

GDC2015 報告会

「ゲームから社会へ、社会からゲームへ、 人工知能技術の拡がり」

三宅 陽一郎
(IGDA日本ゲームAI専門部会代表)

@miyayou
<https://www.facebook.com/youichiro.miyake>

y.m.4160@gmail.com
<http://blogai.igda.jp>

<http://www.linkedin.com/in/miyayou>

2015.3.21

<http://www.igda.jp/?p=408>

カリフォルニア大学バークレー校(見学)



GDC2015 各社講演資料集(速報)

- GDC講演資料集1

<http://blogai.igda.jp/article/115160418.html>

- GDC講演資料集2

<http://blogai.igda.jp/article/115279691.html>

- 殆どは GDC Vault (一部有料)に上がるはずです。

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

GDC報告会資料集2008-2014(三宅)

GDC2014

http://igda.sakura.ne.jp/sblo_files/ai-igdajp/igdaj/YMiyake_IGDA_GDC_Report_2014_4_5.pdf

GDC2013

http://igda.sakura.ne.jp/sblo_files/ai-igdajp/igdaj/YMiyake_IGDAJ_GDC_Report_2013_4_13.pdf

【記事】試行錯誤やインタラクションで「学習」するAI

<http://www.inside-games.jp/article/2013/04/29/66067.html>

GDC2012

http://igda.sakura.ne.jp/sblo_files/ai-igdajp/igdaj/YMiyake_IGDA_GDC_Report_2013_3_31.pdf

GDC2011

<http://www.slideshare.net/youichiromiyake/igda-gdc2011ai>

http://igda.sakura.ne.jp/sblo_files/ai-igdajp/igdaj/YMiyake_IGDAJ_GDC_2011_4_16.pdf

GDC2010

http://igda.sakura.ne.jp/sblo_files/ai-igdajp/GDC2010/YMiyake_GDC_Report_2010_4_3.pdf

GDC2009

<http://digrajapan.org/?wpdmact=process&did=NS5ob3RsaW5r>

http://igda.sakura.ne.jp/sblo_files/ai-igdajp/dcaj/YMiyake_DCAJ_2009_3_31.pdf

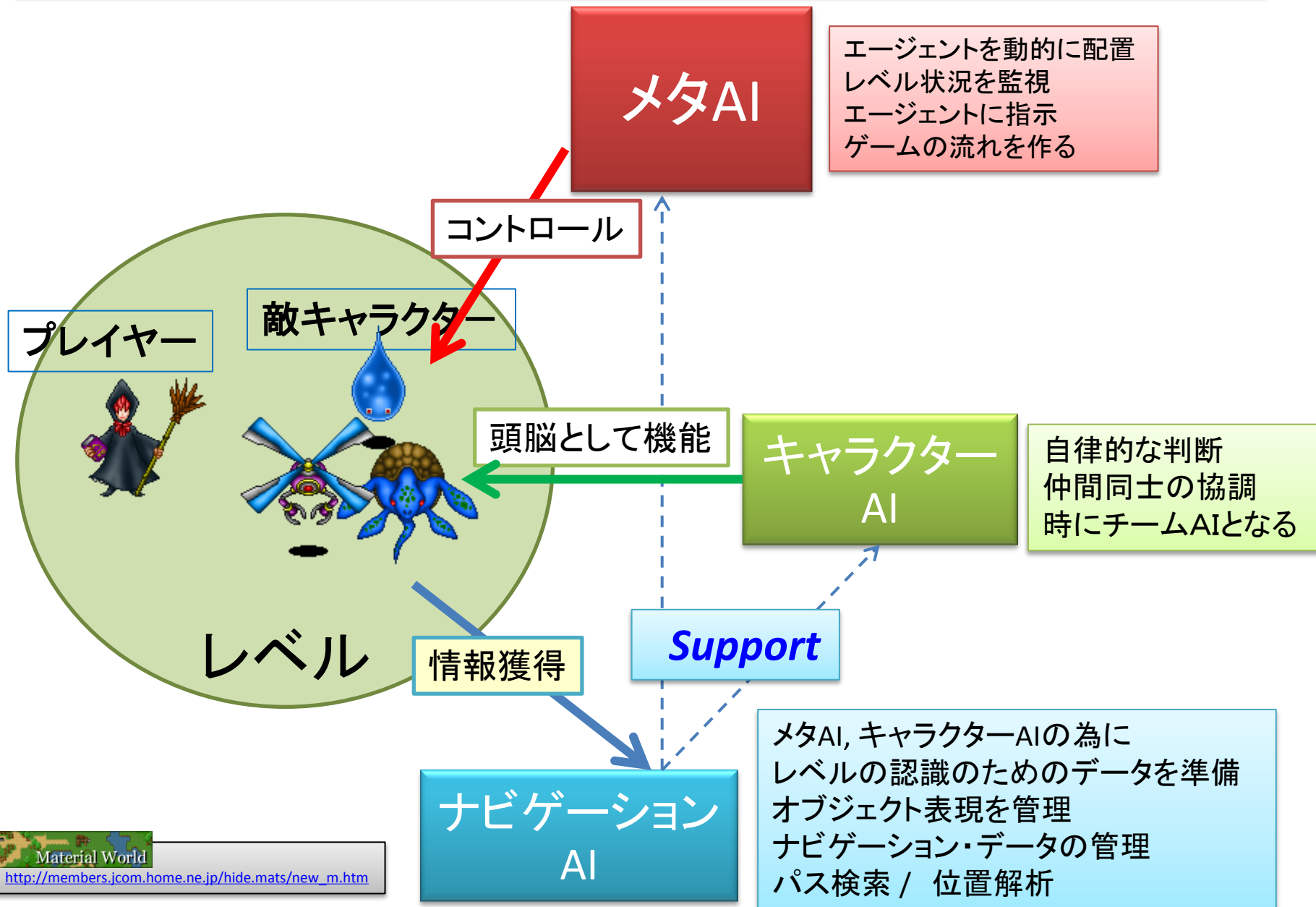
GDC2008

http://igda.sakura.ne.jp/sblo_files/ai-igdajp/AI/IDGA_GDC08_Miyake_public.pdf

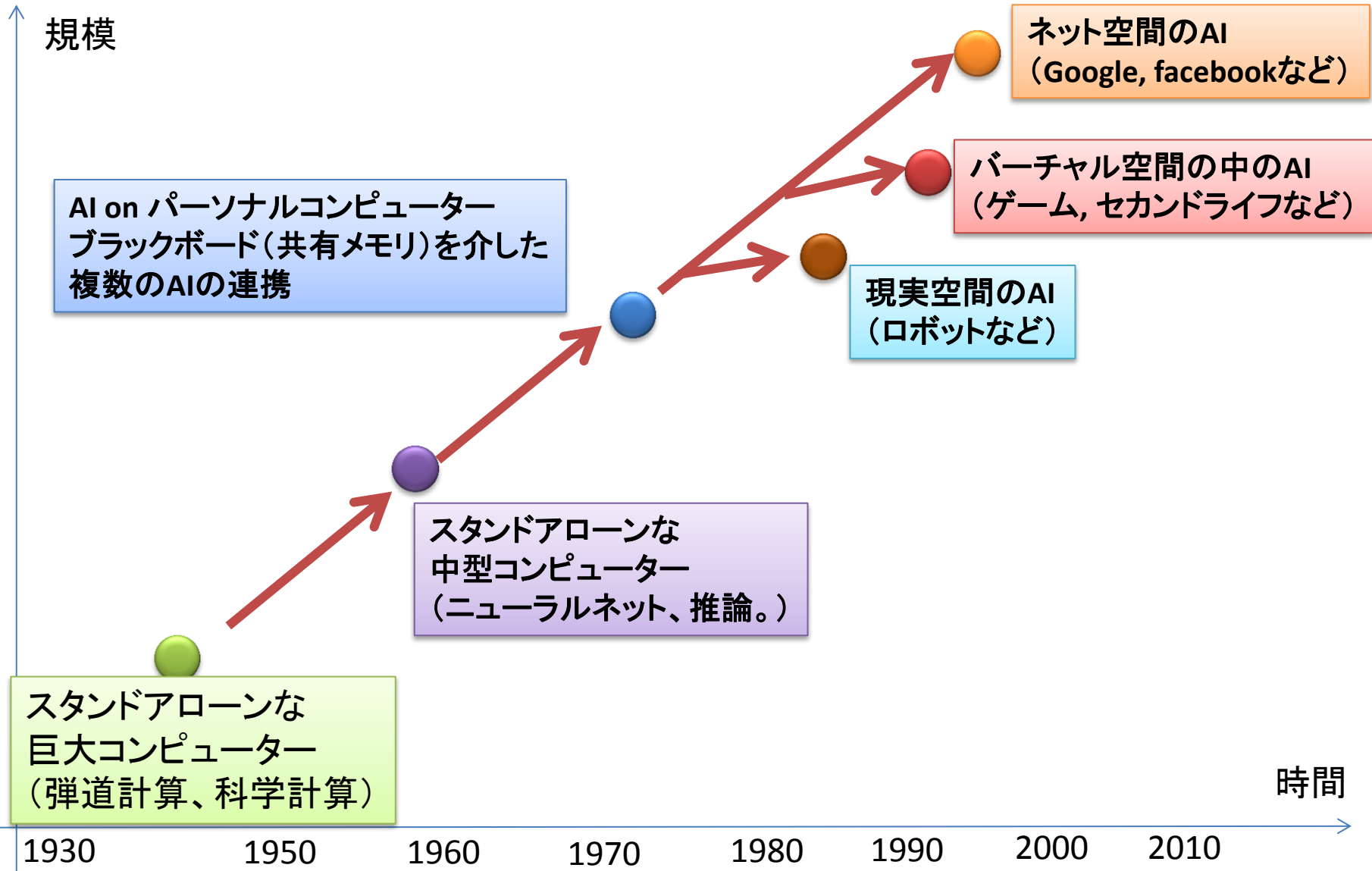
ゲームAI技術

- AI技術 = キャラクターの意思決定など。
- プロシージャル技術(自動生成/自動配置技術)

現代ゲームAIの仕組み



人工知能の現代までの流れ



2011年までのGDCのゲームAI全体の傾向

2011年： ゲームAIの大型から小型化 & 大型AI技術の固定化

*有名AI開発者の独立(Halo3, The Sims 3, Killzone 2...)Darkspore,
The Sims: Medieval, StarCraft 2, Havok AI, Kynapse, Insomniac PathEngine*

2010年： 大型ゲームAIの完成 & キャラクター制御の高度化

Uncharted 2, Sprinter Cell:Conviction, Killzone2, The Sims 3, Bioshock 2, FIFA
http://igda.sakura.ne.jp/sblo_files/ai-igdajp/GDC2010/YMiyake_GDC_Report_2010_4_3.pdf

2009年： キャラクターAI(アニメーション&パスファインディング) & メタAI

Morpheme 2.0, Warhammer Online(Kynapse), Halo Wars, Left 4 Dead

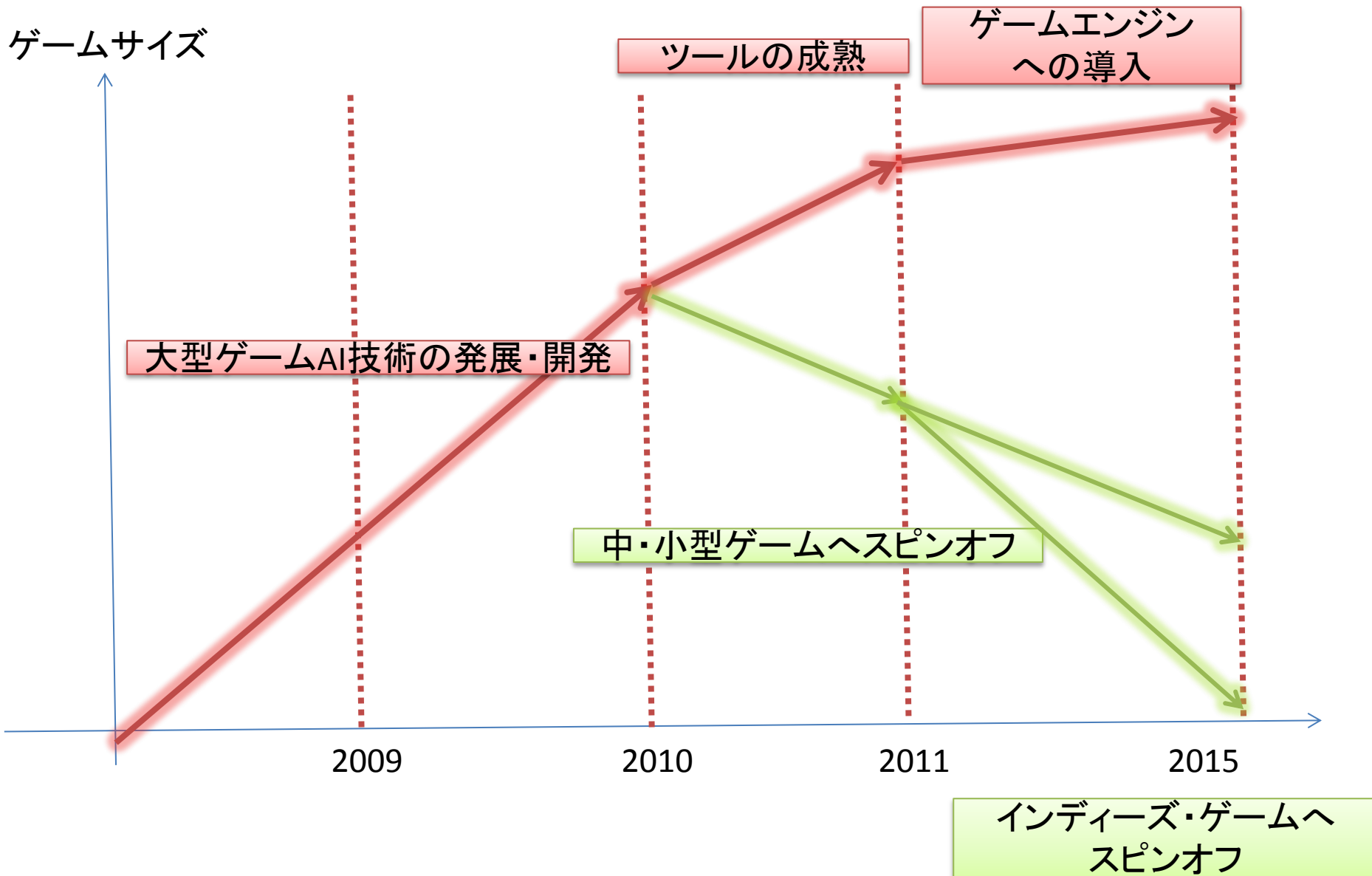
<http://www.digrajapan.org/modules/mydownloads/images/study/20090411.pdf>

2008年： 大型ゲームAI & プロシージャル

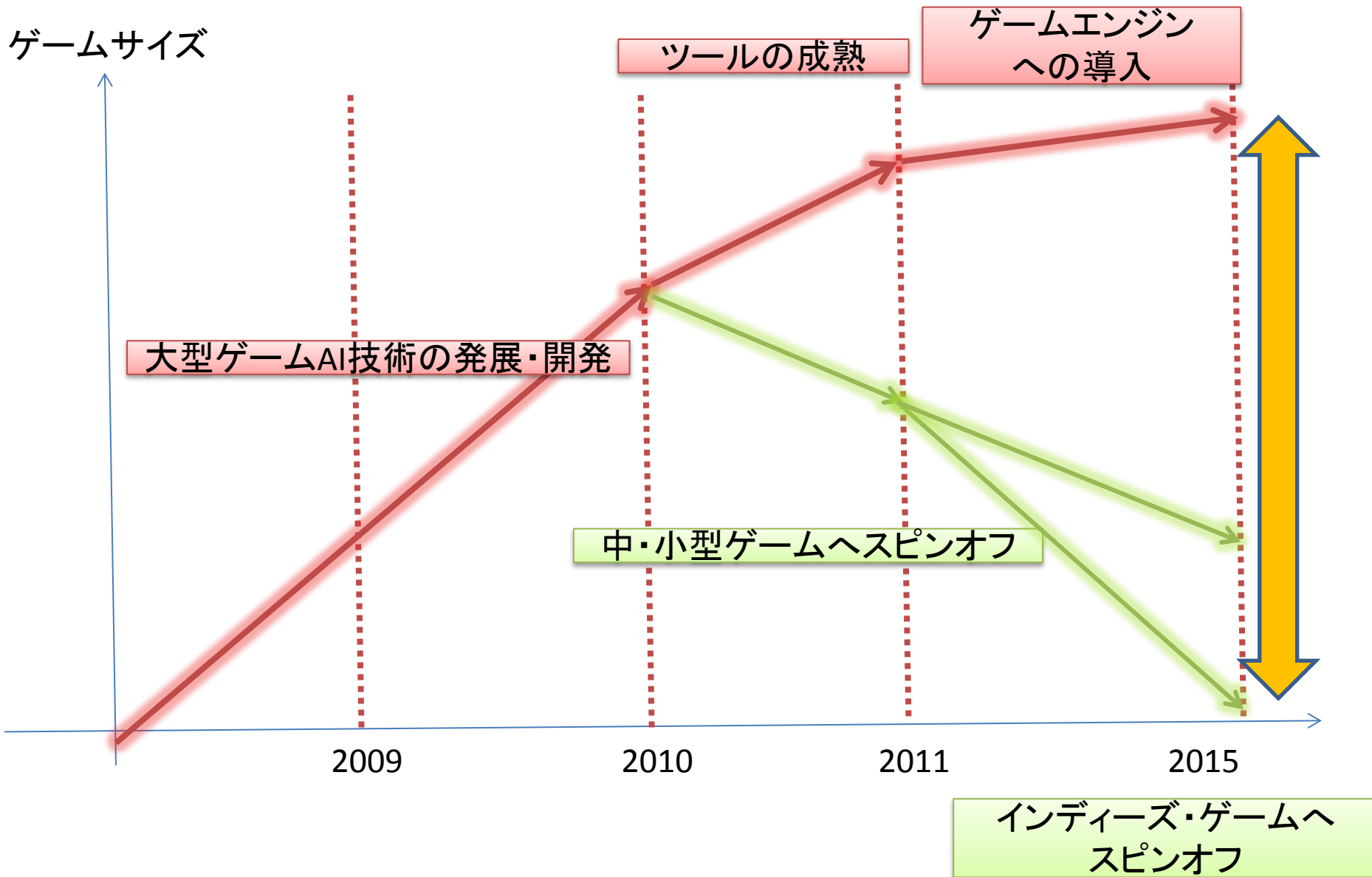
Spore, Halo3, Assassin's Creed...

http://igda.sakura.ne.jp/sblo_files/ai-igdajp/AI/IDGA_GDC08_Miyake_public.pdf

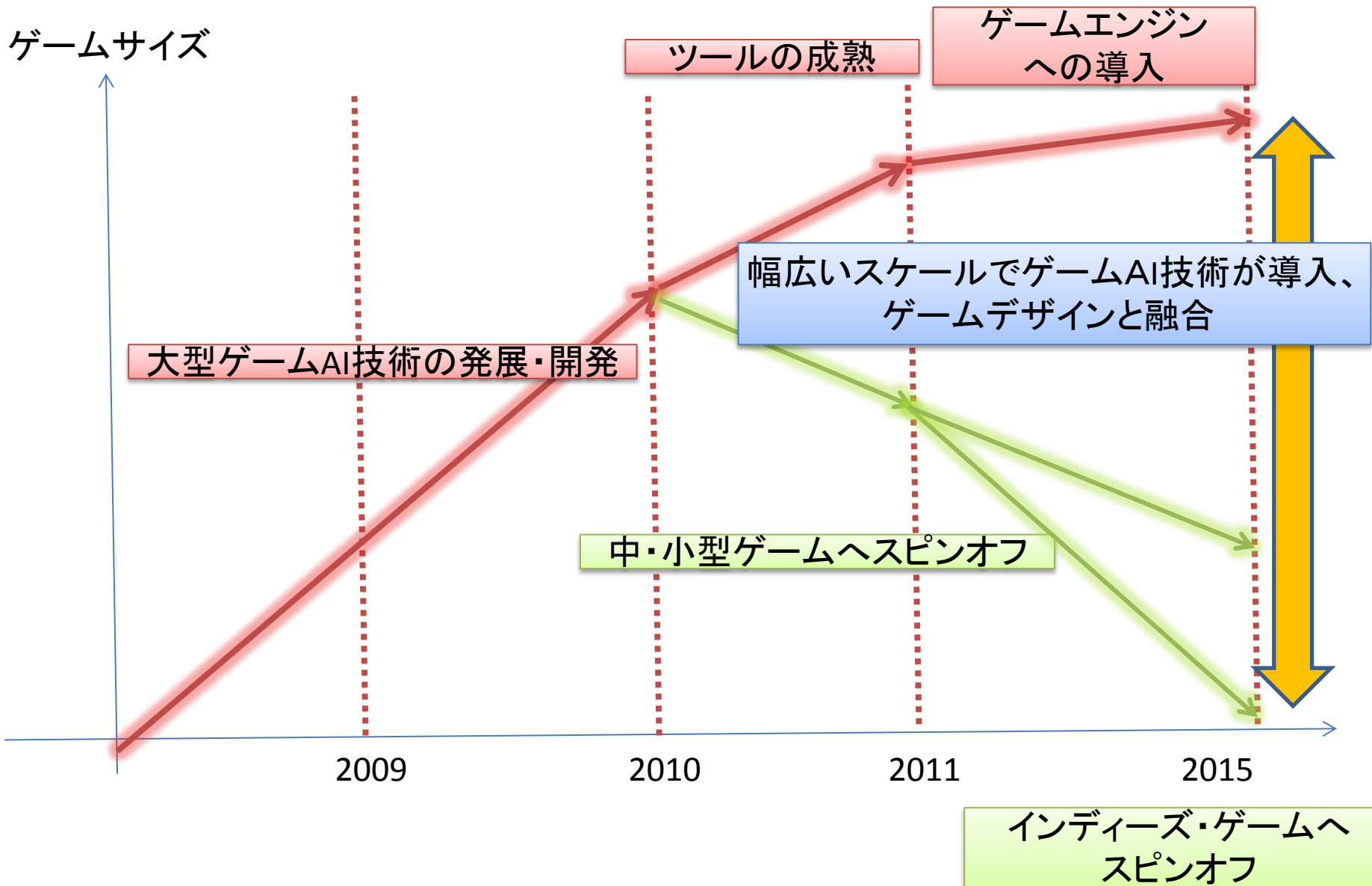
GDC2015までのゲームAI全体の傾向



GDC2015までのゲームAI全体の傾向



GDC2015までのゲームAI全体の傾向



AI/プロシージャル技術は ゲームエンジンや基礎システムに

2015年の
GDCのAIの
特徴

- 学習や自然言語処理の応用を期待 → 殆どない
- これまで技術がゲームエンジンや基礎システムにきちんと入る

現代のゲームエンジン群

- EA Frostbite
- Ubi Dunia Engine/ ArtFramework
- Naughty Dog Engine
- RED Engine (CDProjekt)
- Unity Framework
- Unreal Engine 4

応用事例

- Tomb Raider (GOAP)
- Watch Dogs
- Assassin's Creed Unity
- Sunset Overdrive
- FARCRY4
- Eldritch

GDC 2015 AI + エンジン事例

- アサシンズクリード・ユニティのオンラインにおけるAI
- Naughty Dog Engine
- FARCRY4 のプロシージャル技術
- DUNIA ENGINE のパイプライン
- ENDRICH のプロシージャル技術
- JPS高速化アルゴリズム
- Guild War の Utility システム
- Watch Dogs のキャラクター制御
- Unreal Engine 4 KITE におけるAI
- Tomb Raider の Goal-Oriented Action Planning (GOAP)

GDC 2015 AI + エンジン事例

- アサシンズクリード・ユニティのオンラインにおけるAI
- Naughty Dog Engine
- FARCRY4 のプロシージャル技術
- DUNIA ENGINE のパイプライン
- ENDRICH のプロシージャル技術
- JPS高速化アルゴリズム
- Guild War の Utility システム
- Watch Dogs のキャラクター制御
- Unreal Engine 4 KITE におけるAI
- Tomb Raider の Goal-Oriented Action Planning (GOAP)

本講演で取り上げるテーマ

プロシージャル的なレベル/AI の作り方

Unreal Engine 4 KITE Demo

A Boy and His Kite: An Animated Short (Epic, Unreal Engine 4)

<https://www.youtube.com/watch?v=JNgsbNvkNjE>



【技術解説動画】

GDC 2015: Creating the Open World Kite Real-Time Demo in Unreal Engine 4

<https://www.youtube.com/watch?v=clakekAHQx0>

Unreal Engine 4 KITE Demo

Landscape



UNREAL
ENGINE

Result: textured, lit material view

GDC 2015: Creating the Open World Kite Real-Time Demo in Unreal Engine 4
<https://www.youtube.com/watch?v=clakekAHQx0>

植物は手動か自動配置。 マテリアルを観ながら自動配置も。



植物の生成・成長のシミュレーションで レベルを生成

Simulation

- Natural looking
 - Clustering
 - Self Thinning
 - Growth Curves
- Fast iteration
- Artist intervention



UNREAL
ENGINE

GDC 2015: Creating the Open World Kite Real-Time Demo in Unreal Engine 4
<https://www.youtube.com/watch?v=clakekAHQx0>

密度などを指定しつつ 植物が群生する

Simulation

Create a 'ecosystem'
containing species

Add a few instances
of each species,
based on density
settings



UNREAL
ENGINE

GDC 2015: Creating the Open World Kite Real-Time Demo in Unreal Engine 4
<https://www.youtube.com/watch?v=clakekAHQx0>

植物は年齢を重ねて、実を拡げる

Simulation

Each step we age the plants and spread seeds



UNREAL
ENGINE

GDC 2015: Creating the Open World Kite Real-Time Demo in Unreal Engine 4
<https://www.youtube.com/watch?v=clakekAHQx0>

植物の競合ルールを敷く。

Simulation

Use species rules to decide overlaps

- Relative Size
- Shade
- Altitude
- Slope



UNREAL
ENGINE

GDC 2015: Creating the Open World Kite Real-Time Demo in Unreal Engine 4
<https://www.youtube.com/watch?v=clakekAHQx0>

こんな感じになります！

Simulation

Final result!



UNREAL
ENGINE

GDC 2015: Creating the Open World Kite Real-Time Demo in Unreal Engine 4
<https://www.youtube.com/watch?v=clakekAHQx0>

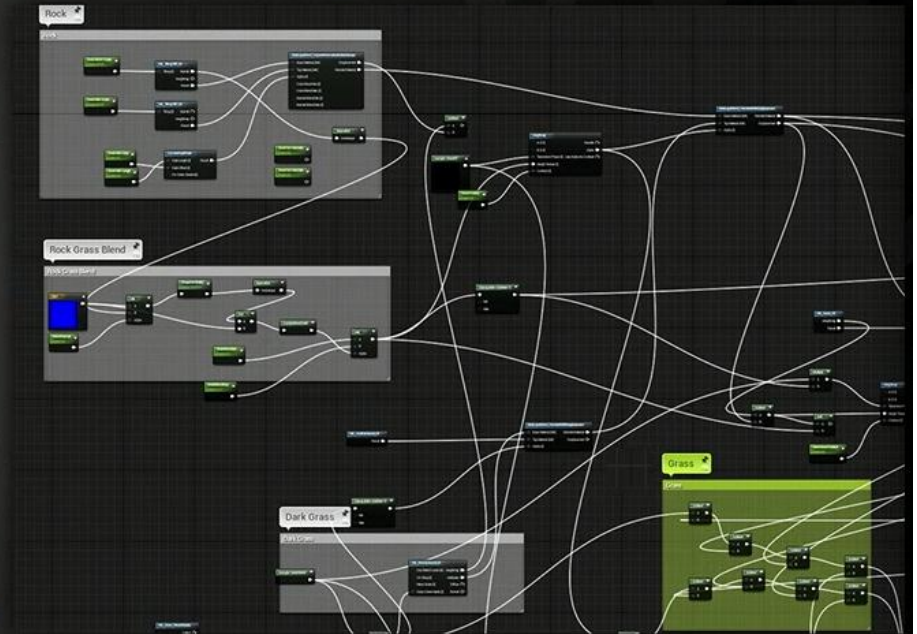
草配置グラフ

Grass Placement

Using weight maps did not work well

Use the material graph to define grass density!

Rasterize terrain on CPU to find instance locations



UNREAL
ENGINE

GDC 2015: Creating the Open World Kite Real-Time Demo in Unreal Engine 4
<https://www.youtube.com/watch?v=clakekAHQx0>

ダイナミックな動物の活動もテスト

Fauna

- Test how our dynamic systems scale in large worlds



UNREAL
ENGINE

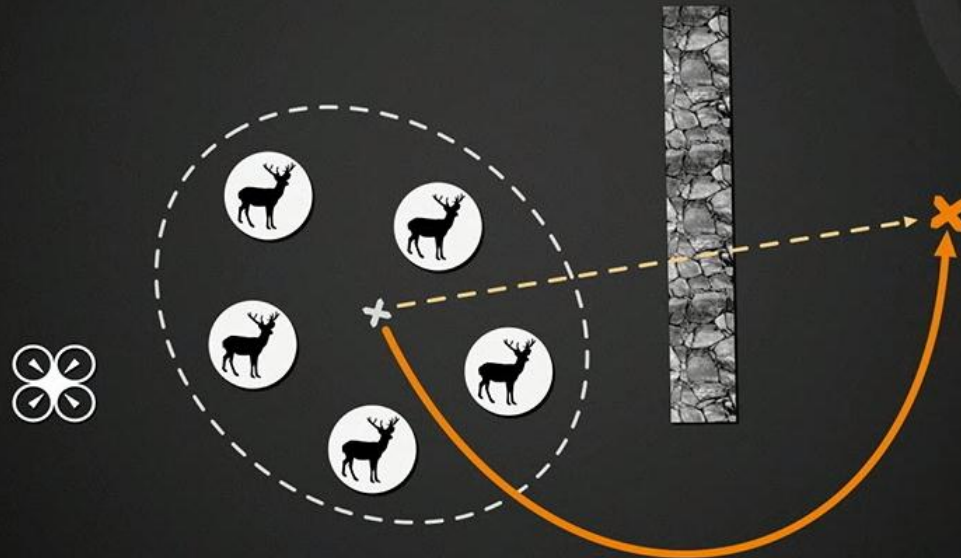
Movement, AI, Animation etc.

GDC 2015: Creating the Open World Kite Real-Time Demo in Unreal Engine 4
<https://www.youtube.com/watch?v=clakekAHQx0>

カメラドローンへ向かって走る

AI

- Find path from centre of herd away from drone



UNREAL
ENGINE

GDC 2015: Creating the Open World Kite Real-Time Demo in Unreal Engine 4
<https://www.youtube.com/watch?v=clakekAHQx0>

群制御でAIは木を避ける。

AI

- Crowd simulation used to avoid trees and other deer
- Quadruped challenges
 - Turning on the spot
 - Takes time to accelerate

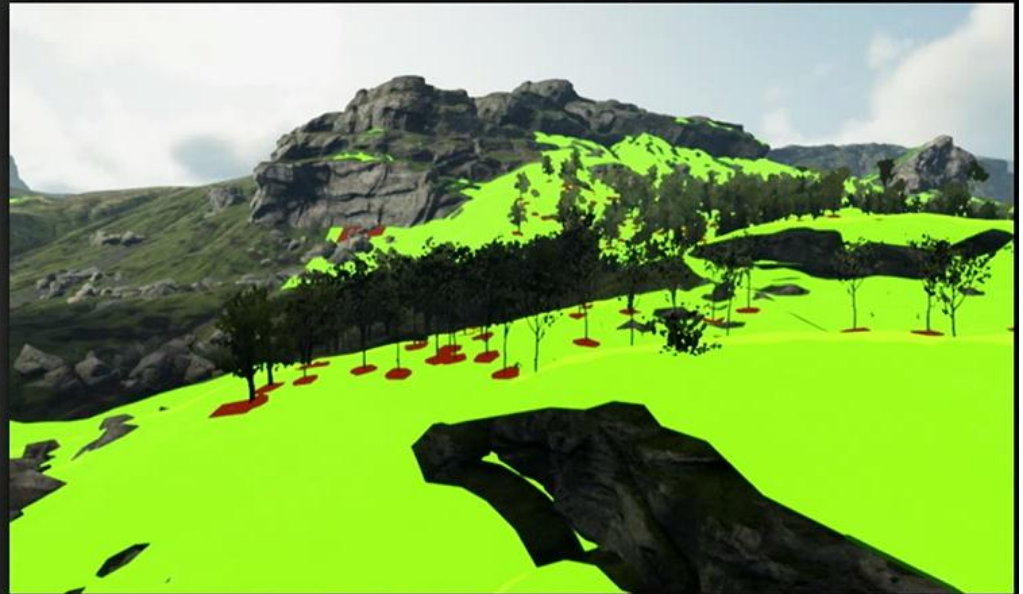


UNREAL
ENGINE

ナビゲーション・メッシュを生成

Navigation Mesh

- Need navigation data
 - Avoid trees and cliffs!
 - New efficient movement



UNREAL
ENGINE

GDC 2015: Creating the Open World Kite Real-Time Demo in Unreal Engine 4
<https://www.youtube.com/watch?v=clakekAHQx0>

ナビゲーション・メッシュは動的に生成

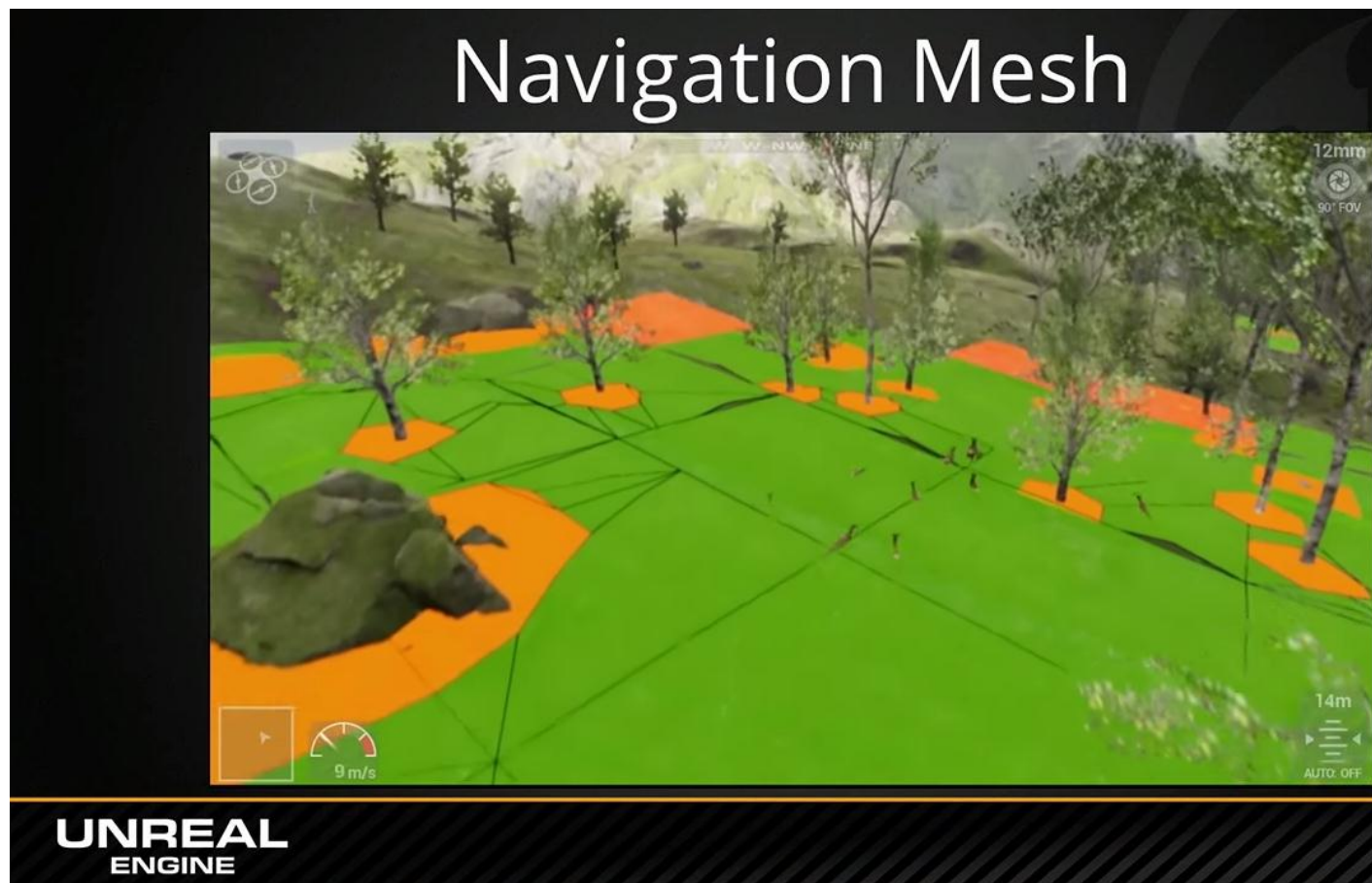
Navigation Mesh

- World too large to pre-calculate
 - Would use >5GB of memory!
- Generate tiles on demand
 - Spread work across multiple cores

UNREAL
ENGINE

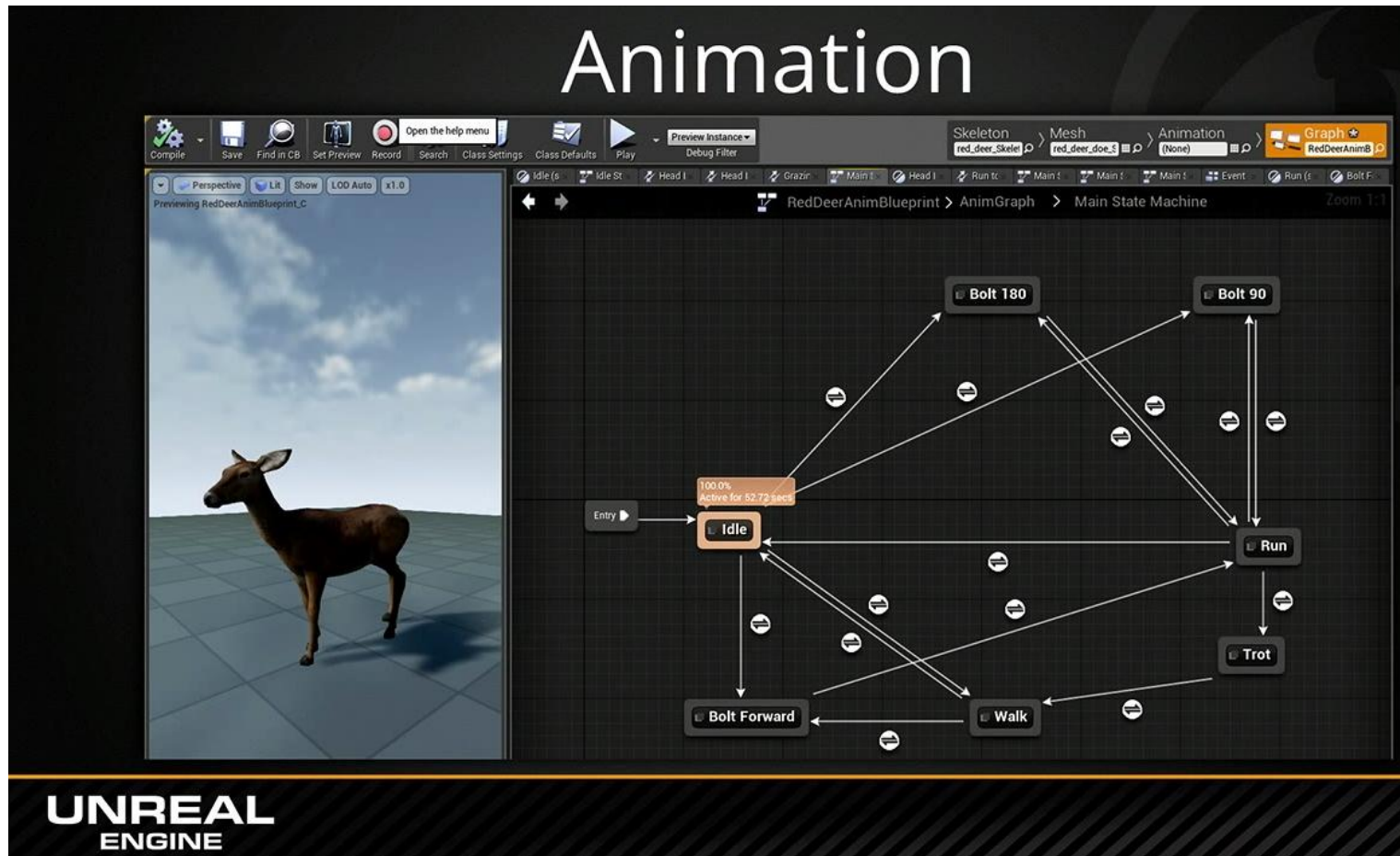
GDC 2015: Creating the Open World Kite Real-Time Demo in Unreal Engine 4
<https://www.youtube.com/watch?v=clakekAHQx0>

ナビゲーション・メッシュ自動生成



GDC 2015: Creating the Open World Kite Real-Time Demo in Unreal Engine 4
<https://www.youtube.com/watch?v=clakekAHQx0>

アニメーション・グラフ



GDC 2015: Creating the Open World Kite Real-Time Demo in Unreal Engine 4
<https://www.youtube.com/watch?v=clakekAHQx0>

多数のNPCを出しても最適化されている

Animation



UNREAL
ENGINE

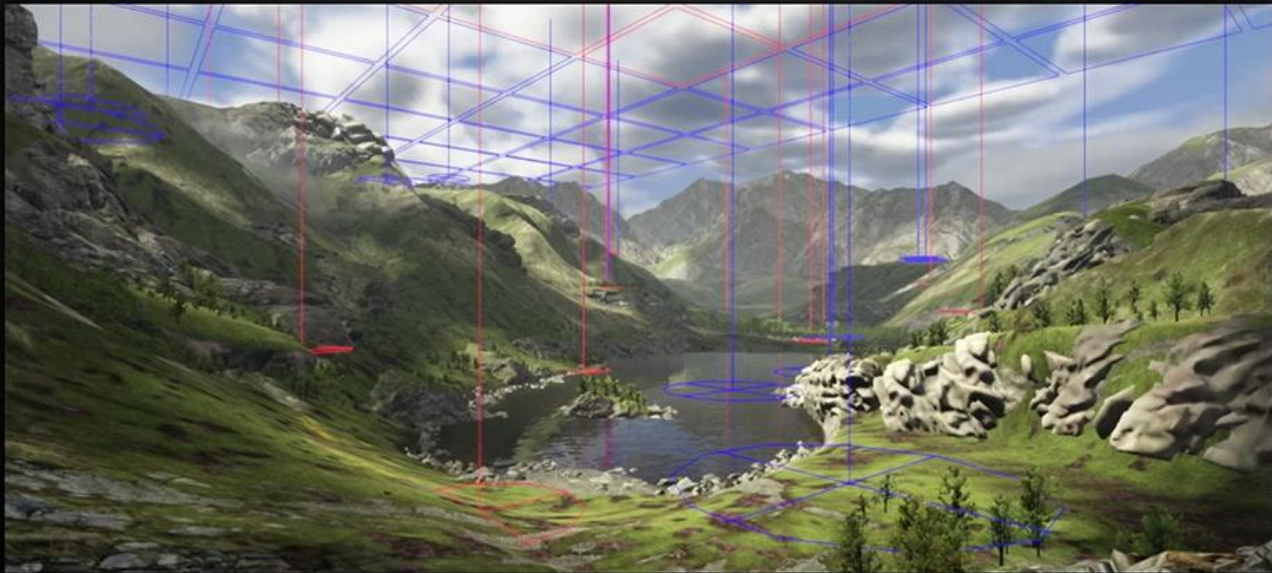
Optimizations: 300 deer

GDC 2015: Creating the Open World Kite Real-Time Demo in Unreal Engine 4
<https://www.youtube.com/watch?v=clakekAHQx0>

各セルごとに最適なスポーニングポイントを ルールで決定

Spawning

- Divide world into cells
- Rules determine best locations in each cell



UNREAL
ENGINE

GDC 2015: Creating the Open World Kite Real-Time Demo in Unreal Engine 4
<https://www.youtube.com/watch?v=clakekAHQx0>

Procedural and Automation Techniques for Design and Production of Sunset Overdrive

David Santiago | Technical Artist, Insomniac Games

Location: Room 305, South Hall

Date: Thursday, March 5

Time: 5:30pm - 6:30pm

Format: Session

Track:  Programming,  Design

Pass Type: All Access Pass, Main Conference Pass - **Get your pass now!**

Vault Recording: Video

Audience Level: All

 Like  Share

Much of the open world of Sunset Overdrive was created using procedural systems and automation. Using a few curves, an artist can lay out a city of streets, a network of railroad tracks, or the ocean from the shores to the horizon. In minutes, a new area of the game with final assets is playable in engine. The rapid turnaround allowed designers to test many possible street layouts and elevations, and view final (or close to final) geometry, before involving environment artists. Once turned over, environment artists could concentrate on the buildings, instead of hand-modeling custom ground. This session explains the ground and many other systems used to create Sunset City.

Takeaway

Attendees will learn various procedural and automation techniques that give the artists as much control as they want techniques such as curve, attribute and rule-driven asset placement or exclusion. The talk will also discuss the creative reasons we left building construction to the artists on this project, instead of automating it.

Intended Audience

Artists, designers, programmers and producers will benefit from the techniques and the creative and efficient benefits of our implementation.

HOUDINI を利用した自動生成レベルデザイン
(INSOMIAC, SUNSET OVERDRIVE)

<http://schedule.gdconf.com/session/procedural-and-automation-techniques-for-design-and-production-of-sunset-overdrive>

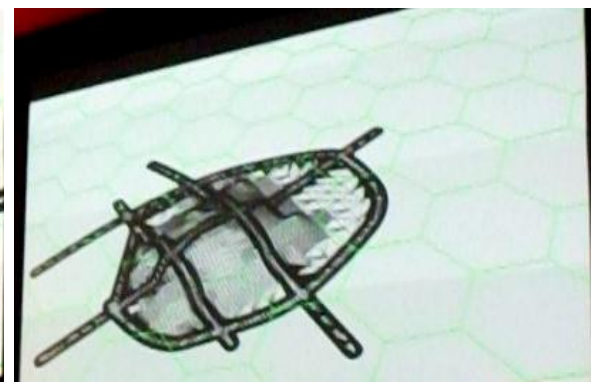
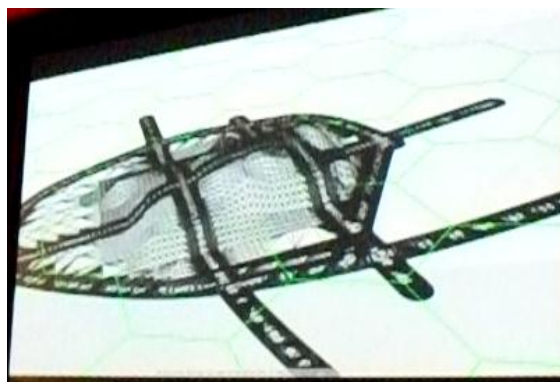
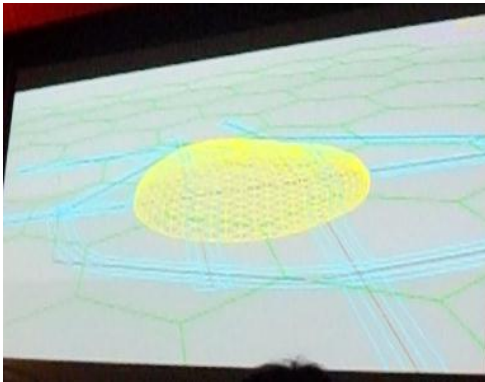
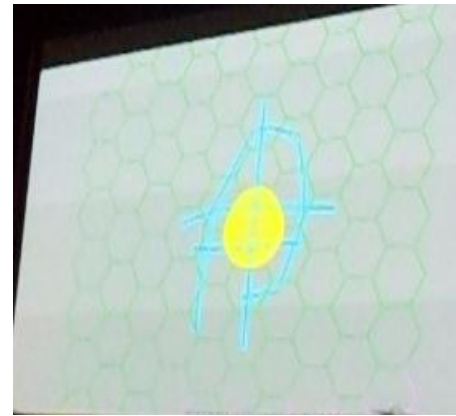
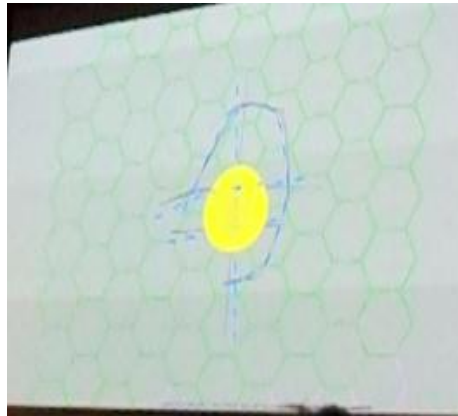
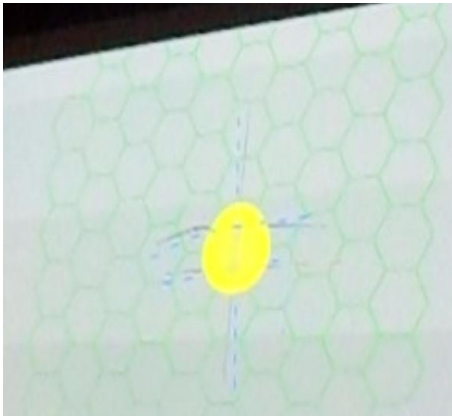
Houdini を利用した自動生成レベルデザイン (Insomiac, Sunset Overdrive)



Houdini の
プロシージャルな
特性を利用して、
レベル製作を
補佐する。



Houdini を利用した自動生成レベルデザイン (Insomiac, Sunset Overdrive)

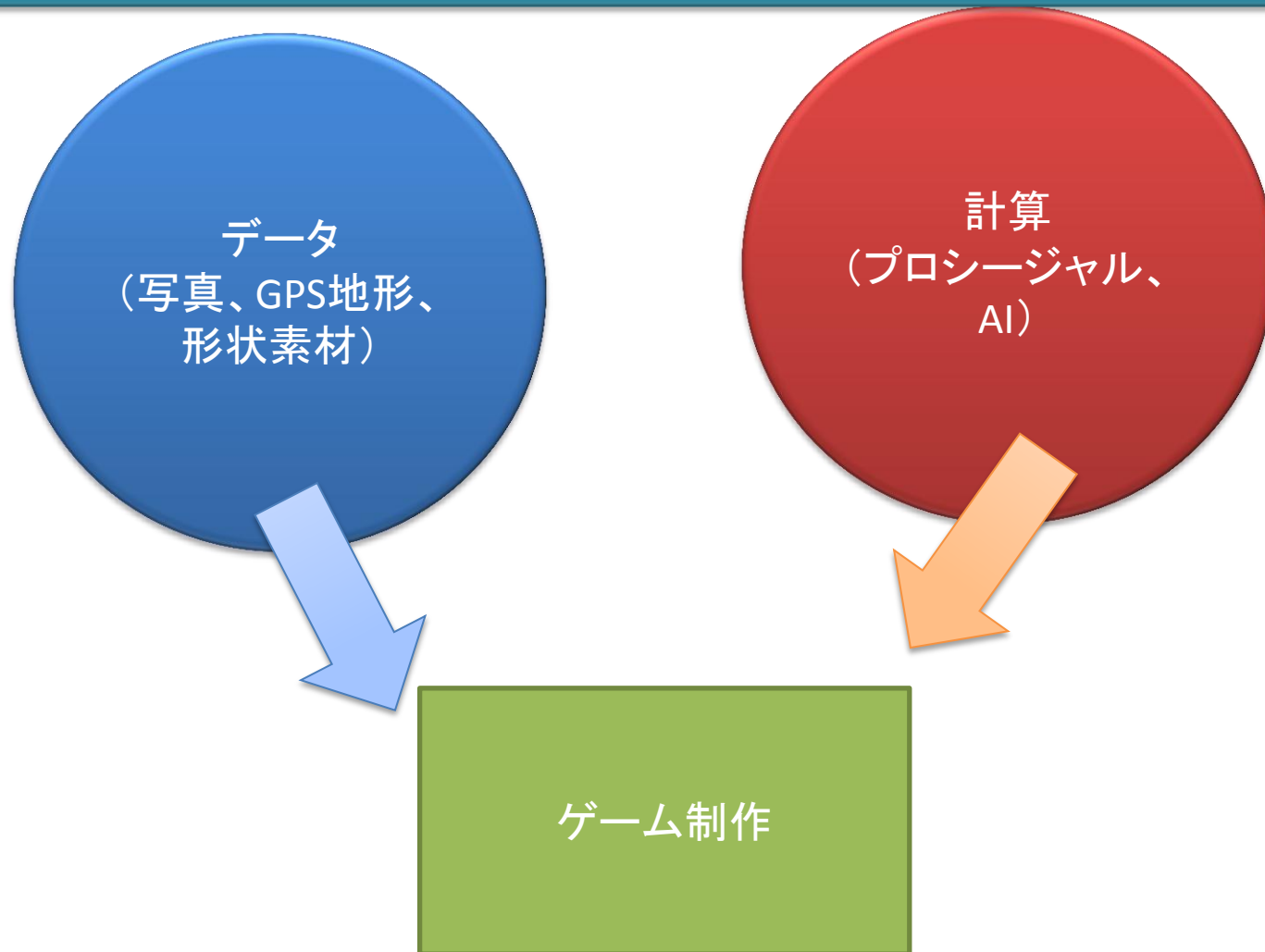


曲線で描いた線が道路になる。中央の楕円体が道路に滑らかにつながる。

プロシージャル・エンジンとしての Houdini Engine (記事)

- Sponsored: Go Procedural - A Better Way to Make Better Games (Gamasutra)
- http://www.gamasutra.com/view/news/233899/Sponsored_Go_Procedural_A_Better_Way_to_Make_Better_Games.php

プロシージャル技術を開発過程へ



In Your Hands: The Character (of Watch_Dogs)

David Therriault | Technical Team Lead Programmer - Character, Camera & Behaviours -

Watch Dogs, Ubisoft Montreal

Location: Room 306, South Hall

Date: Wednesday, March 4

Time: 2:00pm - 3:00pm

Format: Session

Track:  Programming

Pass Type: All Access Pass, Main Conference Pass - **Get your pass now!**

Vault Recording: Video

Audience Level: Intermediate

 Like  Share

Games showcase characters capable of tons of interactions in their environment and it is a known passion to try to make them look as perfect as possible! It's not easy to share features among different types of characters like players, combat & crowd ones to enhance their versatility and their surroundings connection in a systemic open world. This presentation will illustrate the shared gameplay animation architecture by starting from the structure of the multi-layered online-replicated state machine logic composed of dynamic animation trees using the [M(otion)-A(ction)-S(tance)-K(inesic)] stack developed from the ground up on Watch_Dogs. Passing through data segmentation helping motion input responsiveness, voxel generated jump links fluid usage, stylized ragdolls flying in the air and procedural additive experiments, it will expose tips & errors on how a third person character can be able to do free-running, to use his smartphone to hack a bridge and to shoot around simultaneously.

Takeaway

The audience will have a detailed explanation of the multi-layering state machine logic called [M(otion)-A(ction)-S(tance)-K(inesic)] including concrete examples and advantages for each of these 4 layers while taking into account the player reactive input needs and the richness of the animation data for great realized moments on characters in-game.

Intended Audience

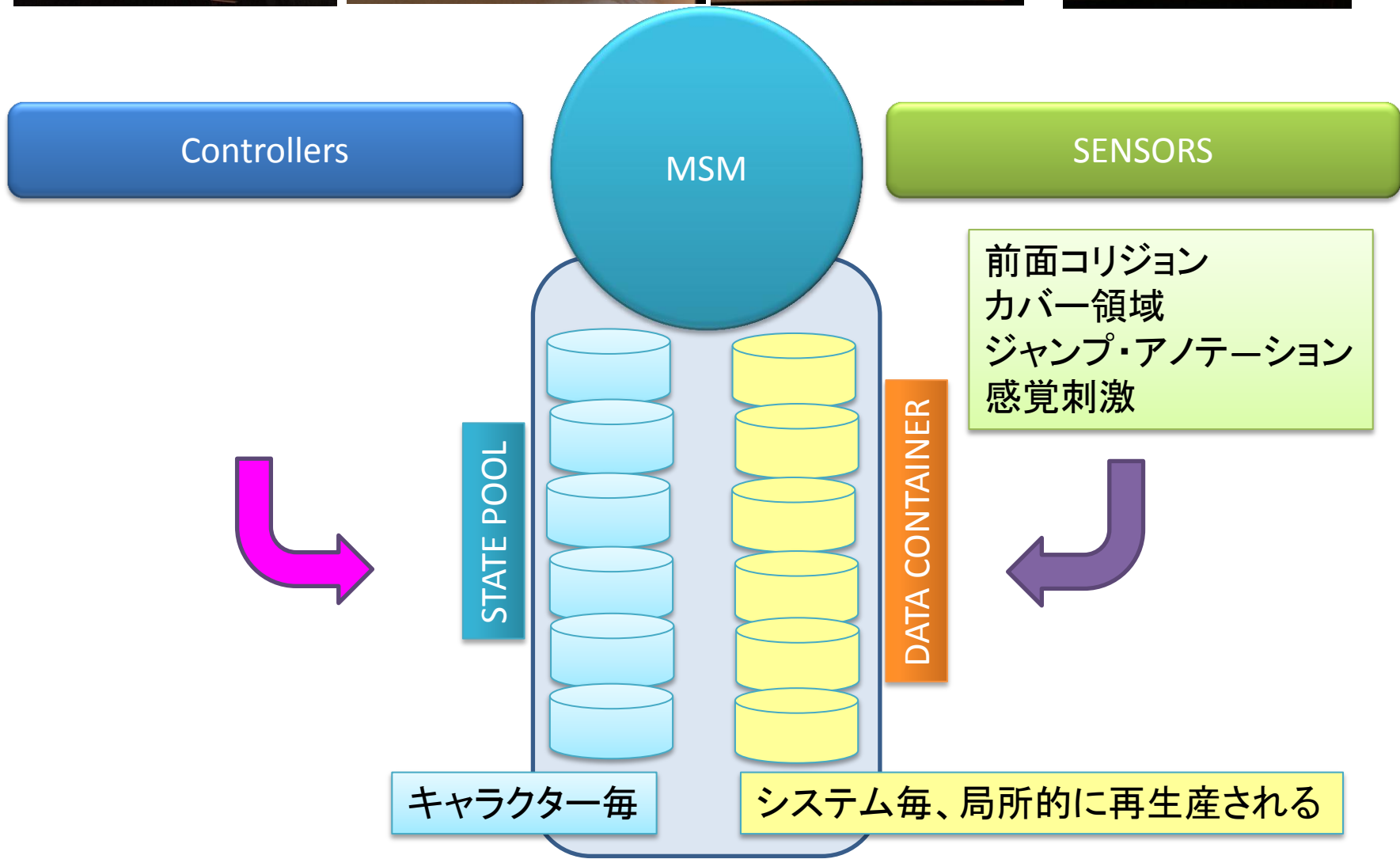
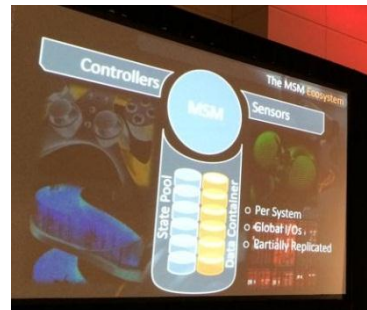
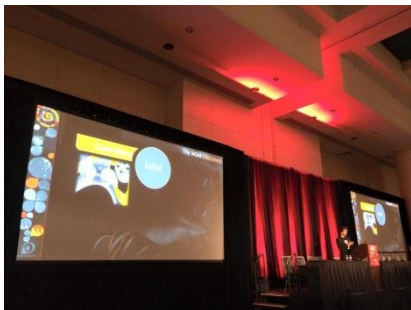
Mostly targetting experienced Gameplay and Ai Programmers, Technical Animators and Technical Designers with a passion for well architected, realized and comfortable animation systems. It can be also really interesting for any programmer with an interest for gameplay animation mechanics.

WATCH DOGS の
プレイヤー・キャラクター制御

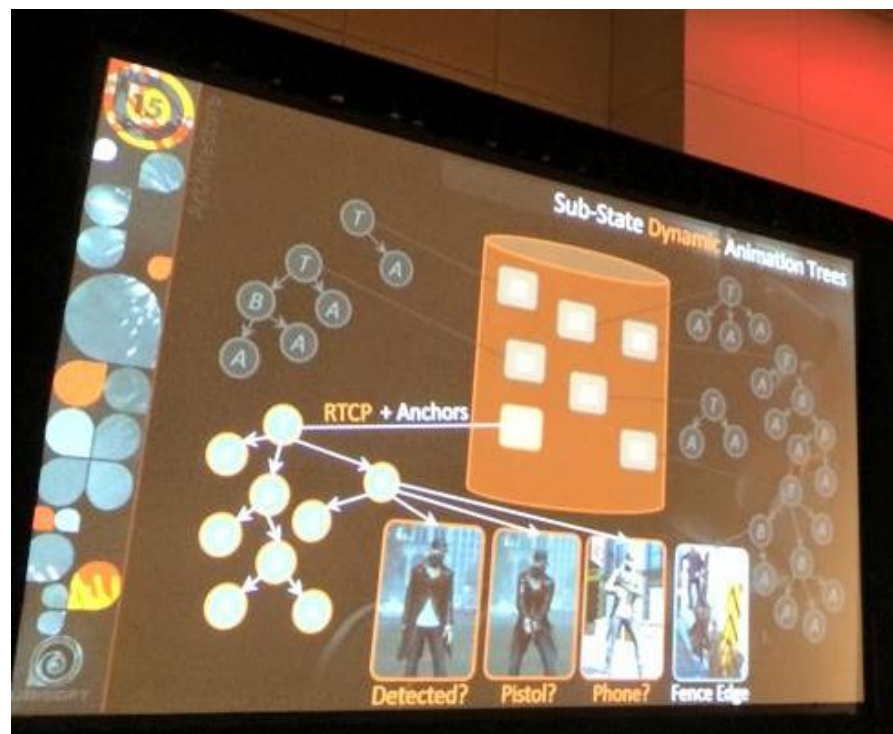
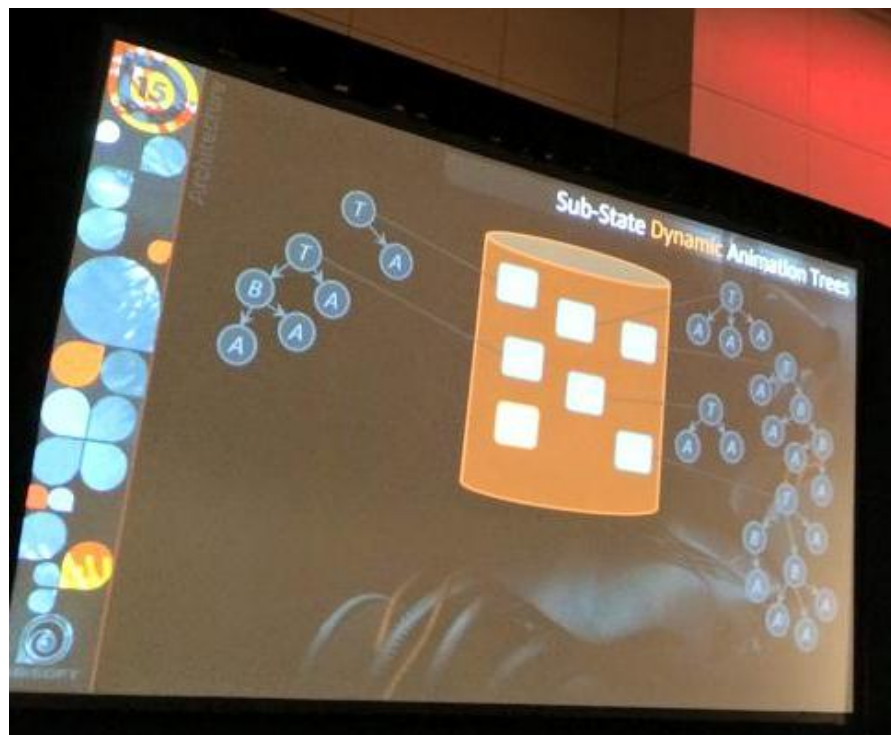
<http://schedule.gdconf.com/session/in-your-hands-the-character-of-watch-dogs>

Watch Dog のプレイヤー・キャラクターシステム MSM (State Machine)



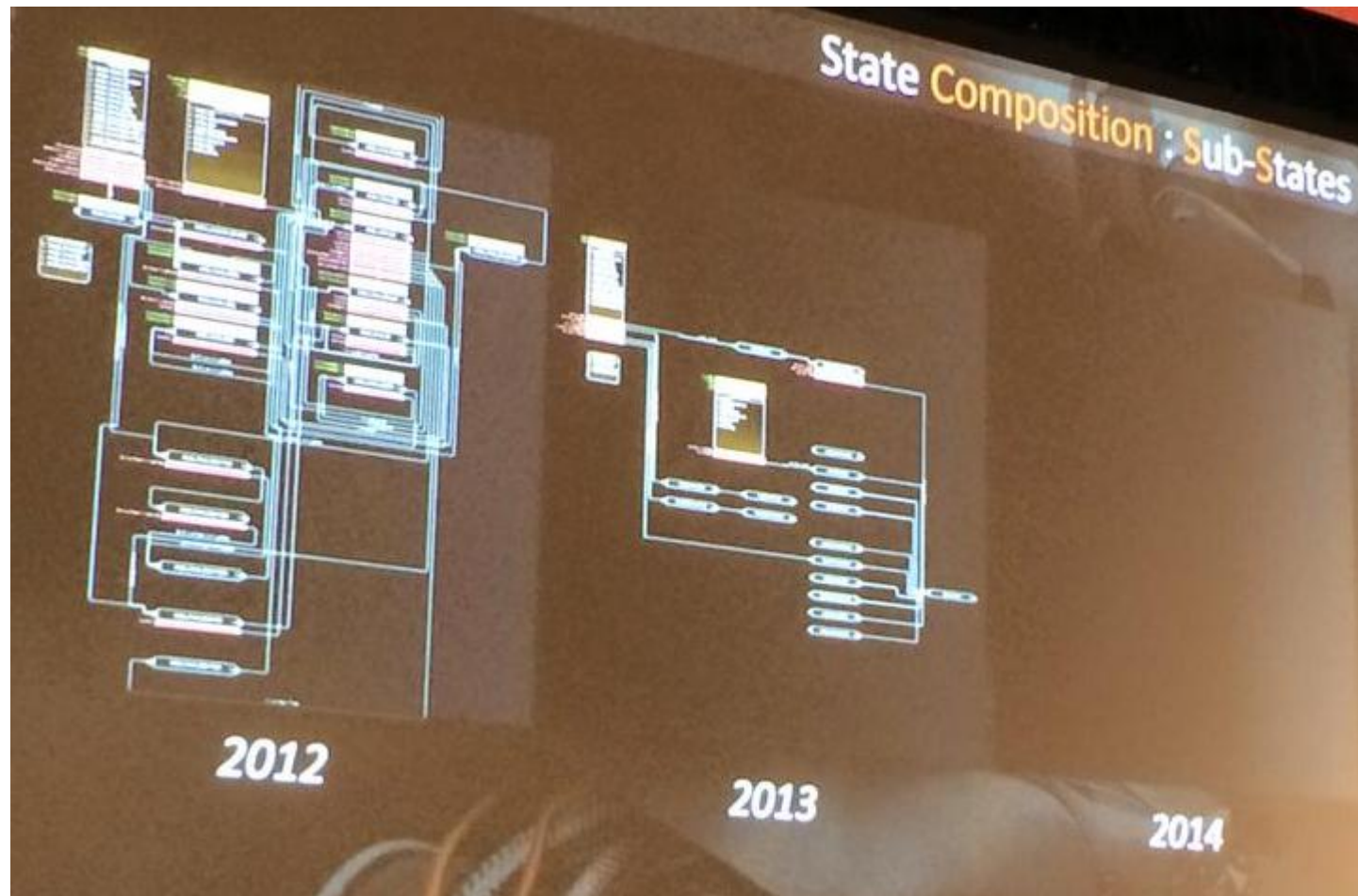


Sub-State Dynamic Animation

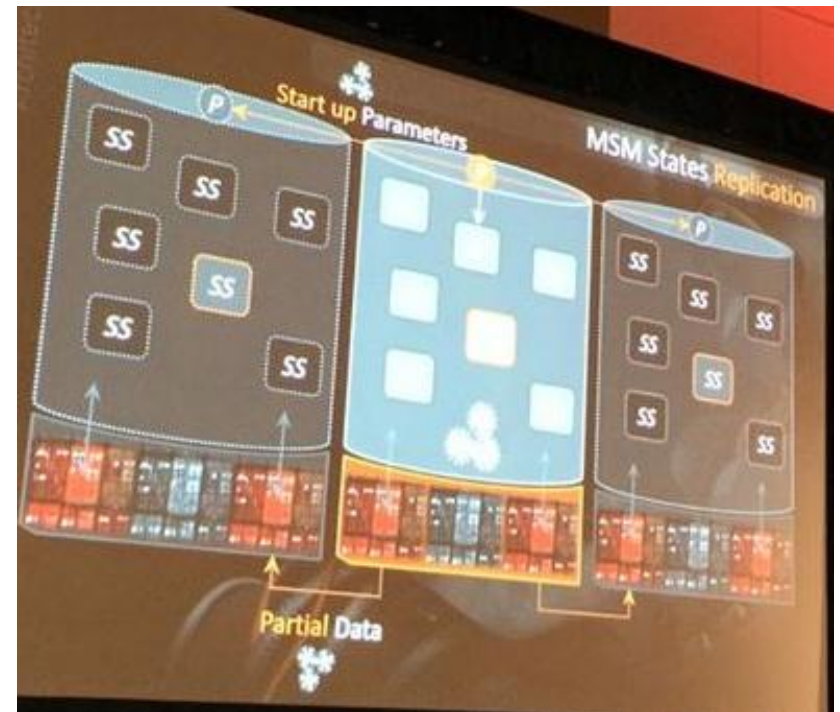


State の中にはアニメーション・ツリーが入っている。

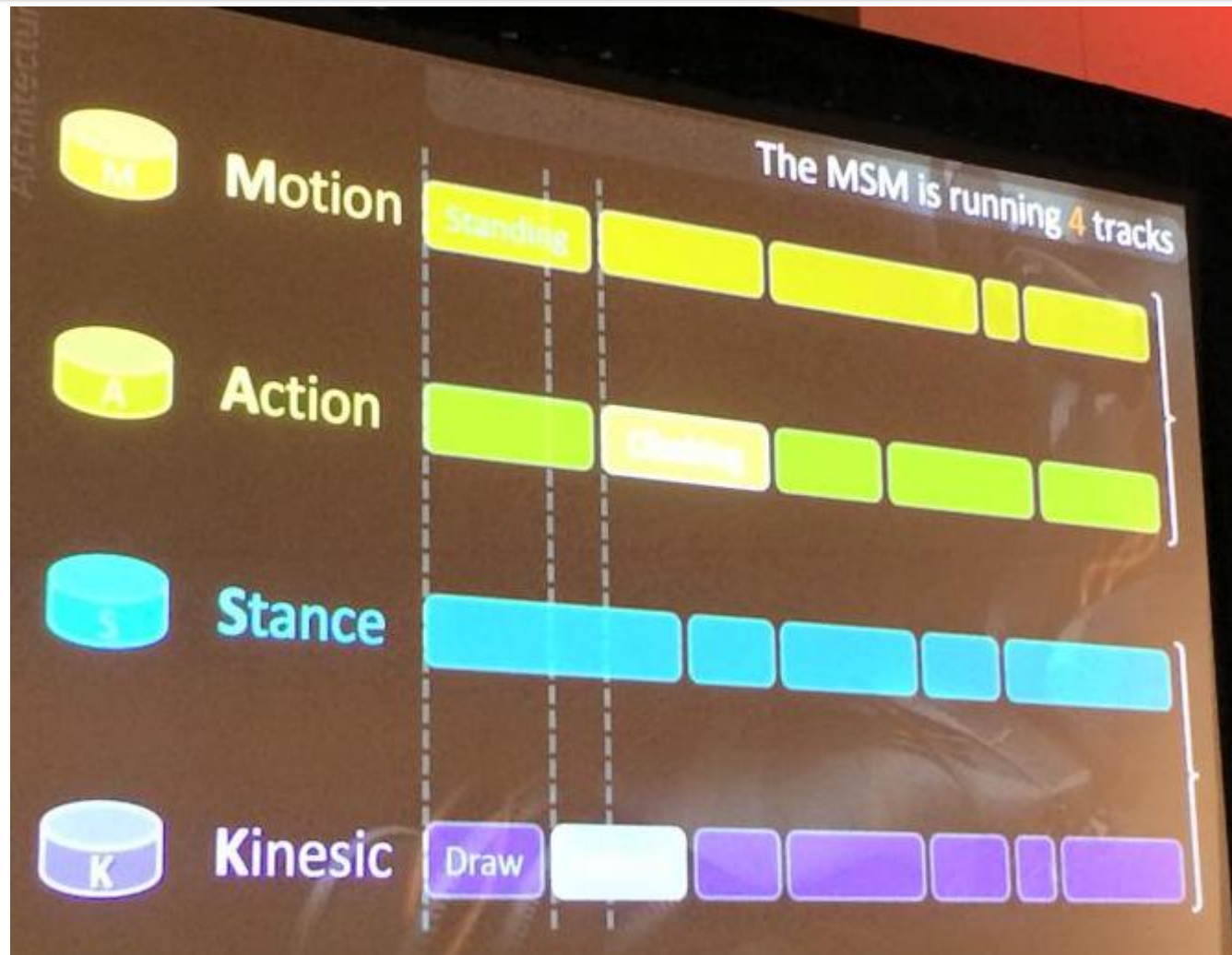
Sub-Stateをよりシンプルに



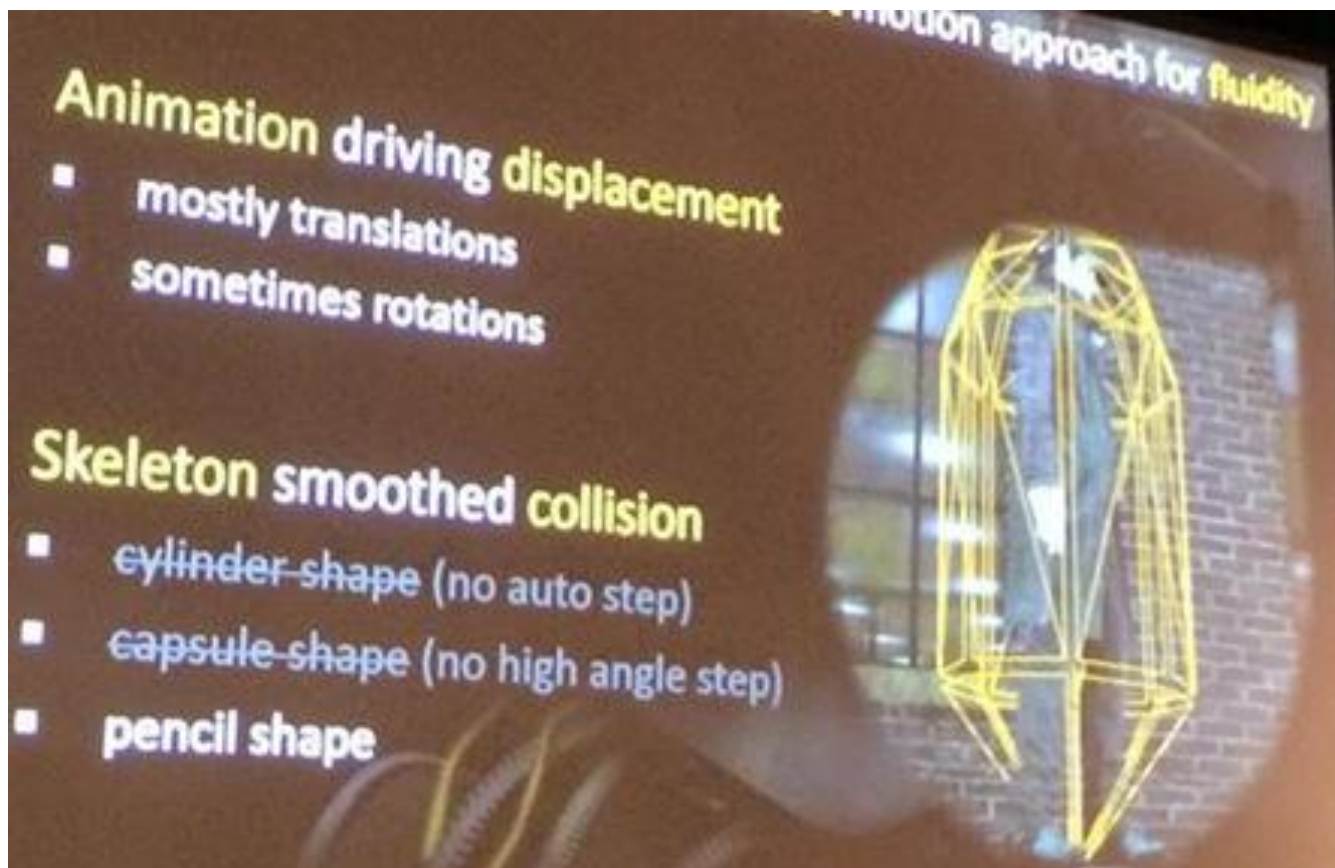
State を使い回す



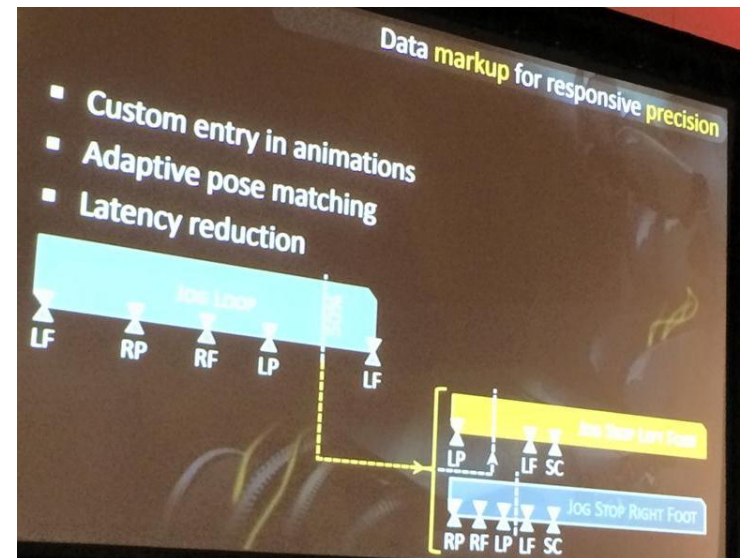
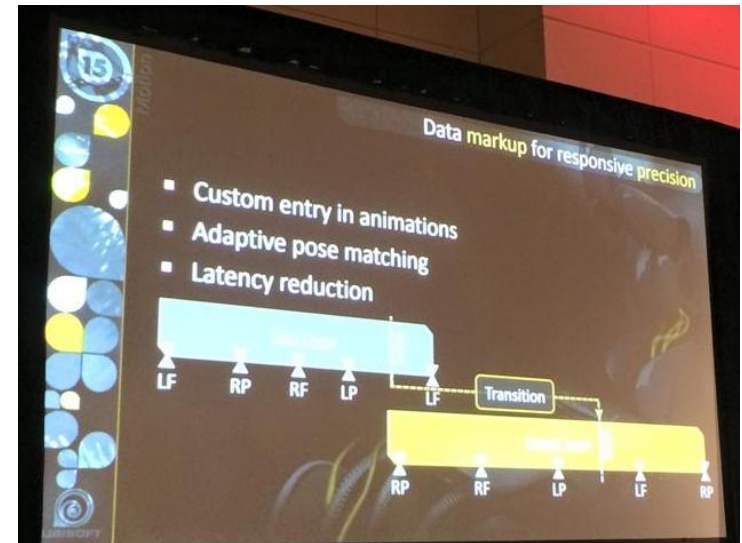
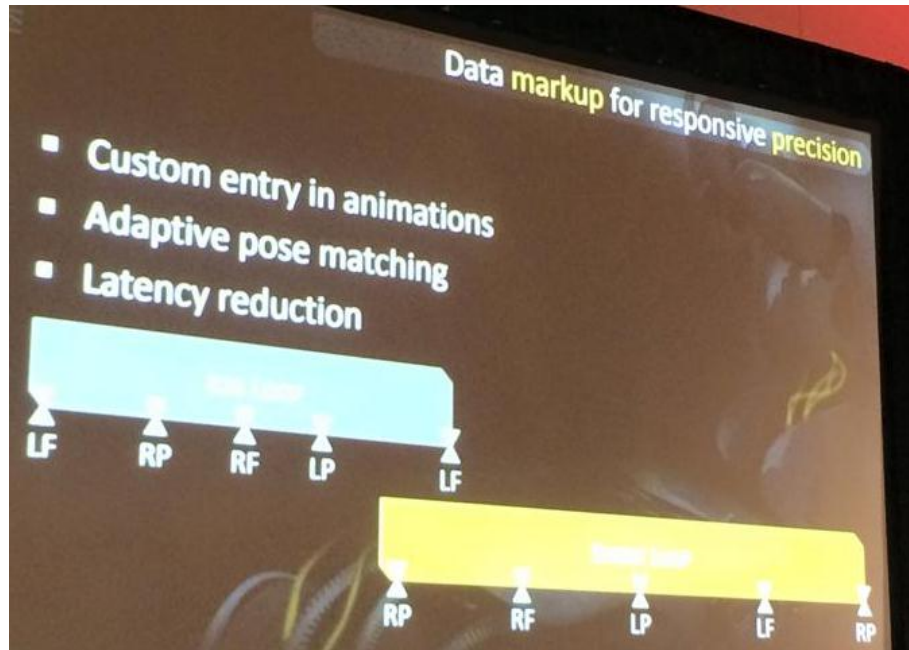
MSMは4つのレイヤーを制御する



ペンシル型キャラクター・コリジョン



アニメーションデータにマークアップ してマッチングを行う



Data segmentation for start precision

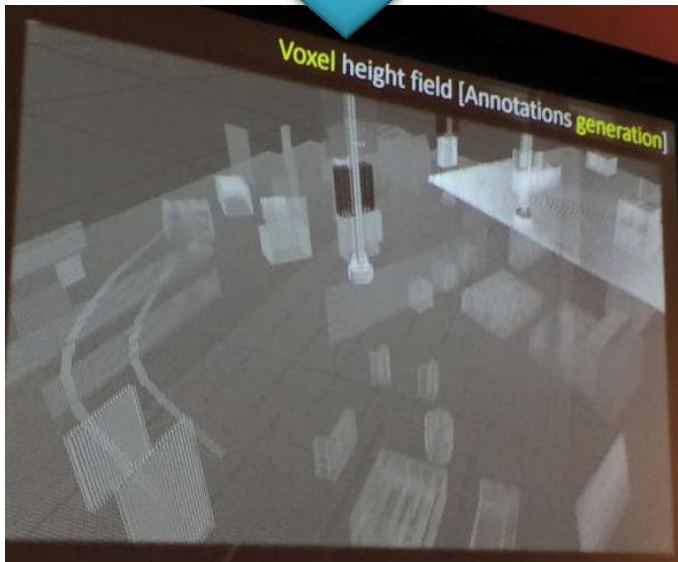
- Optical illusion : Head motion reactivity
- Read mind for direction & speed : Fixed delay
- Anticipation of ~6@8 frames (*playtest*)

START

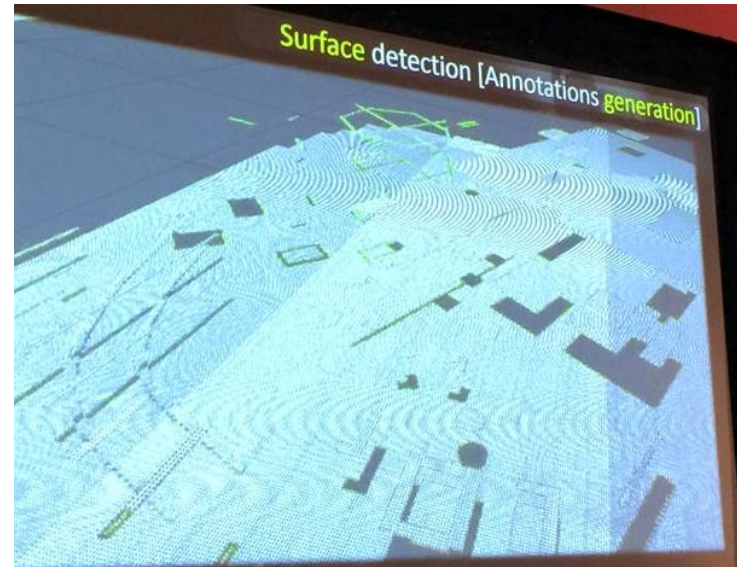
プレイヤー・キャラクターの運動のための地形解析



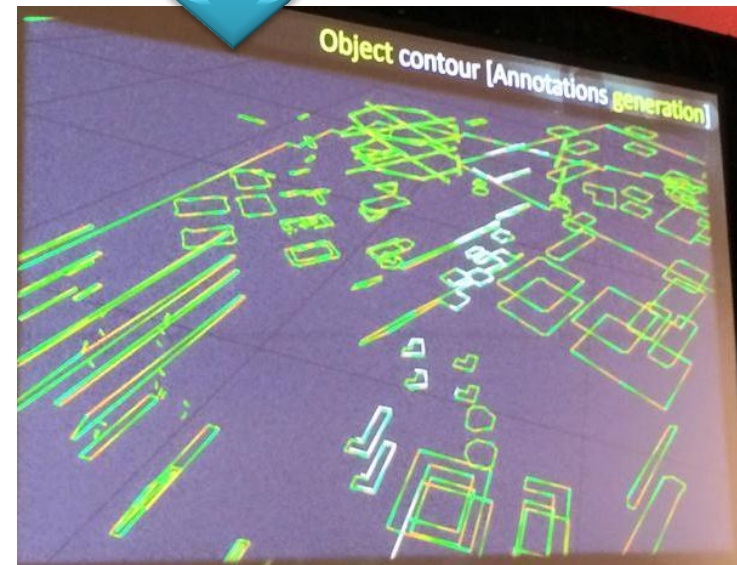
コリジョンモデル



ボクセル高さフィールド

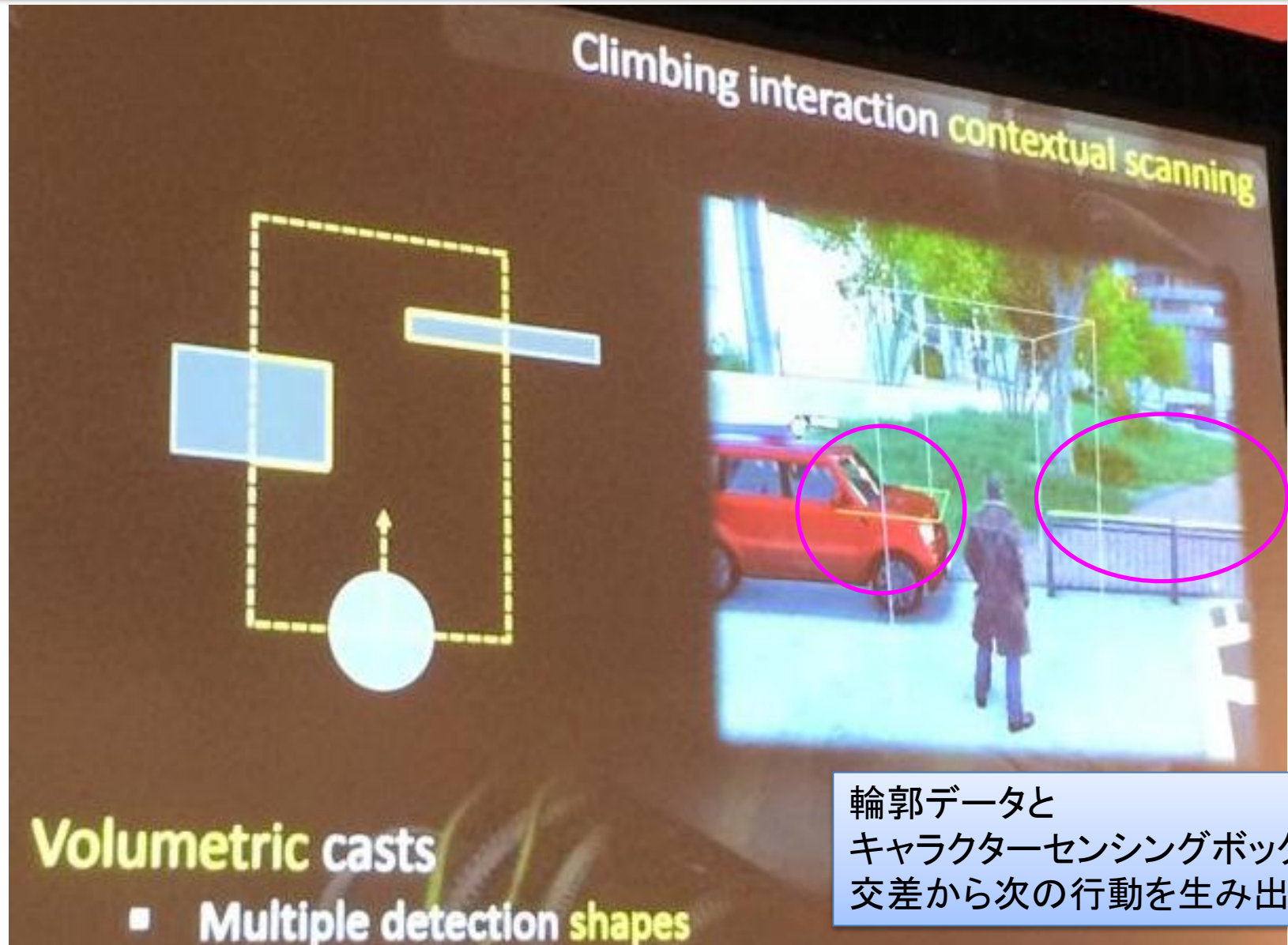


平面による断面



輪郭を取ったマップ

プレイヤー・キャラクターの運動のための地形解析

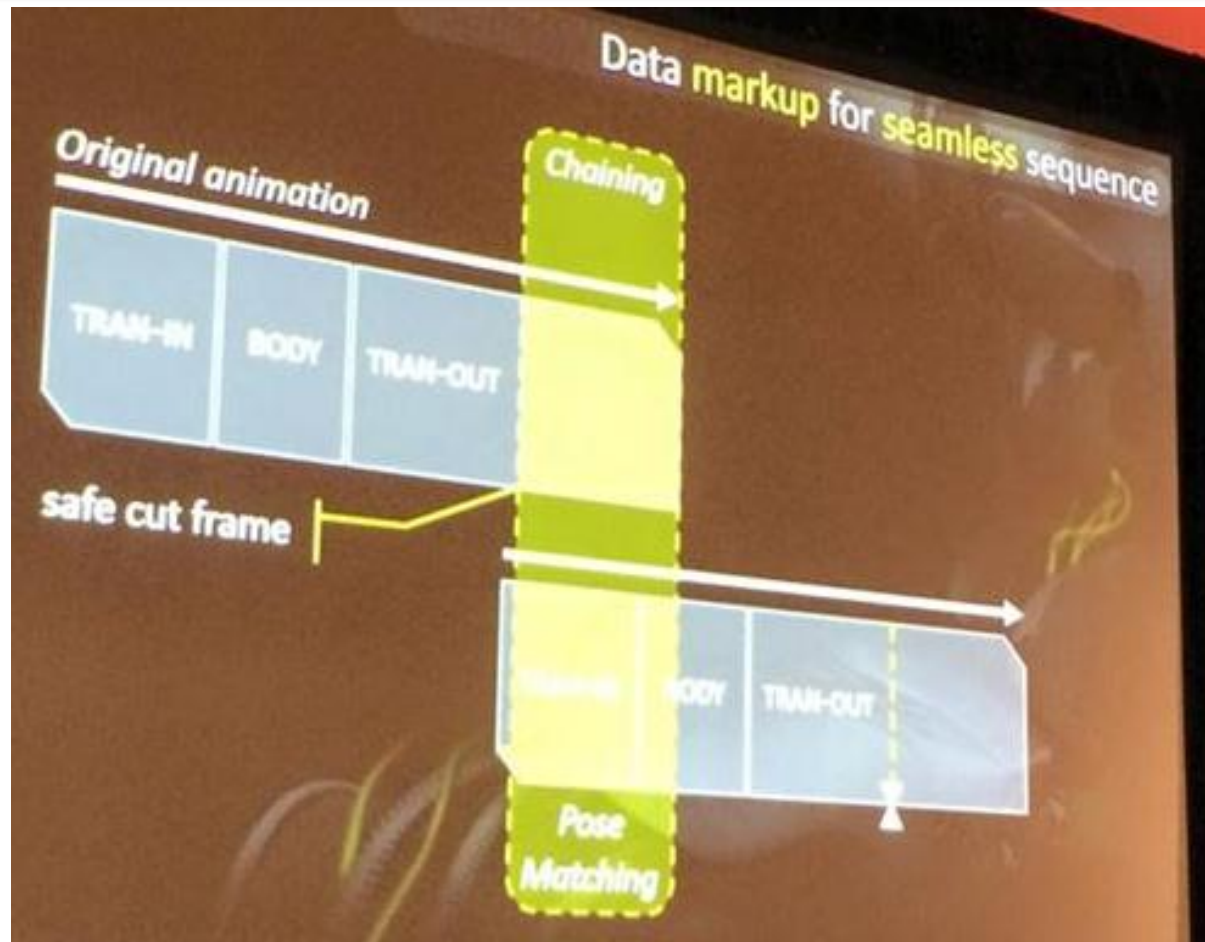


プレイヤー・キャラクターの運動のための 地形解析→ 実際の運動



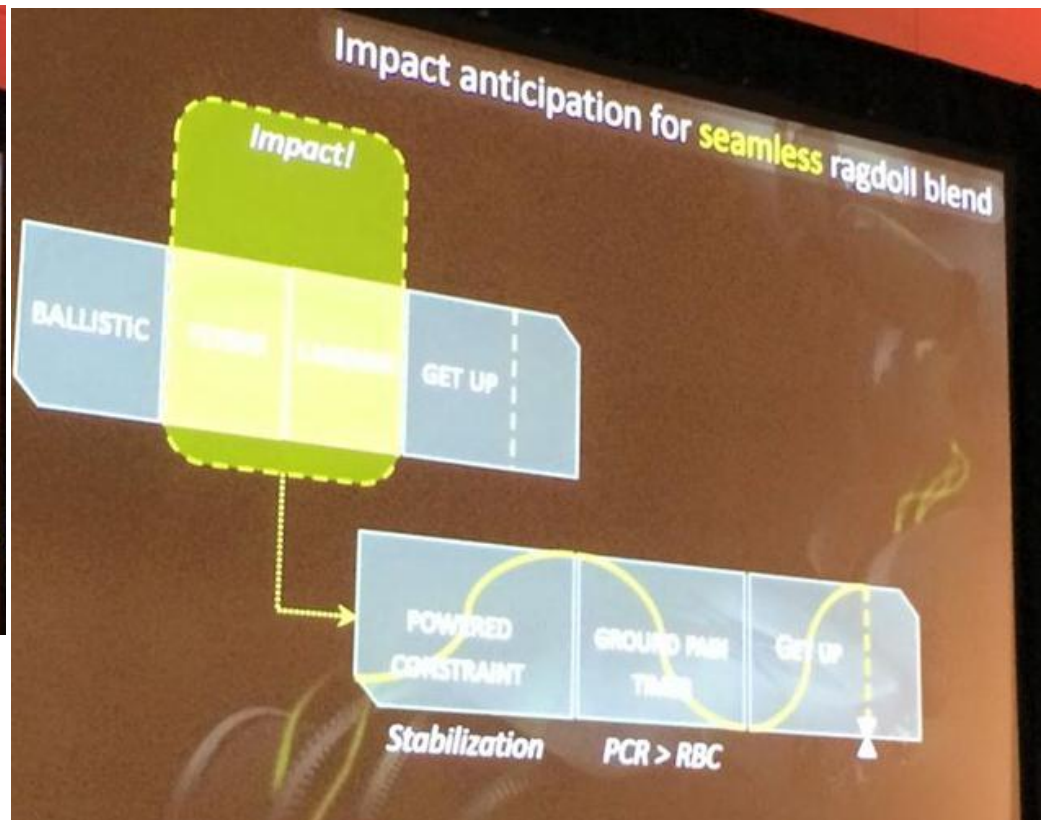
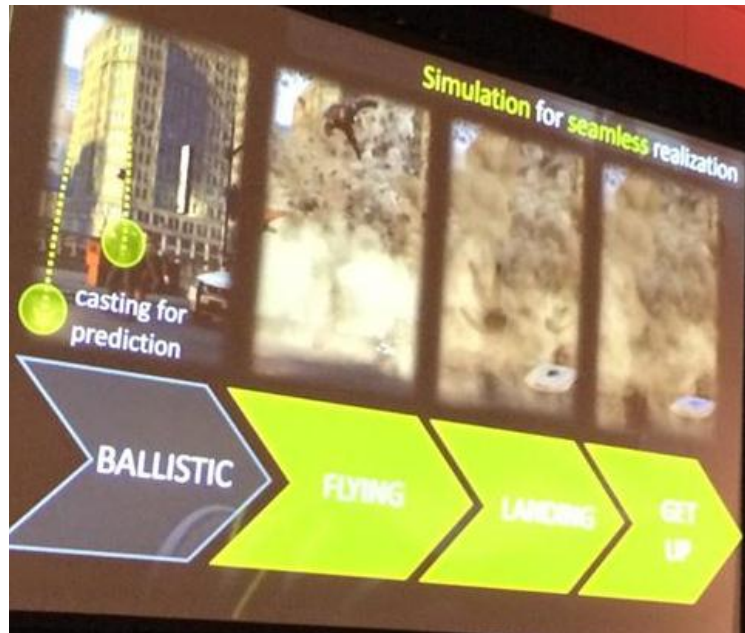
キャラクターの状態を遷移
(TRAN IN → 行動(BODY) → TRAN OUT)

データをマークアップして チェイニング(連鎖)によってモーションをつなぐ



二つのアニメーションをポーズマッチングでつなぐ
= チェイニング(連鎖)

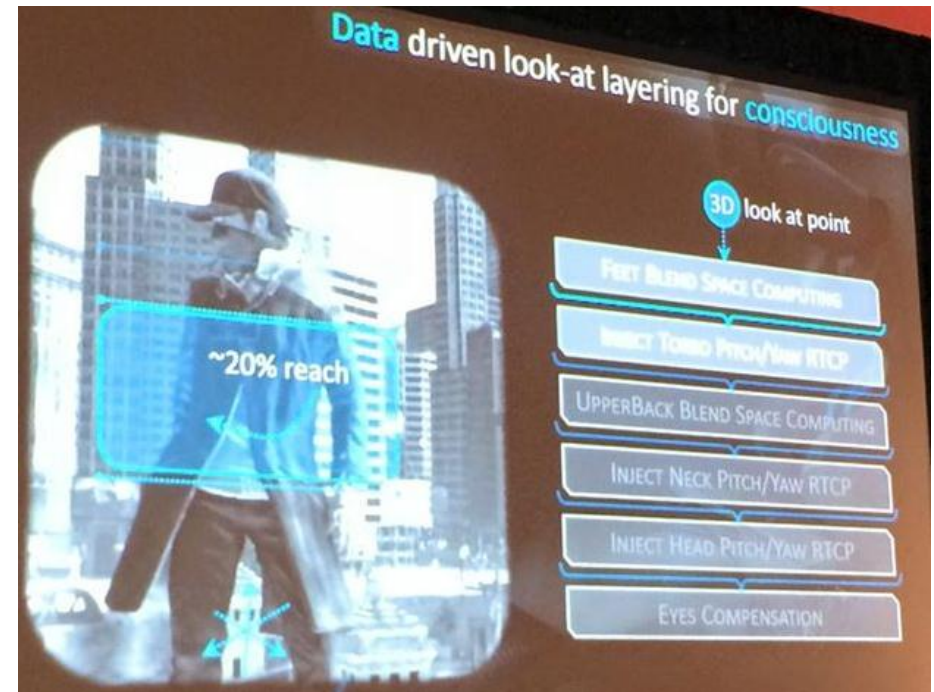
衝撃を受けた場合はラグドールに遷移



衝撃処理・IK処理



吹っ飛ぶ予定の軌跡をボリュームで
事前に検査する



Look at point の処理

レイヤー構造のアニメーションとレンダリング

Pose modification layering for diversity

Additive layering on full body

- different weights for each bone
- bone weights different for each speed
- feet IK resolution post layering
- ...not only useful for slopes!



FEET BLEND SPACE COMPUTING

```
// Get current referential
Quaternion CRightFoot( sourceResult->m_displacementTransform.GetRotation()
    * (modelCommonRaw->GetModelTransform( rightFootIdx, skeletonRaw, jointTransformArray ).GetRotation() * InvRightFoot) );

Quaternion CLeftFoot( sourceResult->m_displacementTransform.GetRotation()
    * (modelCommonRaw->GetModelTransform( leftFootIdx, skeletonRaw, jointTransformArray ).GetRotation() * InvLeftFoot) );

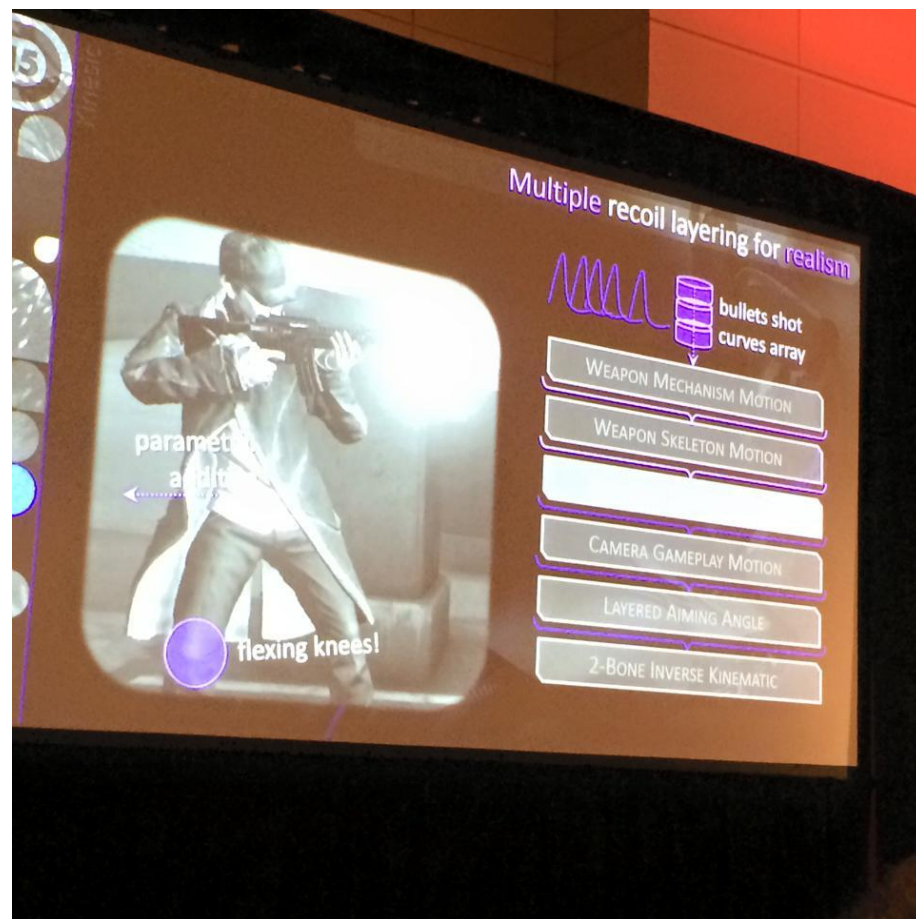
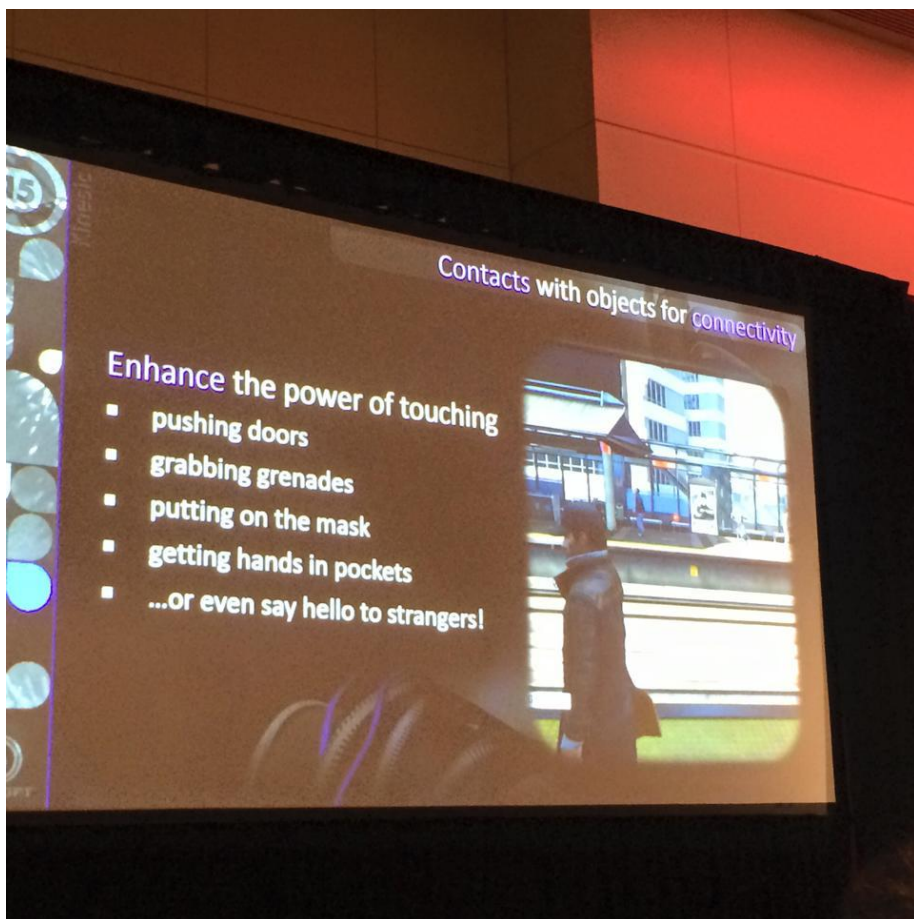
Quaternion feetRot;
feetRot.QuaternionSlerp( CRightFoot, CLeftFoot, 0.5f );
feetRot.RemoveAllExceptAxis( CVector3( 0.f, 0.f, 1.f ) );
```

UPPERBACK BLEND SPACE COMPUTING

```
// Get current referential
ndQuat upperBack( sourceResult->m_displacementTransform.GetRotation()
    * (modelCommonRaw->GetModelTransform( upperBackIdx, skeletonRaw, jointTransformArray ).GetRotation() * InvUpperBack) );

m_lookatWorldTransform = ndPosQuatTransform( upperBack, neckPos );
```

リコイル処理



多層レイヤーを介して
銃の衝撃のリコイル処理を行う

Massive Crowd on Assassin's Creed Unity: AI Recycling

Francois Cournoyer | AI Programmer, Ubisoft

Location: Room 2016, West Hall

Date: Wednesday, March 4

Time: 3:30pm - 4:30pm

Format: Session

Track:  Programming

Pass Type: All Access Pass, Main Conference Pass - **Get your pass now!**

Vault Recording: Video

Audience Level: Intermediate

 Like  Share

In this presentation, we will introduce a new technique of AI level of detail that has allowed us to create thousands of replicated, persistent, interactive NPCs for Assassin's Creed: Unity. We will go over how we created a pooling system that allowed us to swap from low-res NPCs to high-res NPCs without the player noticing. This talk will then explain how to create a simplistic AI brain to animate the thousands of NPCs. With the limit of 40 real AIs and 120 high resolution models, we could successfully create a scene where 10,000 crowd NPCs are on screen at the same time.

Takeaway

Attendees will learn how to develop AI systems using new console capabilities, in order to have a massive number of entities present in their game world. We will cover many optimization techniques done, as well as the mistakes that were made during the development process.

Intended Audience

This presentation is geared toward AI programmers and engine programmers. Basic knowledge of AI systems is required.

Postmortem: Developing Systemic Crowd Events on Assassin's Creed Unity

Christine Blondeau | Game Designer, Ubisoft Montreal

Location: Room 3016, West Hall

Date: Friday, March 6

Time: 10:00am - 11:00am

Format: Session

Track:  Design,  Programming

Pass Type: All Access Pass, Main Conference Pass - **Get your pass now!**

Vault Recording: Video

Audience Level: All

 Like  Share

With next generation consoles, players expect more from open world games. They expect deeper immersion with AI and more emergent gameplay that will add replay value. In ACU, we accomplished this by developing non-player-centric systemic events, capable of creating spontaneous gameplay opportunities for the players. This lecture looks back at the development of the system from both a game designer and a programmer's perspective, and how we overcame some of the challenges presented by systemic gameplay via concrete examples. This talk will present the initial vision for the system, our design intentions, problems we met as we iterated on the system, and the solutions we used, integration with other crowd systems, level and mission design concerns, as well as avenues of improvement.

Takeaway

Attendees will learn about the pitfalls that come with designing systemic gameplay, the lessons we learned on Assassin's Creed Unity, and the tools and solutions we used during development.

Intended Audience

Game designers, level designers and gameplay programmers interested in AI, systemic gameplay and open world design. While this session does not delve into highly technical explanations, it will appeal especially to the more technically inclined designers.

<http://schedule.gdconf.com/session/massive-crowd-on-assassins-creed-unity-ai-recycling>

<http://schedule.gdconf.com/session/postmortem-developing-systemic-crowd-events-on-assassins-creed-unity>

ASSASIN'S CREED UNITY のAIと群衆 オンライン上の同期の仕組み



Networking Gameplay and AI in Assassin's Creed Unity

Charles Lefebvre | AI Programmer, Ubisoft

Location: Room 3020, West Hall

Date: Friday, March 6

Time: 10:00am - 11:00am

Format: Session

Track:  Programming

Pass Type: All Access Pass, Main Conference Pass - **Get your pass now!**

Vault Recording: Video

Audience Level: Intermediate

 Like  Share

Making Assassin's Creed Unity a completely unified, networked multiplayer game was a huge challenge, as none of the existing systems were network-friendly. We needed to support join in-progress co-operative gameplay with host migration in a streaming open world. This required us to design replication frameworks that were lightweight in terms of memory, CPU and bandwidth. This talk will present you the solutions that we are using to replicate the gameplay and the AI. It will encompass various systems, like the spawning, the gameplay mechanisms, and the AI of NPCs. It will also introduce the tools we have developed to make network replication easier for programmers.

Takeaway

Attendees will have an overview of our solution for replicating an existing gameplay engine. They will be able to compare it with their own solutions, or decide if our solution could fit in their framework.

Intended Audience

The target audience includes gameplay and AI programmers. Basic knowledge of game networking techniques is recommended.

<http://schedule.gdconf.com/session/networking-gameplay-and-ai-in-assassins-creed-unity>



ASSASIN'S CREED UNITY のAIと群衆 オンライン上の同期の仕組み

資料(PPT欄外のコメントに詳細が書かれています)

Charles Lefebvre

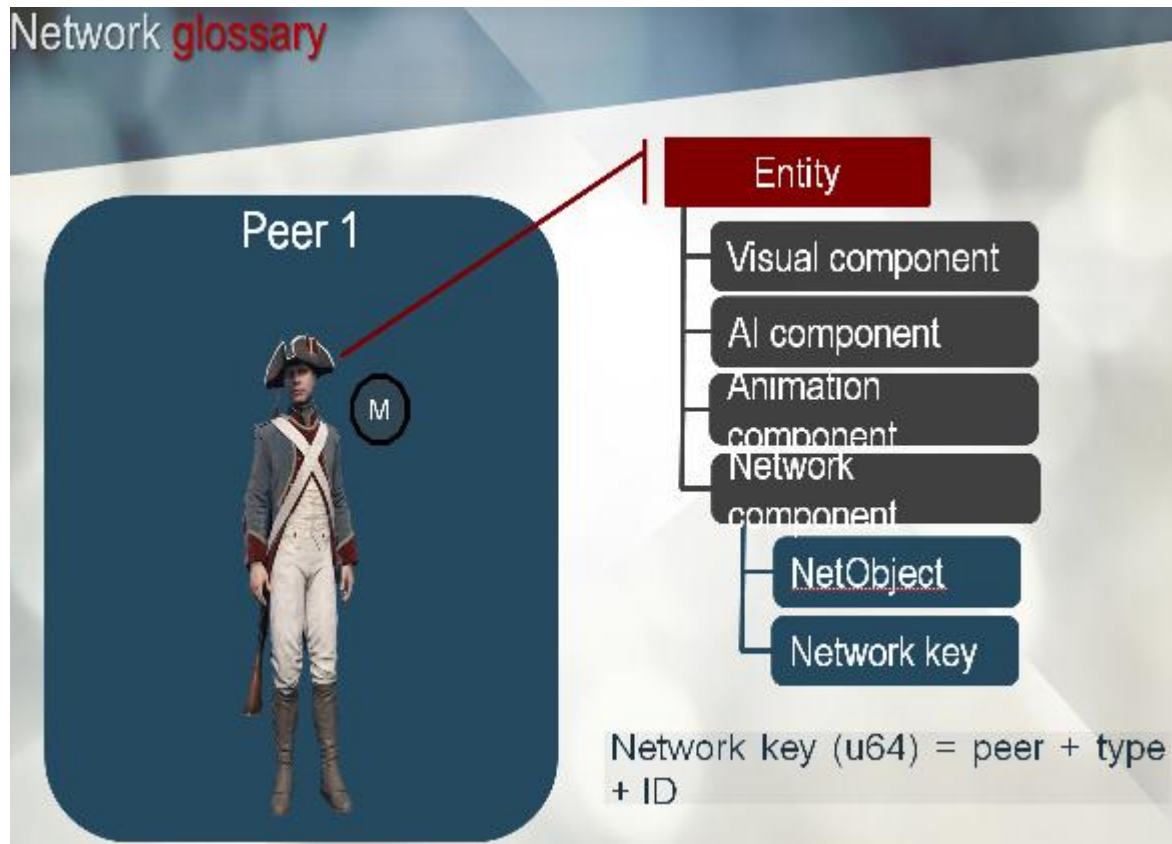
“Gameplay and AI networking in Assassin’s Creed Unity”

<http://www.authorstream.com/Presentation/charleslefebvre-2424121-gdc-gameplay-replication-acu/>

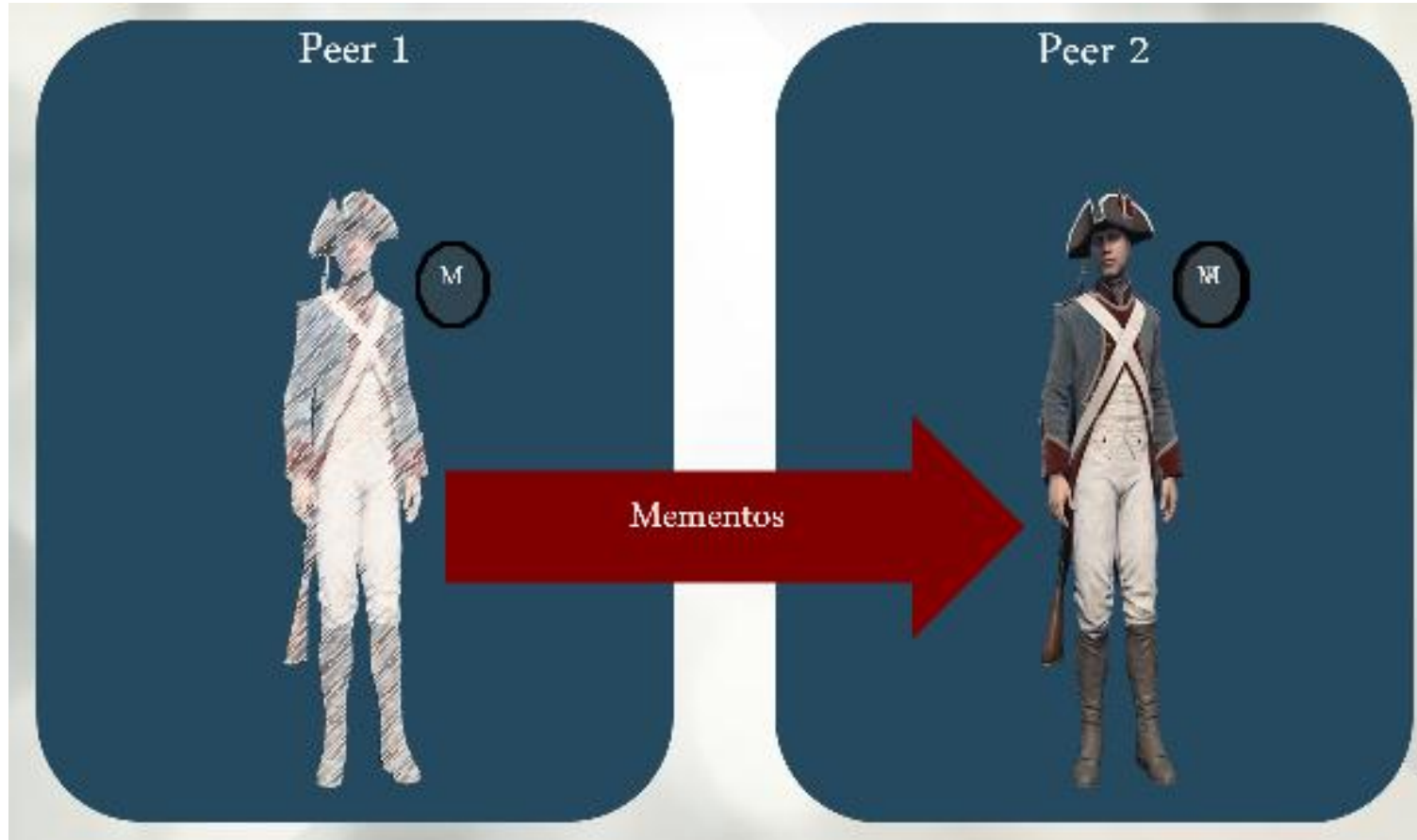
<https://t.co/YeYvvDyFYt>

<https://twitter.com/FrozenInMTL/status/577294278306705408>

オンライン処理のための NetObject と Network Key

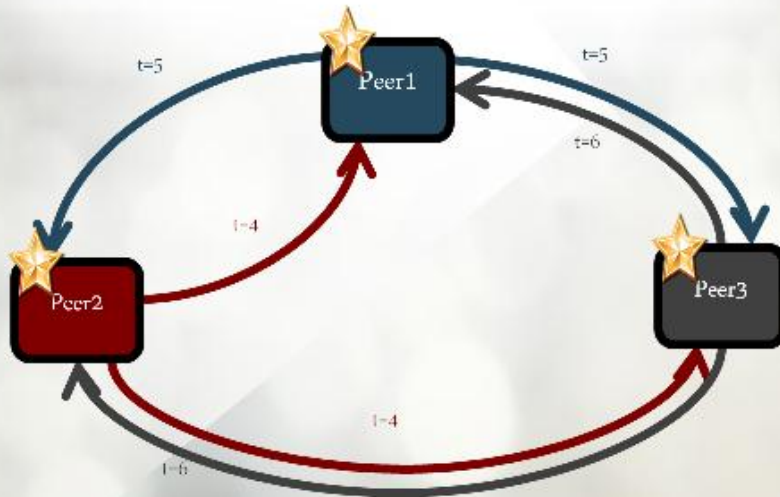


ユーザーから別のユーザへ同期



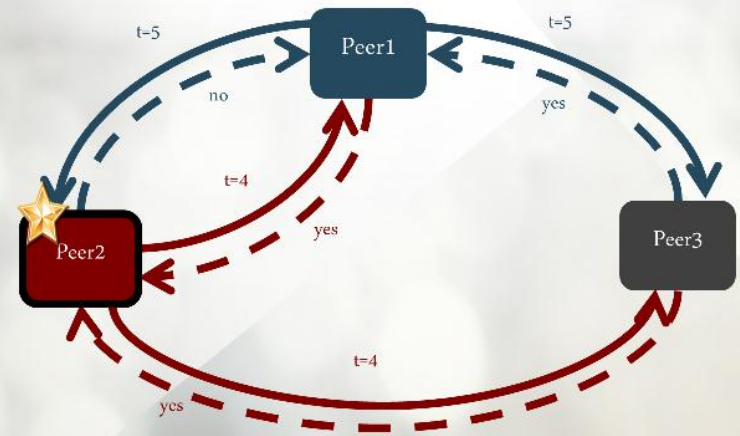
NetToken でAI同期を処理 プレイヤーの同期も同様

Can also be used to wait for all players



Network tools: **NetTokens**

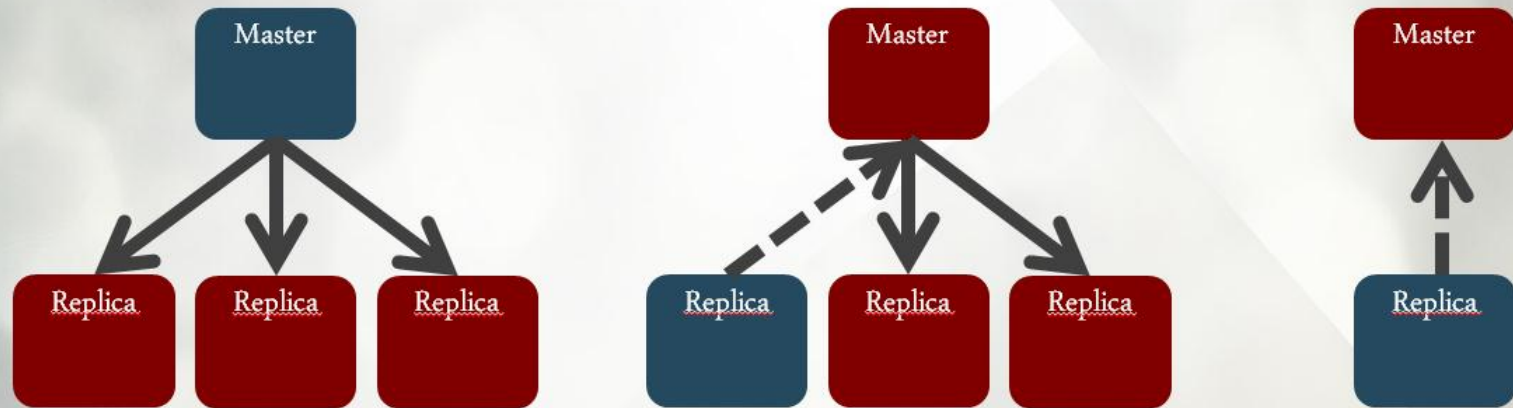
NetTokens: Layer for peer to peer conflict resolution



Master と Replica が組織されるが、
処理の発端から RPC がマスターに渡される。

Network tools: **remote procedure call**

Flow examples:

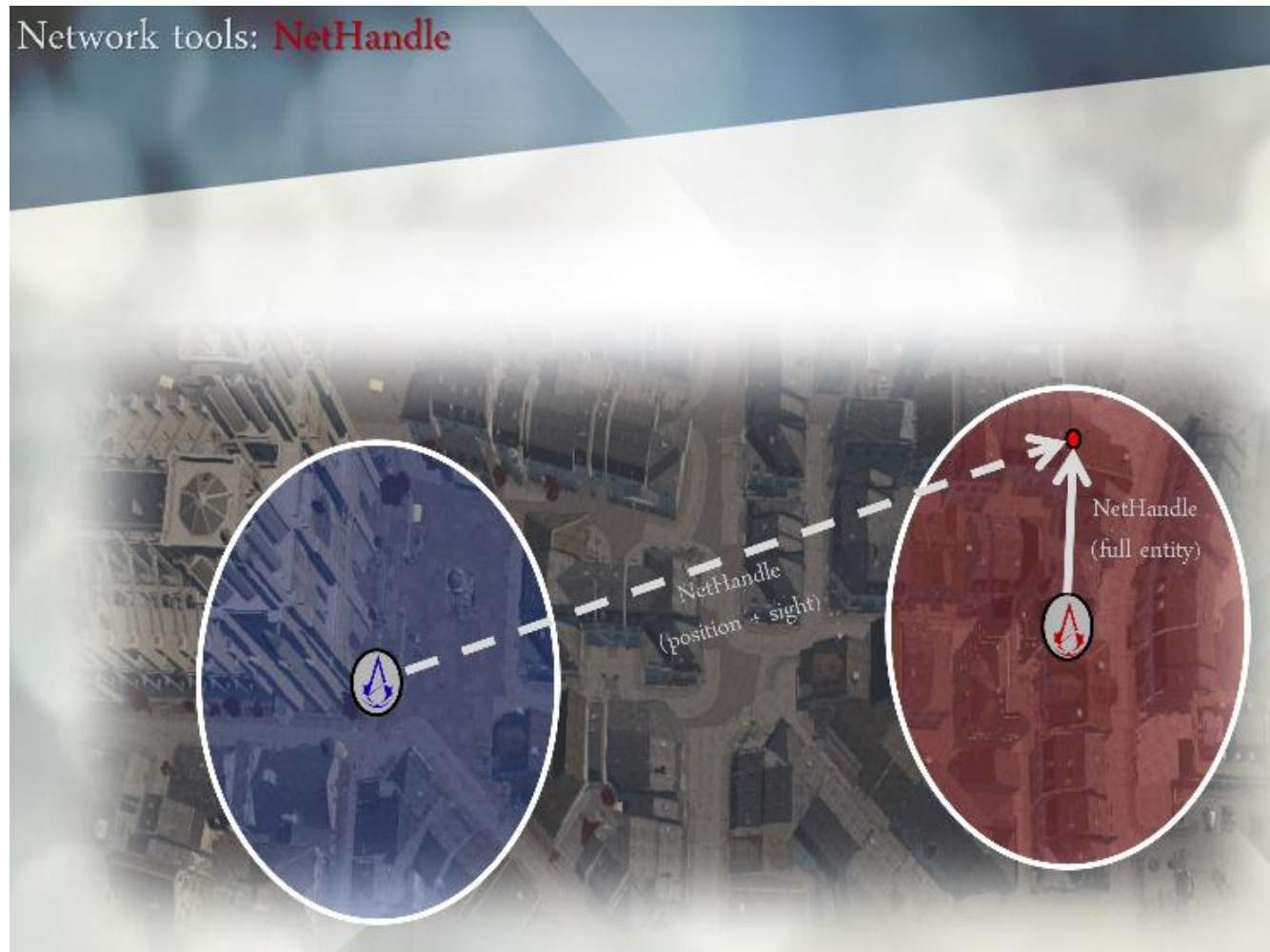


NetHandle 構造体を渡し合う

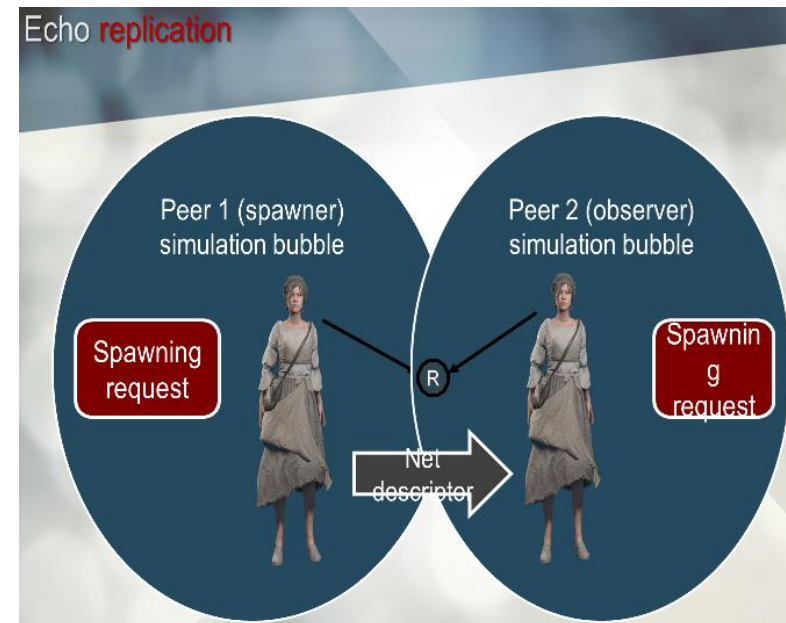
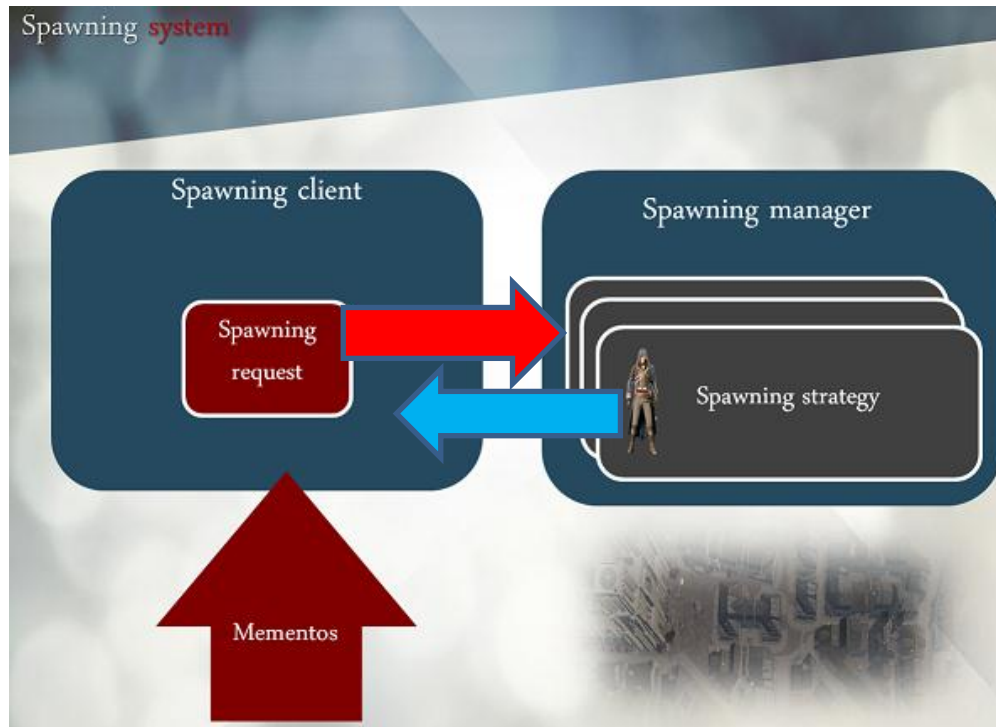
```
struct NetHandle
{
    Handle<Entity> m_Entity;
    NetKey m_NetKey;

    bool IsValid();
    Entity * GetEntity();
    bool GetPosition(vec4& position);
    bool GetSight(vec4& position);
}
```

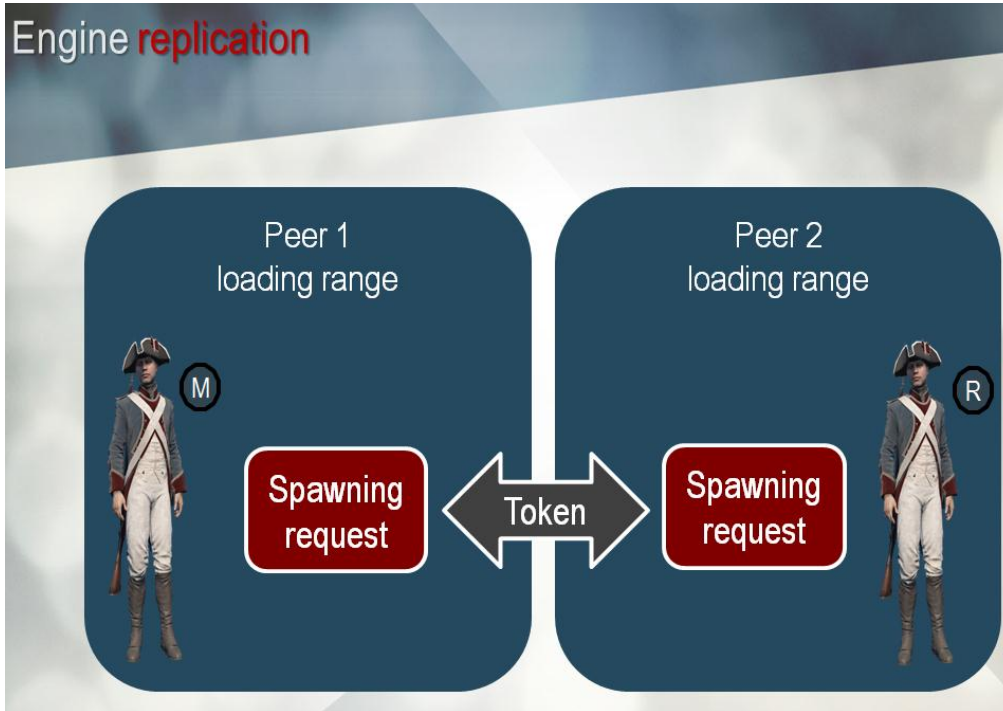
Network tools: **NetHandle**



Spawning request が Manger に投げられて、そこからキャラクー生成命令が複数のPeerへ出る。



Engineにおける複製



それぞれのエンジンにおいて、
ゲームプレイシステムが、
エンティティをスポーンするか決定する。

ゲームプレイ・コーディネーターは、
複数のピアでアクティブになることが
できる。ピアの数と同じぐらいの
エンティティ数に抑えておきたい。

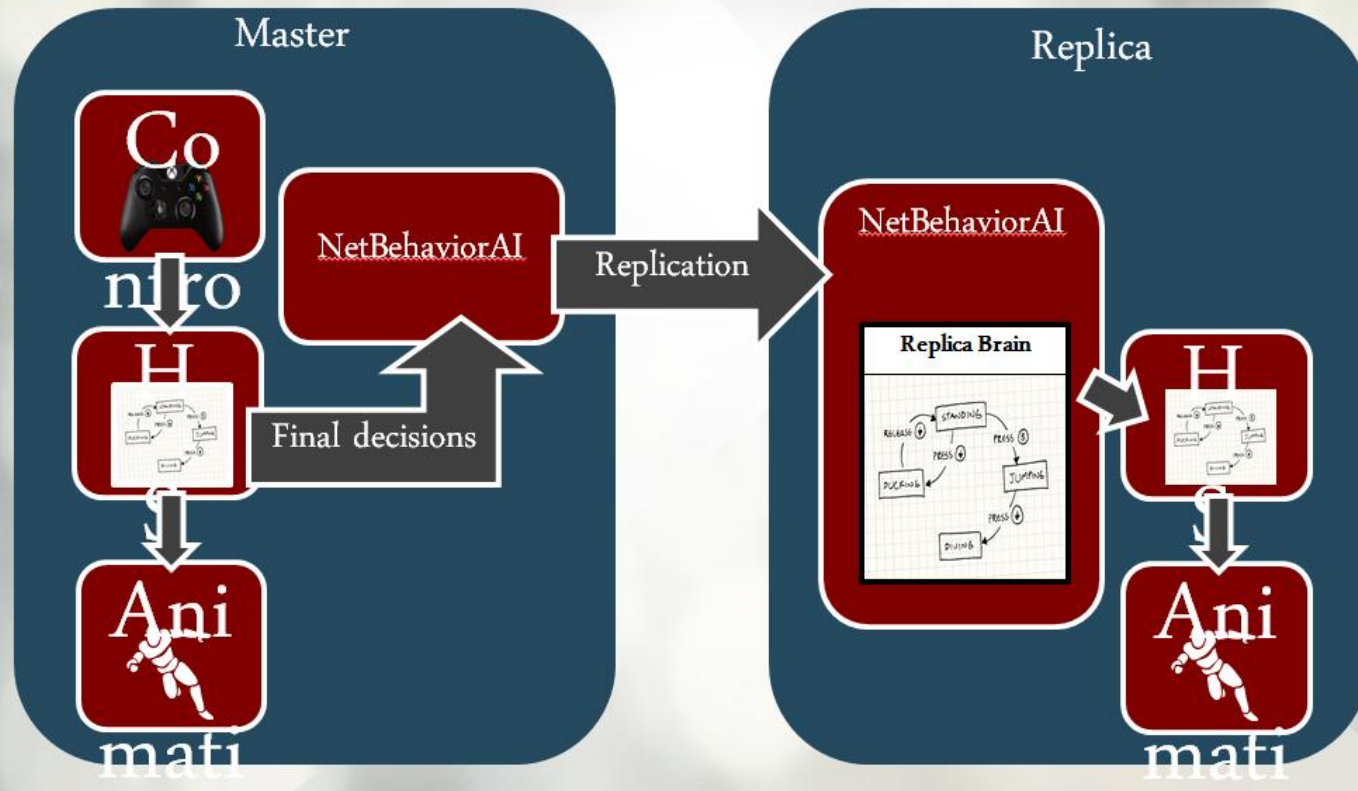
スポーニングのリクエストは、
全てのピアの上に投げられる。
スポーニングリクエストには
トークンが必要で、結果、マスターが
生成される。トークンはまた、
エンティティのネットキーをセットする
ためにも使われる。他のピアは、
レプリカを生成する。

生成シードは、スポーニングクライアント
のIDと
共に生成される。すべてのピアの対して
知られてるNPCタイプは
何もブロードキャストされない。

プレイヤーのレプリカ

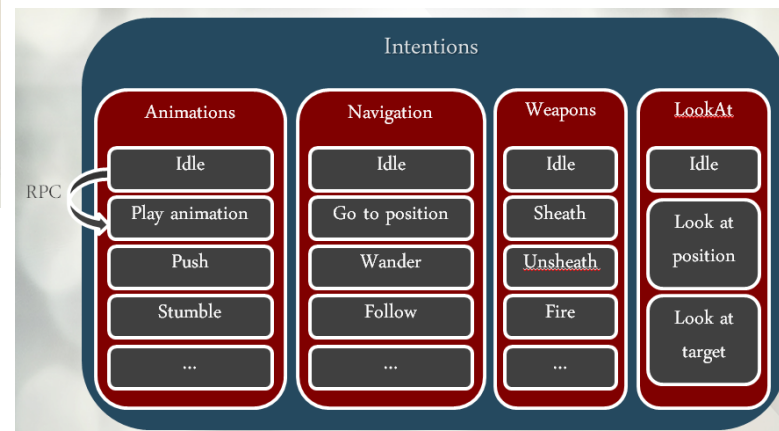
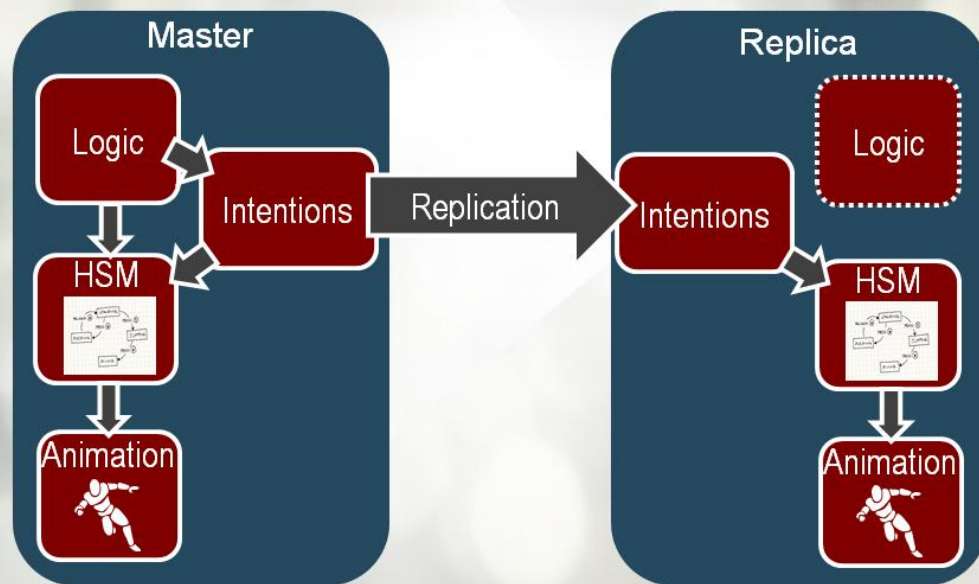
HSM (Human State Machine) の状態で同期

Player replication **flow**

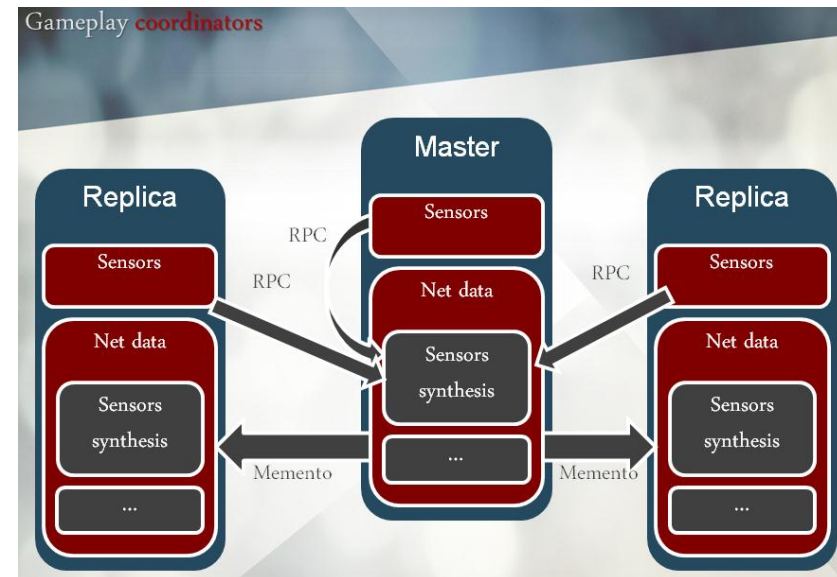
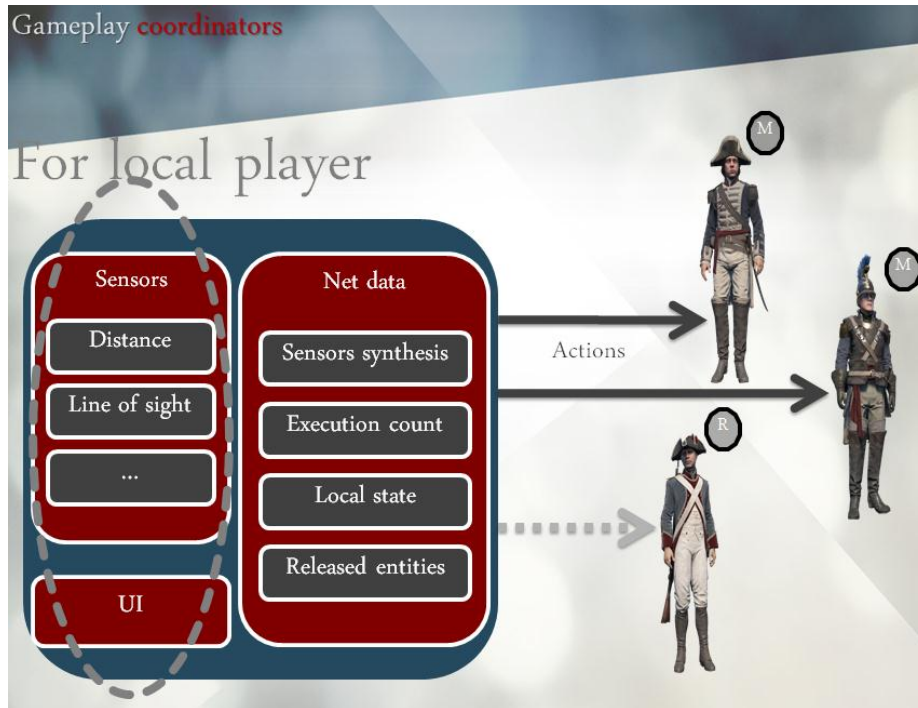


AIはIntention Layer で同期

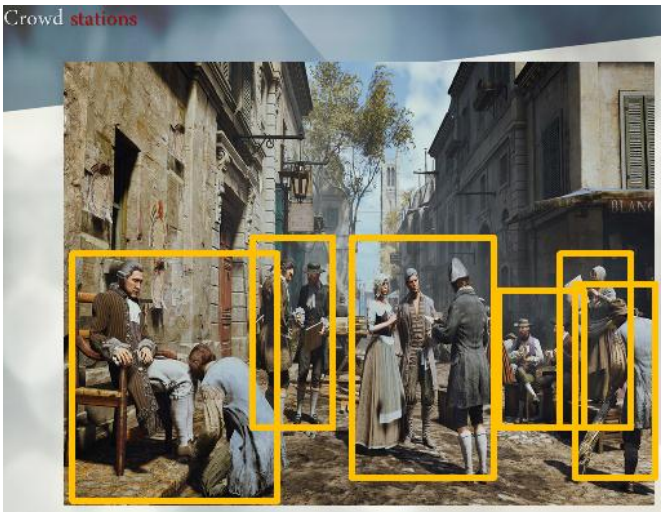
A.I. replication **flow**



各ローカルのAIのセンサー情報を マスターが統合し共有する。



移動しないが動作する数人の群衆(＝ステーション)は
雰囲気作りが主で基本的に同期しないが、
キャラクターとのインタラクションでステートが変わると共有する。



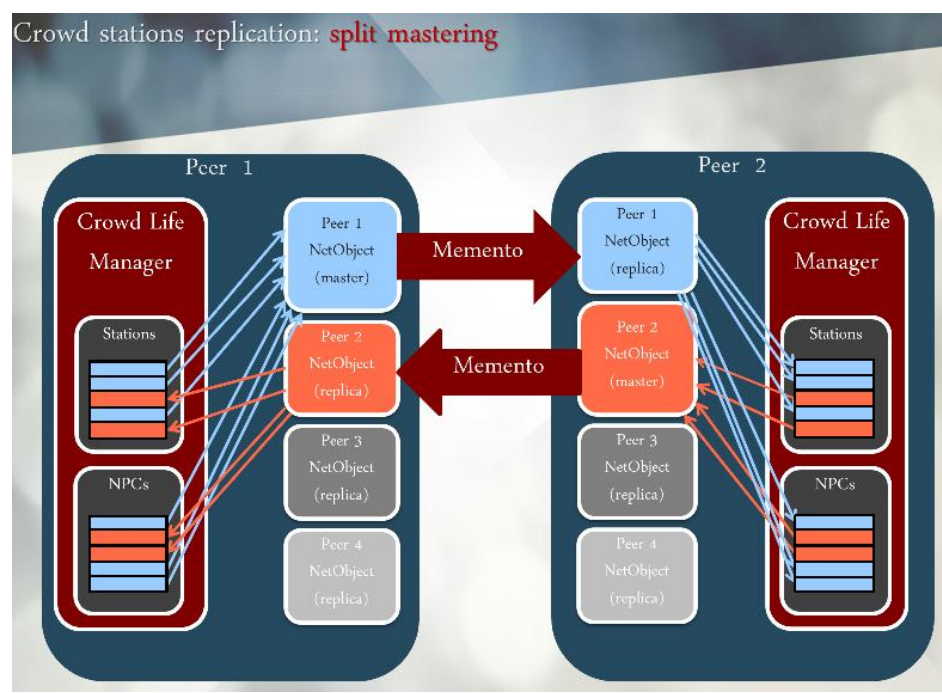
Crowd stations **replication**



What to replicate ?

- NPCs
- Station state

Master = token winner



Frostbite: Implementing a Scripting Solution for Your Editor

Matthew Doell | Technical Art Director, Electronic Arts

Location: Room 2016, West Hall

Date: Thursday, March 5

Time: 11:30am - 12:30pm

Format: Session

Track:  Programming

Pass Type: All Access Pass, Main Conference Pass - **Get your pass now!**

Vault Recording: Video

Audience Level: Advanced

 Like  Share

The scripting solution used in Frostbite tools is described along with production examples from Battlefield 4, Need for Speed, Plants vs. Zombies: Garden Warfare and Dragon Age: Inquisition. Building and maintaining a solution that can be shared by a wide range of projects requires adaptable technology and a community of developers that actively share across teams and contribute back to the core. One of the Frostbite technologies that has been used effectively across many game teams is Frostbite Script referred to as FBScript. This scripting solution provides the ability to quickly create custom tools for each project, and easily share new tools between teams.

Takeaway

Tool and application developers will take away a better understanding of how to design and implement scripting as a feature for .Net applications by looking at how a scripting solution was implemented and maintained over multiple versions of the Frostbite editor.

Intended Audience

Software programmers, technical artists and game designers are using this solution at EA. People working in similar roles at other companies will find value in the design, implementation and examples. A prerequisite would be knowledge of scripting languages of any kind (Python, MEL, JavaScript, C#, etc.)

FROSTBITE SCRIPT

EA「FROSTBITE ENGINE」とバックエンド

<http://schedule.gdconf.com/session/frostbite-implementing-a-scripting-solution-for-your-editor>

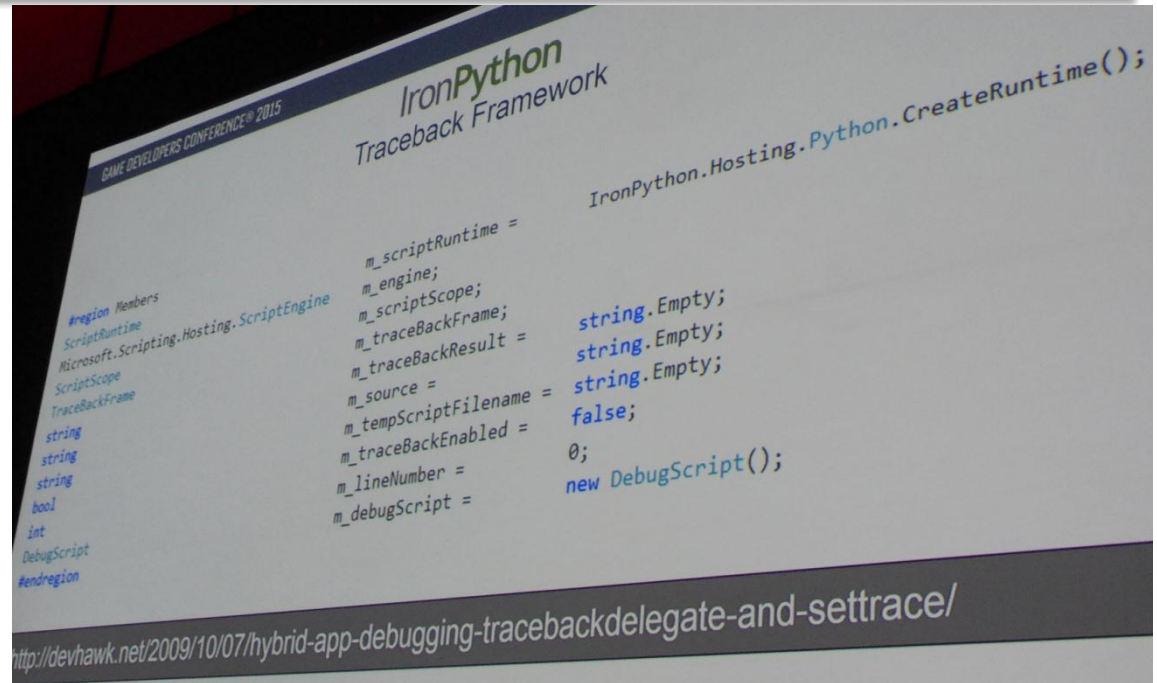
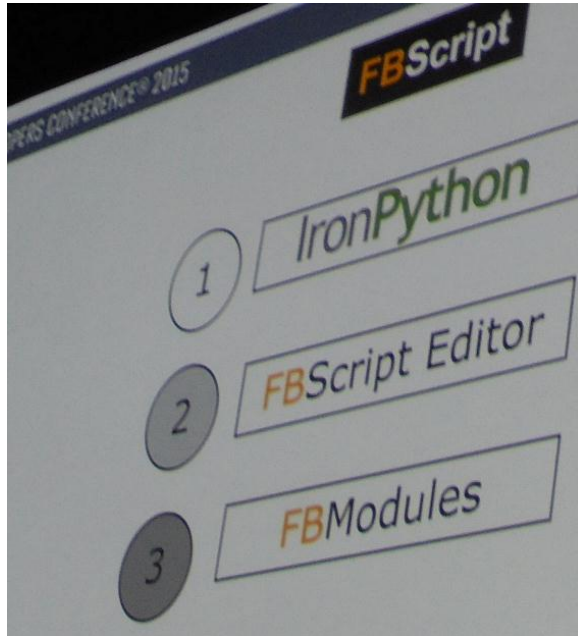
Frostbite ENGINE (2007-)

FrostBite ENGINE = Battlefield用エンジン(D.I.C.E.)

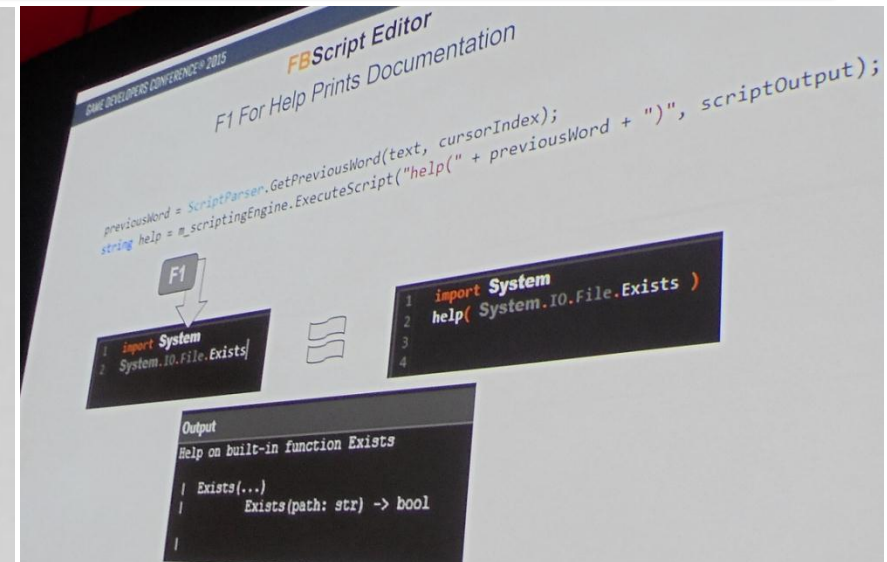
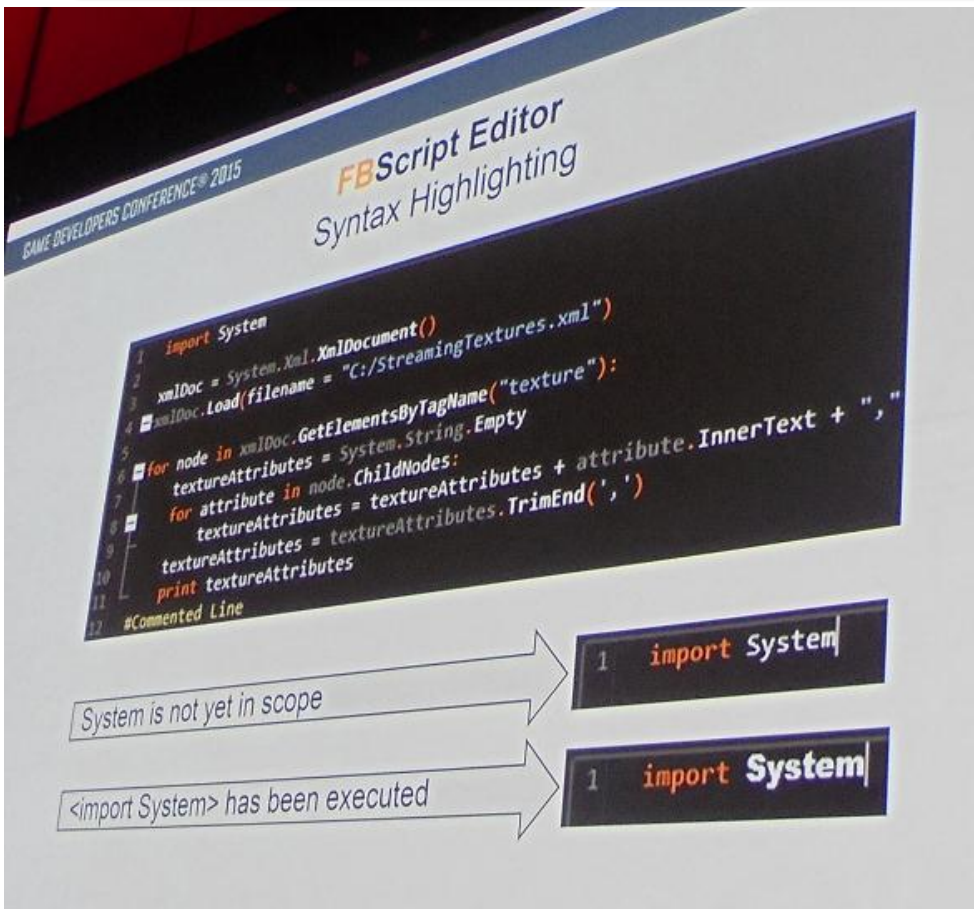


EA 標準開発環境

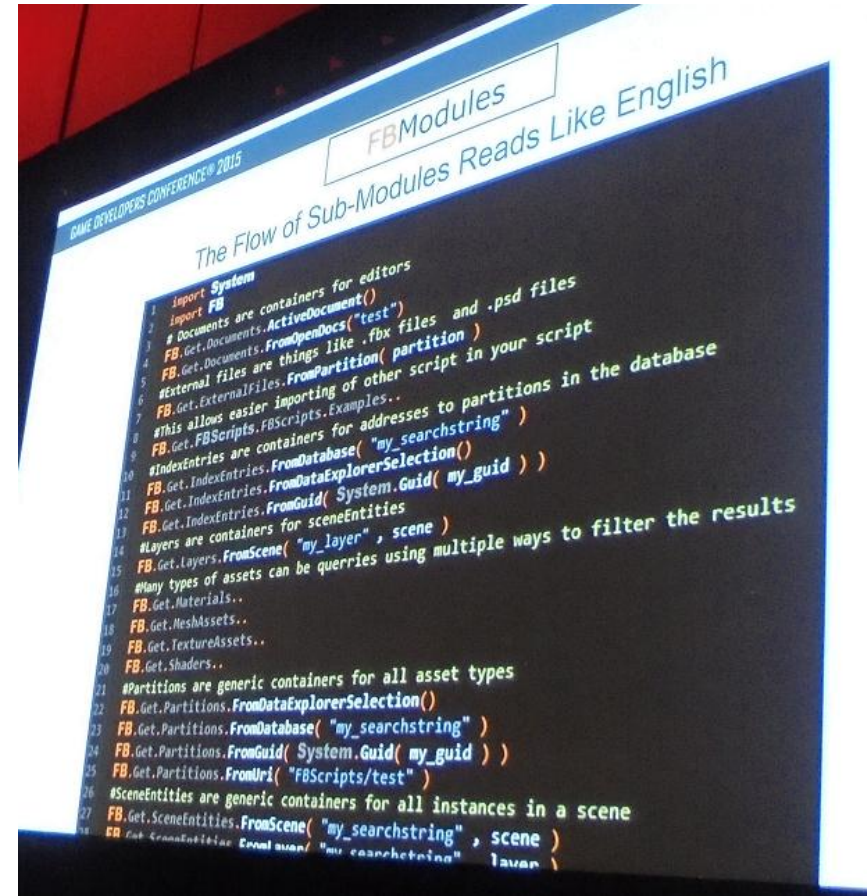
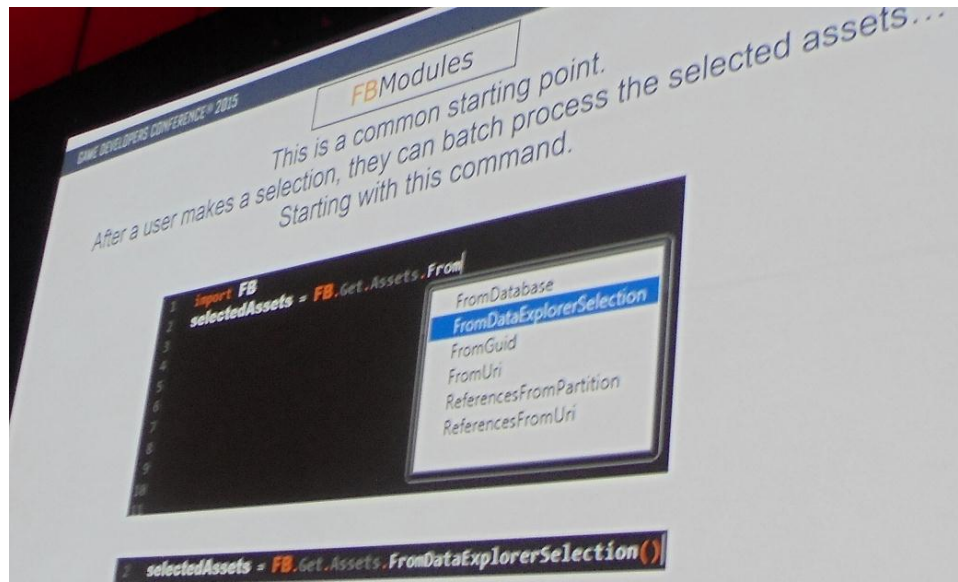
FBScript, FBModules



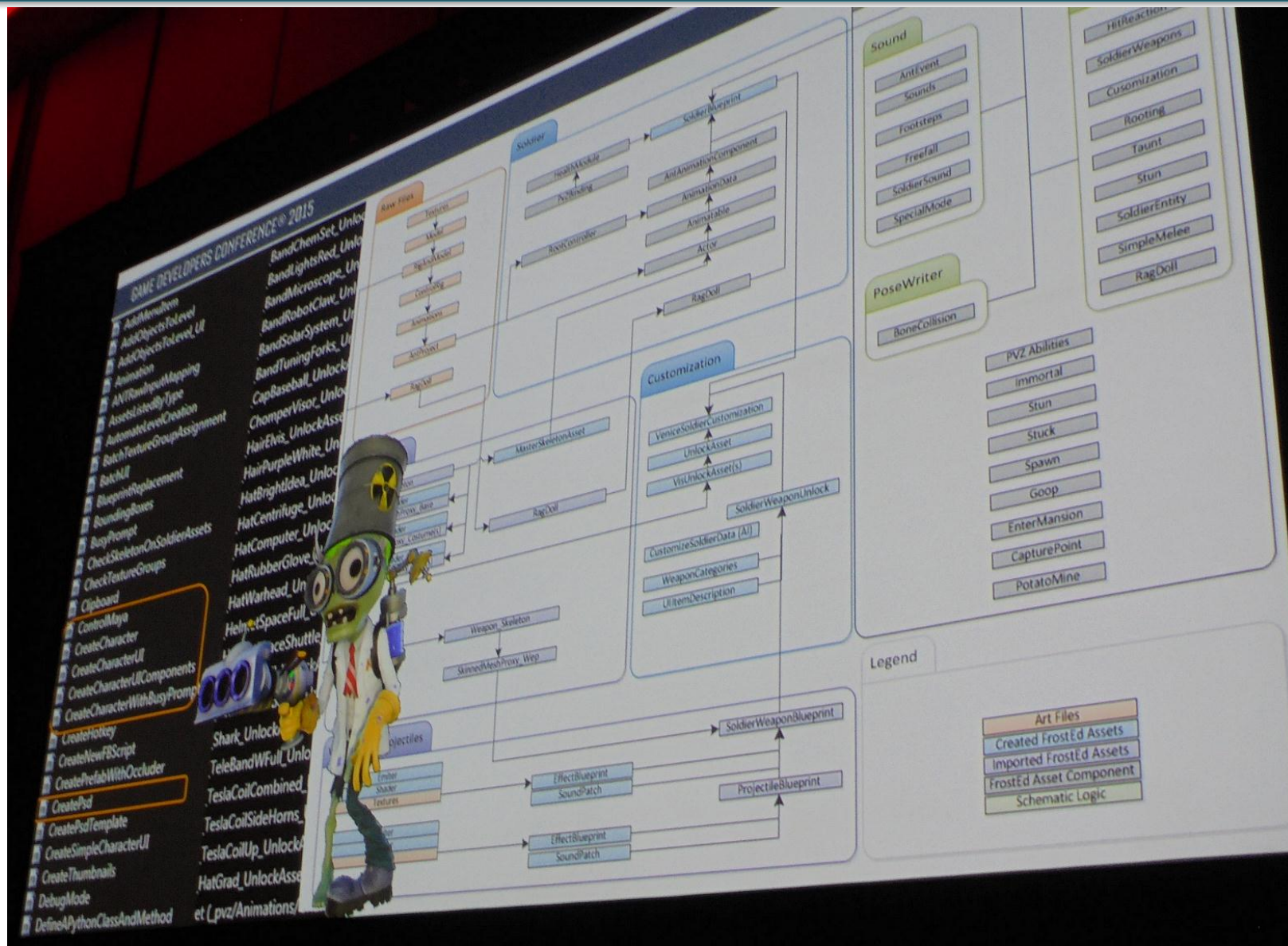
FBScript Editor



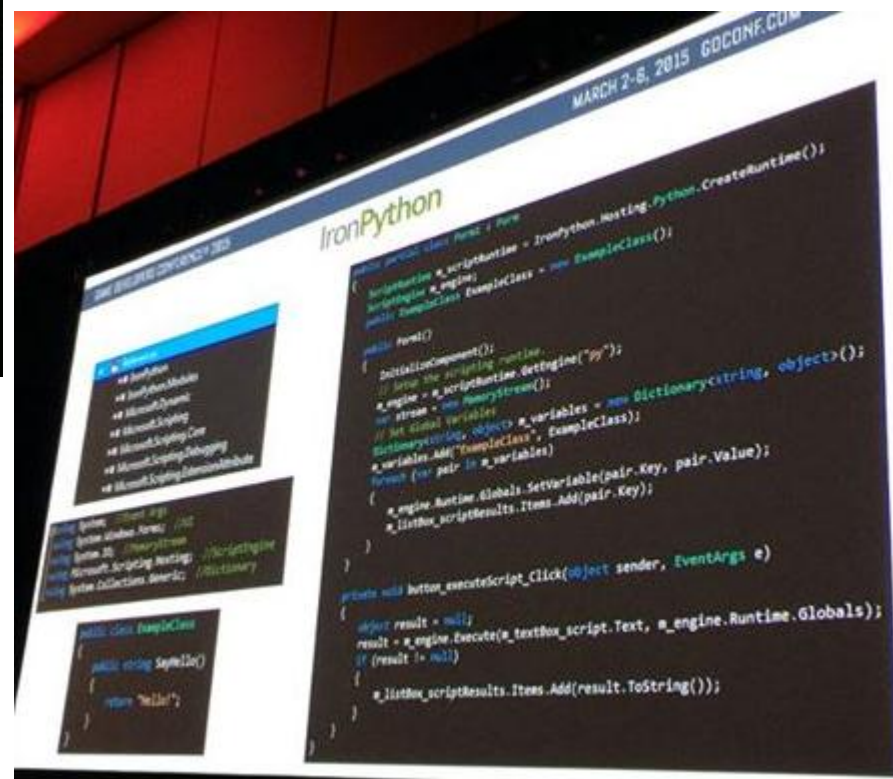
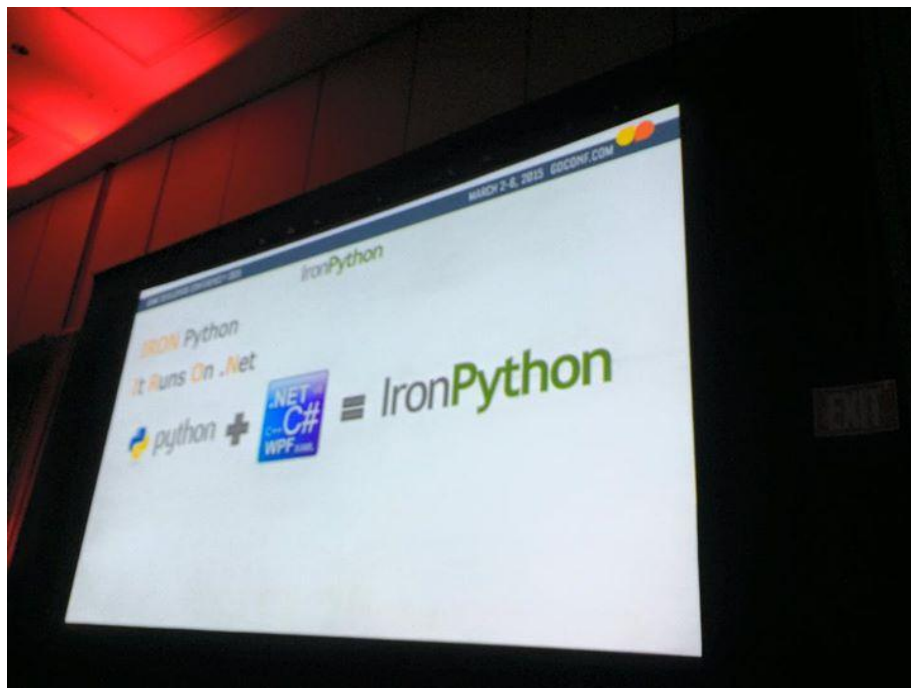
FB Modules



ゾンビゲームでフロストバイトを使っているところ



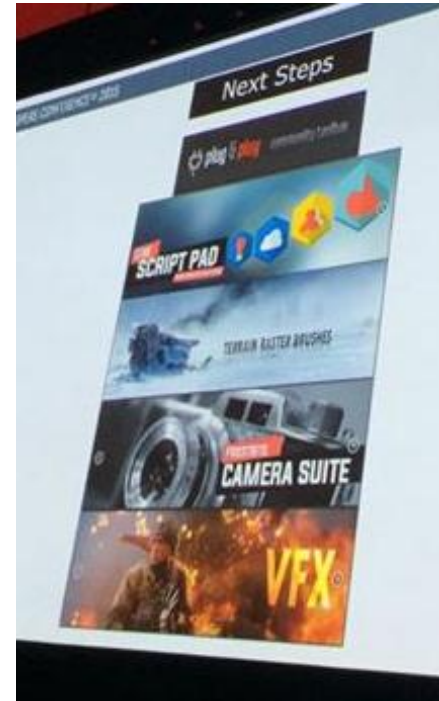
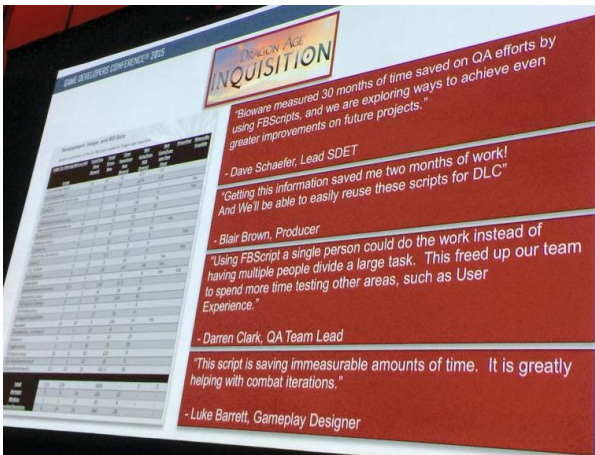
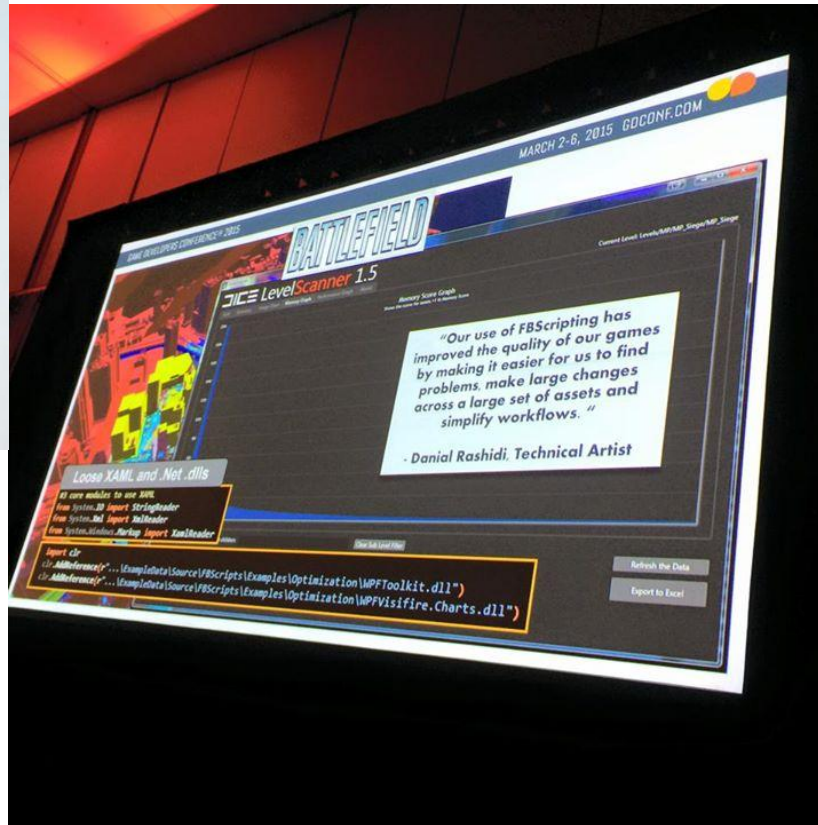
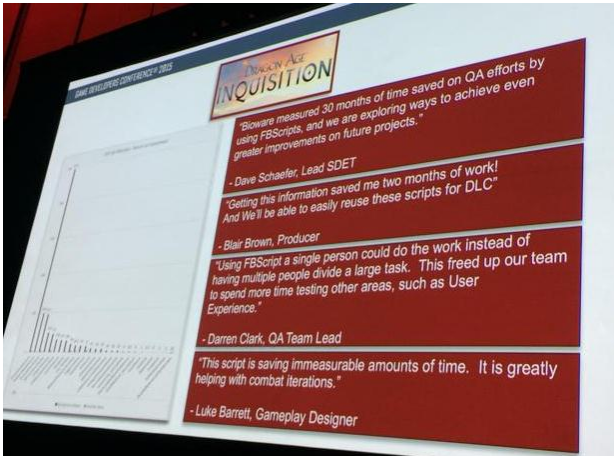
IronPython を使っている



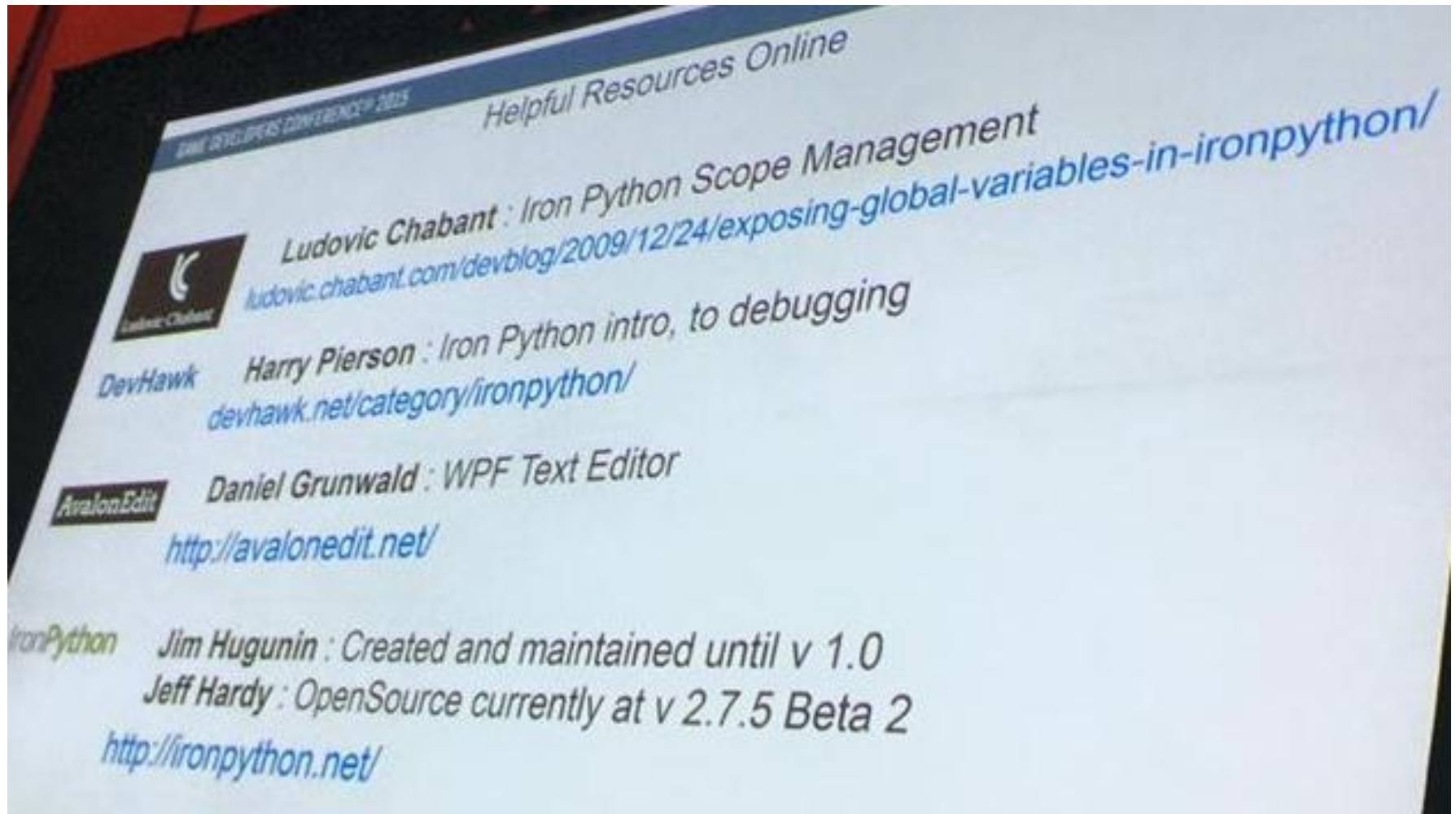
Frostbite Editor



Battlefield, Dragon Age INQUISITION
でも使用



外部のツールも積極的に使用

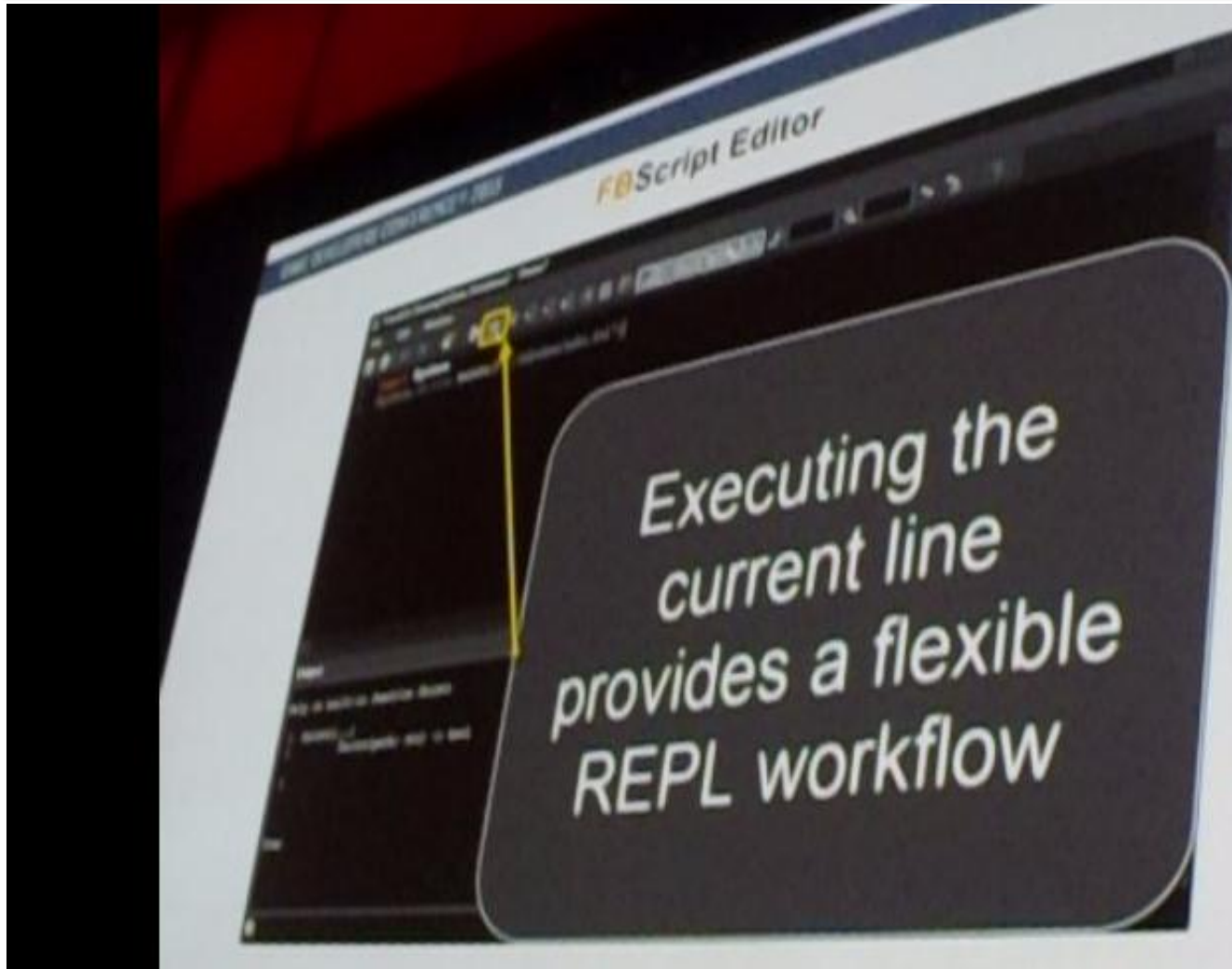


Frostbite Tool

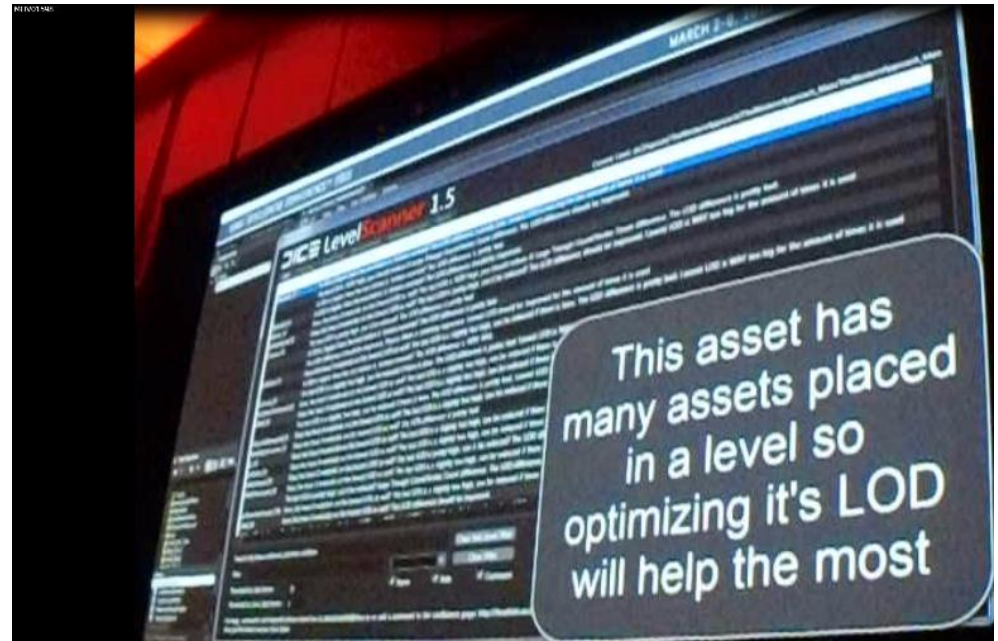
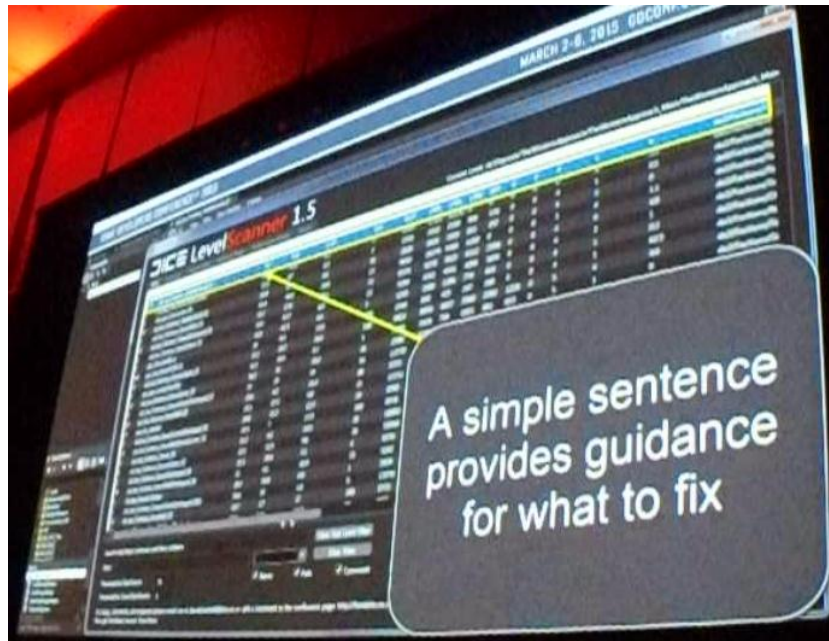


Level Editor

Frostbite Script Editor



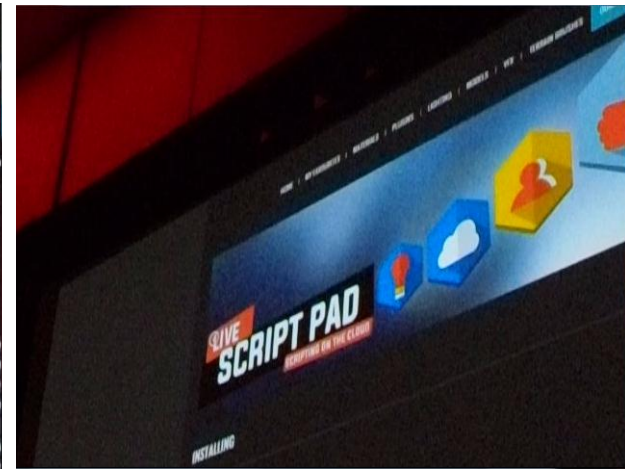
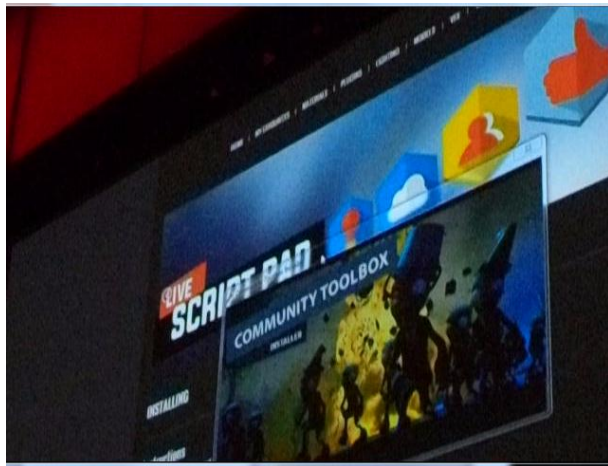
Frostbite Level Scanner



Level Scanner

そのレベルにあるアセットの一覧を表示して、ポリゴン数や、置かれている回数など、統計情報を一覧表示してくれる。

Frostbite Community Toolbox



社内のCOMMUNITY TOOL BOX があり、
そこから共通ツールを引き出して、
その場でロードして使うことができる。

Fast Iteration for Far Cry 4 - Optimizing Key Parts of the Dunia Pipeline

Remi Quenin | Engine Architect, Ubisoft

Location: Room 2016, West Hall

Date: Friday, March 6

Time: 10:00am - 11:00am

Format: Session

Track:  Programming

Pass Type: All Access Pass, Main Conference Pass - **Get your pass now!**

Vault Recording: Video

Audience Level: Intermediate

 Like  Share

With five platforms to support and twice as much content as Far Cry 3, Far Cry 4 presented a big challenge for pipeline efficiency. To reach the maximum achievable quality in such a massive open-world environment, iteration is key. This talk will discuss game-changing improvements that we applied to our pipeline that resulted in spectacular day-to-day workflow improvements for the entire production. We'll explain how we massively reduced the compile time of our 2.5M lines of C++ code to under a minute. We'll describe how we built a network cache for "binarized" assets on commodity PCs, outperforming million-dollar commercial hardware. We'll show how we used this asset store to speed up our data builds, and how we accelerated our package distribution to be instantaneous for hundreds of concurrent users. Finally, we'll show the tools we've built to allow patching a running instance of the game with locally modified assets for rapid content iteration.

Takeaway

The attendee will see that great iteration times are possible even on the largest of AAA games. The techniques used to optimize production will be detailed, showing how it is possible to get the latest data package, compile your own executable, and play on console - all in under two minutes.

Intended Audience

The presentation is aimed at programmers and architects, but may be of interest to technical content creators.

UBIのエンジン群

<http://schedule.gdconf.com/session/fast-iteration-for-far-cry-4-optimizing-key-parts-of-the-dunia-pipeline>

Ubiのエンジン群

- DUNIA ENGINE = FarCry, Assassin's Creed
- 小型・中型 ENGINE = UbiArt Framework

[GDC 2014]アーティストの意思をすぐに反映できるエンジン「UbiArt Framework」でゲームを制作する“新たなプロセス”を紹介
<http://www.4gamer.net/games/999/G999903/20140322007/>

DUNIA PIPELINE



Optimizing key parts of the

DUNIA

PIPELINE

FARCRY4
Fast iteration for



DUNIA
ENGINE™

Rémi QUENIN
Engine Architect - FarCry 4
Ubisoft Montréal
@azagoth



Fast Iteration for FARCRY4 – Optimizing key parts of the DUNIA PIPELINE
<https://docs.google.com/file/d/0B8l3tRzjCaGVT2tRaWt0U1JQT1k/edit>

DUNIA PIPELINE

THE DUNIA PIPELINE

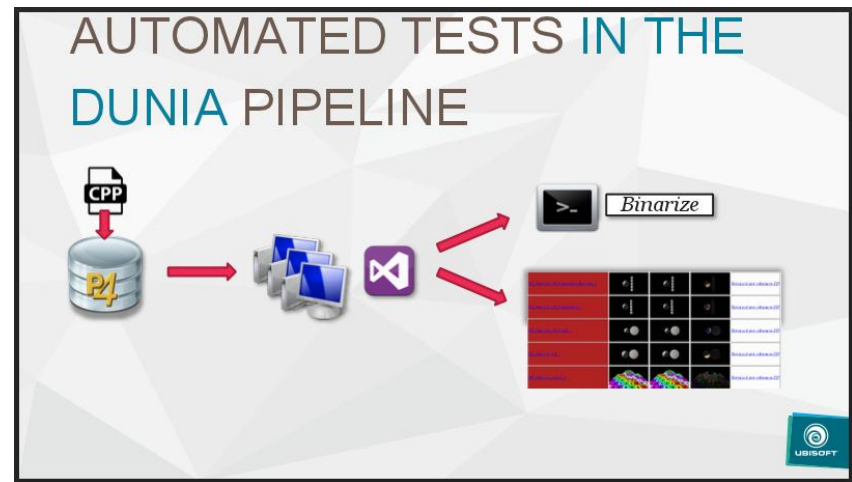
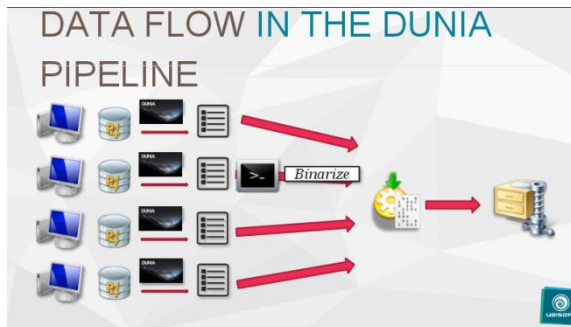
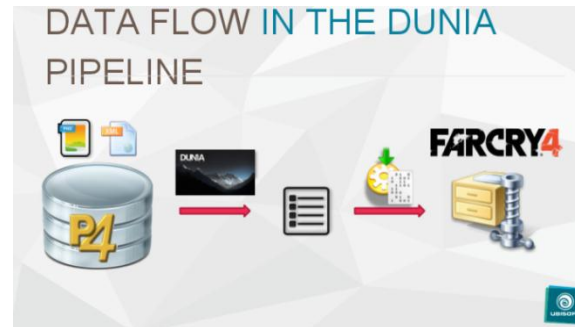
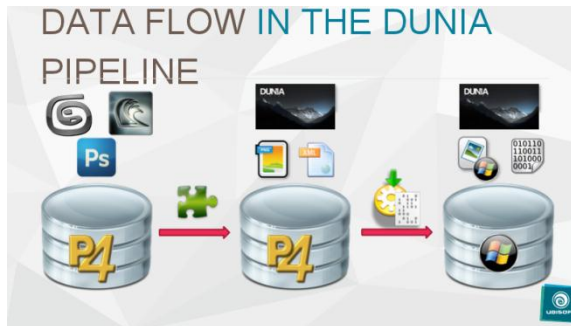
- 250 GiB of package per Day
Official + Test Maps
- Over 3.7 Million Lines Of Code
2.5 M. C++ Runtime
- 6 studios, 450 people in Montréal
Getting data everyday
- ~500 checkins per day
350 data, 150 code (+sound)



Fast Iteration for FARCRY4 – Optimizing key parts of the DUNIA PIPELINE

<https://docs.google.com/file/d/0B8l3tRzjCaGVT2tRaWt0U1JQT1k/edit>

パイプライン上の自動テスト

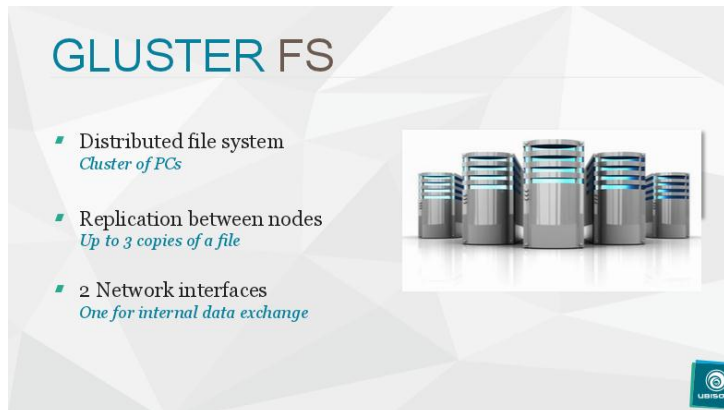
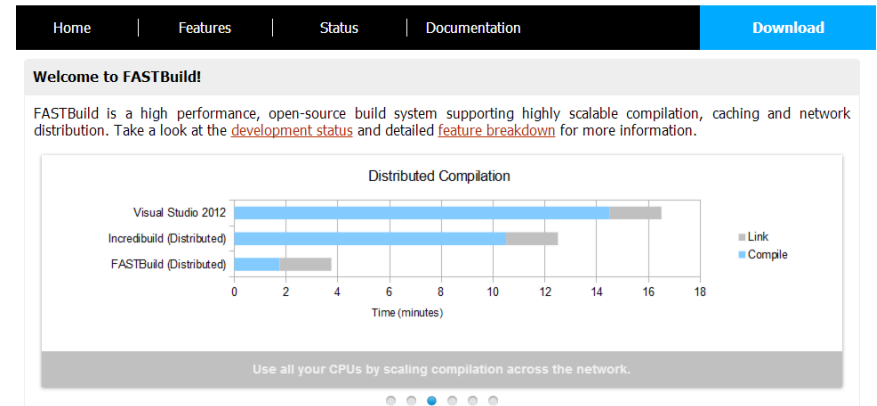


Fast Iteration for FARCRY4 – Optimizing key parts of the DUNIA PIPELINE

<https://docs.google.com/file/d/0B8l3tRzjCaGVT2tRaWt0U1JQT1k/edit>

高速化・最適化

- FASTBUILDの使用
- ファイル共有システム
- NIGHTLY BUILD
- CREATE PATCH



Fast Iteration for FARCRY4 – Optimizing key parts of the DUNIA PIPELINE
<https://docs.google.com/file/d/0B8l3tRzjCaGVT2tRaWt0U1JQT1k/edit>

Level Design in a Day: Procedural Level Design in Eldritch

David Pittman | Developer, Minor Key Games

Location: Room 3005, West Hall

Date: Tuesday, March 3

Time: 11:15am - 12:15pm

Format: Tutorials & Bootcamps

Track:  Design,  Production

Pass Type: All Access Pass, Summits, Tutorials & Bootcamps Pass - **Get your pass now!**

Vault Recording: Video



Minor Key Games co-founder David Pittman presents a detailed rundown of the theory and algorithms behind the procedural level design in Eldritch. How do you achieve a desired experience when level creation is offloaded to an algorithm? What role does a level designer play in a game with randomly generated levels? Why use procedural level design at all? David examines these and other topics through the lens of 3D action roguelike Eldritch. Attendees will learn how and why Eldritch's level generation was implemented, as well as general reasons why procedural algorithms might be embraced by level designers.

ELDRITCH のプロシージャル/AI技術

<http://schedule.gdconf.com/session/level-design-in-a-day-procedural-level-design-in-eldritch>

Eldritch のプロシージャル/AI技術

GAME DEVELOPERS CONFERENCE® 2015

MARCH 2-6, 2015 GDCONF.COM

Overview

- Build a 3D maze
- Fill it with room modules
- Add enemies and items



3Dダンジョン自動生成。
(モジュールの組み合わせを計算する)

各部屋をモジュール化。

エネミーの自動配置・スポーニング。

GAME DEVELOPERS CONFERENCE® 2015

MARCH 2-6, 2015 GDCONF.COM

Results



GAME DEVELOPERS CONFERENCE® 2015

MARCH 2-6, 2015 GDCONF.COM

Module transforms

- Rotation about z-axis
- Reflection about x- or y-axis



Modules may be rotated about the vertical axis to fit into the maze. So a module with exits to the North South and East could be spun into any of the positions on the top row.

Modules can also be mirrored about the horizontal axes. This produces redundant configurations (note that each configuration on the bottom row also exists on the top), so it isn't strictly necessary. But mirrored modules are visibly different from unmirrored ones, so doing this stretches our modules a bit further.

Mirroring also adds a small constraint; I couldn't, for example, spell a word out on voxels because the word would read backward when the module was mirrored.

制限付きのジェネレーション

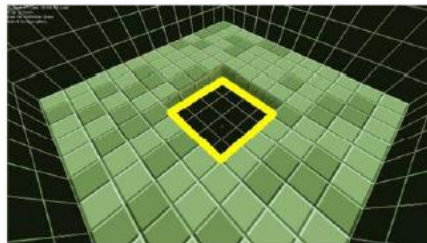
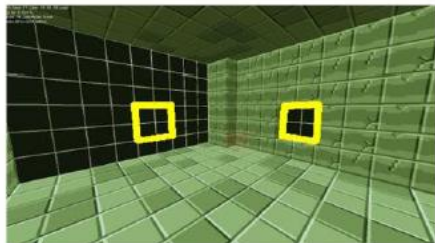
GAME DEVELOPERS CONFERENCE® 2015

MARCH 2-6, 2015 GDCONF.COM



Module design

- Handbuilt voxel modules
- 6 exits = 64 configurations
- Must connect to any other module



部屋の中も自動生成。

制限付き。

Procedural Generation
with restriction.

Room modules are built from voxels in a simple in-engine tool. Each module is 12 x 12 x 8 voxels (at a 1m resolution).

To guarantee that any two adjacent modules are traversable, modules are required to have an opening in at least a safe region (shown here in yellow).

Eldritch のスポウニング

GAME DEVELOPERS CONFERENCE® 2015

MARCH 2-6, 2015 GDCONF.COM

Spawners

- Basic
- Random chance
- Weighted random entity



```
[Spawner_BankArtifacts]
Chance           = 0.75
NumSubSpawners   = 2
SubSpawner0      = "Spawner_Artifact"
SubSpawner0Weight = 2.0
SubSpawner1      = "Spawner_Artifact5"
SubSpawner1Weight = 3.0
```

I could also change what entity was spawned, using a weighted list. In this example, the coins which a player finds in a bank vault have a 75% chance of spawning at all, and a 3:2 chance to be a big pile of coins instead of a single coin.

I also used weighted randomness for enemies. I would place generic enemy AI spawners, and then write a weighted table of enemies to get a balanced distribution of grunts and tougher enemies.

スポーナーをグループ化して管理。

ダンジョンで出て来る
各種の敵、アイテムの数が決まっている。

プレイヤーから一定距離
離れたスポーナーを
ランダムだが、場所によって
優先選択。

GAME DEVELOPERS CONFERENCE® 2015

MARCH 2-6, 2015 GDCONF.COM

Spawner placement

- Loot in dead ends (1 exit)
- Traps in hallways (2 exits)
- Enemies at junctions (3+ exits)

Rendering the World of Far Cry 4

Stephen McAuley | Technical Lead, Ubisoft Montreal

Location: Room 305, South Hall

Date: Friday, March 6

Time: 3:00pm - 4:00pm

Format: Session

Track:  Programming

Pass Type: All Access Pass, Main Conference Pass - **Get your pass now!**

Vault Recording: Video

Audience Level: Intermediate

 Like  Share

Rendering Kyrat, the beautiful world of Far Cry 4, inspired the development of many new graphical features to use the power of the new generation of hardware. First, the foundations were laid down by lighting and material improvements. An anisotropic BRDF was created, high-resolution directional sky occlusion was added, and the global illumination system was extended. Next, the vegetation system was rewritten from scratch, resulting in individual leaves being modelled up close to tessellated, displaced billboards in the distance, complete with physics simulation. Finally, overall image quality was improved through the use of temporal stabilization using motion vectors, SMAA and our "flip quad" super-sampling.

Takeaway

Attendees will learn how to improve the quality of their lighting and shading through techniques such as our anisotropic BRDF. They will discover how to render vegetation at both near and far distances. Finally, they will learn how to implement our AA and temporal stability techniques to enhance the overall image quality of their games.

Intended Audience

The intended audience includes graphics programmers working on both current-gen and cross-generational titles, although technical artists might also find some of the content relevant. Knowledge of physically-based shading and basic computer graphics is required.

FARCRY4 の植物生成

<http://schedule.gdconf.com/session/rendering-the-world-of-far-cry-4>

SpeedTree (middleware)を使用。

VEGETATION

- New vegetation system focused on trees and bushes.
- Create trees using SpeedTree and import into engine:



FARCRY4



Steve McAuley, "Rendering the World of Far Cry 4"
<https://t.co/CuI93nkhqh>

幹と葉は別のデータ

VEGETATION

- Trees consist of a trunk plus leaf clusters:



Everything is rendered with alpha test; no alpha blending is used.

Steve McAuley, "Rendering the World of Far Cry 4"
<https://t.co/CuI93nkhqh>

LODを持つデータとして生成

VEGETATION: LEVEL OF DETAIL

- Trunks and leaf clusters have three LODs before a tree becomes an impostor.
- Leaf cluster LODding generated offline:
 - Calculate percentage of cluster visible:
 - 12 distances.
 - 9 viewpoints.

LOD 0

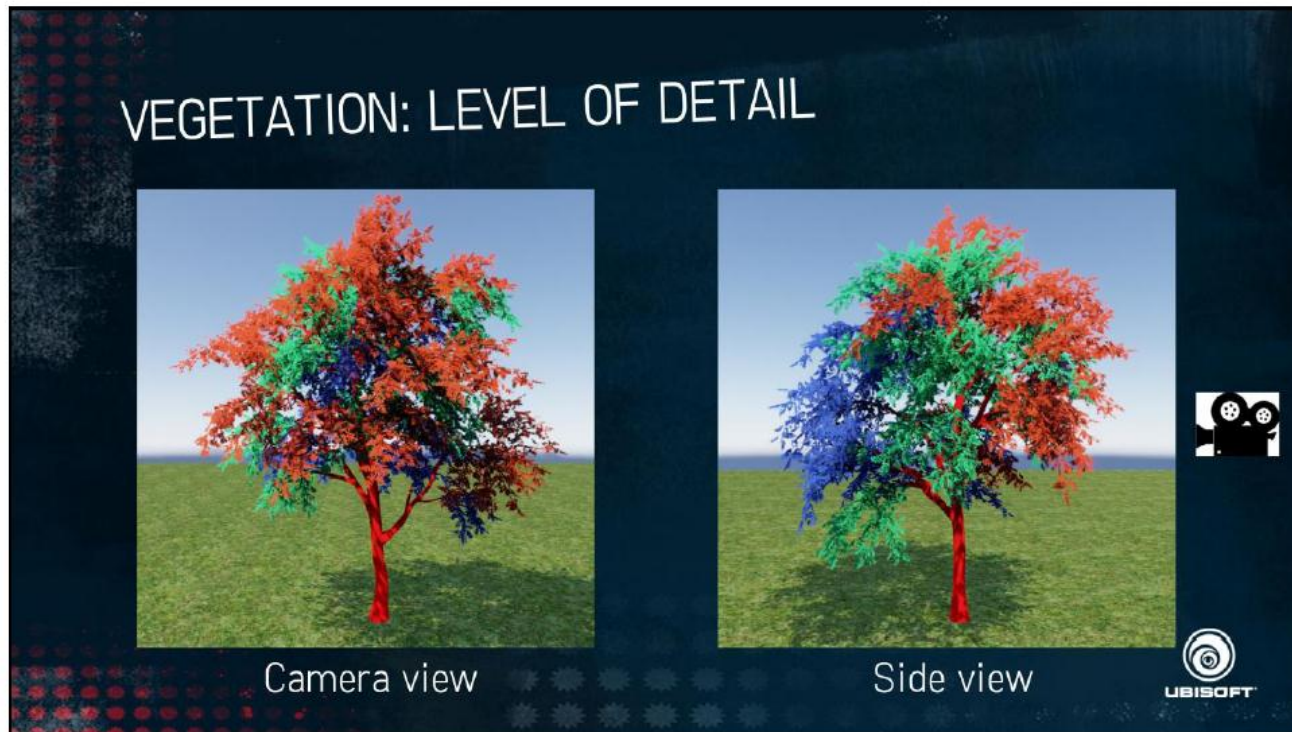
LOD 1

LOD 2



Steve McAuley, "Rendering the World of Far Cry 4"
<https://t.co/Cul93nkhqh>

カメラ位置によるLOD



If we fix our culling camera then look around the side, we'll see that leaf clusters at the back of the tree have lower LODs.

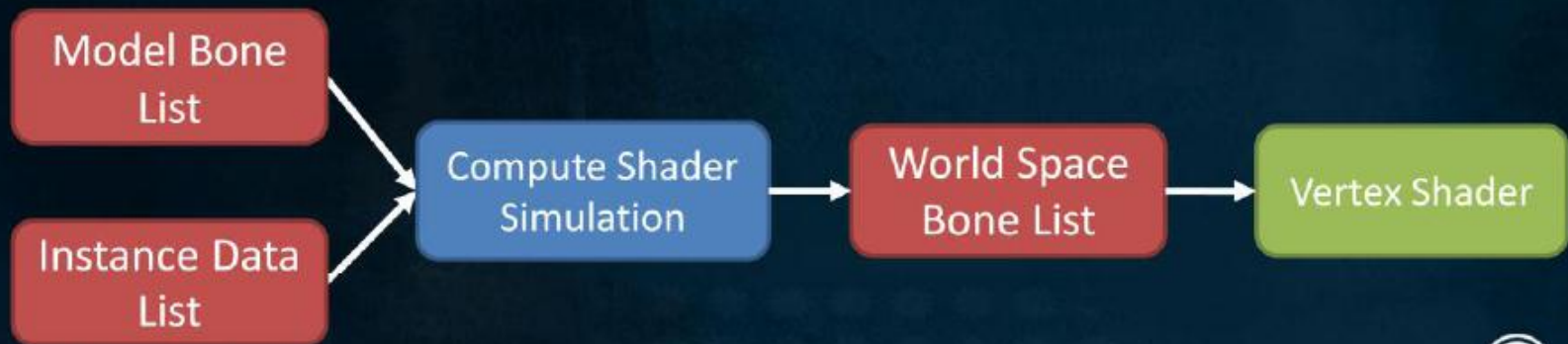
Obviously, please remember that we generate these leaf cluster LODs not just at various viewpoints around the camera, but also at different distances too, so far away the whole tree would turn blue.

Steve McAuley, "Rendering the World of Far Cry 4"
<https://t.co/Cul93nkhqh>

ワールドヘインスタンス

VEGETATION: SIMULATION

- Physics and movement simulated on GPU.
- Instanced bones transformed into world space bones.



Steve McAuley, "Rendering the World of Far Cry 4"
<https://t.co/Cul93nkhqh>

Parallelizing the Naughty Dog Engine Using Fibers

Christian Gyrling | Lead Programmer, Naughty Dog

Location: Room 306, South Hall

Date: Wednesday, March 4

Time: 5:00pm - 6:00pm

Format: Session

Track:  Programming

Pass Type: All Access Pass, Main Conference Pass - **Get your pass now!**

Vault Recording: Video

Audience Level: All

 Like  Share

This talk is a detailed walkthrough of the game engine modifications needed to make The Last of Us Remastered run at 60 fps on PlayStation 4. Topics covered will include the fiber-based job system Naughty Dog adopted for the game, the overall frame-centric engine design, the memory allocation patterns used in the title, and our strategies for dealing with locks.

Takeaway

Useful low-level technologies that will allow for better use of multi-core hardware by minimizing contention and maximizing core occupancy.

Intended Audience

Programmers with influence over the overall direction of their company's game engine(s).

NAUGHTY DOG ENGINE

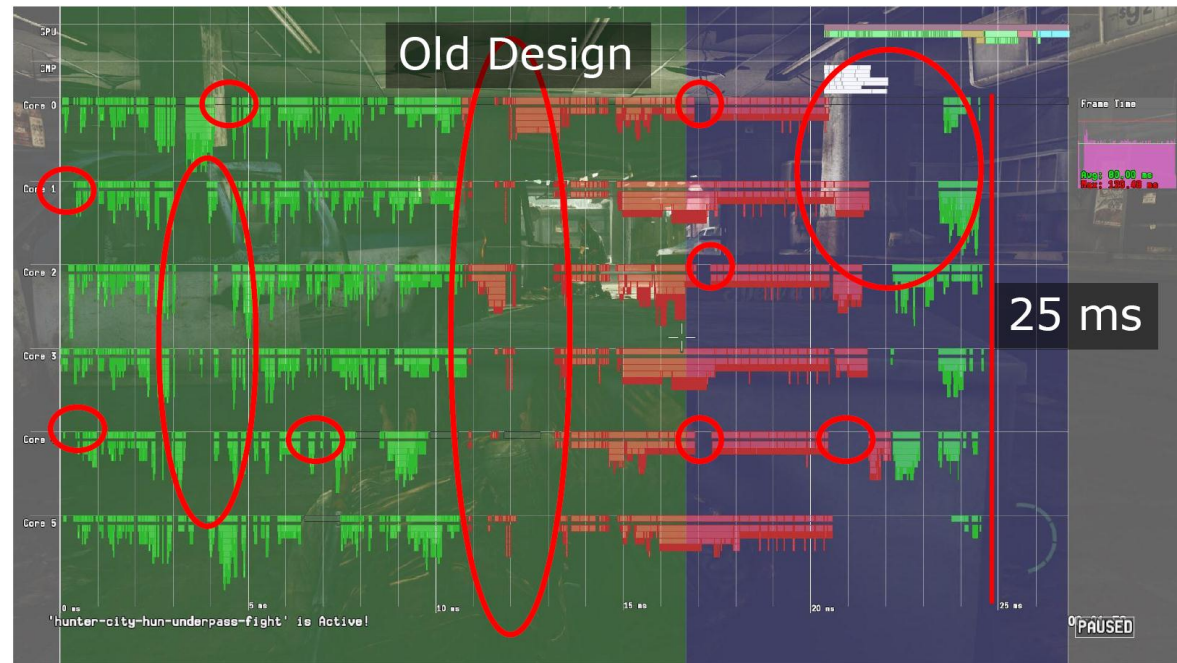
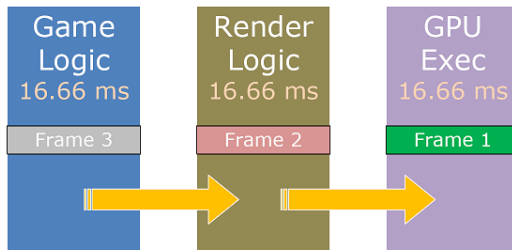
GDC 2015 Presentation – Parallelizing the Naughty Dog Engine

<http://www.swedishcoding.com/2015/03/08/gdc-2015-presentation-parallelizing-the-naughty-dog-engine/>

<http://schedule.gdconf.com/session/parallelizing-the-naughty-dog-engine-using-fibers>

最初はシリアルに処理

Engine pipeline – Feed forward

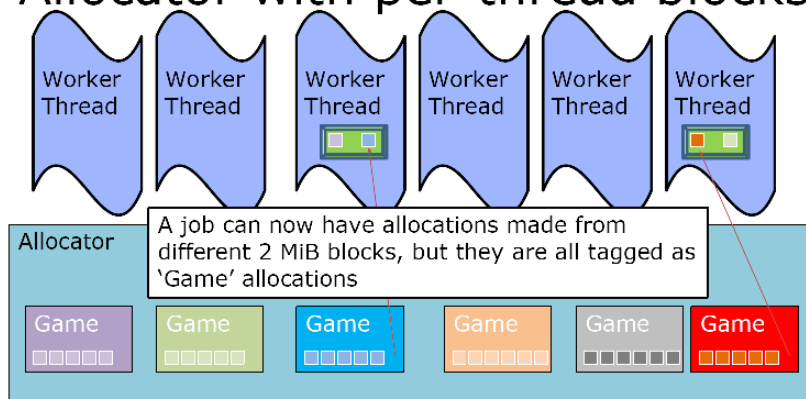


並列化する



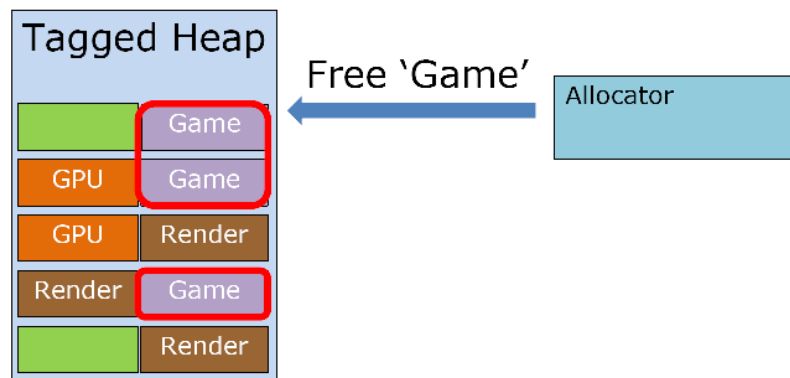
アロケーションとワーカースレッド

Allocator with per-thread blocks



アロケーターが
「ゲーム」データを
2MiB毎に準備して、
ワーカースレッドが
処理して行く。

Bulk free



Goal-Oriented Action Planning: Ten Years Old and No Fear!

Chris Conway | Lead AI Engineer, Crystal Dynamics

Peter Higley | Senior Software Engineer, Monolith Productions

Eric Jacopin | Professor, Center of Research, Saint-Cyr Military Academy

Location: Room 306, South Hall

Date: Tuesday, March 3

Time: 10:00am - 11:00am

Format: Session

Track:  AI Summit

Pass Type: All Access Pass, Summits, Tutorials & Bootcamps Pass - **Get your pass now!**

Vault Recording: Video

 Tweet  Like  Share

Monolith released their game F.E.A.R. in 2005. At GDC 2006, Jeff Orkin showed us how his Goal-Oriented Action Planner system (GOAP) powered the AI for F.E.A.R. in a way that was new and fresh. In the past 10 years since, GOAP has had its ups and downs, still trying to find its way among the traditional AI algorithms. While architecturally simple, it has had to adapt to handle the demands of each new generation of AAA games. This multi-studio lecture will show how Monolith has continued to grow their implementation of GOAP in their games, such as the recent Shadow of Mordor. Additionally, we will see how Crystal Dynamics handles debugging and reporting of the (often novel or unexpected) GOAP behaviors in their Tomb Raider titles. Lastly, Éric Jacopin will show how data analytics can be used for AI components such as GOAP to gather metrics and patterns.

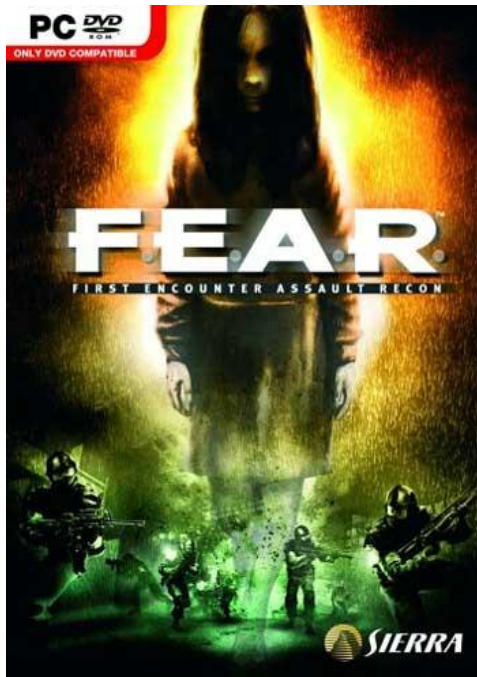
Takeaway

Attendees will see peeks into how GOAP has evolved over the past 10 years, its current state, and how it can be used in their own titles.

プランニング技術

<http://schedule.gdconf.com/session/goal-oriented-action-planning-ten-years-old-and-no-fear>

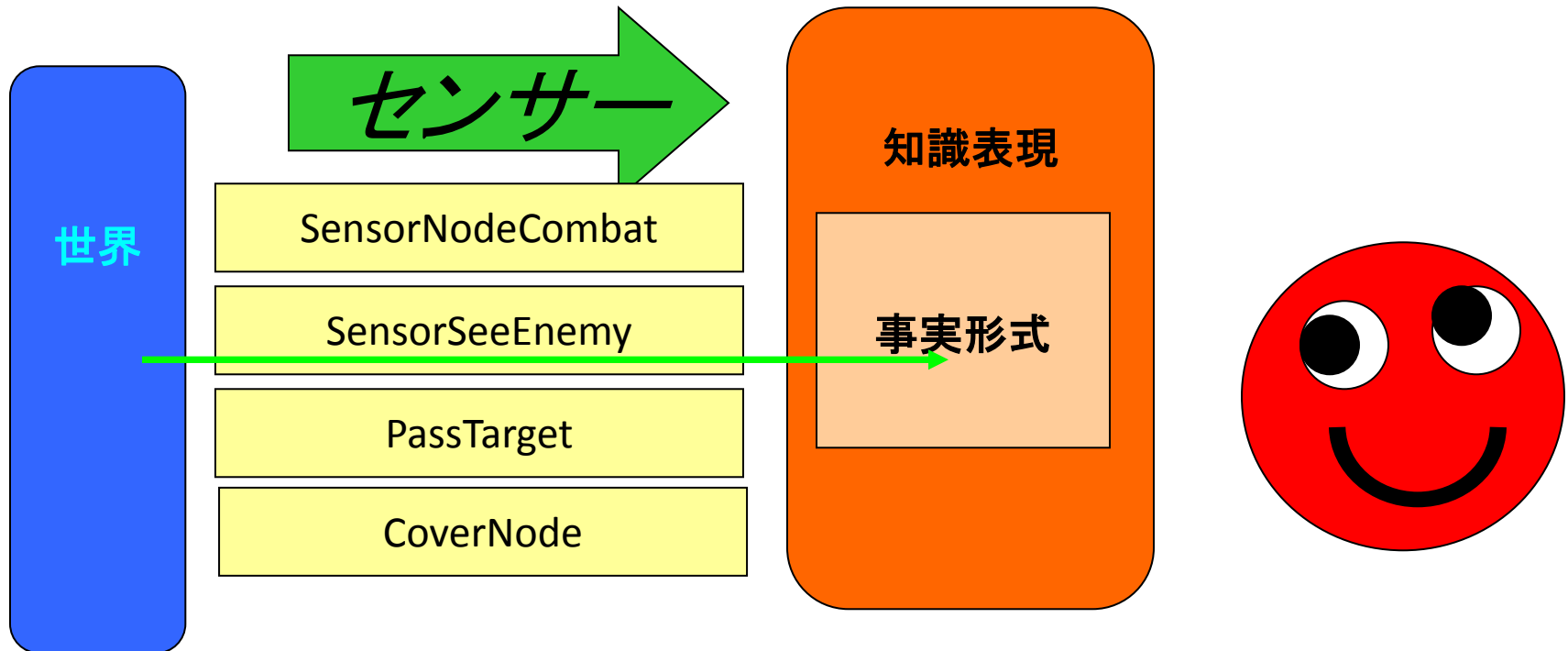
F.E.A.R.におけるゴール指向プランニング



- ◆ Genre: Horror FPS
- ◆ Developer: Monolith Production
- ◆ Publisher: SIERRA
- ◆ Hardware: Windows
- ◆ Year: 2004

F.E.A.R のCOMの感覚(センサー)

センサー = 五感からの情報、及び、五感から得られるはずの情報を模擬する機能



レイキャスト(視線チェック)、パス検索など重たい処理からなる関数

SensorNodeCombat 隠れる、或いは、隠れながら攻撃できる場所を探す。
SensorSeeEnemy 敵が見えるかチェック
PassTarget 戦術ポイントまでのパスを見つけ、かつそのパスが敵から安全であることをチェックする。
CoverNode 隠れることができるノード

統一事実記述形式

キャラクター

オブジェクト

任務

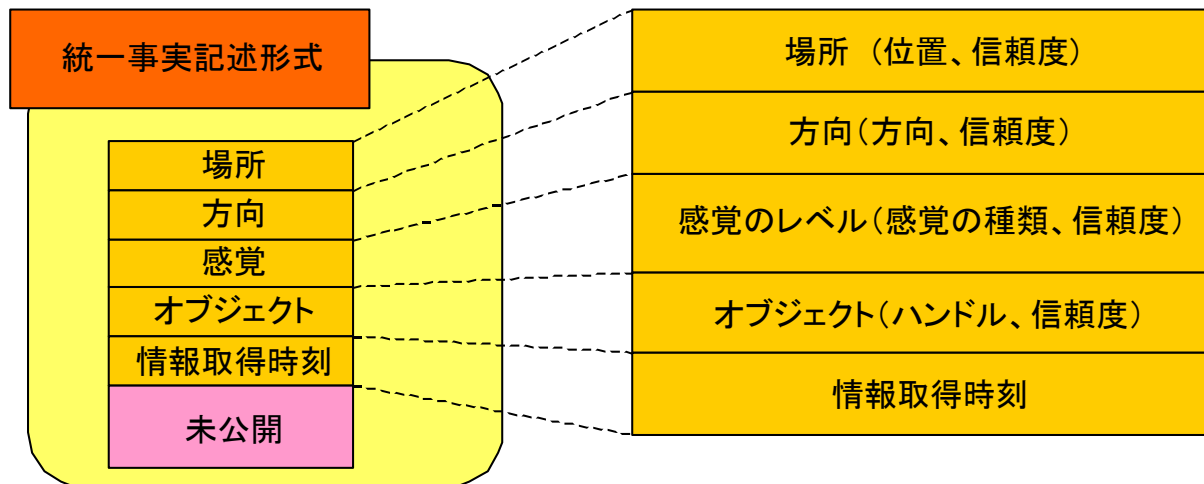
事件

パス

欲求

ノード

全て以下の形式(フォーマット)で記述する。

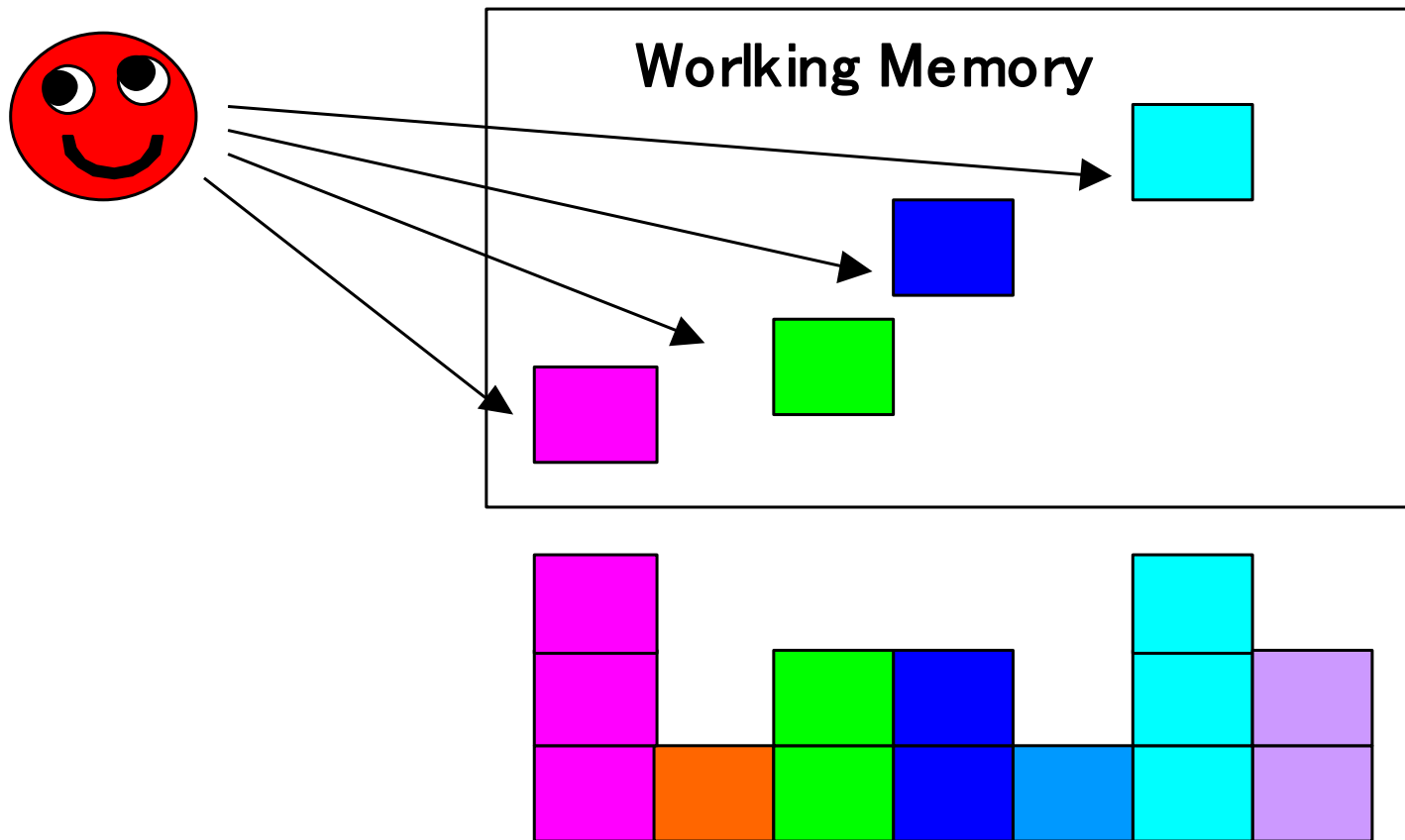


```
WorkingMemoryFact
{
    Attribute<Vector3D>    Position
    Attribute<Vector3D>    Direction
    Attribute<StimulusType> Stimulus
    Attribute<Handle>      Object
    Attribute<float>       Desire
    ...
    float                  fUpdateTime
}
```

全部で本当は16個の属性がある

```
Attribute<Type>
{
    Type  Value
    float fConfidence
}
```

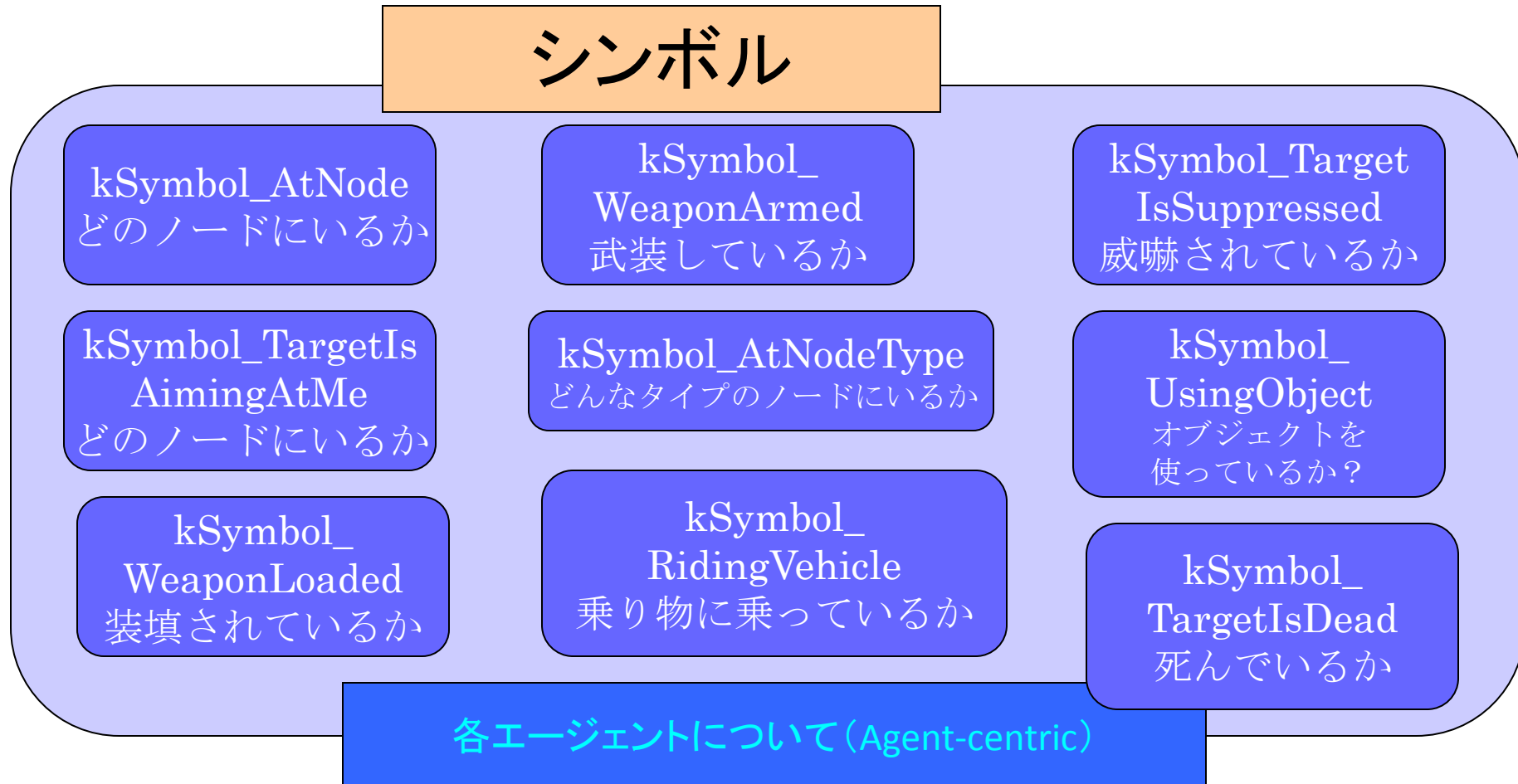
自分の記憶領域に認識した事実を蓄積する



プランニングのための知識

エージェントの認識する世界をもっとシンプルに表現したい

➡ 20個のシンボルで世界を集約して表現する



上記のシンボルは、対象とするエージェントについての情報。

kTargetIsDead = true
この兵士Aは死んだ

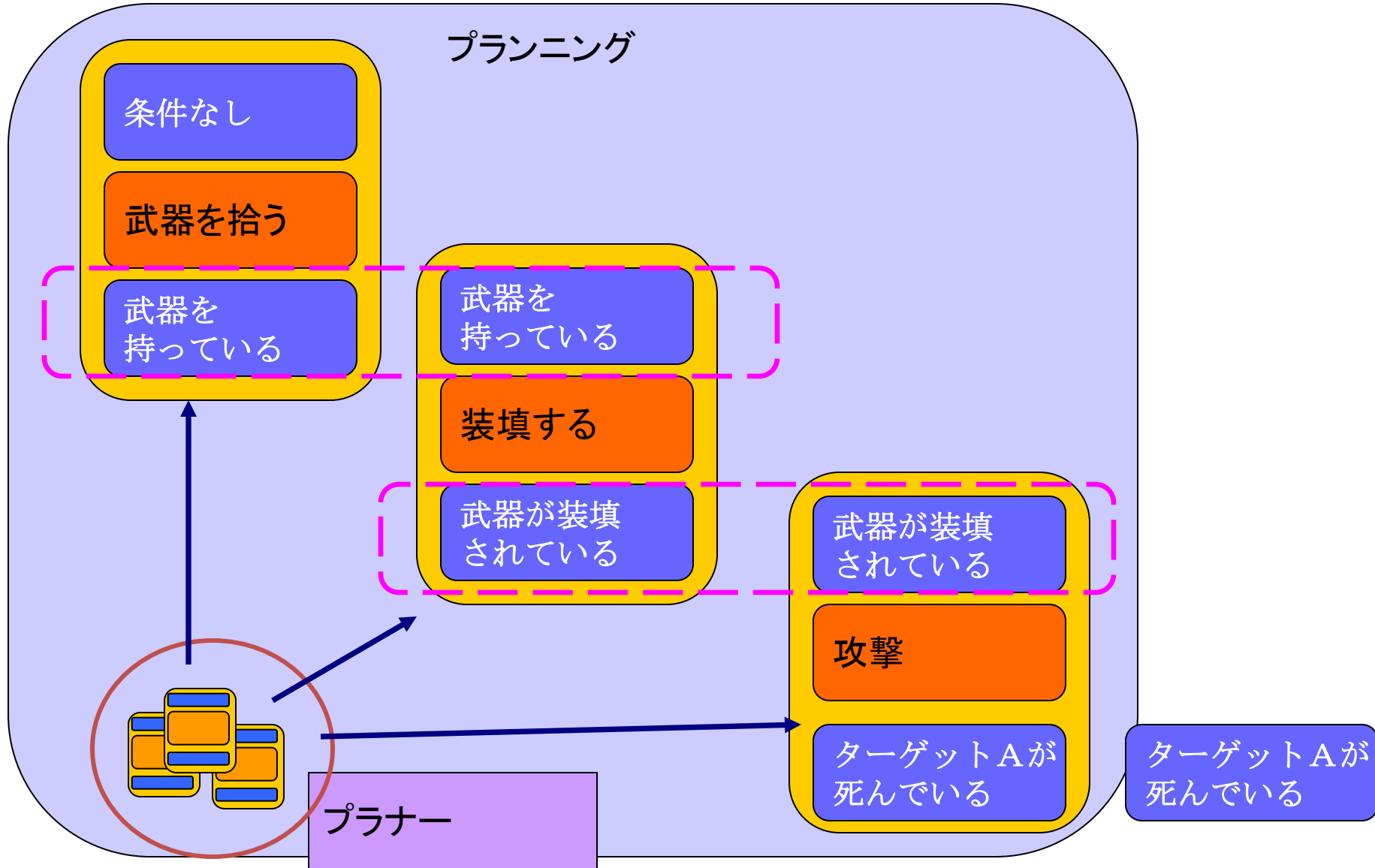
シンボル

kTargetAtme = true
この兵士Bは自分を狙っている



kWeaponIsLoaded = false
私Cの武器は装填済みでない

F.E.A.R.のプランニング(=GOAP) シンボルによる連鎖プランニング



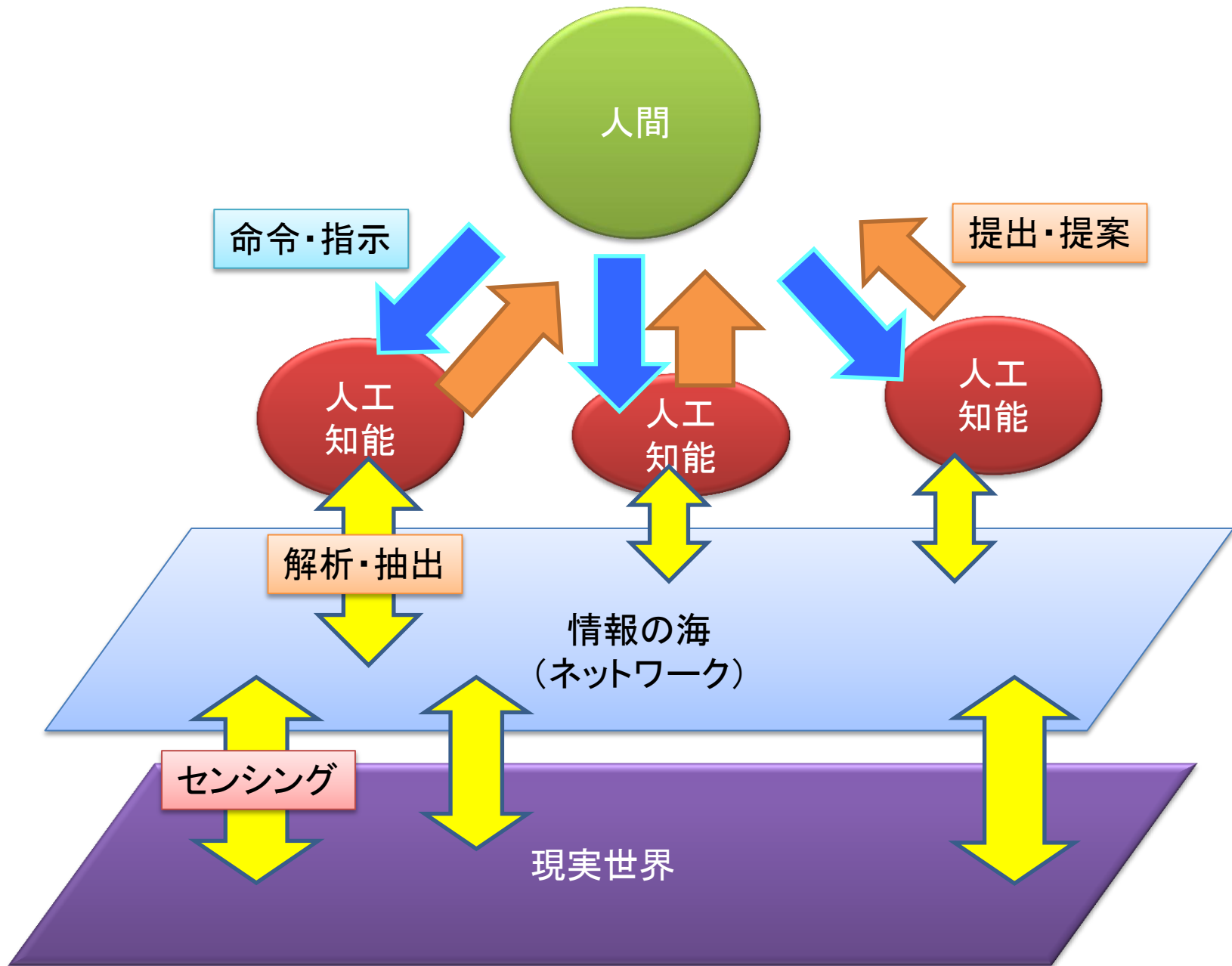
Tomb Raider の事例

- ビヘイビアレベルのGOAPを導入

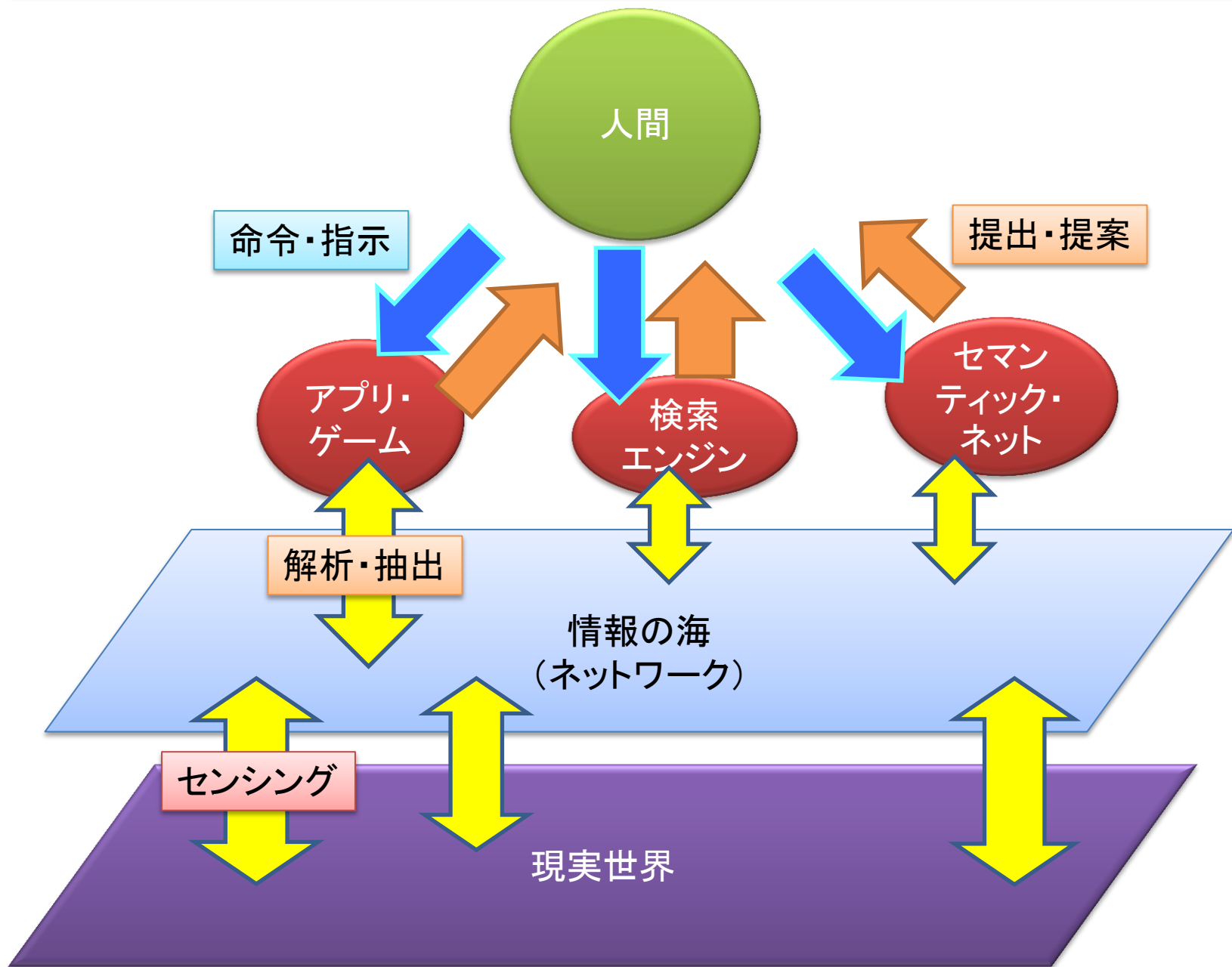
まとめ

これからのAIの動向

ネット空間の人工知能



ネット空間の人工知能



現実とバーチャルの境界線

ネット空間

純粋な
ソフト
ウェア

普通の
ソフトウェア

現実空間（物理空間）

ヴァーチャル空間/現実空間の境界

現実とバーチャルの境界線

ネット空間

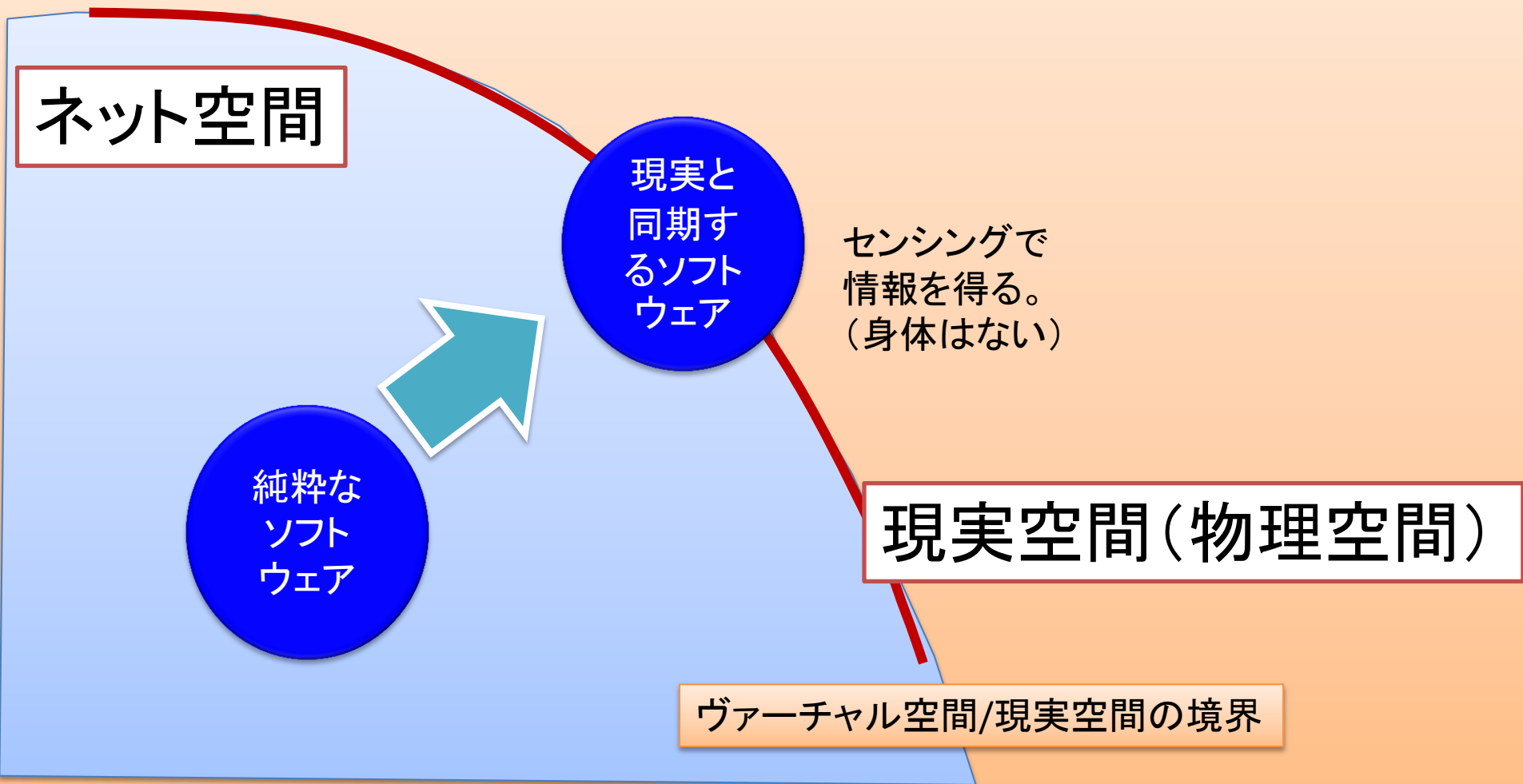
現実と
同期す
るソフト
ウェア

センシングで
情報を得る。
(身体はない)

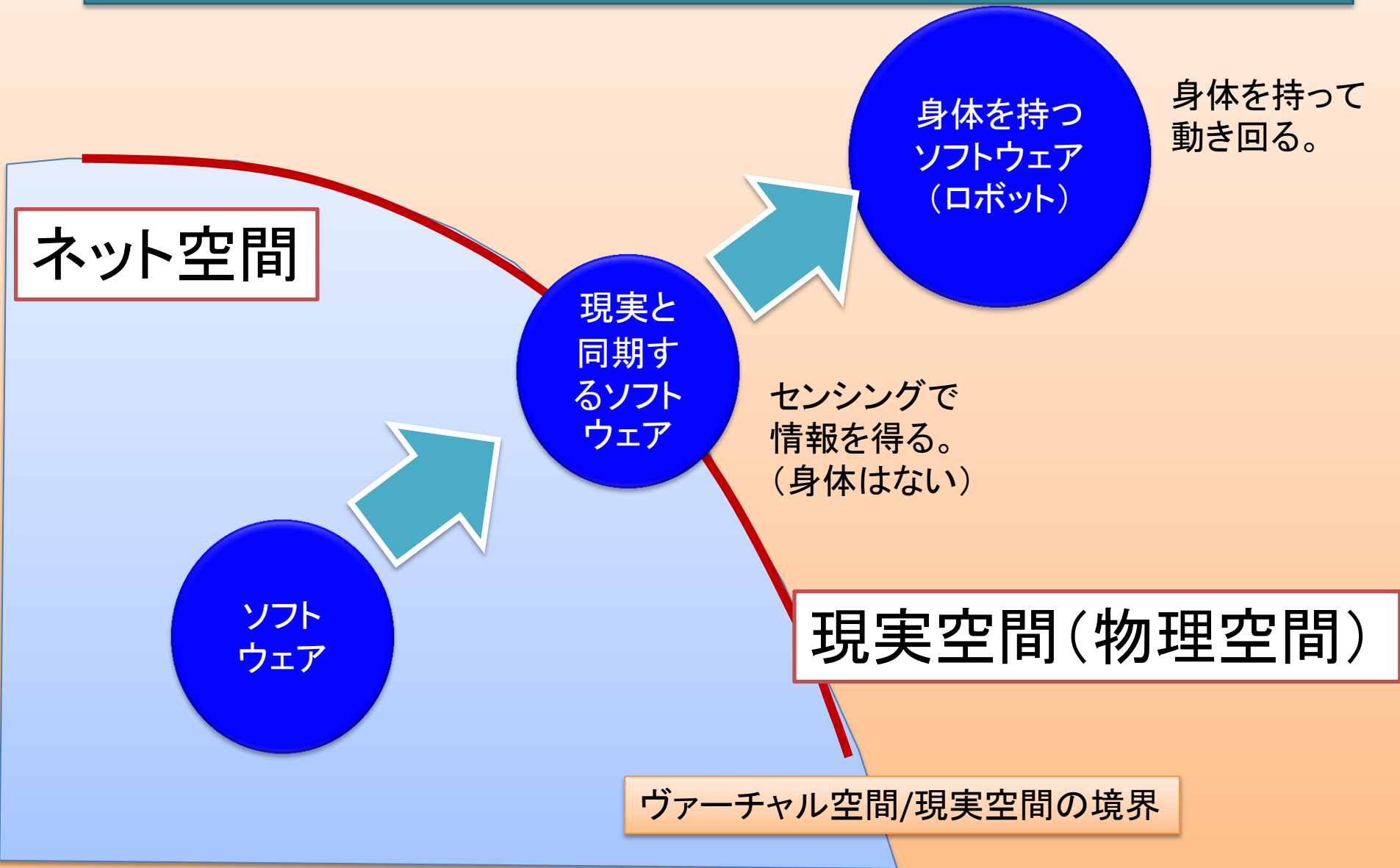
純粋な
ソフト
ウェア

現実空間(物理空間)

ヴァーチャル空間/現実空間の境界



現実とバーチャルの境界線



現実とバーチャルの境界線

身体を持つ
ソフトウェア

身体を持って
動き回る。

ヴァーチャ

ソフトウェアは身体を持ち、
現実空間を動き回るようになる。

これは一つのソフトウェアの
進化の形である。

ソフト
ウェア

現実空間（物理空間）

ヴァーチャル空間/現実空間の境界

現在起こっていること ～ネット空間から現実空間への回帰

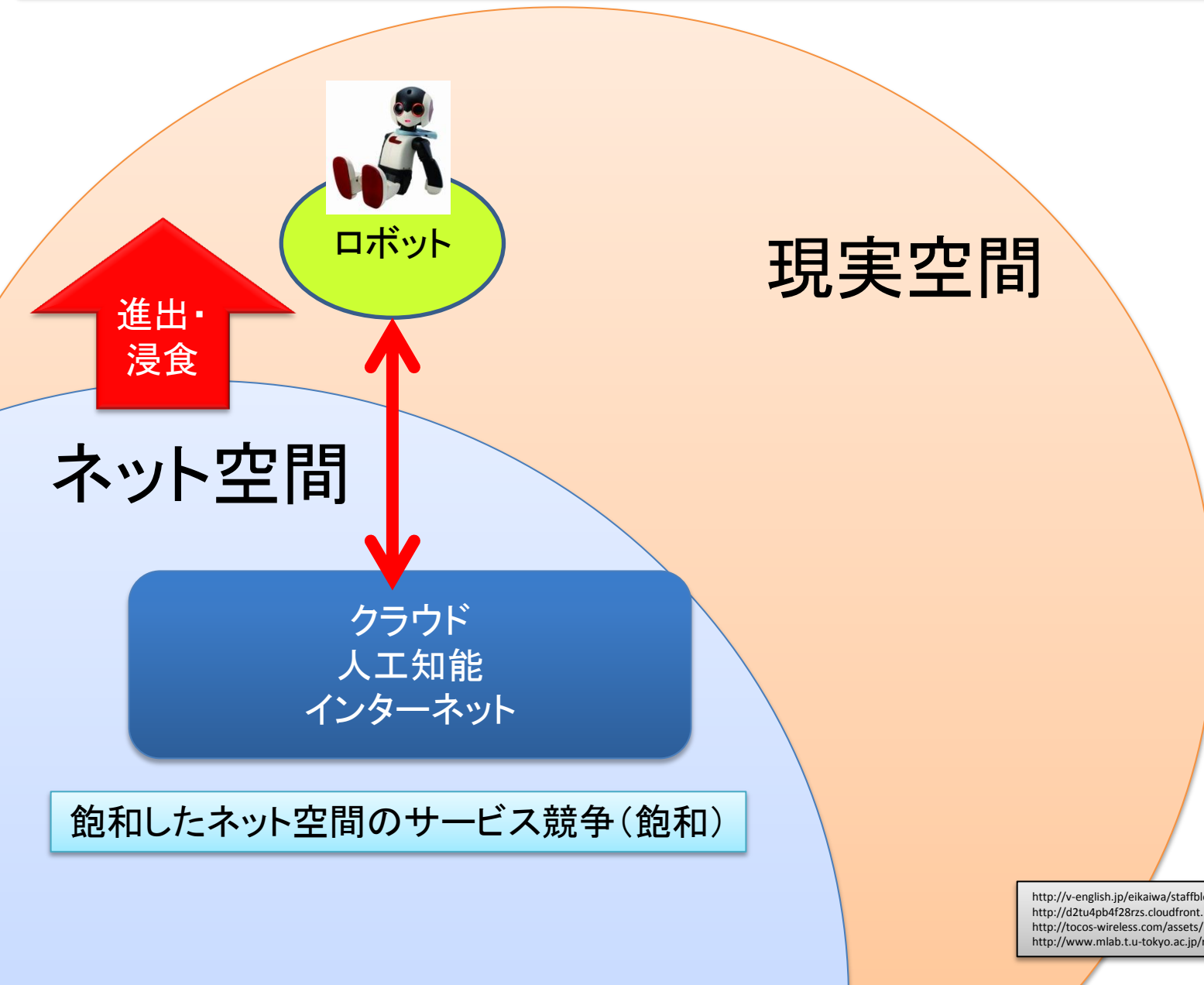
現実空間

ネット空間

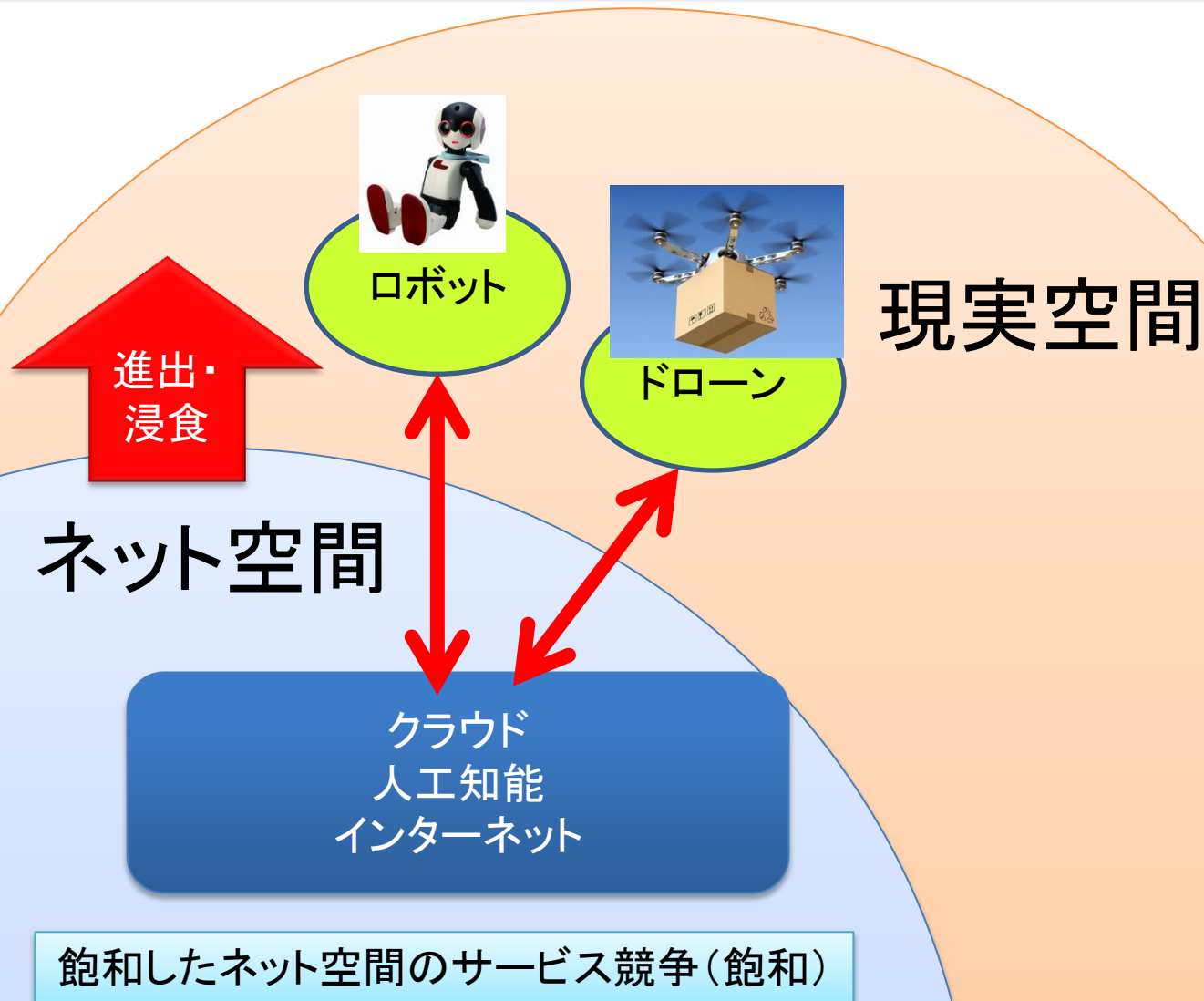
クラウド
人工知能
インターネット

飽和したネット空間のサービス競争(飽和)

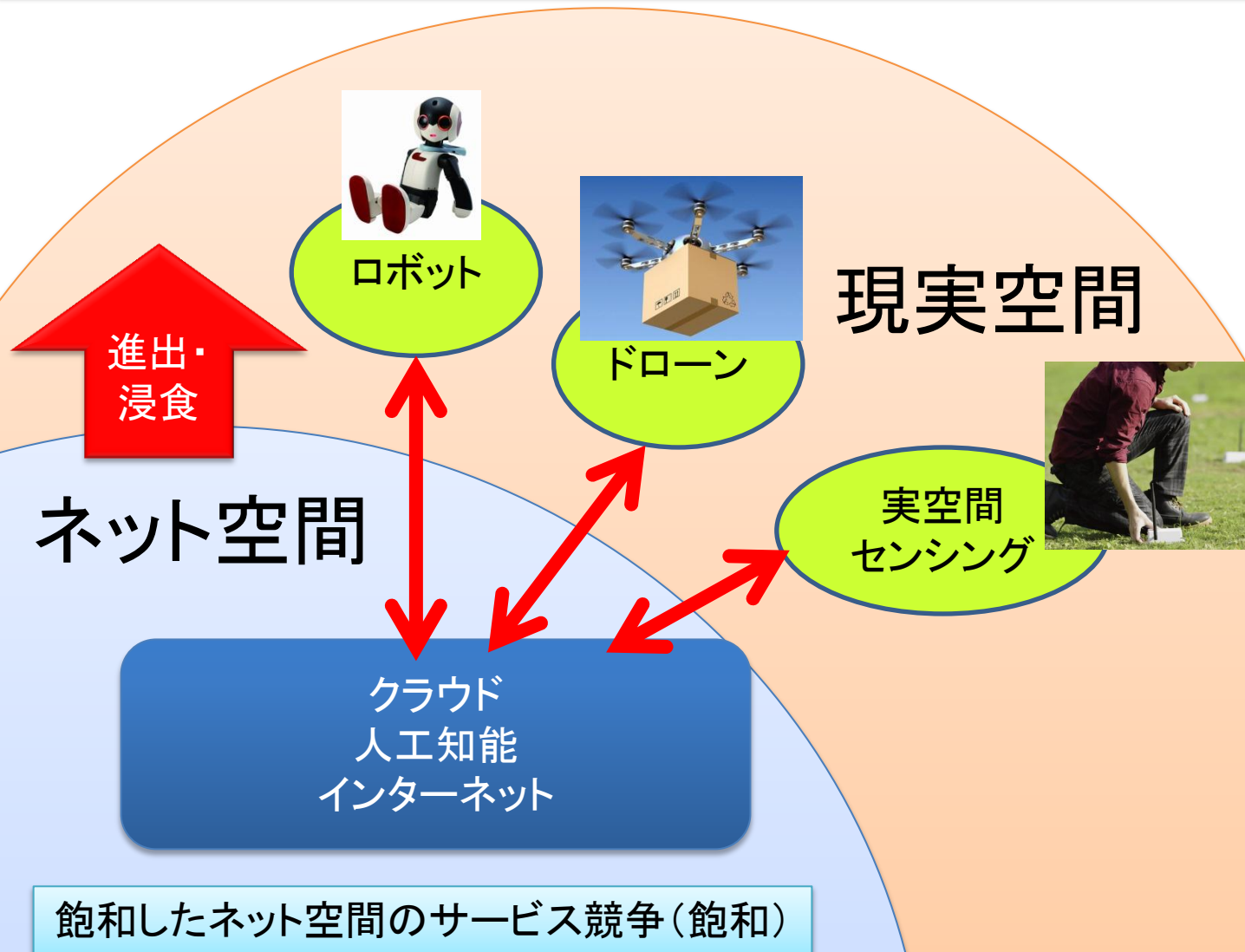
現在起こっていること ～ネット空間から現実空間への回帰



現在起こっていること ～ネット空間から現実空間への回帰

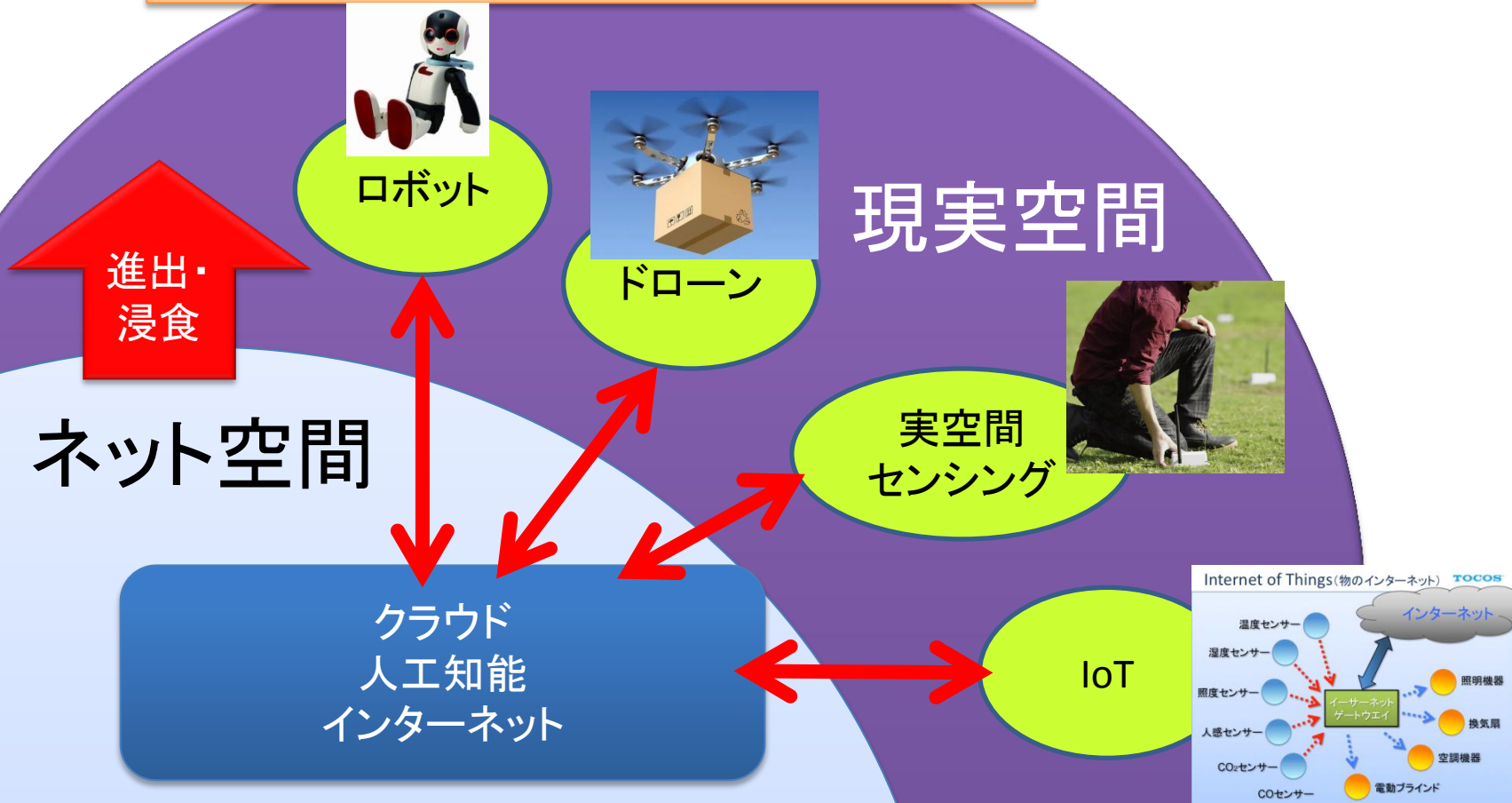


現在起こっていること ～ネット空間から現実空間への回帰



現在起こっていること ～ネット空間から現実空間への回帰

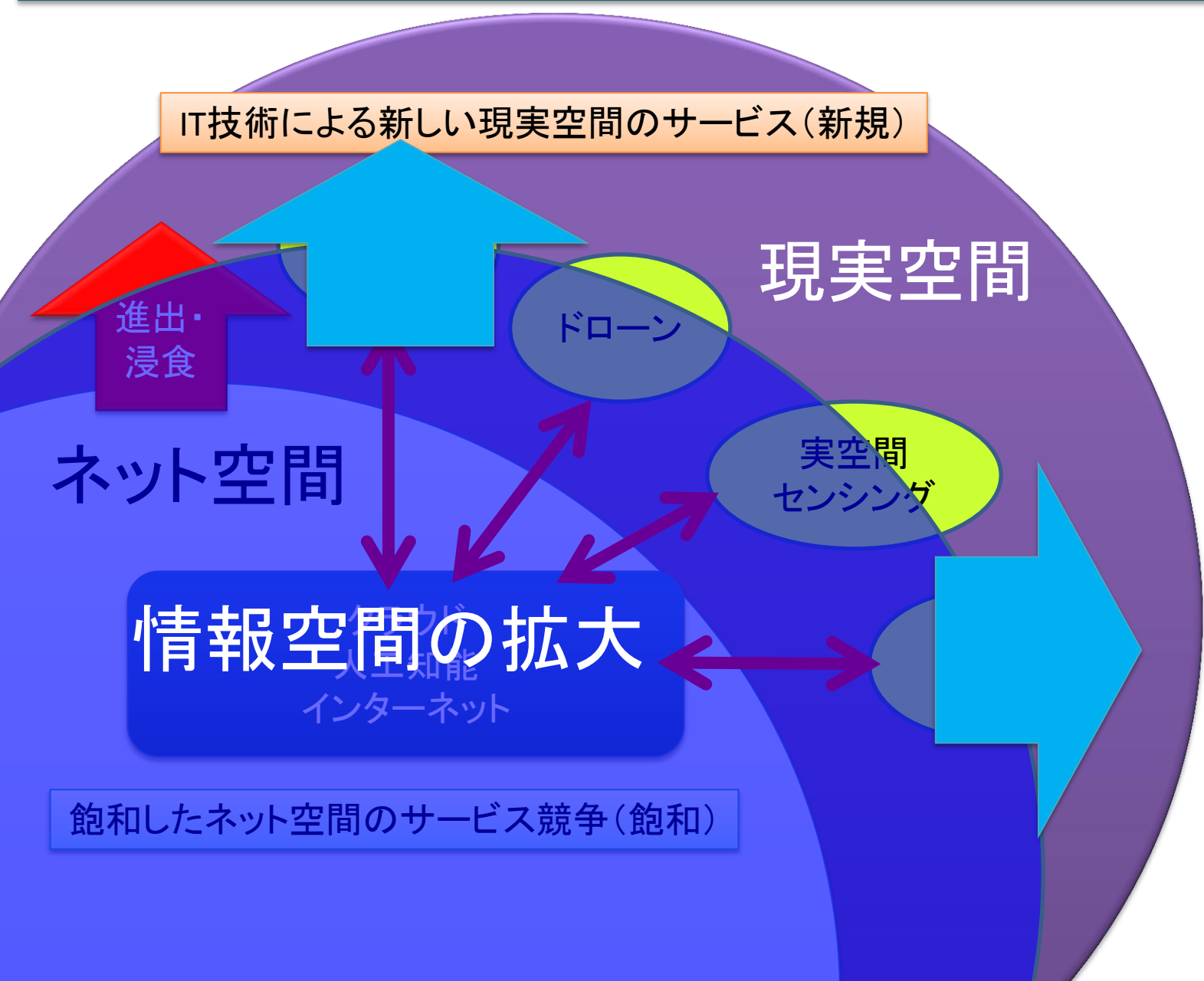
IT技術による新しい現実空間のサービス(新規)



飽和したネット空間のサービス競争(飽和)

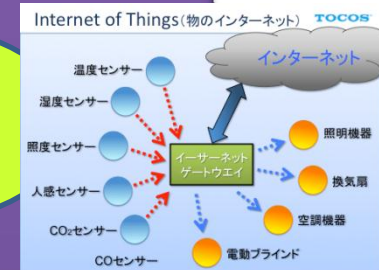
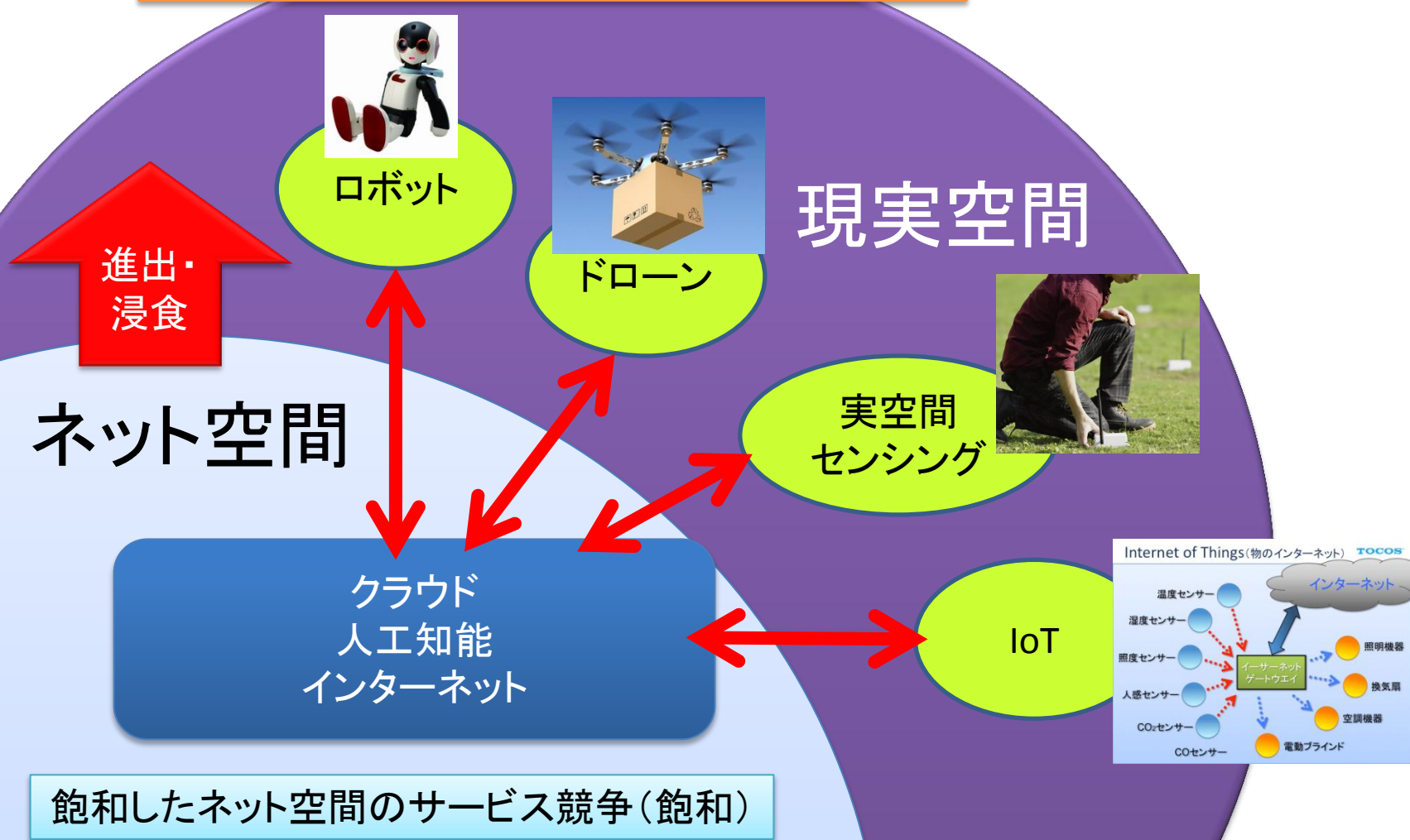
http://v-english.jp/eikaiwa/staffblog/files/2014/07/20140710_fb.jpg
http://d2tu4pb4f28rzs.cloudfront.net/site_img/photo/2013-03-04-055321.jpg
<http://tocos-wireless.com/assets/loT/InternetofThings.png>
http://www.mlab.t.u-tokyo.ac.jp/research/2014/network/img/index_ss.png

現在起こっていること ～ネット空間から現実空間への回帰

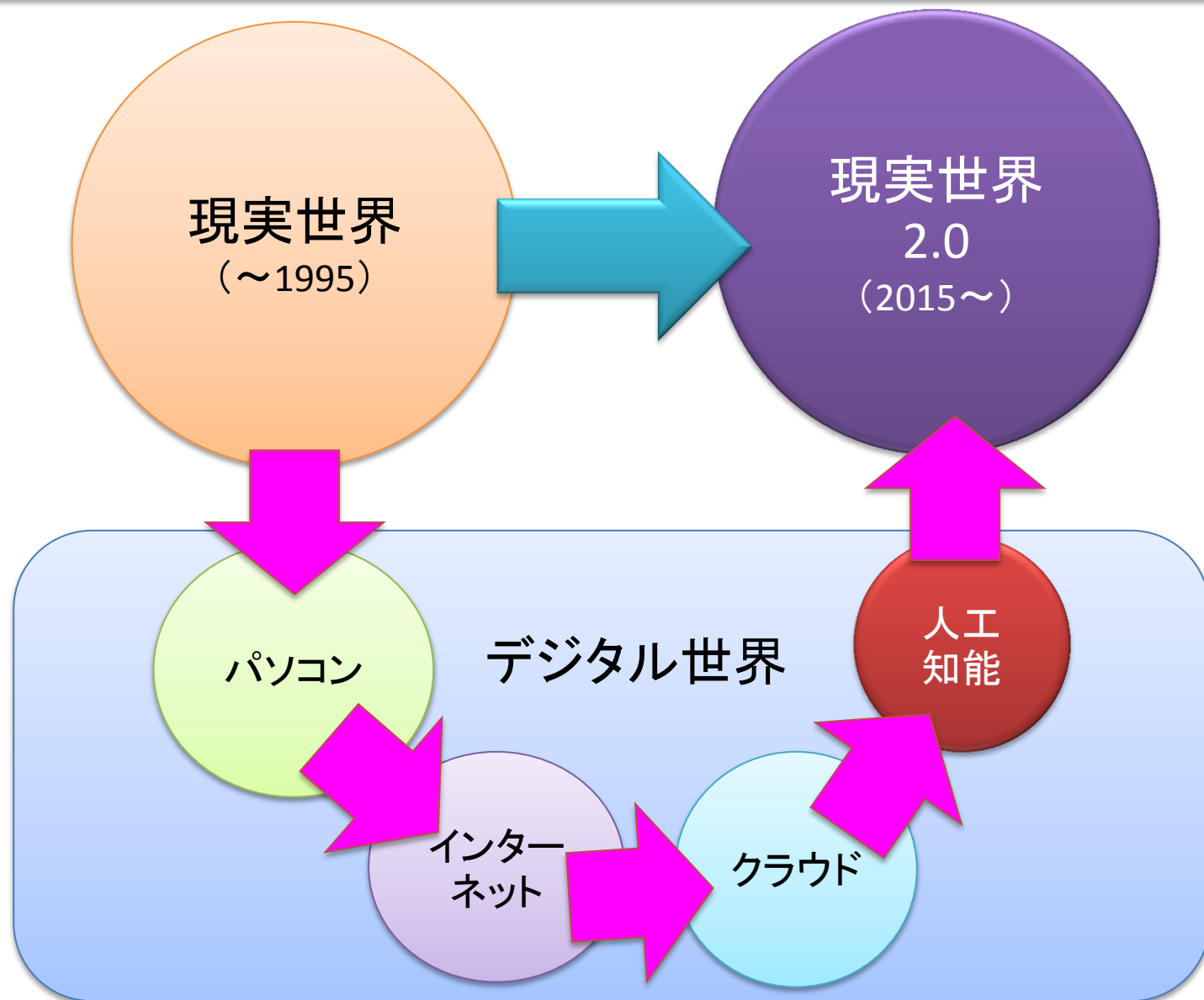


現在起こっていること ～ネット空間から現実空間への回帰

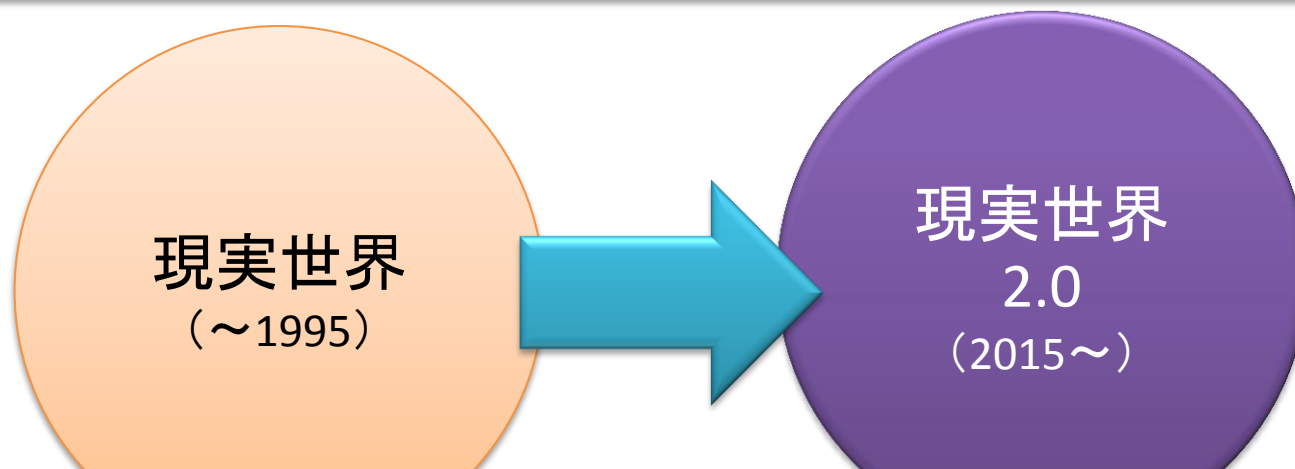
IT技術による新しい現実空間のサービス(新規)



人工知能による新しい空間



人工知能による新しい空間

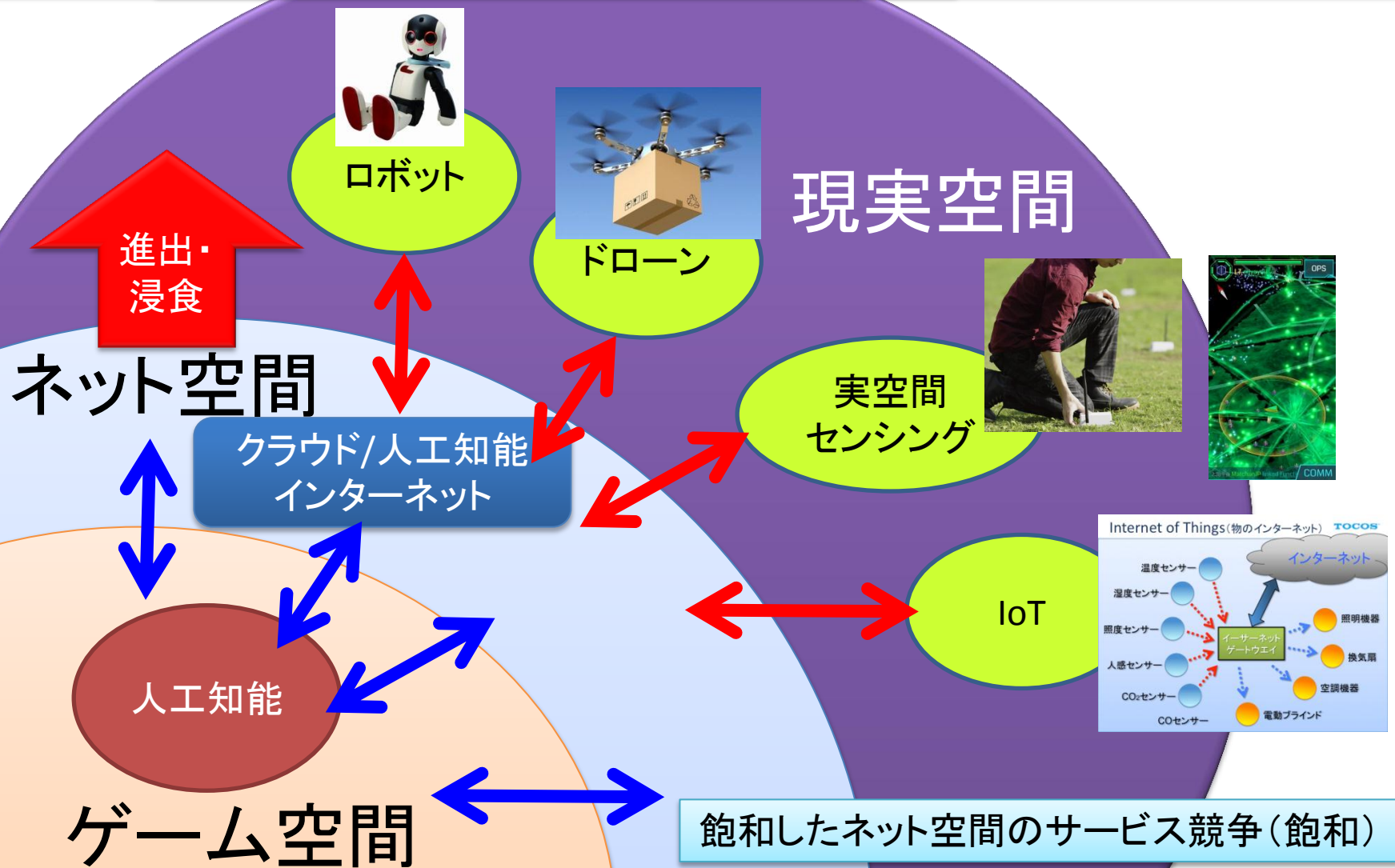


キーワードは「実世界指向」

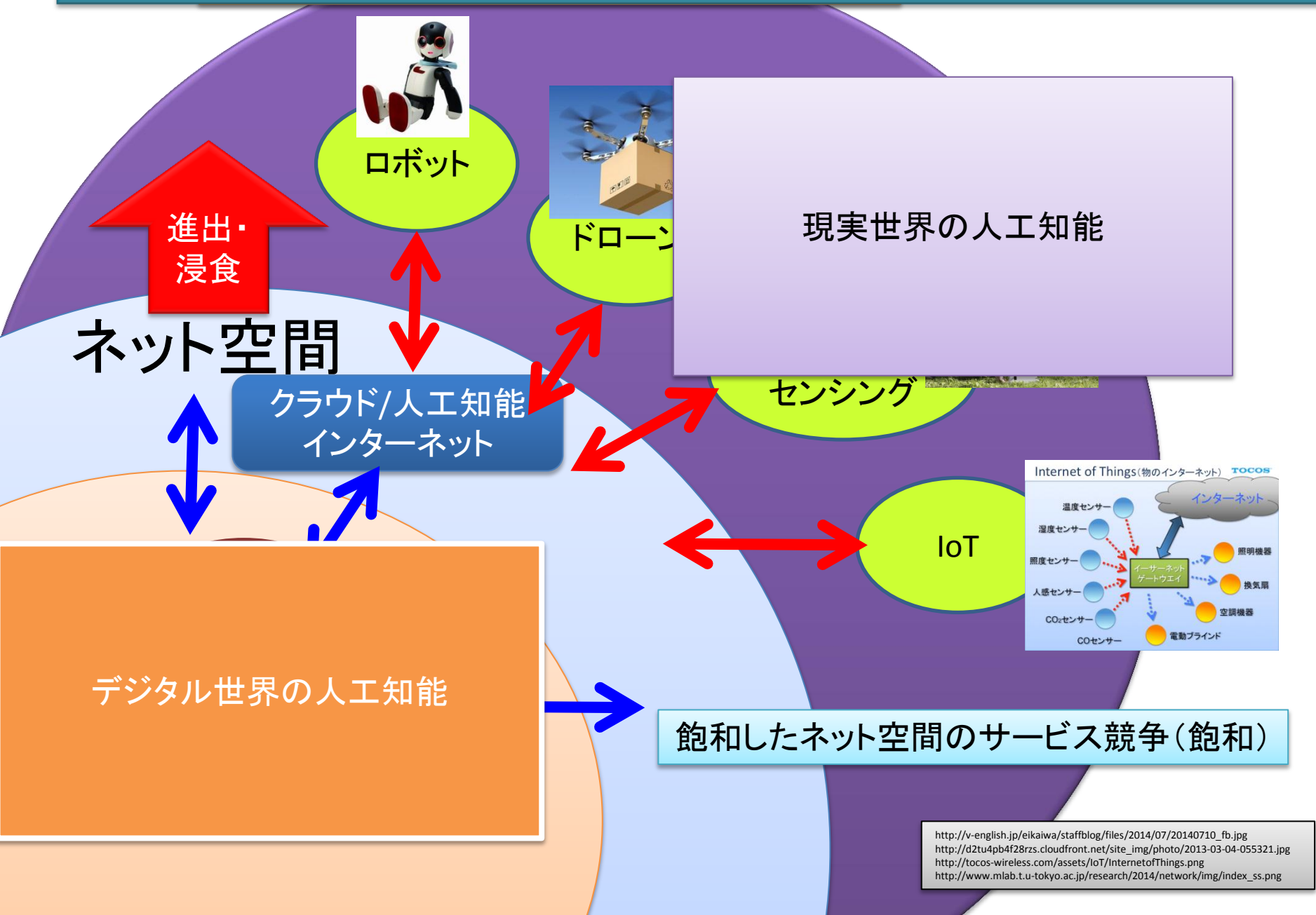
ソフトウェアは身体を持って現実に出る。
それは現実世界を変貌させて行く。

「現実世界 2.0」

現在起こっていること ～ ネット空間から現実空間への回帰



ネット空間から現実空間へ、現実世界からネット空間へ



ネット空間から現実空間へ、現実世界からネット空間へ



ロボット

進出・
浸食

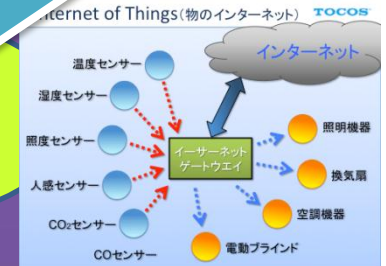
ネット空間

現実世界の人工知能

未開拓

ゲーム空間

IoT



デジタル世界の人工知能
成熟

飽和したネット空間のサービス競争(飽和)

現実世界とデジタル世界をまたぐゲーム空間



ロボット

進出・
浸食

ネット空間

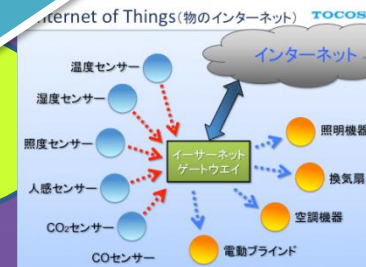
現実世界の人工知能

未開拓

ゲーム空間

デジタル世界の人工知能
成熟

飽和したネット空間のサービス競争(飽和)



深層学習とは何か？

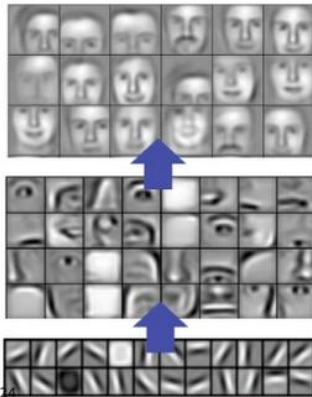
DEEP LEARNING

Deep Learning = 多層型ニューラルネットワーク

深層学習の場合 顔の学習の例

- 深いレイヤーはパーツを組み合わせた全体の学習

深いレイヤー
全体的な特徴
より抽象的なもの



- 浅いレイヤーは単純なパーツを学習

浅いレイヤー
小さい部品
より具体的なもの

16

Preferred Infrastructure

表現学習をとことんやったら？

DistBelief [J. Dean+, NIPS 13]

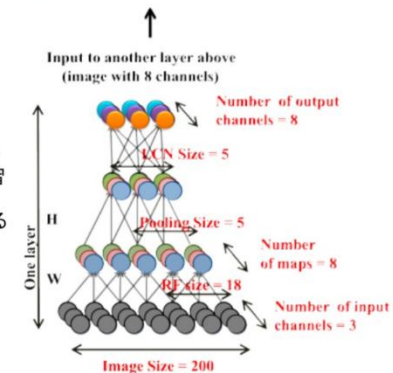
非同期での大規模学習をサポート

- オンラインの並列学習 (SGD)
- バッチ学習の並列学習 (L-BFGS)
- 16000コアでの並列学習が可能

Youtubeから得られた 200 x 200 の画像 1000万枚に対して教師無し学習

(AutoEncoderと似たRICAと呼ばれる方法でDNNを学習)

次項以降で学習で得られた最上部のニューロンが反応する画像を紹介



19

Preferred Infrastructure

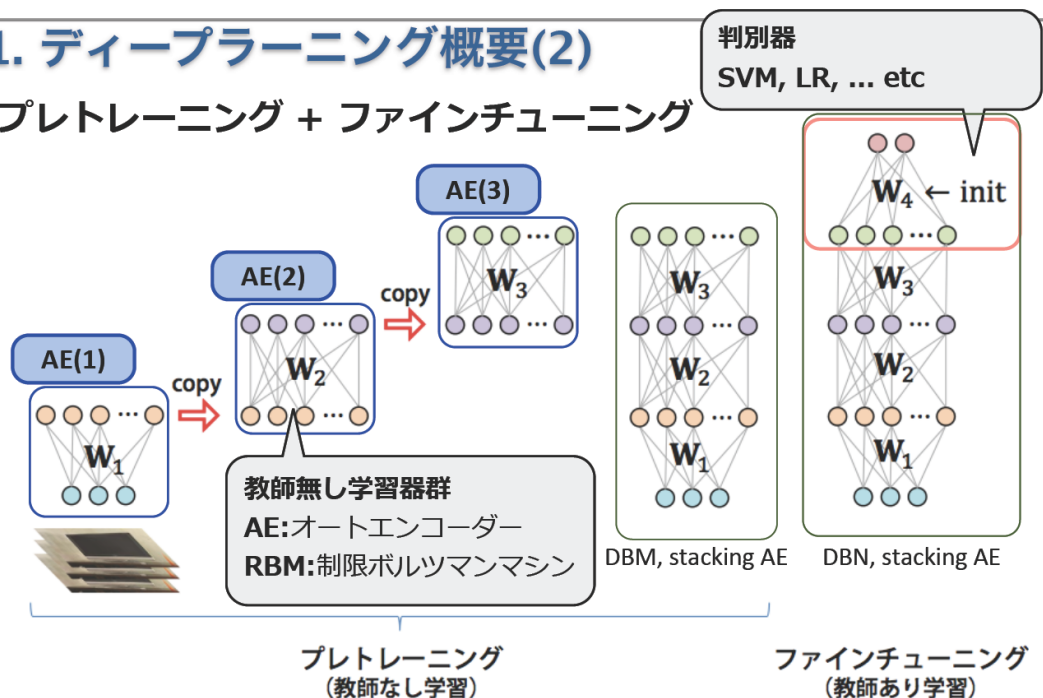
岡野原大輔「一般向けのDeep Learning」

<http://www.slideshare.net/pfi/deep-learning-22350063>

階層型ニューラルネット＋階層型学習

1. ディープラーニング概要(2)

プレトレーニング＋ファインチューニング



http://www.vision.is.tohoku.ac.jp/files/9313/6601/7876/CVIM_tutorial_deep_learning.pdf

Analysis

7

Yurie Oka「実装ディープラーニング」

<http://www.slideshare.net/yurieoka37/ss-28152060>

Deep Q-Learning

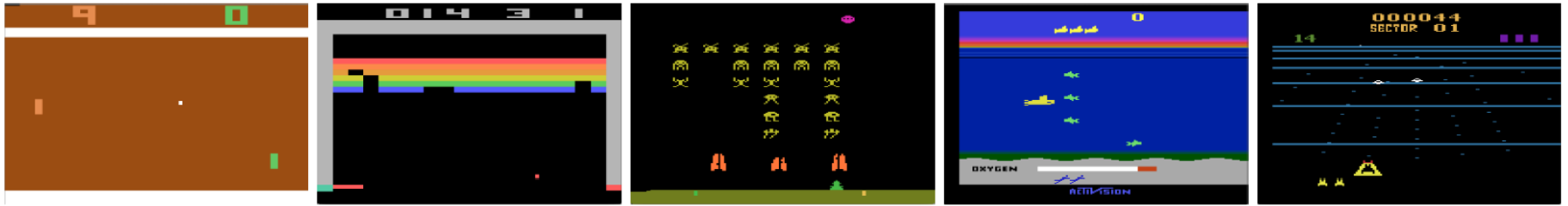


Figure 1: Screen shots from five Atari 2600 Games: (Left-to-right) Pong, Breakout, Space Invaders, Seaquest, Beam Rider

画面を入力
操作はあらかじめ教える
スコアによる強化学習

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves,
Ioannis Antonoglou, Daan Wierstra, Martin Riedmiller (DeepMind Technologies)

Playing Atari with Deep Reinforcement Learning

<http://www.cs.toronto.edu/~vmnih/docs/dqn.pdf>

操作と点数による強化学習アルゴリズム

Algorithm 1 Deep Q-learning with Experience Replay

Initialize replay memory $\mathcal{D}$ to capacity N

Initialize action-value function Q with random weights

for episode = 1, M **do**

 Initialize sequence $s_1 = \{x_1\}$ and preprocessed sequenced $\phi_1 = \phi(s_1)$

for $t = 1, T$ **do**

 With probability ϵ select a random action a_t

 otherwise select $a_t = \max_a Q^*(\phi(s_t), a; \theta)$

 Execute action a_t in emulator and observe reward r_t and image x_{t+1}

 Set $s_{t+1} = s_t, a_t, x_{t+1}$ and preprocess $\phi_{t+1} = \phi(s_{t+1})$

 Store transition $(\phi_t, a_t, r_t, \phi_{t+1})$ in $\mathcal{D}$

 Sample random minibatch of transitions $(\phi_j, a_j, r_j, \phi_{j+1})$ from $\mathcal{D}$

 Set $y_j = \begin{cases} r_j & \text{for terminal } \phi_{j+1} \\ r_j + \gamma \max_{a'} Q(\phi_{j+1}, a'; \theta) & \text{for non-terminal } \phi_{j+1} \end{cases}$

 Perform a gradient descent step on $(y_j - Q(\phi_j, a_j; \theta))^2$ according to equation 3

end for

end for

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves,
Ioannis Antonoglou, Daan Wierstra, Martin Riedmiller (DeepMind Technologies)

Playing Atari with Deep Reinforcement Learning

<http://www.cs.toronto.edu/~vmnih/docs/dqn.pdf>

學習過程解析

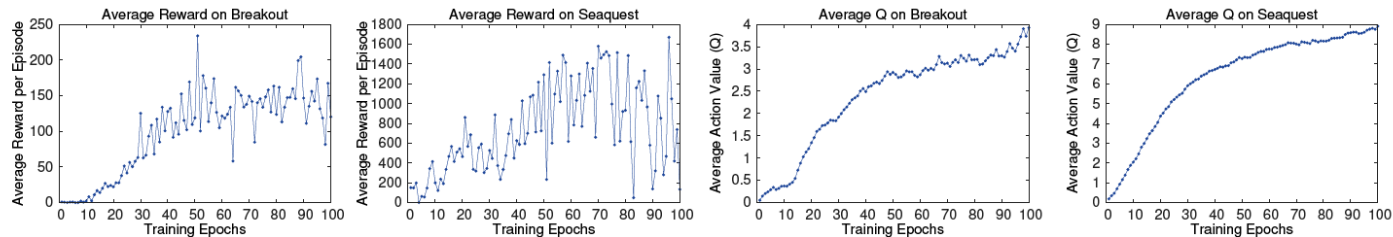


Figure 2: The two plots on the left show average reward per episode on Breakout and Seaquest respectively during training. The statistics were computed by running an ϵ -greedy policy with $\epsilon = 0.05$ for 10000 steps. The two plots on the right show the average maximum predicted action-value of a held out set of states on Breakout and Seaquest respectively. One epoch corresponds to 50000 minibatch weight updates or roughly 30 minutes of training time.



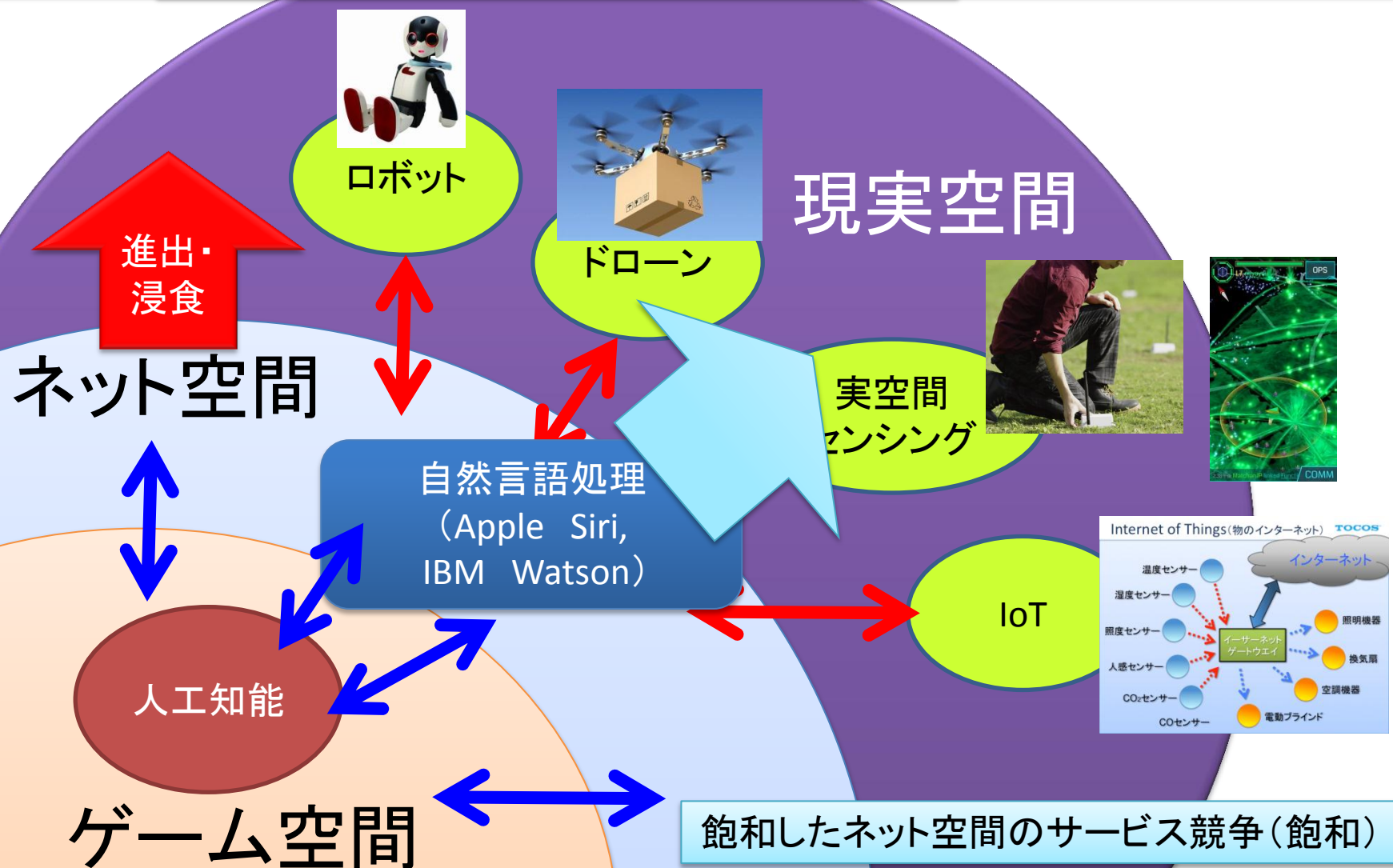
Figure 3: The leftmost plot shows the predicted value function for a 30 frame segment of the game Seaquest. The three screenshots correspond to the frames labeled by A, B, and C respectively.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves,
Ioannis Antonoglou, Daan Wierstra, Martin Riedmiller (DeepMind Technologies)

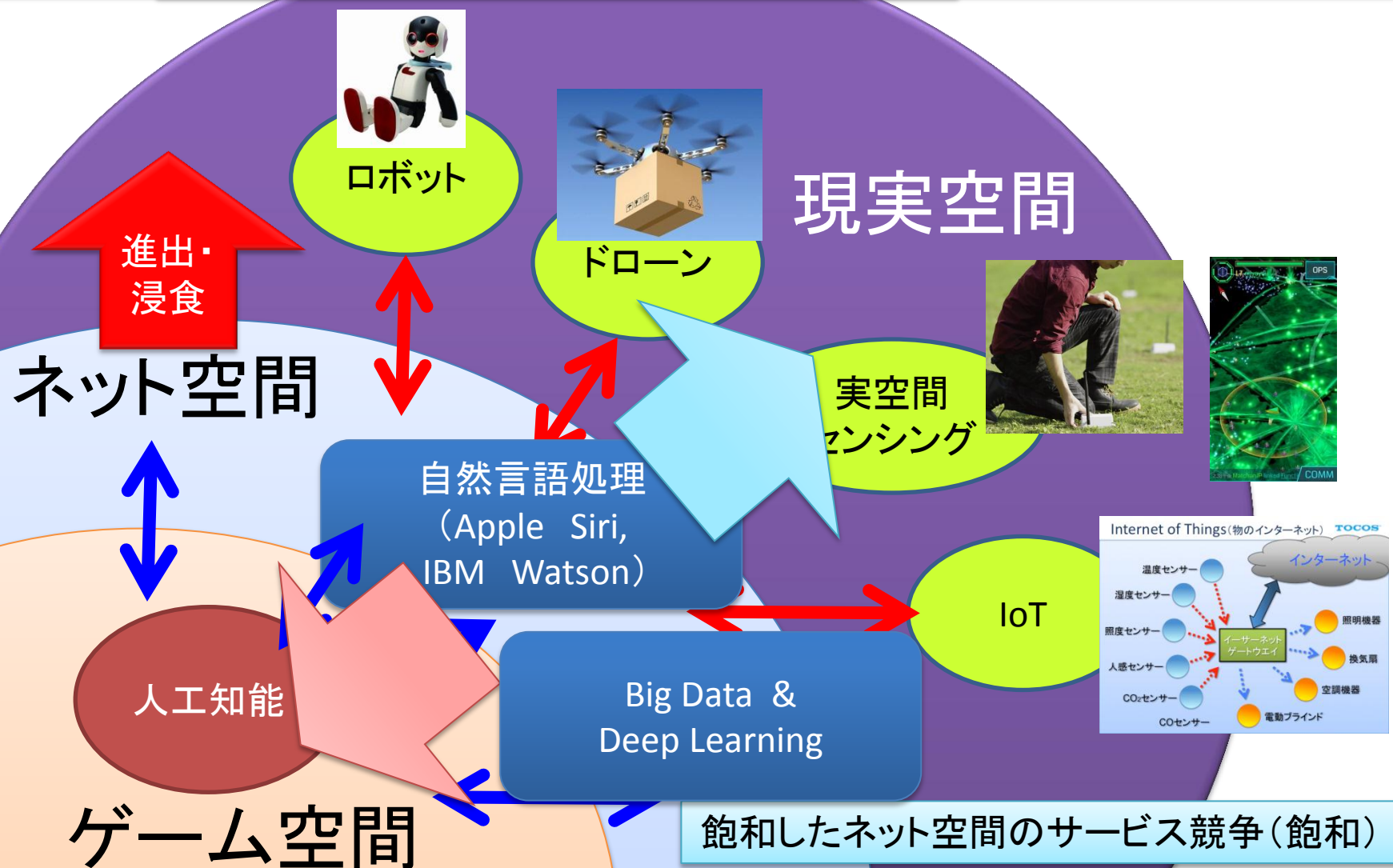
Playing Atari with Deep Reinforcement Learning

<http://www.cs.toronto.edu/~vmnih/docs/dqn.pdf>

現在起こっていること ～ ネット空間から現実空間への回帰



現在起こっていること ～ ネット空間から現実空間への回帰



参考文献

三宅陽一郎

「デジタルゲームにおける人工知能技術の応用の現在」

(人工知能学会誌 2015年 Vol.30 No.1)

https://jsai.ixsq.nii.ac.jp/ej/?action=pages_view_main&active_action=repository_view_main_item_detail&item_id=559&item_no=1&page_id=13&block_id=8



https://jsai.ixsq.nii.ac.jp/ej/?action=pages_view_main&active_action=repository_view_main_item_detail&item_id=556&item_no=1&page_id=13&block_id=8

参考文献

三宅陽一郎 「はじめてのゲームAI」
(技術評論社 WEB+DB PRESS、2012年、Vol.68)



<http://gihyo.jp/magazine/wdpress/archive/2012/vol68>

My GDC Photo on facebook note

- <https://www.facebook.com/notes/youichiro-miyake/gdc2015-%E5%88%B0%E7%9D%80-%E4%B8%80%E6%97%A5%E7%9B%AE-%E6%8A%95%E7%A8%BF%E3%81%BE%E3%81%A8%E3%82%81/10153138006412943>
- <https://www.facebook.com/notes/youichiro-miyake/gdc2015-%E4%BA%8C%E6%97%A5%E7%9B%AE-%E6%8A%95%E7%A8%BF%E3%81%BE%E3%81%A8%E3%82%81/10153137929267943>
- <https://www.facebook.com/notes/youichiro-miyake/gdc%E4%B8%89%E6%97%A5%E7%9B%AE-%E6%8A%95%E7%A8%BF%E3%81%BE%E3%81%A8%E3%82%81/10153140314822943>
- <https://www.facebook.com/notes/youichiro-miyake/gdc2015-%E5%9B%9B%E6%97%A5%E7%9B%AE-%E6%8A%95%E7%A8%BF%E7%9B%AE%E3%81%BE%E3%81%A8%E3%82%81/10153144027357943>
- <https://www.facebook.com/notes/youichiro-miyake/gdc-2015-%E4%BA%94%E6%97%A5%E7%9B%AE-%E6%8A%95%E7%A8%BF%E3%81%BE%E3%81%A8%E3%82%81/10153144905262943>

<https://www.facebook.com/youichiro.miyake>