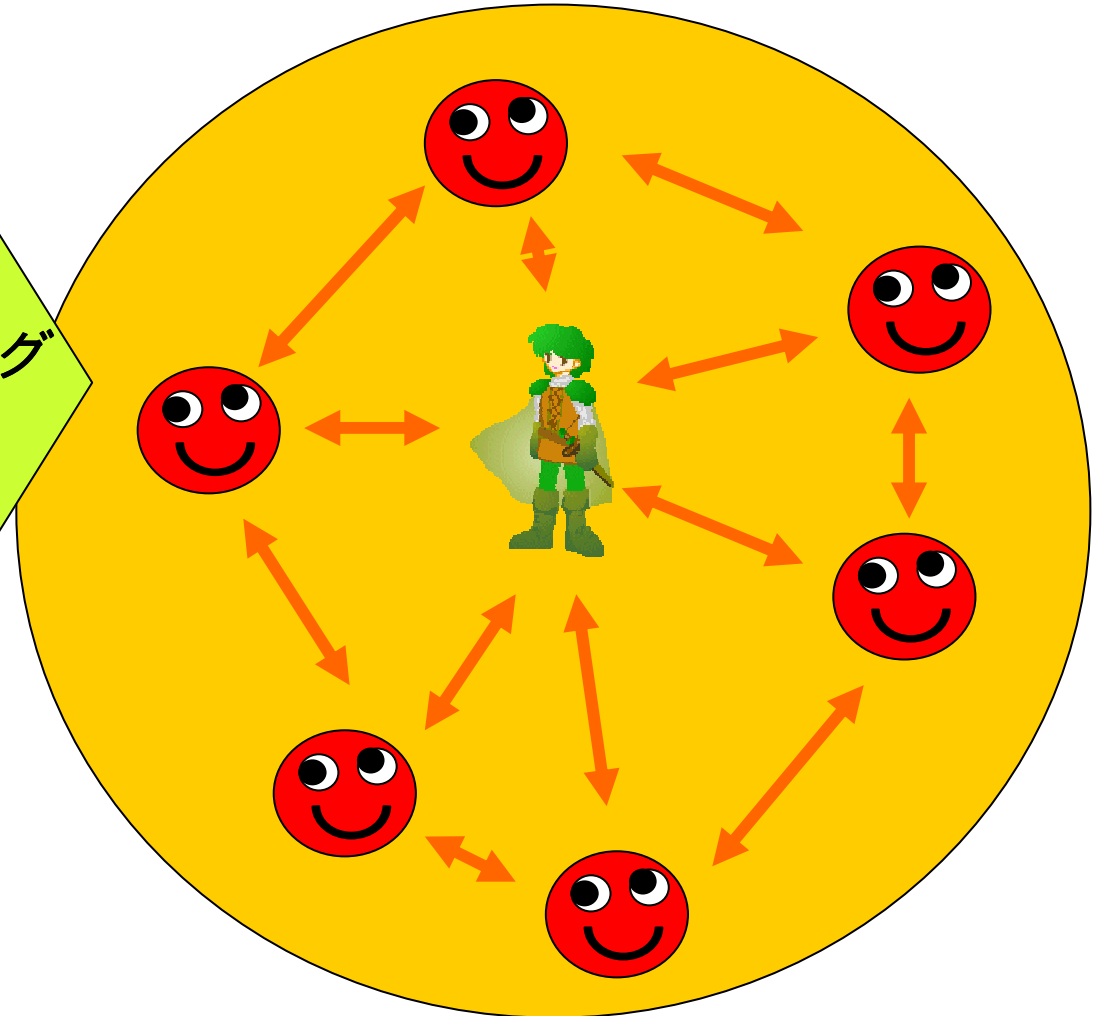
プログラマーとユーザー

エージェントたちがおもてなしするゲーム

マルチエージェントシステムを作ってユーザーに参加して貰う

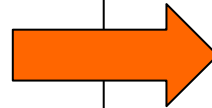


プログラミング
実装



マルチエージェント・システムを利用しよう (未開拓)

エージェントをユーザーに
ぶつける



マルチエージェント・システム
にユーザーが参加する

デジタルゲームの世界は、
「ユーザーとエージェントの対立構造」から、
「エージェントの協調によるマルチエージェント・システムを
どうゲームの中に利用して行くか」
が将来的な課題となるだろう。

エージェントを協調させてゲームの中に社会を作る
マルチエージェント・システム(Multi-agent system)

第2部 まとめ

第2部まとめ

デジタルゲームにおける人工知能技術

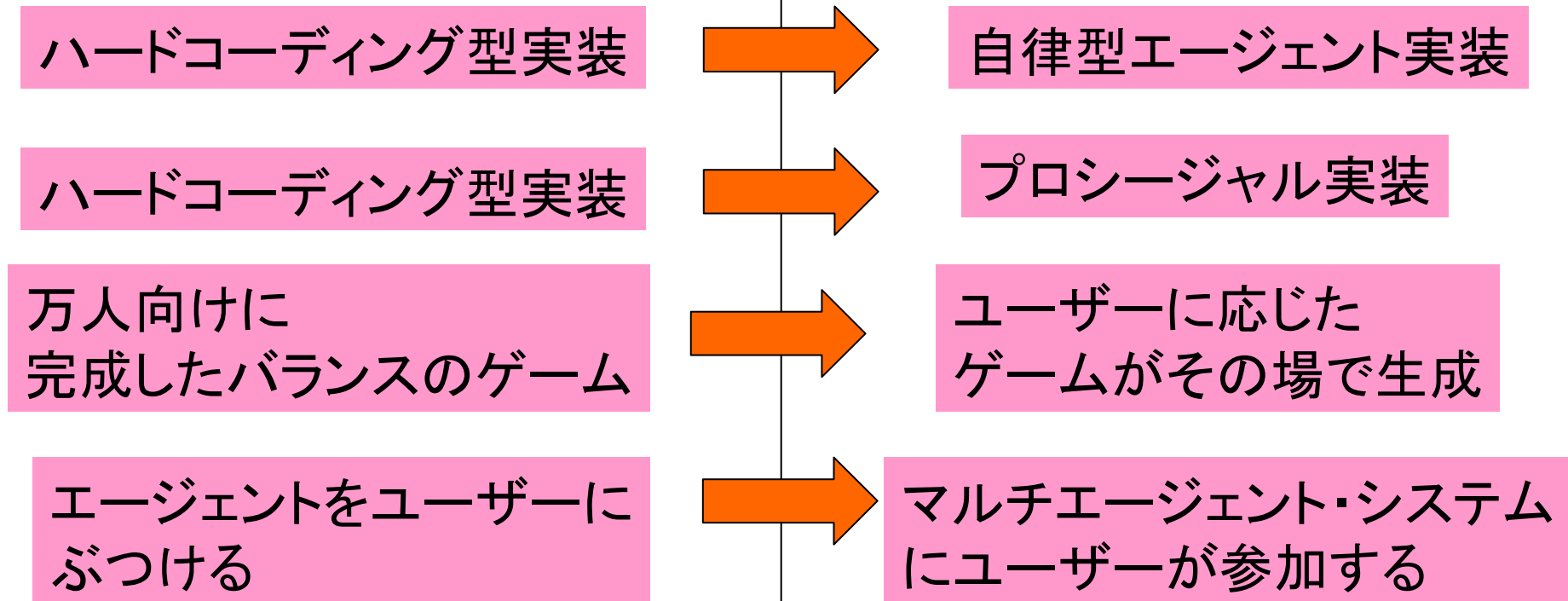
第一章 自律型エージェント (Autonomous Agent)

第二章 プロシージャル (Procedural)

第三章 メタAI (Meta-AI)

第四章 マルチエージェント (Multi-agent)

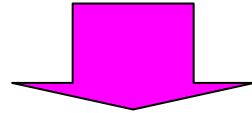
第2部まとめ



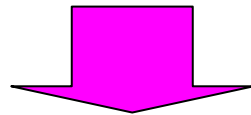
自律発展する世界の中の、自律的に判断・運動するAI、かつ、それらのAIが連携して、ユーザーと多様なインタラクションを生成する世界。また、それらのゲーム世界全体がユーザーのアクションの履歴に応じて変化して行く世界。

これからのゲームの行方

現状： 西欧の暴力ゲームに引きずられる形



自律発展する世界の中の、自律的に判断・運動するAI、かつ、それらのAIが連携して、ユーザーと多様なインタラクションを生成する世界。また、それらのゲーム世界全体がユーザーのアクションの履歴に応じて変化して行く世界。



新しいゲームの生成に期待、新しいゲームクリエイターの誕生

[付録] ゲームAIの学習・研究の仕方

- (I) 日本語の文献が殆どない、或いは、資料は英語でよいものが多いので英語は必須。
- (II) ゲームAIの教科書は英語で多く出版されている。
- (III) WEB上にゲームAIでよい英語論文が見つかる。
- (IV) 日本語の資料は少ないが質が高い。

(I) 参考文献(日本語)

- (1) 「FSM」「プランニング」「評価値法」など、
ゲームAIの基礎技術については、



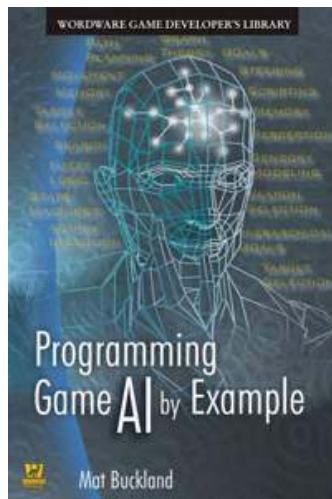
オライリー・ジャパン

「**実例で学ぶゲームAIプログラミング**」
(**Mat Buckland** 著、**松田晃一** 訳)

の解説が優れています。

ソースコードはWEB

<http://www.wordware.com/files/ai/>



できれば原書で読みましょう！

(I)参考文献(日本語)

(2)

- ①「世界表現」「プランニング」については、IGDA日本のHPの「ダウンロード」から、三宅が書いた第1, 2, 5, 6回セミナーの教科書、CEDEC2006の資料がDLできます。

<http://www.igda.jp/>

(センサーの実装の仕方、記憶の利用法などを知りたい方は必読)
上記サイト復旧中のため一時的に以下のフォルダから

<http://server02.joeswebhosting.net/~ig1347/modules/mydownloads/>

- ② デジタルコンテンツ協会

デジタルコンテンツ制作の先端技術応用に関する調査研究報告書(第3章)

http://www.dcaj.org/report/2007/ix1_07.html

(PDFファイルがダウンロード出来ます。)

- ③ 人工知能学会誌 Vol. 23 No. 1 (2008 年 1 月) 「ゲームAI特集」
「デジタルゲームにおける人工知能技術の応用」(三宅)

(II)参考文献(英語)

WEB

Mat Buckland

ai-junkie <http://www.ai-junkie.com/ai-junkie.html>

Craig Reynolds

RAYNOLDS <http://www.red3d.com/>

リンク集 <http://www.red3d.com/cwr/games/>

Steven Rabin

GameAI <http://www.gameai.com/>

CGF-AI

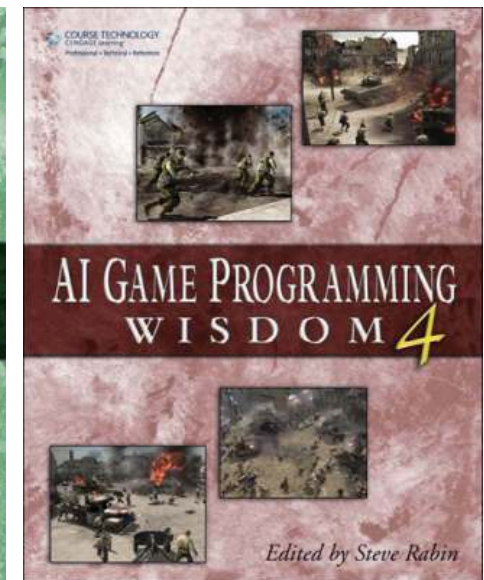
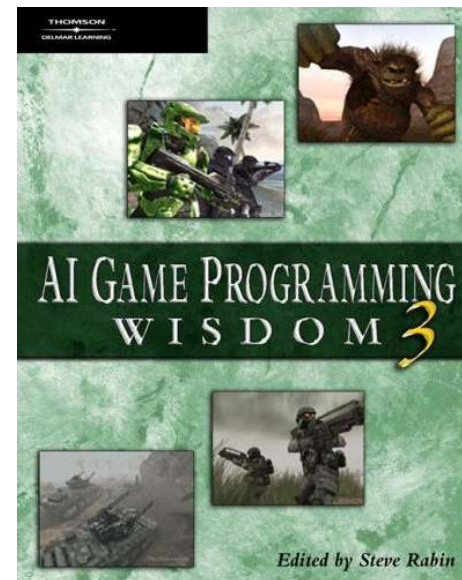
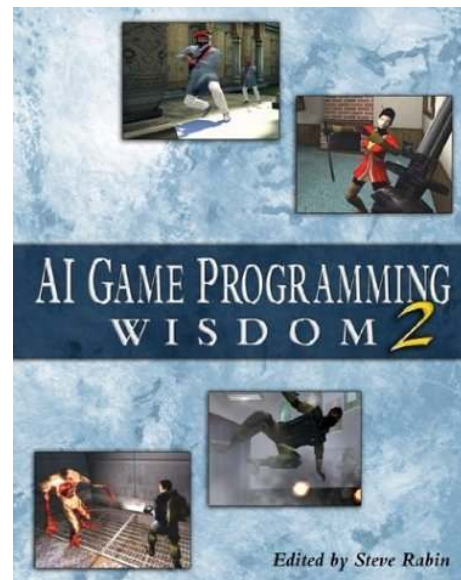
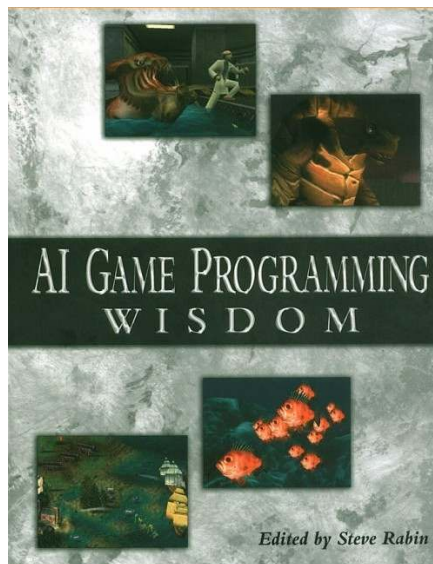
CGF-AI <http://www.cgf-ai.com/>

リンク集 <http://www.cgf-ai.com/links.html>

(II)参考文献(英語)

書籍

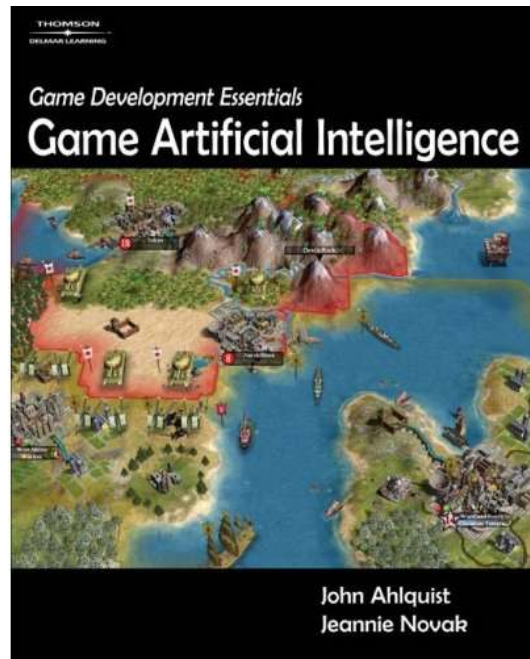
AI Game Programming Wisdom 1 - 4



ゲーム開発者、研究者による、
それぞれのタイトルの実装例、研究成果

(II)参考文献(英語)

書籍



欧米のゲームAIの歴史から最先端までがわかりやすく、解説されています。

文科系の方も理科系の方も読んで楽しめる本です。

John Ahlquist, Jeannie Novak
Game Development Essentials: Game Artificial Intelligence



References for Killzone

- [1] **William van der Sterren** (2001), "Terrain Reasoning for 3D Action Games", http://www.cgf-ai.com/docs/gdc2001_paper.pdf
- [2] **William van der Sterren** (2001) , "Terrain Reasoning for 3D Action Games(GDC2001 PPT)", http://www.cgf-ai.com/docs/gdc2001_slides.pdf
- [3] **Remco Straatman, Arjen Beij, William van der Sterren** (2005) , "Killzone's AI : Dynamic Procedural Combat Tactics", http://www.cgf-ai.com/docs/straatman_remco_killzone_ai.pdf
- [4] **Arjen Beij, William van der Sterren** (2005), "Killzone's AI : Dynamic Procedural Combat Tactics (GDC2005)", http://www.cgf-ai.com/docs/killzone_ai_gdc2005_slides.pdf
- [5] **Damian Isla** (2005), "Dude, where's my Warthog? From Pathfinding to General Spatial Competence", <http://www.aiide.org/aiide2005/talks/isla.ppt>



Reference for Halo & Halo2

- Damian Isla (2005), “Dude, where’s my Warthog? From Pathfinding to General Spatial Competence”,
<http://www.aiide.org/aiide2005/talks/isla.ppt>
http://nikon.bungie.org/misc/aiide_2005_pathfinding/index.html
- Damian Isla (2005), Handling Complexity in the Halo 2 AI, Game Developer's Conference Proceedings.,
http://www.gamasutra.com/gdc2005/features/20050311/isla_01.shtml
- Jaime Griesemer(2002),The Illusion of Intelligence: The Integration of AI and Level Design in Halo,
<http://halo.bungie.org/misc/gdc.2002.haloai/talk.html>
- Robert Valdes(2004), “In the Mind of the Enemy The Artificial Intelligence of Halo2”,
<http://www.stuffo.com/halo2-ai.htm> (現在はclosed)

References for C4 Architecture

- (1) MIT Media Lab Synthetic Characters Group,
<http://characters.media.mit.edu/>
- (2) R. Burke, D. Isla, M. Downie, Y. Ivanov, B. Blumberg,
(GDC2001), “CreatureSmarts: The Art and Architecture of a
Virtual Brain”, <http://characters.media.mit.edu/Papers/gdc01.pdf>
- (3) D. Isla, R. Burke, M. Downie, B. Blumberg (2001)., “A Layered
Brain Architecture for Synthetic Creatures”,
<http://characters.media.mit.edu/Papers/ijcai01.pdf>
- (4) D. Isla, B. Blumberg (2002), “Object Persistence for Synthetic
Characters”, [http://characters.media.mit.edu/Papers/objectPersistence.p
df](http://characters.media.mit.edu/Papers/objectPersistence.pdf)
- (5) Movies of Duncan, <http://web.media.mit.edu/~bruce/whatsnew.html>
- (6) Object Persistence for Synthetic Characters. D. Isla, B.
Blumberg. In the Proceedings of the First International Joint
Conference on Autonomous Agents and Multiagent Systems,
AAIAS2000 [http://characters.media.mit.edu/Papers/objectPersistence](http://characters.media.mit.edu/Papers/objectPersistence.pdf)

References for F.E.A.R.



- 論文 Orkin, J. (2006), [3 States & a Plan: The AI of F.E.A.R.](#), Game Developer's Conference Proceedings.
- Jeff Orkin, Applying Goal-Oriented Action Planning to Games, AI Game Programming Wisdom 2, Charles River Media., 217-228, (2003)
- 参考文献(I) Mat Buckland, “Programming Game AI by Example”, Chapter 9, WORDWARE publishing
(第9章とそのサンプルコードはゴール指向型プランニングの優れた解説です。教科書をお探しの方は、ゲームAIについて最良の書の一つです。推薦します。)
[オライリージャパンより「実例で学ぶゲームAIプログラミング」として翻訳が出版されています。](#)
- 参考文献(II) Jeff Orkin' HP <http://web.media.mit.edu/~jorkin/>
(Jeff Orkin は、米におけるゲームAIにおけるゴール指向型プランニングの推進者の一人。著者のサイトに豊富な情報があります。)
- 参考文献(III) 星野 瑠美子, “F.E.A.R.のAI - 3つの状態とゴール指向プランニングシステム”
http://www.igda.jp/modules/xeblog/?action_xeblog_details=1&blog_id=62
(IGDA Japan サイト内、GDC2006講演の紹介)

ご清聴ありがとうございました。



Photo from <http://www.cyberleaf.com/>

これ以外に、意見や質問があれば、メールかアンケートへ

y.m.4160@gmail.com

(IGDA Japan登録アドレス yoichi-m@pk9.so-net.ne.jp)

<http://www.igda.jp>