



箱庭・試行の軌跡と新たなチャレンジ

～箱庭はなぜ技術屋の心を揺さぶるのか～



森 崇

(永和システムマネジメント)

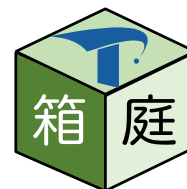


自己紹介

- 森 崇(もり たかし)
- 職歴/専門分野
 - Linuxファイルシステム/カーネル周りのお仕事(10年)
 - 組込みRTOS/ROS周りのお仕事(10年)
 - 新規ビジネス向けWebシステム開発(昨年から～)
- TOPPERS/箱庭WG参加・開発(4年)
 - 組込みソフト/マイコンシミュレータ担当
 - 箱庭コア機能担当
 - 物理シミュレーション/ビジュアライズ
 - その他いろいろ



ROS

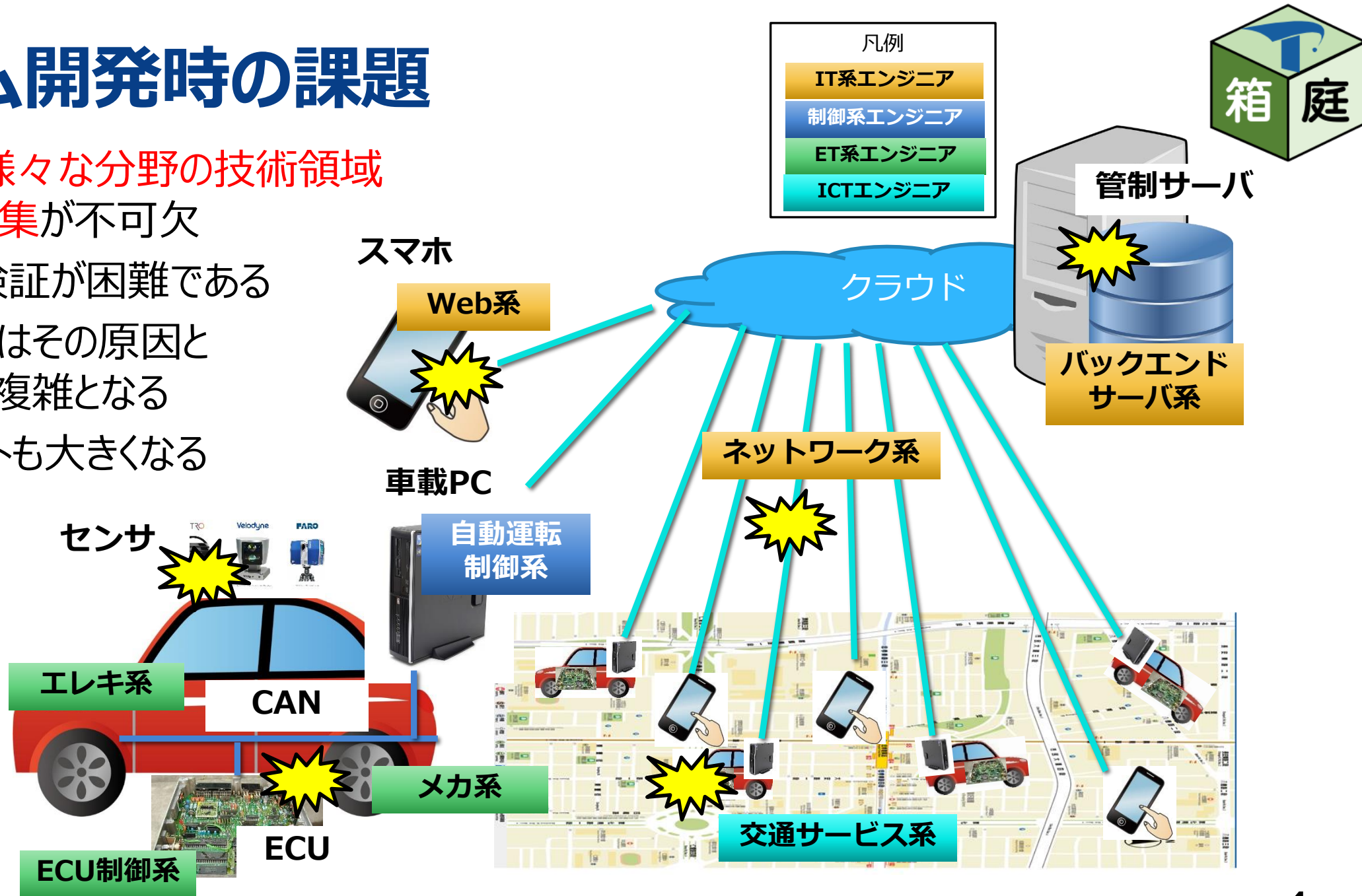


アジェンダ

1. 箱庭コンセプトと目指すところ [2分]
2. 最初の3年でやってきたこと [15分]
3. 今年のチャレンジテーマ [5分]
4. 箱庭はなぜ技術屋の心を揺さぶるのか [3分]
5. 今後の方向性／ロードマップ [3分]

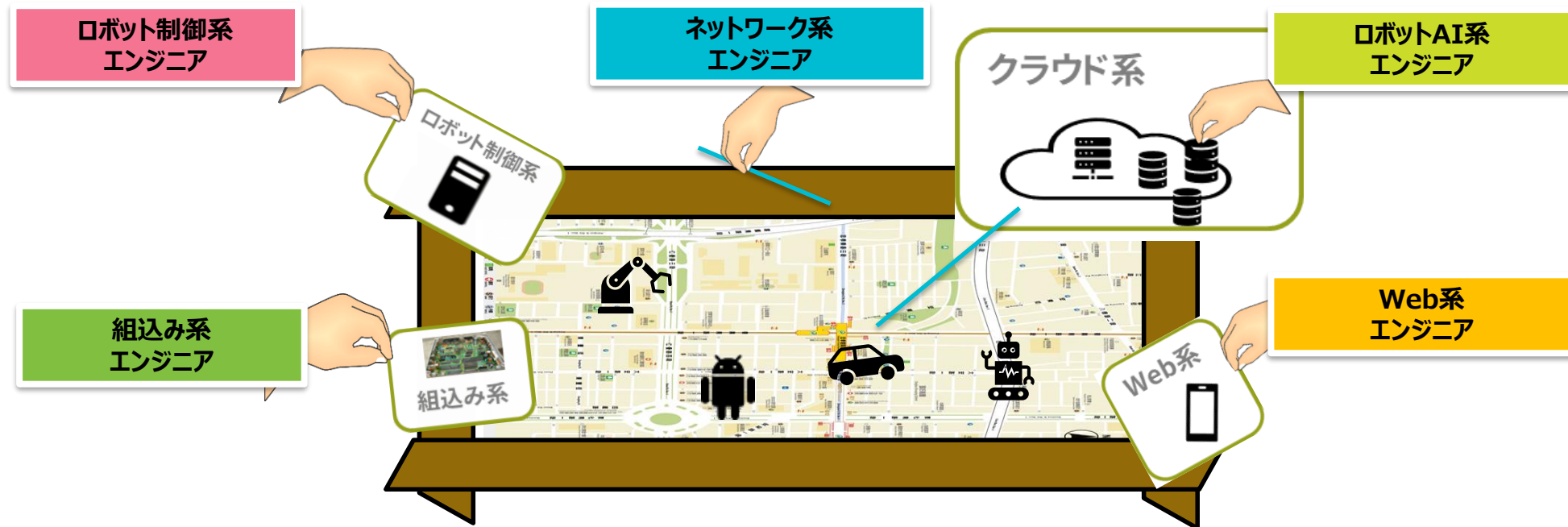
IoTシステム開発時の課題

- IoT開発には様々な分野の技術領域
= 技術者の結集が不可欠
- 結合テストや検証が困難である
- 問題発生時にはその原因と
経路の調査が複雑となる
- 実証実験コストも大きくなる



『箱庭』の狙いとコンセプト

- 箱の中に, **様々なモノ**を**みんなの好み**で配置して, いろいろ試せる!
 - 仮想環境上(**箱庭**)でIoT/ロボット・システムを開発する
- ⇒ 各分野のソフトウェアを持ち寄って, 机上で全体結合&実証実験!



箱庭WGの目指すところ



- 箱庭のターゲット

- IoTのような複雑なシステム(ロボット/物流・宇宙等様々な分野を想定)
 - 様々な機器(リアルタイム/非リアルタイム)がネットワークで接続されたシステム

- 箱庭とは

- 全体結合シミュレーション環境

- 箱庭の利用者

- システム開発者
- システム提供者
- 箱庭アセット(システム構成要素)提供者/開発者

システム開発者

システム提供者

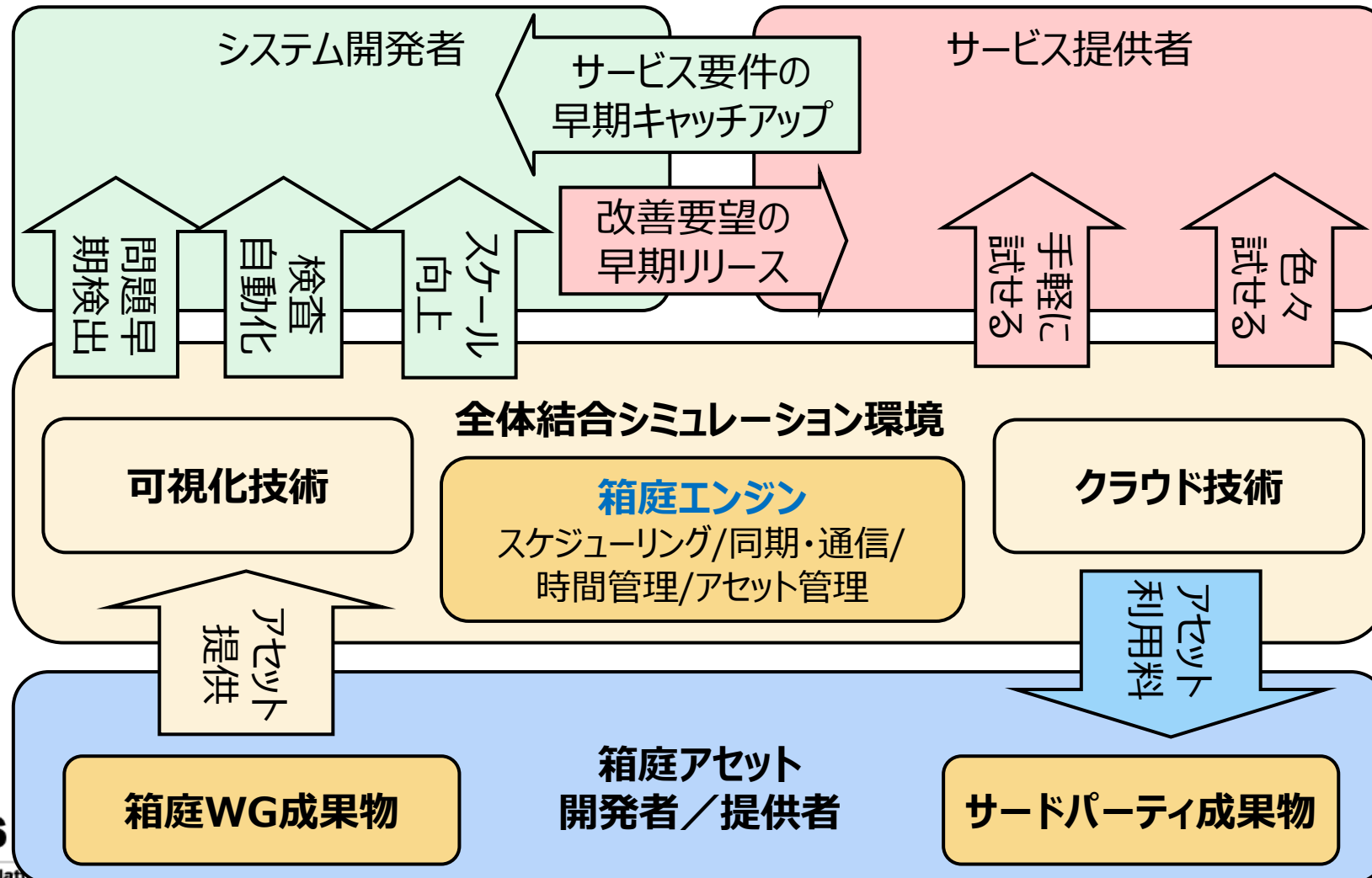
箱庭
(全体結合シミュレーション環境)

箱庭アセット
提供者

箱庭アセット
開発者

『箱庭』の目指すところ

IoTシステム開発者／提供者のためのシミュレーション環境とエコシステムを構築する





最初の3年でやってきたこと

- 2019年
 - mROS × Athrill × Unity でLチカ
 - mROS × Athrill × Unityで車の制御
 - Athrill × ASP3/EV3RT をブラウザでデバッグ
 - Athrill × ASP3/EV3RT でUnityモータ動かす！
 - Athrill × ASP3/EV3RT でロボット動かす！
- 2020年
 - ARM版 Athrill の試行
 - ETロボコンUnityモデル× Athrill(ARM版)連携の模索
 - ARM版 Athrill で mROS 動かしてマルチECU環境へ！
 - UnityとAthrill間で相互に時間同期させることに成功！
 - C#版箱庭コア機能が出来上がり、2台のロボットを走らせることに成功！
 - 様々なセンサやモータを組み込むことに成功
 - mROS使って2台のロボットを動かすことに成功
 - 箱庭教育演習コンテンツ作成した
 - gRPCにハマる
- 2021年
 - Athrillデバイス×gRPCでBluetooth デバイスを作成した
 - 箱庭のdocker対応成功！
 - 箱庭コア機能のgRPC対応始める
 - 箱庭単体ロボットシミュレータのmRuby対応に成功
 - 箱庭通信データのプロトコルバッファ対応に成功
 - 箱庭コア機能の再設計・機能拡張完了！
 - 箱庭コア機能の時間同期がどこまでやれるか実験していた
 - 箱庭上でTurtleBot3を動かすことに成功！
 - Unityが正式にROS対応していたのを知る！
 - 箱庭上で、HackEVをROSで制御できるようになった
 - TurtleBot3とHackEVを同時にROSで動かすことに成功！
 - 道路自動作成ツールの開発を始める
 - クオータニオンの壁に悩まされる
 - 箱庭のROS2対応始まる！
 - 箱庭のROS2対応してqiita記事で一般公開！
 - 道路自動作成ツールの一般公開
 - 箱庭教育コンテンツの一般公開
 - Unityの新しい物理モデルに挑戦！
 - URDF食わせてUnityで動かせるようになった
 - TB3のカメラ対応できた！箱庭仲間に感謝
 - ROS対応版箱庭には、実はこんな隠し機能があったりします
 - 箱庭のWebサービスのアーキテクチャを描く

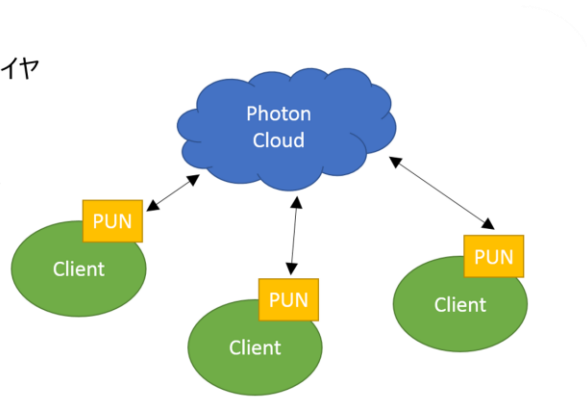
今年のチャレンジテーマ



• 箱庭を実現するゲーム技術

Photon アーキテクチャ

- Client
 - マルチプレイゲームに参加するプレイヤー
- PUN
 - Photon Unity Networking
 - マルチプレイ対応Unityパッケージ
- Photon Cloud
 - マルチプレイゲームを実現する
 - クラウドサービス
 - Client上のプレイヤーを
 - クラウド上で同期する



VRってなに？

言わずと知れたバーチャルリアリティ(VR)
ゲームの世界では手軽に楽しめるようになった！



Oculus Quest2



3D空間にダイブして，臨場感のあるVR体験

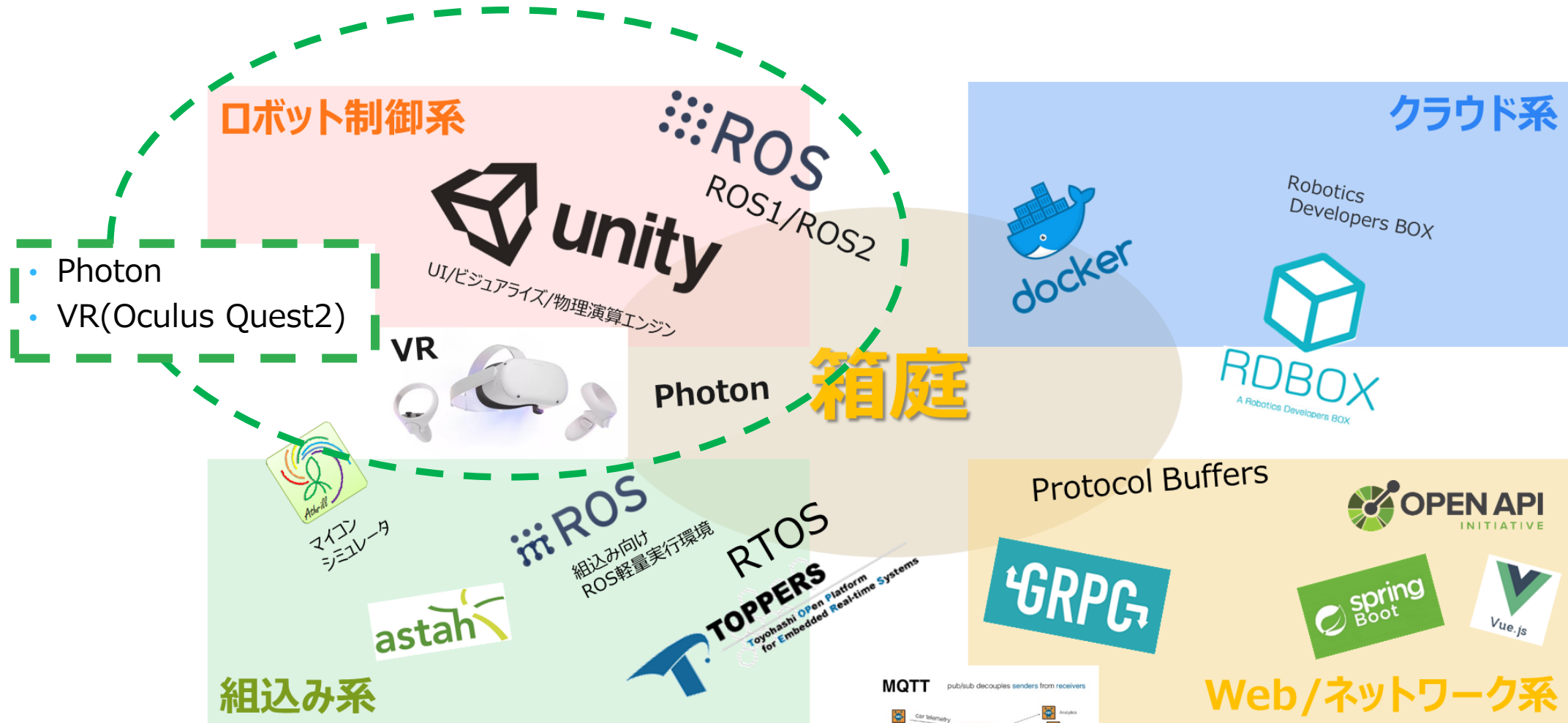
• 箱庭チュートリアル開発

組込み系
エンジニア

ロボット制御系
エンジニア

ロボットAI系
エンジニア

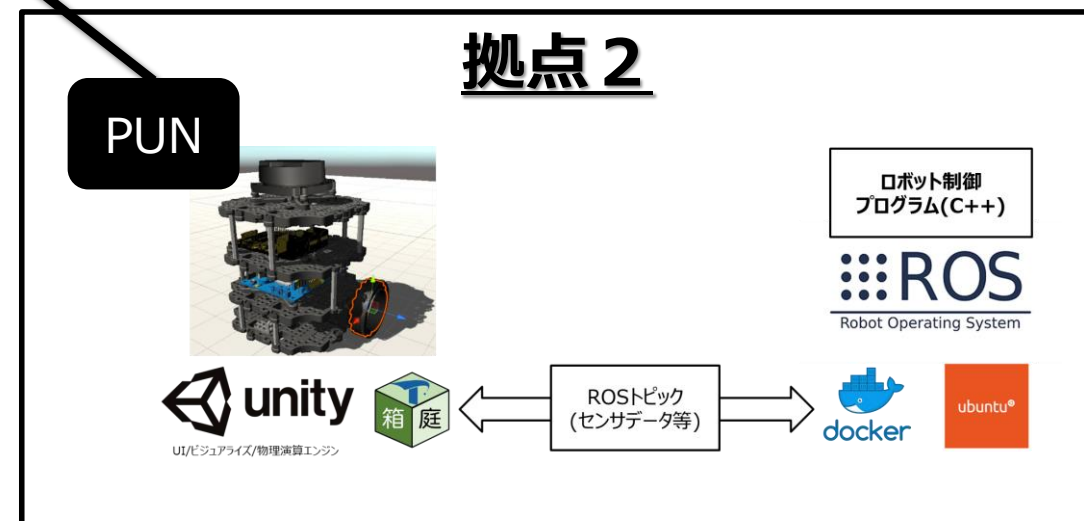
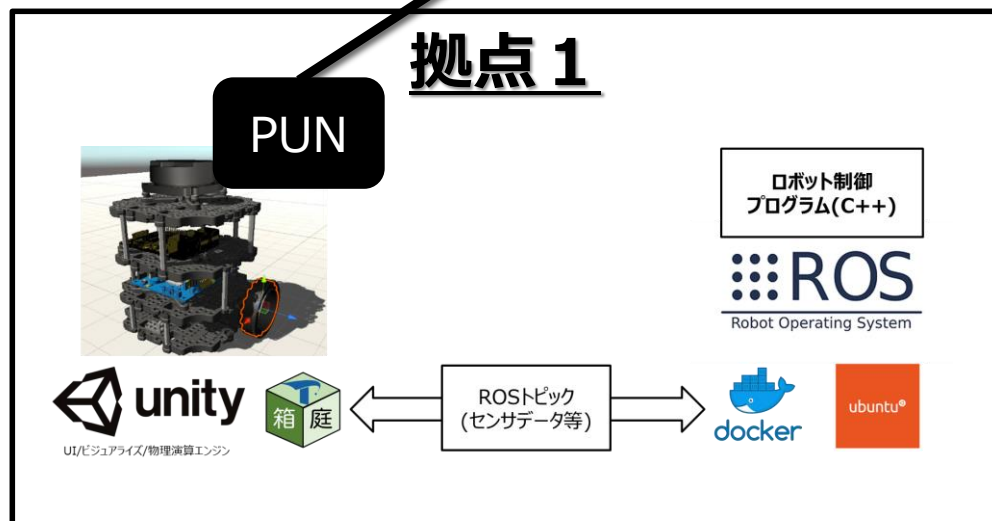
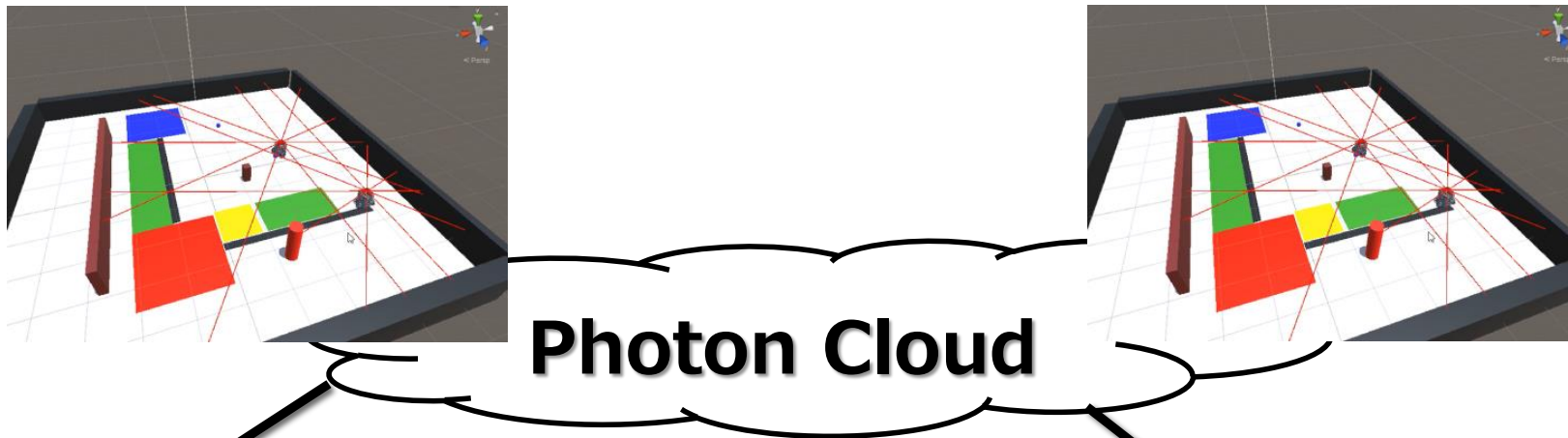
箱庭を実現するゲーム技術



Photonでロボット・マルチプレイする



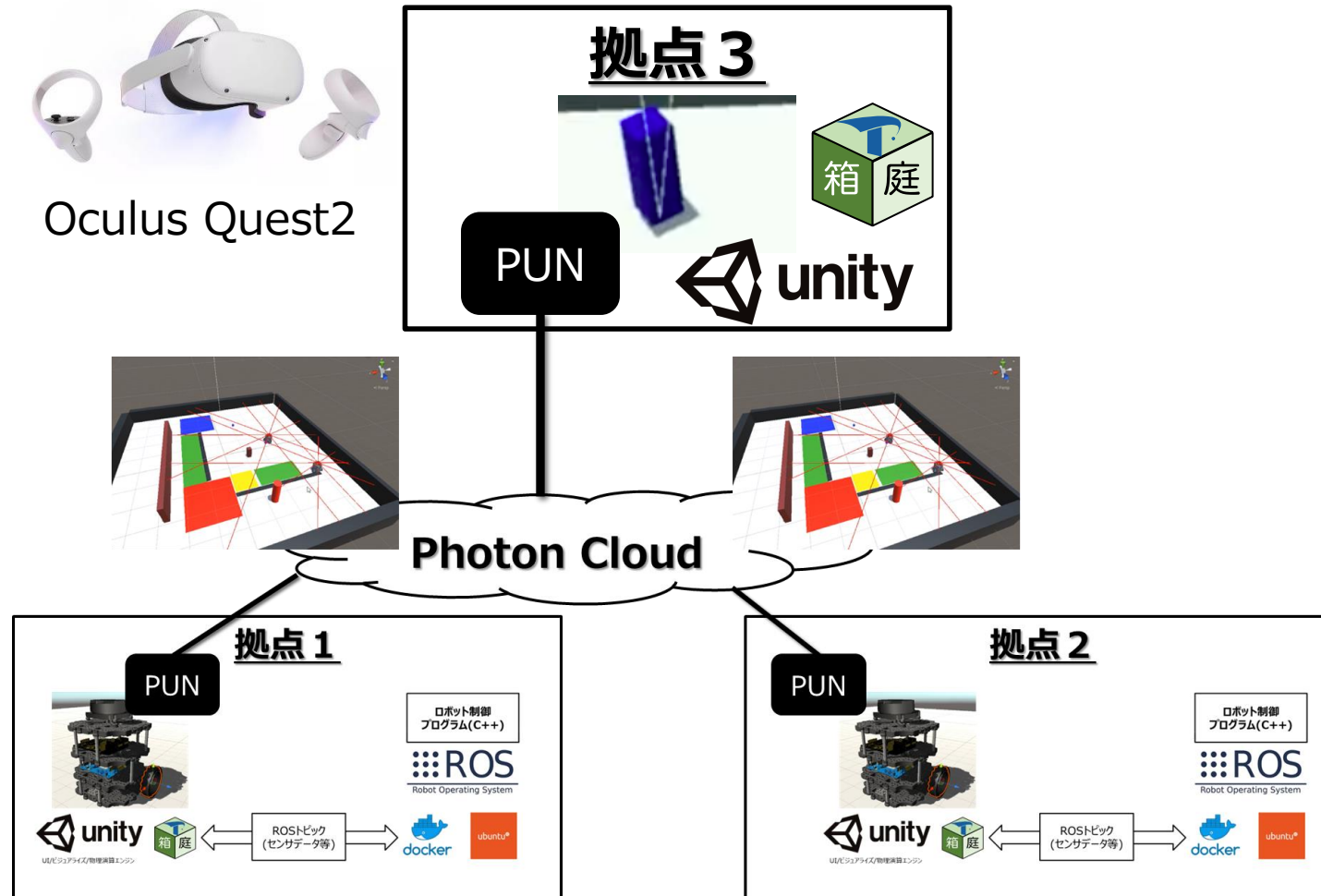
- 複数拠点にいる Turtlebot3 を Photon Cloud で同期します



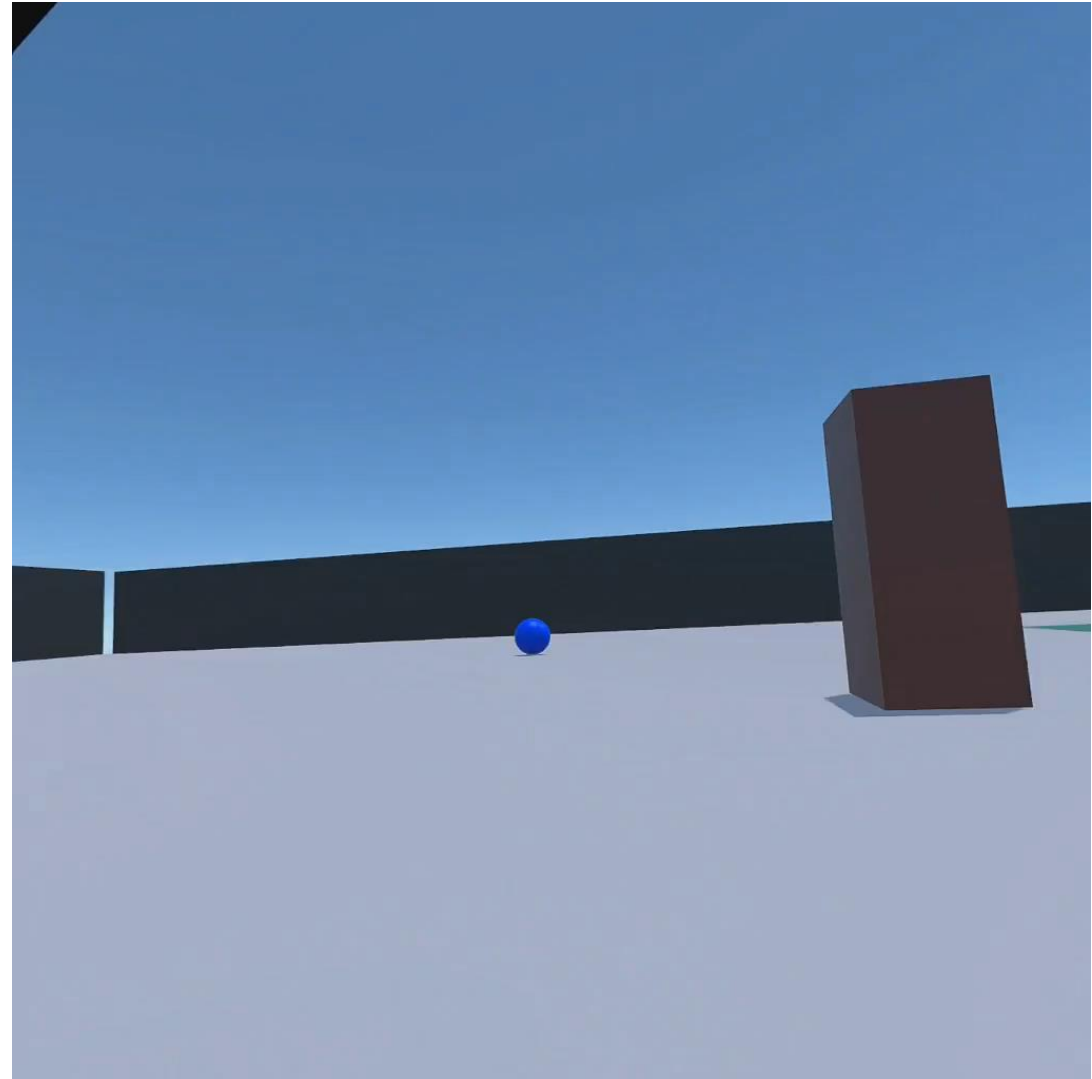
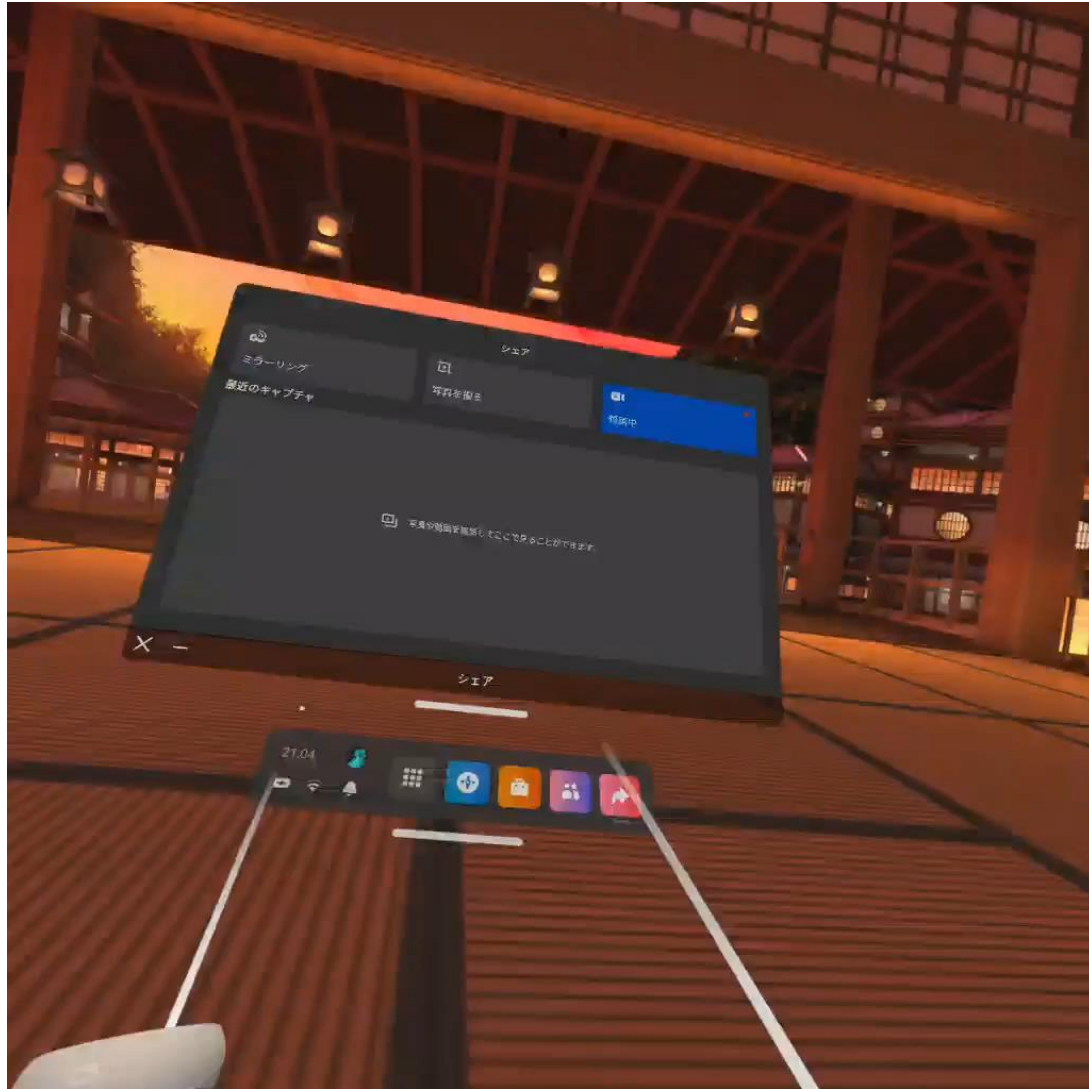
Oculus Quest2で箱庭VR空間にダイブする！



- 箱庭アバターを使って，Photon Cloudにログインして，VR体験します！



Oculus Quest2で箱庭VR空間にダイブする！



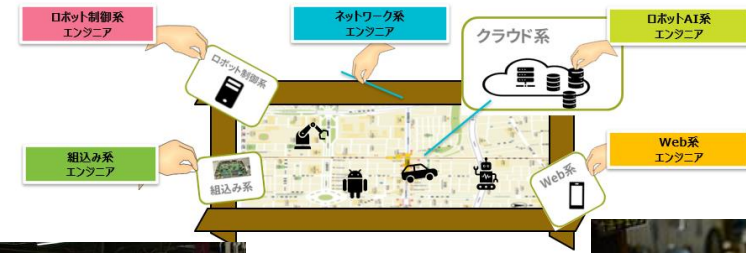
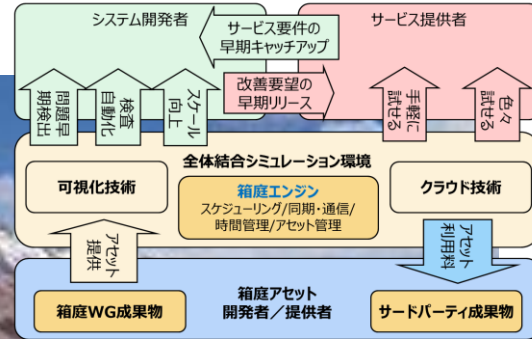
箱庭はなぜ技術屋の心を揺さぶるのか

箱庭の野望は壮大すぎる！

試して/失敗して/やり直す

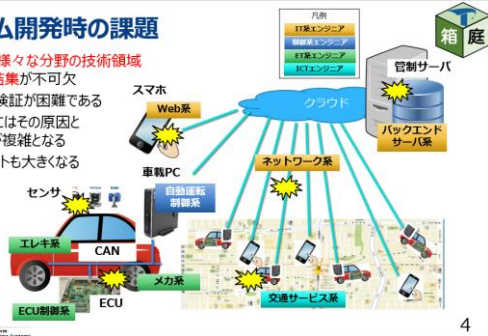
『箱庭』の目指すところ

IoTシステム開発者／提供者のためのシミュレーション環境とエコシステムを構築する



IoTシステム開発時の課題

- IoT開発には様々な分野の技術領域
= 技術者の結集が不可欠
- 結合テストや検証が困難である
- 問題発生時にはその原因と経路の調査が複雑となる
- 実証実験コストも大きくなる



4

今ある技術をどう組み合わせるか？
足りない技術をどう作るか？

今後の方向性／ロードマップ

2019-2021



2022-2024

箱庭チュートリアル

組み系エンジニア

ロボット制御系エンジニア

ロボットAI系エンジニア

2025



組み系エンジニア

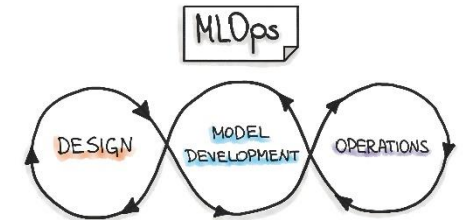
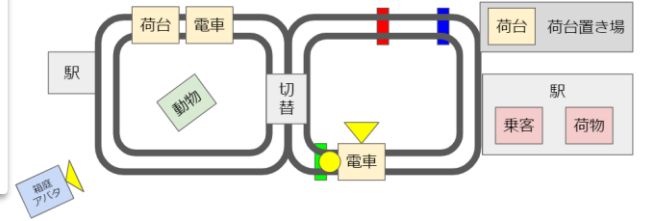
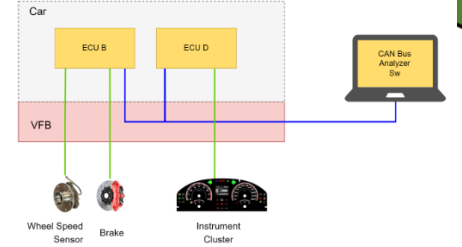
ECU/AUTOSAR開発入門シミュレーション環境

ロボット制御系エンジニア

ロボット制御モデリング/プログラミング入門シミュレーション環境

ロボットAI系エンジニア

ロボット制御学習サイクル自動化入門シミュレーション環境

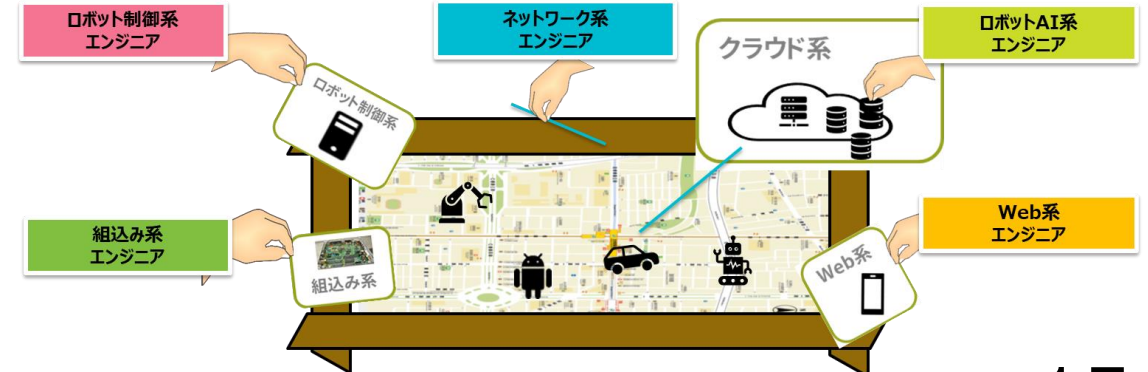


箱庭プロトタイプ
モデル開発

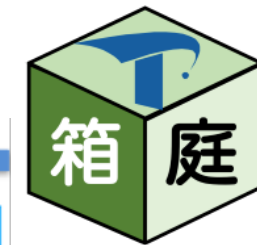
箱庭ユーザを
広める活動

結合環境構築
(箱庭ベース環境)

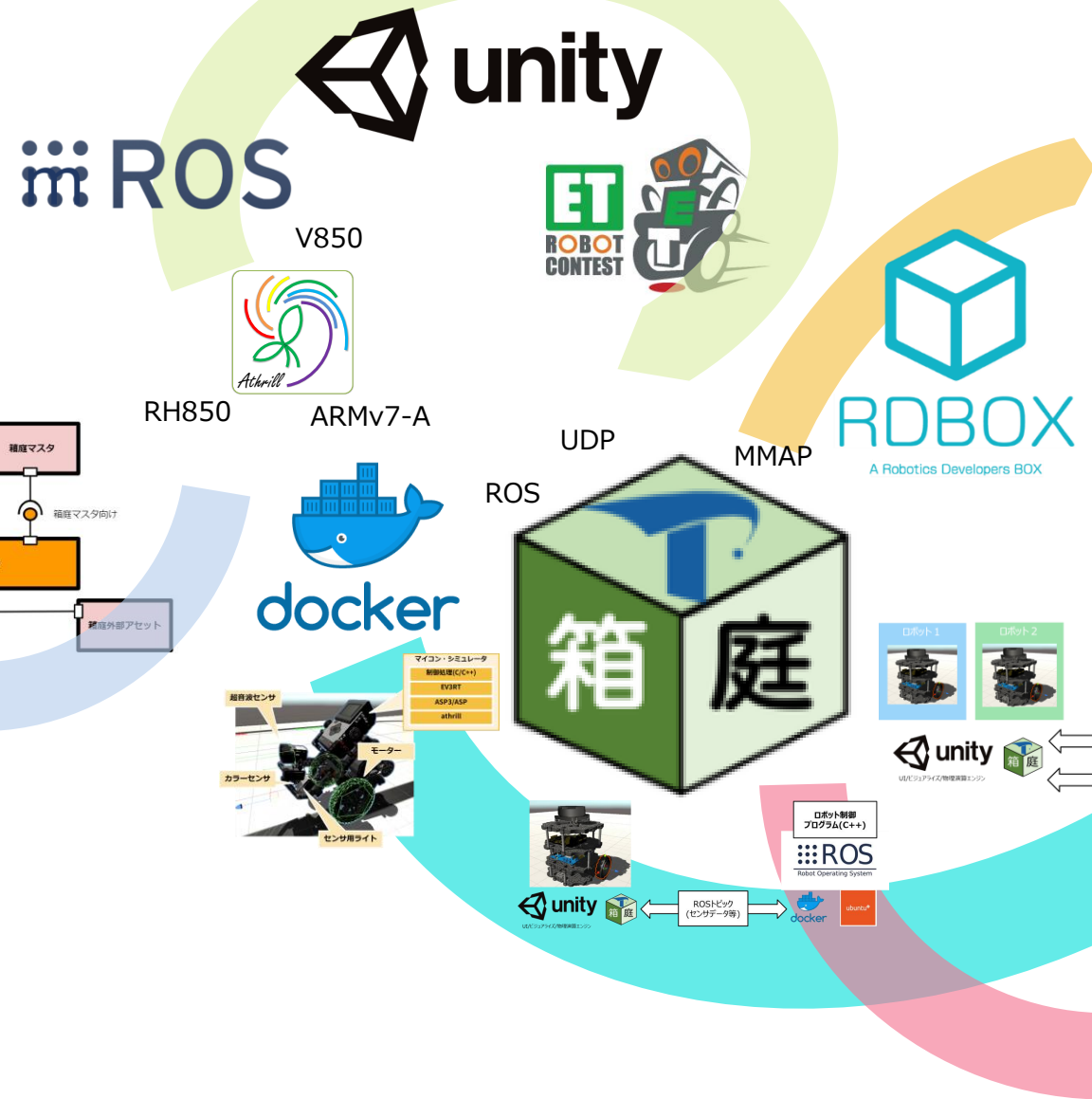
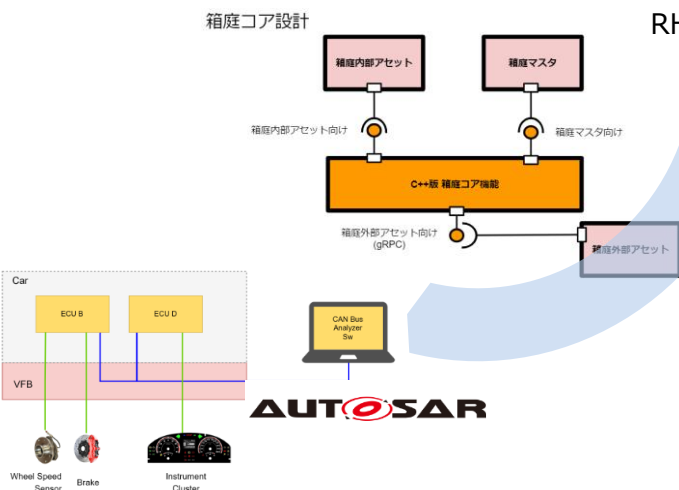
箱庭チュートリアル開発
&&広報活動



箱庭の軌跡



組み込み系エンジニア



ロボット開発の新しいカタチ

仮想世界と現実世界を同じインフラ環境で開発できるようになる！

仮想世界
unity
物理シミュレータ
Athrill
箱庭

現実世界
SBC
外部環境
箱庭

- SBCはAthrillで
- 外部環境はUnityで
- 現実世界の法則は箱庭で
- 開発環境は共通化
 - RDBOX
 - ROS/mROS

クラウドネイティブな情報基盤

ロボットAI系エンジニア

ROS
Robot Operating System

unity
箱庭

docker
ubuntu

ロボット1の制御プログラム(C++)
ロボット2の制御プログラム(C++)

ロボット1のROSBピック
ロボット2のROSBピック

ROS (センサデータ等)

光子 (Photon) We Make Multiplayer Simple
Join 835,120
TRY PHOTON FREE

ロボット制御系エンジニア

お知らせ：箱庭もくもく会

TOPPERS箱庭WGでは、もくもく会を不定期に開催しています。

本記事の内容を実際に試してみたい方や試してみたけど詰まっている方は、ぜひご参加ください。画面共有しながら参加者同士でサポートしたり、一緒に新しい機能を実装したりと“もくもく”しましょう。

箱庭

<https://hakoniwa.connpass.com>



2019年



mROS × Athrill × Unity でLチカ



2019年夏：

Unity習いたてで，athrillとmROS使ってLチカやってました．



<https://twitter.com/kanetugu2020/status/1559008091978444801>

mROS × Athrill × Unityで車の制御

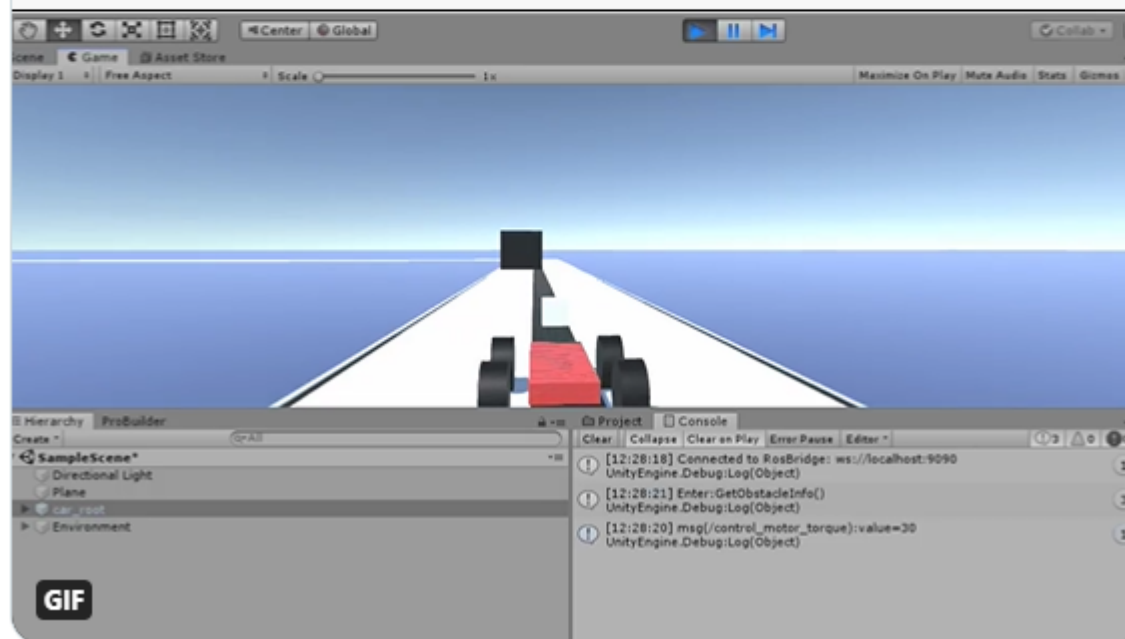


2019年夏：

Unityで上で車作れるようになったので，athrillとmROS
使って制御していたりしていた。（懐かしい）

 森 崇 @kanetugu2020 · 2019年7月28日

unity上で作成した車両のセンシング情報をROSトピックで出版しておいて，
athrill/mROS上で動作する車両制御アプリで障害物検出⇒停止させてみました。



<https://twitter.com/kanetugu2020/status/1155320595925520384>

Athrill × ASP3/EV3RT をブラウザでデバッグ



2019年秋：

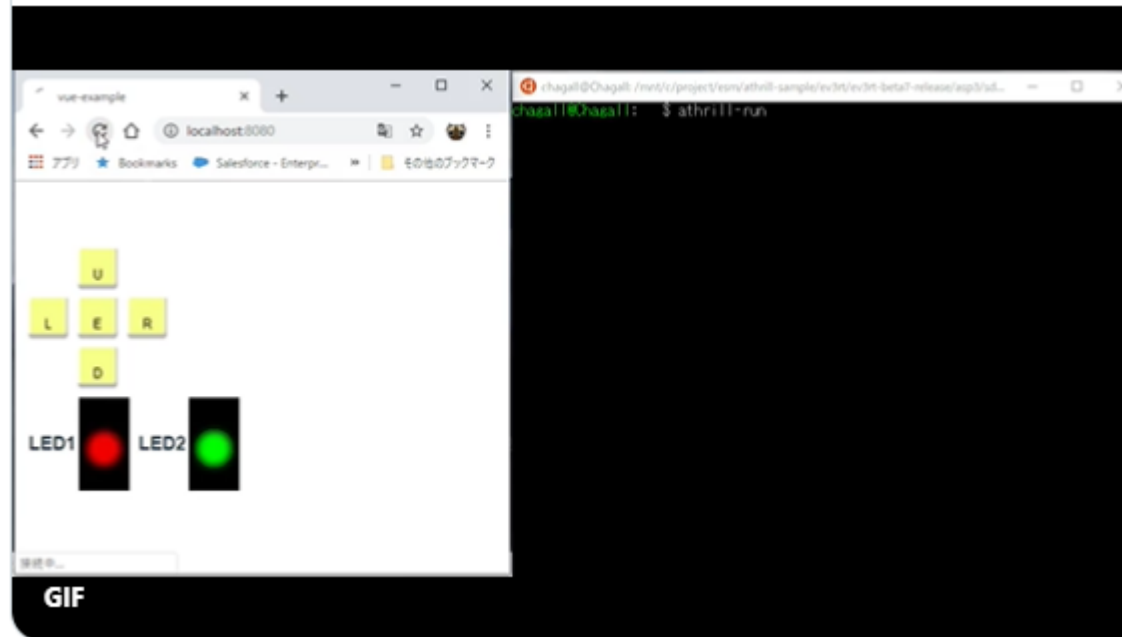
ETロボコン向けのシミュレータを作ろうと思って、
WebブラウザでEV3RTのデバッグしている風景。ブラウザとathrillどうやって通信させてたんやろ？（こんなんやってたんやー。

<https://twitter.com/kanetugu2020/status/1185178718710333440>

🐼 森 崇 @kanetugu2020 · 2019年10月18日

EV3RTをathrill/asp3に移植しています。
ようやく、LEDとプッシュボタンの移植ができました。
Webブラウザ使って動作確認完了。プッシュボタン押下でイベントが上がってきて、イベント処理でLEDの点灯制御しています。

[このスレッドを表示](#)



Athrill × ASP3/EV3RT でUnityモータ動かす！



2019年11月2日：

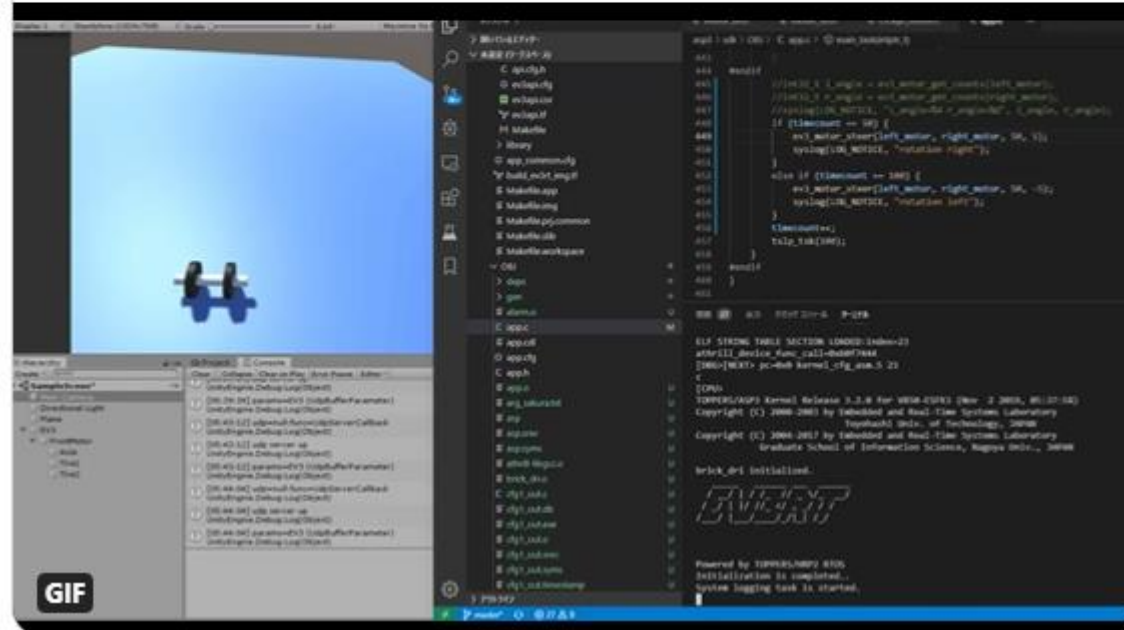
差動モーターをUnity上で作って，Athrill/ASP3/EV3RTで動かすことができた最初のツイート．イケルって思った瞬間．

<https://twitter.com/kanetugu2020/status/1559013573287849984>

森 崇 @kanetugu2020 · 2019年11月2日

マイコンシミュレータ(athrill)とUnityを連携させて，EV3RTのステアリングAPIを使ったモーター制御プログラム動かしたら，Unity上で車輪が右回転⇒左回転してくれた～．

[このスレッドを表示](#)



Athrill × ASP3/EV3RT でロボット動かす！



2019年11月2日12時：
この感動は忘れない！



<https://twitter.com/kanetugu20/status/1559014189267447808>

Athrill ARM 対応始める！



森 崇

@kanetugu2020

...

athrillのARM対応始めました。今日中にADD命令実行できるようにしてみるつもり。

github.com/tmori/athrill-...

午前9:17 · 2019年12月21日 · Twitter Web App

<https://twitter.com/kanetugu2020/status/1208179645964644352>



森 崇

@kanetugu2020

...

athrill の ARM対応, とりあえず add命令動いた。
オリジナル命令: add r1, r2, #10
athrill実行結果: [DONE> core0 pc=0x0 null(null) 0x0:
ADD Rd(R1(0)), Rn(R2(0)), imm12(10):10

午後5:14 · 2019年12月21日 · Twitter Web App

<https://twitter.com/kanetugu2020/status/1208299729030770688>



森 崇

@kanetugu2020

...

"Hello ARM World!"でました～。
実装した命令数 26 個。

github.com/tmori/athrill-...

午後1:50 · 2020年1月1日 · Twitter Web App

<https://twitter.com/kanetugu2020/status/1212234501364965377>

2020年





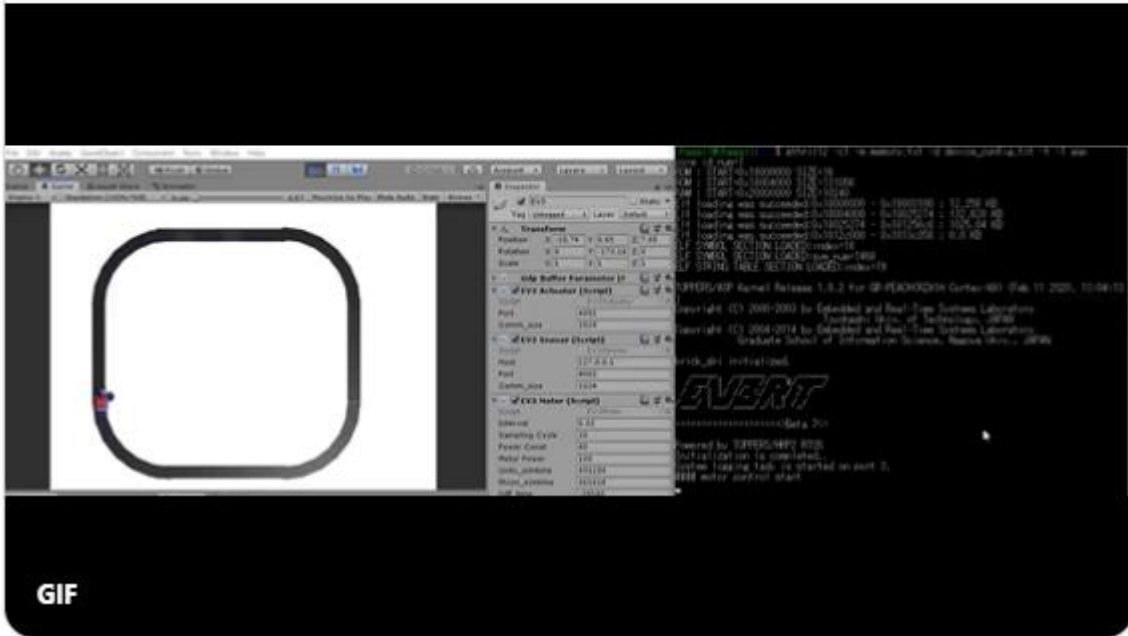
ARM版AthrillでETロボコンシミュレータを動かす！

2020年冬：

実はARM版AthrillでETロボコンシミュレータ作ろうと
していた。

森 崇 @kanetugu2020 · 2020年2月11日

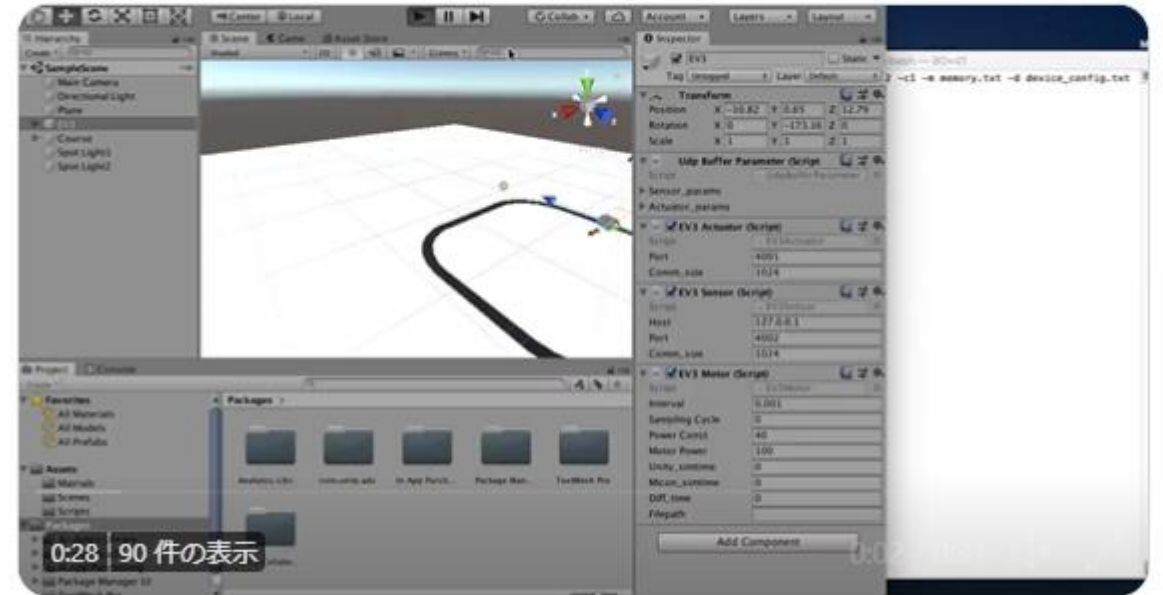
ARM版athrill&EV3RTとETロボコンシミュレータ(Unity)結合したら、動いちゃった。



森 崇

@kanetugu2020

ARM版ETロボコンシミュレータMACでも動いたー。感激！



午後9:13 · 2020年2月27日 · Twitter Web App

<https://twitter.com/kanetugu2020/status/1227082532094500864>

<https://twitter.com/kanetugu2020/status/1233002212890013698>

ARM版Athrillの開発難航する



2020年春：

ARM版Athrill開発難航. 命令セットデバッグをUnityシミュレーションベースでやるのは大変. 個々の命令セットを順番に開発する方が楽だけど時間が掛かる. なので, ビッグバン結合でエイヤっと開発していくスタイルが自分の開発スタイルになってしまった. 品質より動くものを早く作る！

森 崇 @kanetugu2020 · 2020年3月23日

arm版athrillでatofライブラリ動かしたら、デコードエラー連発した。😱あと二個対応したら動くかな？こうやって筋トレ続けないと命令セット増えませんよねー。d(￣￣)

[このスレッドを表示](#)

<https://twitter.com/kanetugu2020/status/1242014263582928897>

ARM版AthrillでmROS動かしてマルチECU環境へ



2020年春：

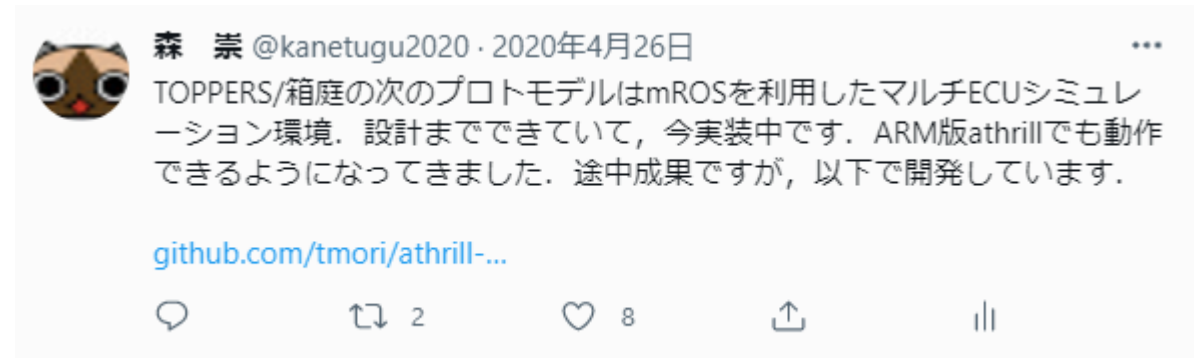
ARM版AthrillでもmROS動かすことできた！

```
森 崇 @kanetugu2020 · 2020年4月5日
ARM版athrillでmROS動いた～！

chagall@Chagall: /mnt/c/project/esm/athrill-sample/mros/mros_ws/hakoniwa-arm-demo/mros_test
chagall@Chagall:~$ athrill-run
OK: found device_config.txt
OK: found memory.txt
OK: found asp.elf
cfg1_out.elf
core id num=1
ROM : START=0x18000000 SIZE=131056
RAM : START=0x20000000 SIZE=2048
RAM : START=0x20200000 SIZE=7168
RAM : START=0x20900000 SIZE=1024
RAM : START=0x20a00000 SIZE=1024
RAM : START=0xfcfe0000 SIZE=1024
RAM : START=0x3fffff80 SIZE=8
ELF SET CACHE REGION:addr=0x18004000 size=151 [KB]
Elf loading was succeeded:0x18000000 - 0x18003100 : 12.256 KB
Elf loading was succeeded:0x18004000 - 0x18029ce0 : 151.224 KB
Elf loading was succeeded:0x18029ce0 - 0x180334d8 : 37.1016 KB
Elf loading was succeeded:0x18038000 - 0x180443bc : 0.0 KB
ELF SYMBOL SECTION LOADED:index=19
ELF SYMBOL SECTION LOADED:sym_num=1740
ELF STRING TABLE SECTION LOADED:index=20
athrill_device_func_call=0x200097f4
[DBG>[NEXT> pc=0x18004400 core_support.$ 61
c
[CPU>
TOPPERS/ASP Kernel Release 1.9.2 for GR-PEACH(RZAIH Cortex-A9) (Apr 5 2020, 20:19:25)
Copyright (C) 2000-2003 by Embedded and Real-Time Systems Laboratory
Toyohashi Univ. of Technology, JAPAN
Copyright (C) 2004-2014 by Embedded and Real-Time Systems Laboratory
Graduate School of Information Science, Nagoya Univ., JAPAN

System logging task is started on port 3.
*****mROS main task start*****
*****mROS pub task start*****
*****mROS sub task start*****
*****mROS slv task start*****
*****mROS mas task start*****
*****mROS Main task finish*****
*****Activate user task*****
[DBG>
```

<https://twitter.com/kanetugu2020/status/1246761169483792384>



<https://twitter.com/kanetugu2020/status/1254326709492396032>

UnityとAthrill間で相互に時間同期させることに成功！



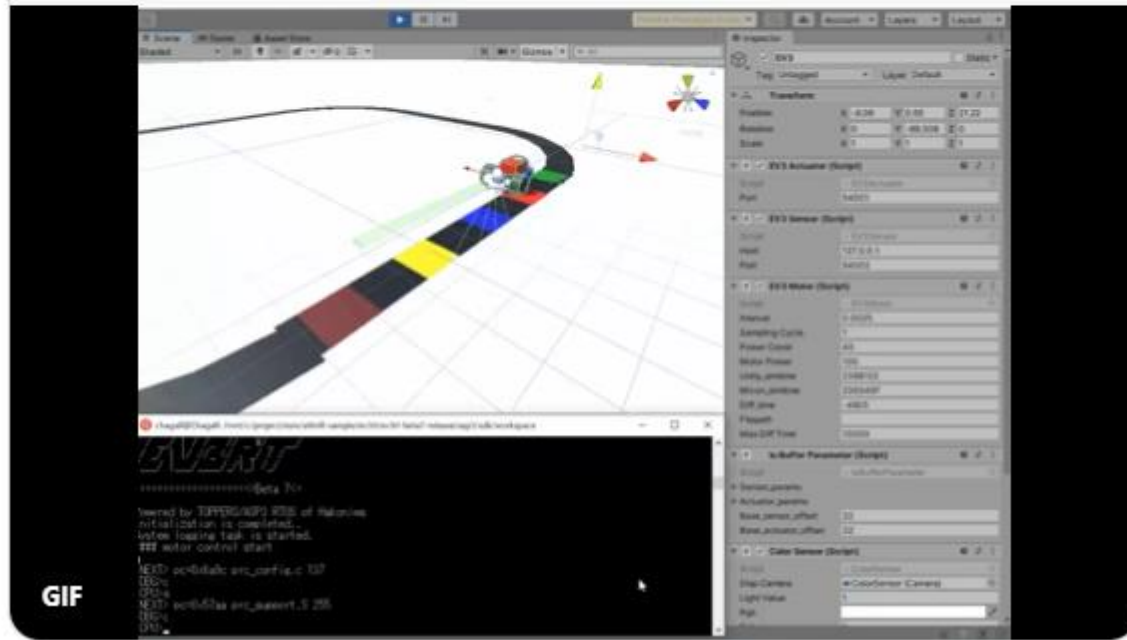
2020年6月：

あまり知られてないけど、自分的には、これ地味に好き。

<https://twitter.com/kanetugu2020/status/1559296207343833092>

森 崇 @kanetugu2020 · 2020年6月4日

TOPPERS箱庭WGのUnity開発者の方に技術的なサポートを頂いたおかげで、Unityとathrillが相互に時間同期できるようになりました。これでathrillでデバッグ⇒ステップ実行でUnityも同期して止まってくれるようになりました～。



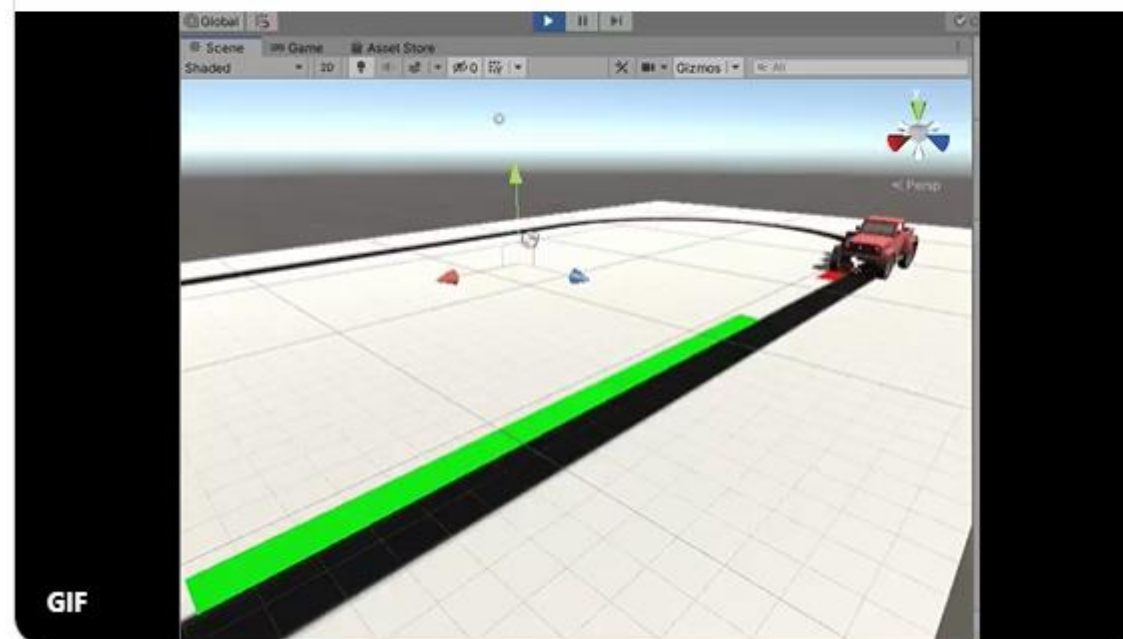
C#版箱庭コア機能が出来上がり， 2 台のロボットを走らせることに成功！



2020年6月：
C#版箱庭コア機能が出来上がり， 2 台のロボットを走らせることに成功！

<https://twitter.com/kanetugu2020/status/1276119750788083718>

森 崇 @kanetugu2020 · 2020年6月25日
TOPPERS箱庭のUnity側のリファクタが終わったので，ロボットの種類を増やして，ロボット台数も増やして遊んでみました。



様々なセンサやモータを組み込むことに成功



森 崇

@kanetugu2020

2020年初夏：
様々なセンサやモータを組み込むことに成功！

<https://twitter.com/kanetugu2020/status/1282220971470905344>



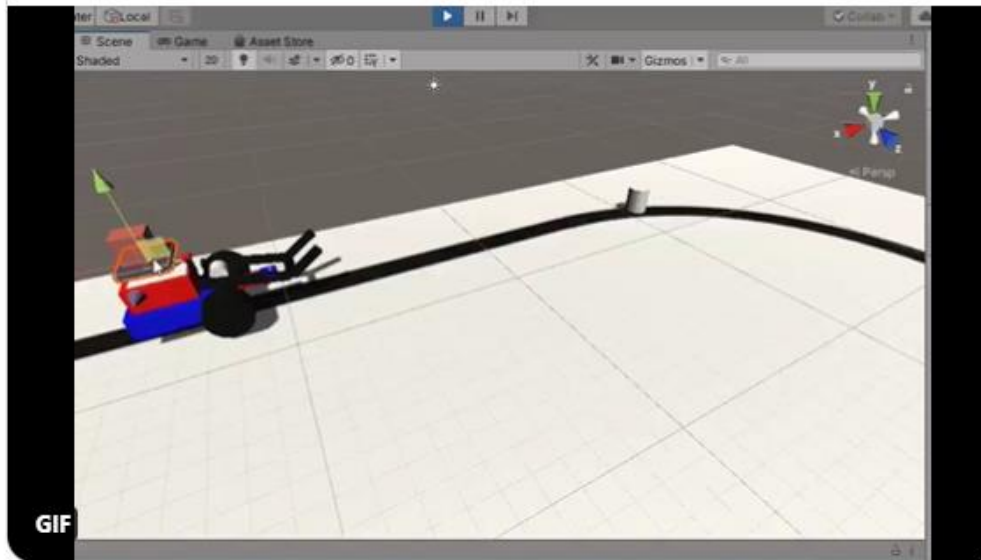
森 崇 @kanetugu2020 · 2020年7月12日

今週末の成果です！以下の機能追加しました。

1. タッチセンサ
2. ジャイロセンサ
3. アーム・モータ

ソースは以下で公開していました。UnityPackageは、準備でき次第公開させていただきます。

github.com/toppers/hakoni...



mROS使って 2 台のロボットを動かすことに成功



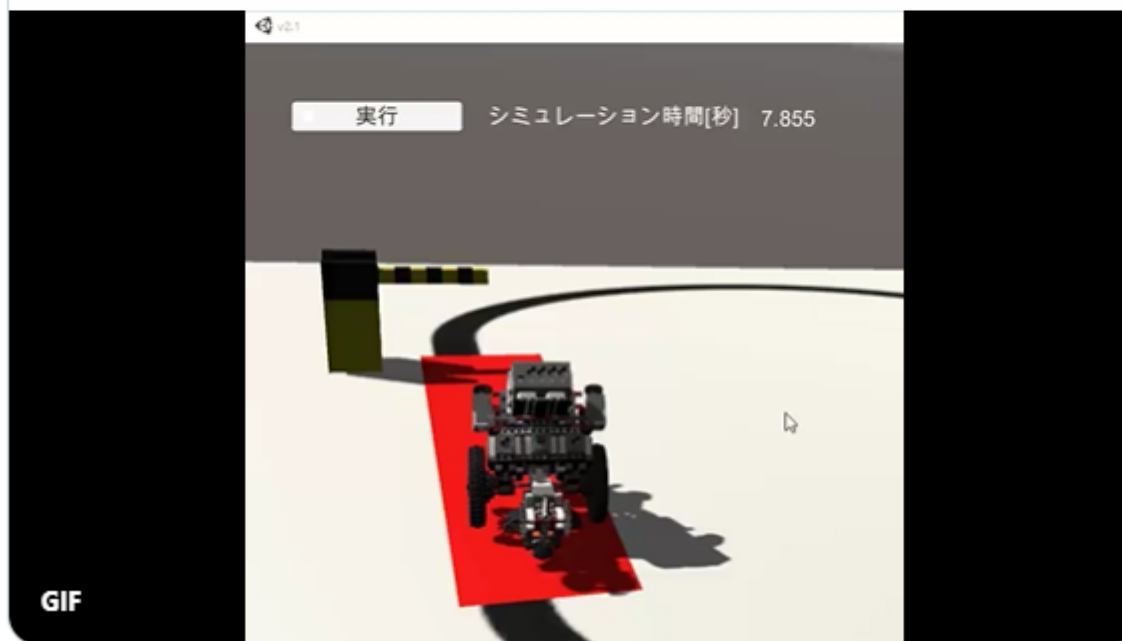
2020年夏：

mROS使って 2 台のロボットを動かすことに成功．踏切ロボットサイコー．箱庭のROS対応はココから始まった．

<https://twitter.com/kanetugu2020/status/1294536745044701184>

森 崇 @kanetugu2020 · 2020年8月15日

TOPPERS箱庭アセット 2 個(踏切, HackEV)を使って, 複数ロボットのシミュレーションやってみました. ロボット間の通信はmROS使ってます. 踏切状態をROSトピックで出版し, HackEVがそれを読み取って制御できるようになりました. 1 PCで2台実行するにはまだ課題残ってますので, 公開はもう少し先です...



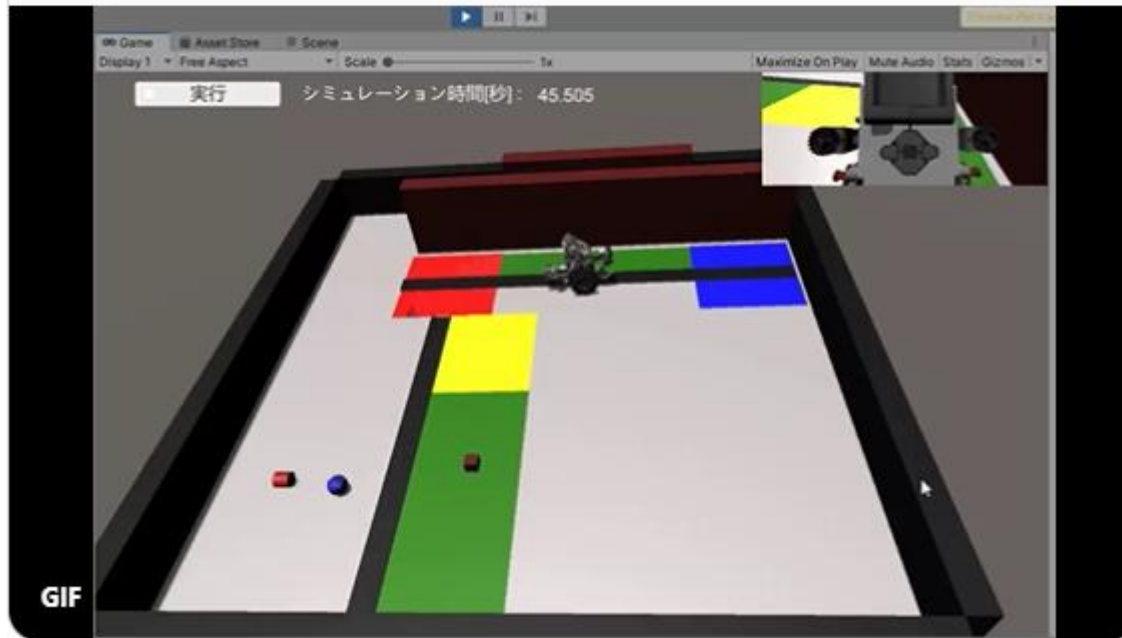
箱庭教育演習コンテンツ作成した



2020年9月：
箱庭教育演習コンテンツ作成した

森 崇 @kanetugu2020 · 2020年9月2日

9/27 に行われるTOPPERS開発者会議(toppers.jp/devconf2020.ht...)で、単体ロボットシミュレータのハンズオンやります。こんな演習課題を考えていますので、ぜひご参加をご検討ください～(まだ正解がうまく作れてないけれど・・・)。



<https://twitter.com/kanetugu2020/status/1301057244537077760>

gRPCにハマる



2020年冬：
gRPCにハマる

 森 崇 @kanetugu2020 · 2020年12月30日

athrillの周辺デバイスを様々な言語で実装できるための仕組みを作りました。
athrillの周辺デバイスの処理をgrpc連携することで、grpcサポート言語(ruby, python等)での処理記述ができるようになります。以下、サンプル用のデバイスをrubyで実装したものを公開しました。
[github.com/toppers/athrill...](https://github.com/toppers/athrill)

<https://twitter.com/kanetugu2020/status/1344130739164434433>

2021年



Athrillデバイス×gRPCでBluetooth デバイスを作成した

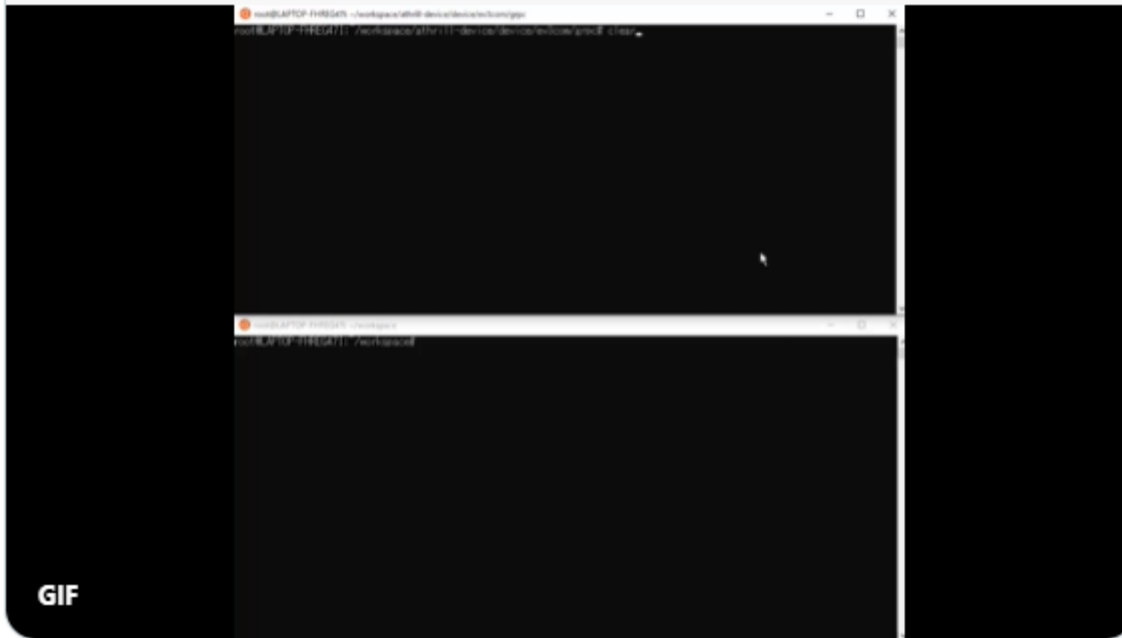


2021年1月：

Athrillデバイス×gRPCでBluetooth デバイスを作成した

 森 崇 @kanetugu2020 · 2021年1月31日

EV3RT用のathrill-deviceにbluetoothデバイスを追加しました。EV3RTのアプリから、bluetoothのシリアルI/Oを実行すると、サーバー側(rubyプログラム)でシリアルデータを受信できるようになりました。そのうち、一般公開します。



<https://twitter.com/kanetugu2020/status/1355820512950185986>

箱庭のdocker対応成功！



2021年2月：
箱庭のdocker対応成功！

 森 崇 @kanetugu2020 · 2021年2月11日

WSL2/docker で, TOPPERS/箱庭 単体ロボットシミュレータを実行できるようにしました. 面倒なネットワーク周りのコンフィグ設定も自動化できた.

qiita.com/kanetugu2018/i...

<https://twitter.com/kanetugu2020/status/1359802986541715460>



箱庭コア機能のgRPC対応始める

2021年2月：

箱庭コア機能のgRPC対応始める。宣言大事。

森 崇 @kanetugu2020 · 2021年2月21日

箱庭コア機能の試作始めました。とりあえず、grpcベースでどこまでやれるか突き進んでみる。

github.com/toppers/hakoni...

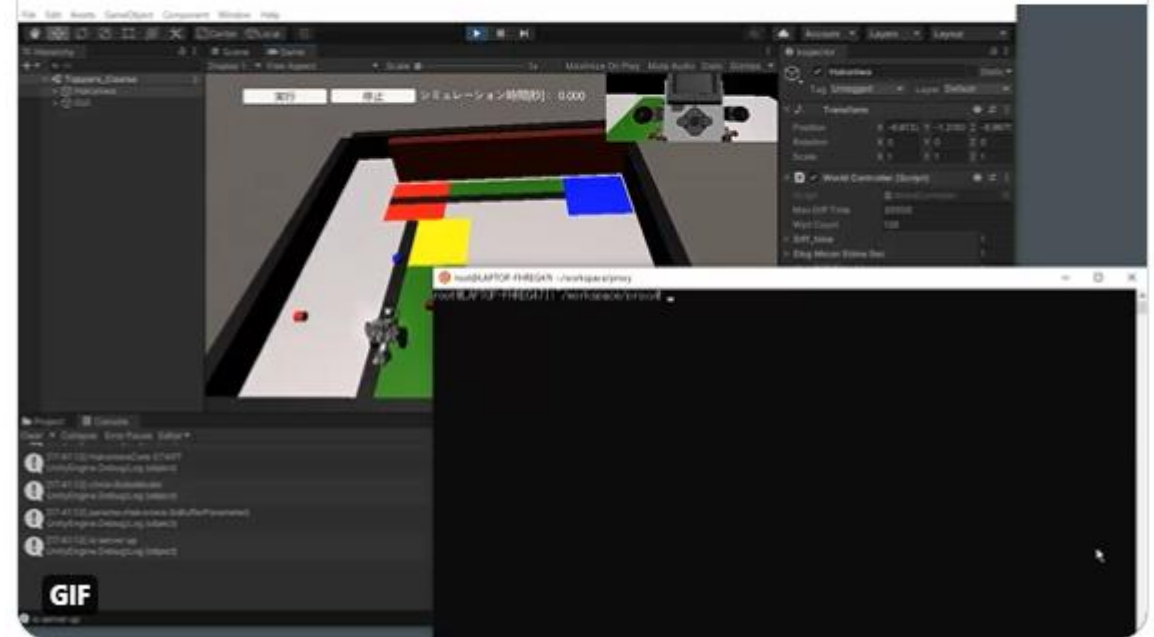
<https://twitter.com/kanetugu2020/status/1363398302050316289>

2021年3月：

箱庭コア機能のgRPC対応成功。いろいろ便利になった。

森 崇 @kanetugu2020 · 2021年3月9日

箱庭コア機能と箱庭プロキシ連携して、シミュレーションの再実行できるようになりました。



<https://twitter.com/kanetugu2020/status/1369207897033371654>

箱庭単体ロボットシミュレータのmRuby対応に成功



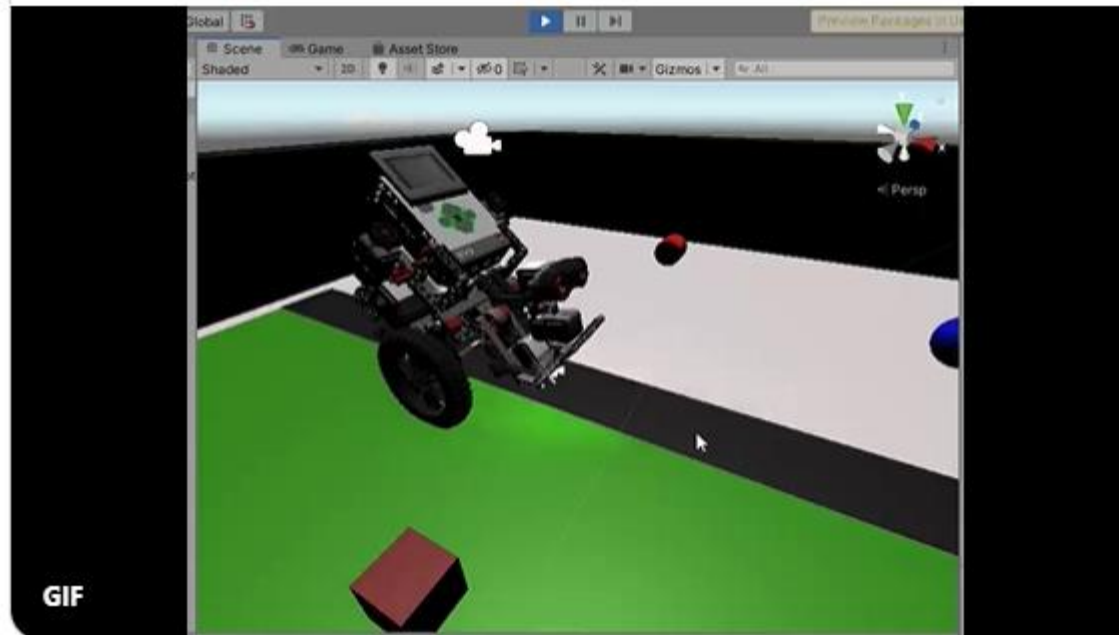
2021年3月：
mRuby対応やってました！

森 崇 @kanetugu2020 · 2021年3月13日
箱庭単体ロボットシミュレータがmruby版で動作できることを確認しました！

こちらの mruby版のプログラムを利用させていただきました(感謝)
github.com/yamanekko/base...

動作結果は、こんな感じで、結構スムーズに動いてます。

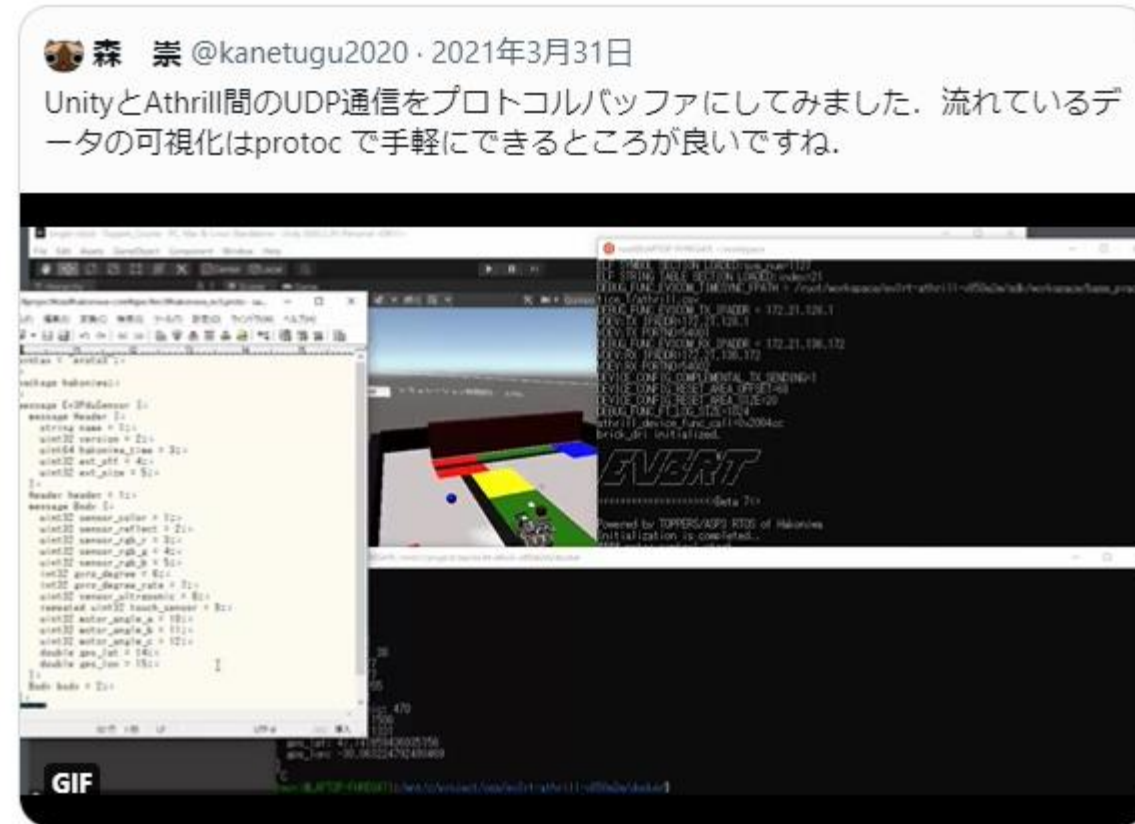
<https://twitter.com/kanetugu2020/status/1370536995101929477>



箱庭通信データのプロトコルバッファ対応に成功

2021年3月：
箱庭通信データ，実はプロトコルバッファでもやれる
ようにしています。

<https://twitter.com/kanetugu2020/status/1377188114272165893>



箱庭コア機能の再設計・機能拡張完了！

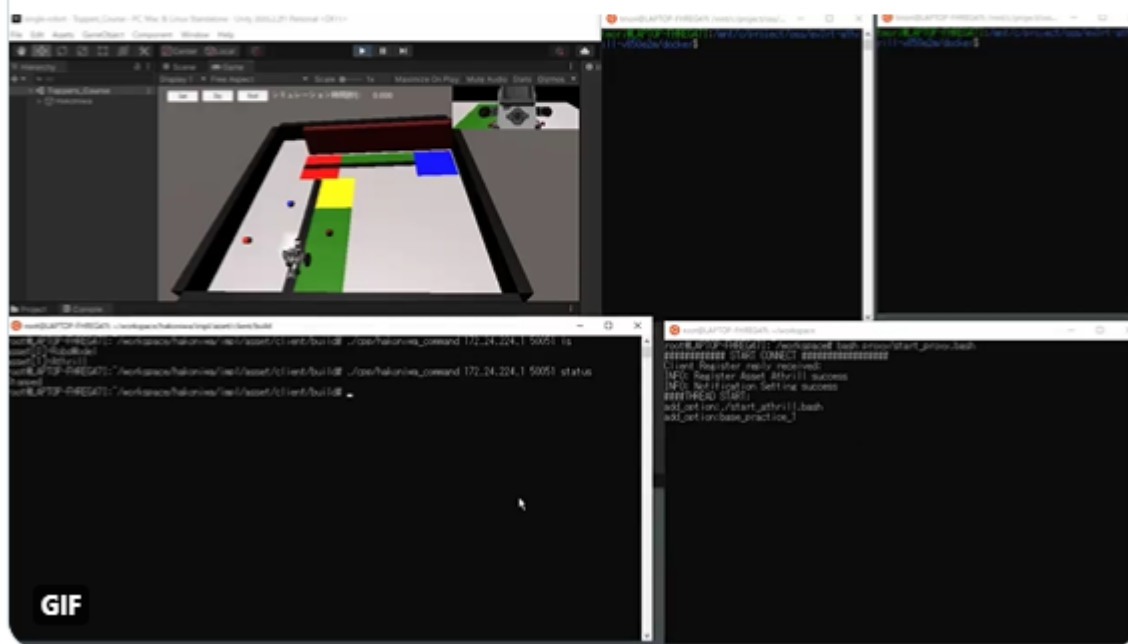


2021年4月：
箱庭コア機能の再設計・機能拡張完了！

森 崇 @kanetugu2020 · 2021年4月3日

箱庭コア機能の再設計・機能拡張でやりたかったことがほぼ出来上がりました。

- 箱庭アセット・プロキシ追加(画面右下)
 - 箱庭コマンド追加(画面左下)
 - 箱庭コアとアセット間の通信をプロトコルバッファ化する(画面右上)
- プロトコルバッファベースなので、protocコマンドで簡単に可視化可能。



<https://twitter.com/kanetugu2020/status/1559310893602652160>

箱庭コア機能の時間同期がどこまでやれるか実験していた



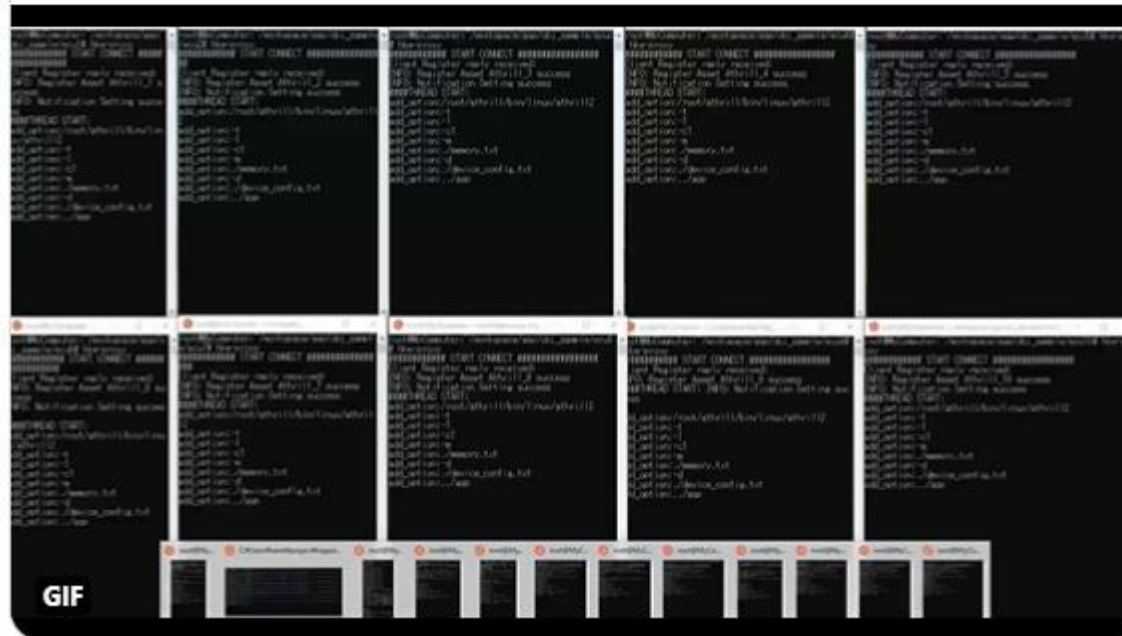
2021年4月：

箱庭コア機能の時間同期がどこまでやれるか実験していた。

森 崇 @kanetugu2020 · 2021年4月24日

箱庭コア機能使って、10個のECUの時間同期ができました。100msec周期でログ出力していますが、同じタイミングでカウントアップされていますー。
(10個端末準備するのが大変でした。)

<https://twitter.com/kanetugu2020/status/1385780796263321608>



箱庭上でTurtleBot3を動かすことに成功！



2021年4月：
箱庭上でTurtleBot3を動かすことに成功！

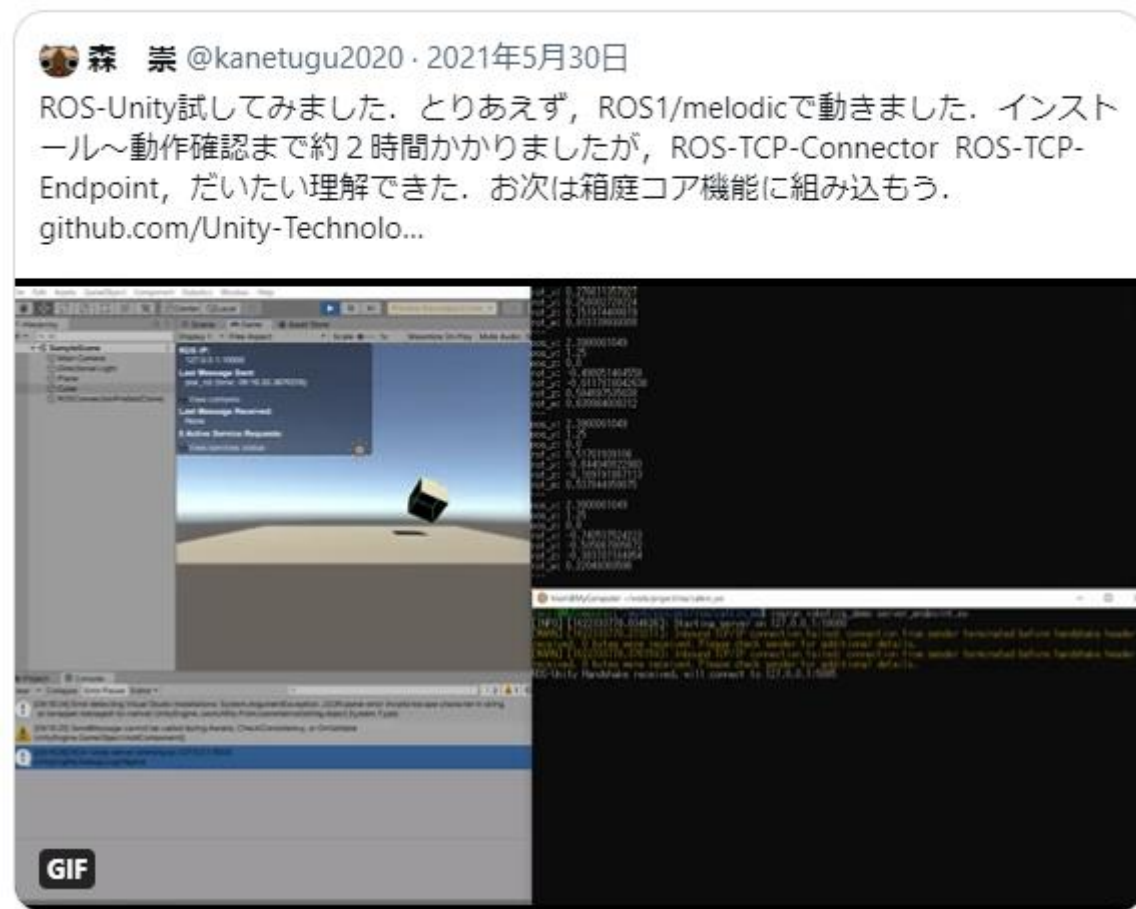


<https://twitter.com/kanetugu2020/status/1386611735595945988>

Unityが正式にROS対応していたのを知る！



2021年5月：
Unityが正式にROS対応していたのを知る！



<https://twitter.com/kanetugu2020/status/1398796788241342465>

箱庭上で，HackEVをROSで制御できるようになった

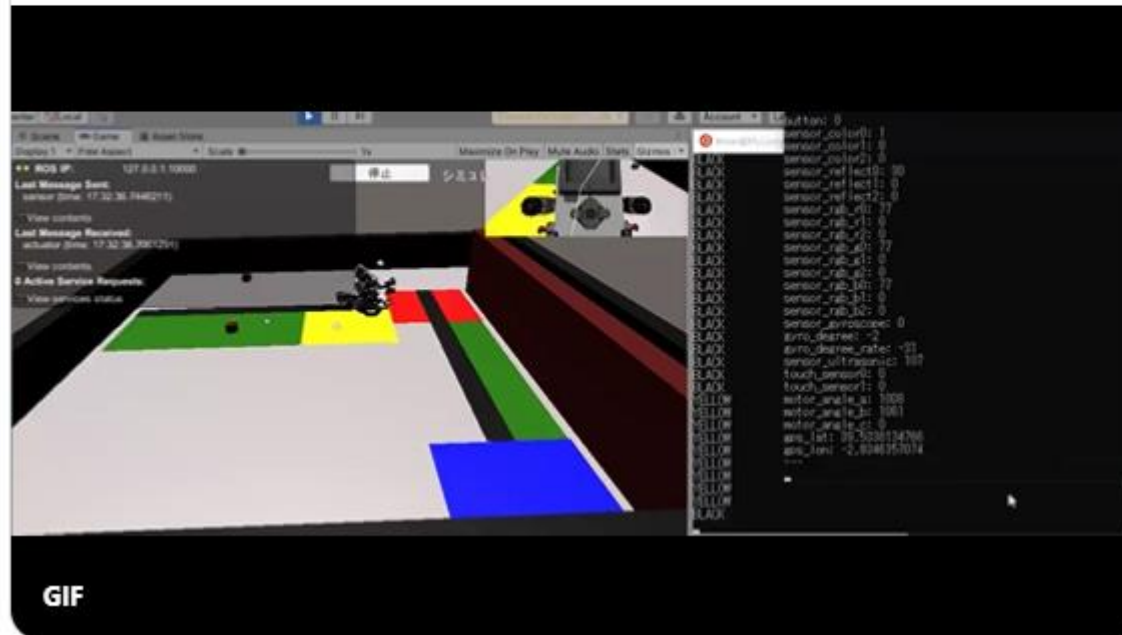


2021年6月：
箱庭上で，HackEVをROSで制御できるようになった

森 崇 @kanetugu2020 · 2021年6月26日
箱庭コア機能にROSトピックのpub/sub対応して，ETロボコン向けのデモを移植して動かしてみました。なんとか動きましたー！全部，ROSで動かしています！！

[github.com/tmori/unity-ro...](https://github.com/tmori/unity-ros)

<https://twitter.com/kanetugu2020/status/1559315691731185665>



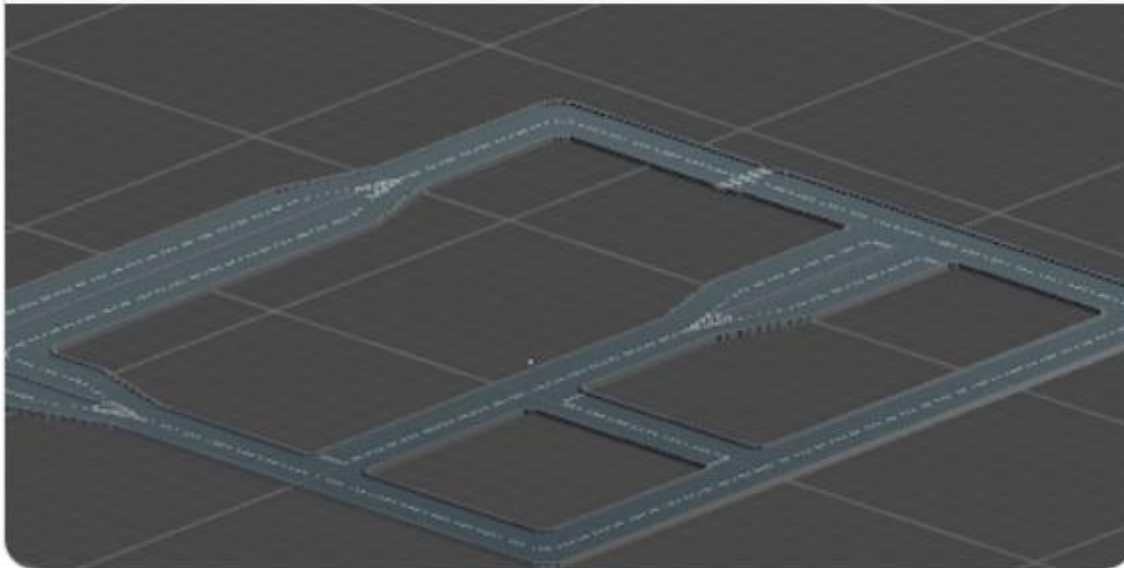
道路自動作成ツールの開発を始める



2021年7月：
道路自動作成ツールの開発を始める

 森 崇 @kanetugu2020 · 2021年7月15日

Unityのフリーのアセットを使って、道路を自動生成してみた。パーツの形状に応じた微調整が厄介だったけど、なんとか自動化に成功した。



<https://twitter.com/kanetugu2020/status/1415469879042269184>

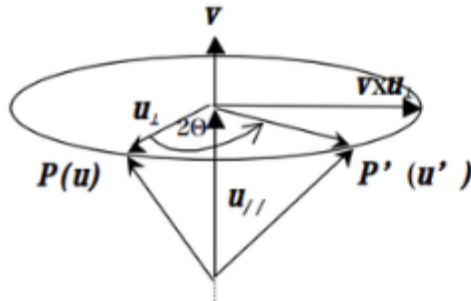
クォータニオンの壁に悩まされる

2021年7月：
クォータニオンの壁に悩まされる

森 崇 @kanetugu2020 · 2021年7月16日

クォータニオンを勉強しようと思って、ググっていたら平鍋さんの記事を見つけ。すごくわかりやすく解説されてました。

[qiita.com/kenjihiranabe/...](https://qiita.com/kenjihiranabe/)



すなわち、単位ベクトル $\mathbf{v} = (v_x, v_y, v_z)$ 回りの θ 回転を表す Quaternion q は、 $\theta = 2\Theta$ と置けば、

$$\begin{aligned} \mathbf{q} &= (v_x \sin \theta/2, v_y \sin \theta/2, v_z \sin \theta/2, \cos \theta/2) \\ &= (\sin \theta/2 \mathbf{v}, \cos \theta/2) \end{aligned}$$

<https://twitter.com/kanetugu2020/status/1415872191208460290>

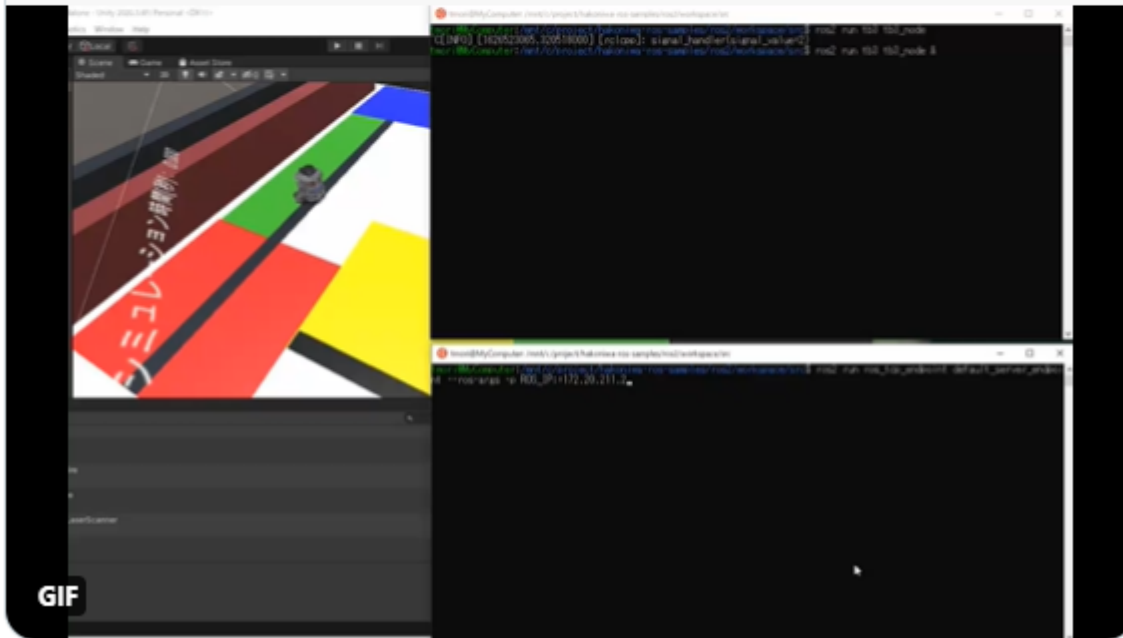
箱庭のROS2対応始まる！



2021年7月：
箱庭のROS2対応始まる！

森 崇 @kanetugu2020 · 2021年7月17日

UnityのROS2対応に伴い，早速，箱庭もROS2対応やってみました．とりあえず，突貫で動くところまでもっていきました～．もう少しソース綺麗にしたら一般公開します．



<https://twitter.com/kanetugu2020/status/1416367480797155329>

箱庭のROS2対応してqiita記事で一般公開！

2021年7月：
箱庭のROS2対応してqiita記事で一般公開！

森 崇 @kanetugu2020 · 2021年7月22日

TOPPERS/箱庭でROS1/ROS2対応がほぼ終わりましたので、qiita記事で手順等公開しました。

qiita.com/kanetugu2018/i...

<https://twitter.com/kanetugu2020/status/1418064276418306051>

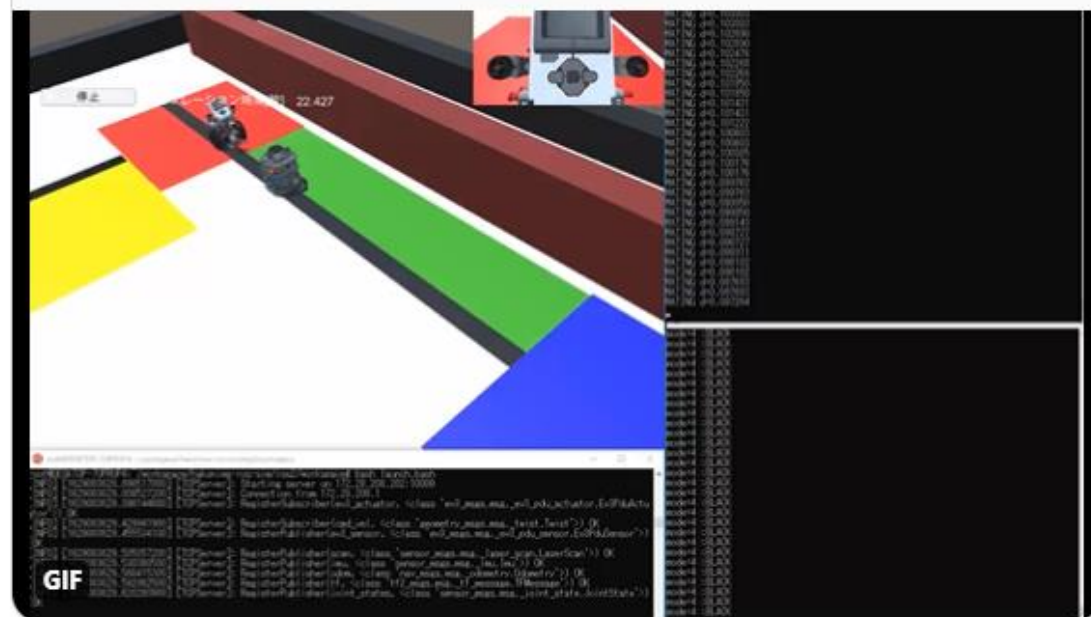
2021年8月16日：
ちょうど一年前

森 崇 @kanetugu2020 · 2021年8月16日

箱庭ROS2対応、以下の対応をしました。

- 箱庭プロキシ対応(時間同期/遠隔操作)
 - シミュレーション再現性と操作性があがります
- docker対応
 - 箱庭ROS2の導入手順が簡易化されます

もう少し早めに終わらせるつもりだったのに、思わぬバグ対応に追われてしまった。．．(一通り直した)



午前8:25 · 2022年8月16日 · Twitter Web App

道路自動作成ツールの一般公開



2021年8月：

道路自動作成ツールの一般公開。まだ日の目を見てないことに気づいた今日この頃。なんとか生かしたい。

 森 崇 @kanetugu2020 · 2021年8月29日

道路自動生成ツールの一般公開しましたー。
導入手順/利用手順をqiita記事で紹介しています。

qiita.com/kanetugu2018/i...

<https://twitter.com/kanetugu2020/status/1431890107800305666>

箱庭教育コンテンツの一般公開



2021年9月：
箱庭教育コンテンツの一般公開.



<https://twitter.com/kanetugu2020/status/1440833401380999170>

Unityの新しい物理モデルに挑戦！

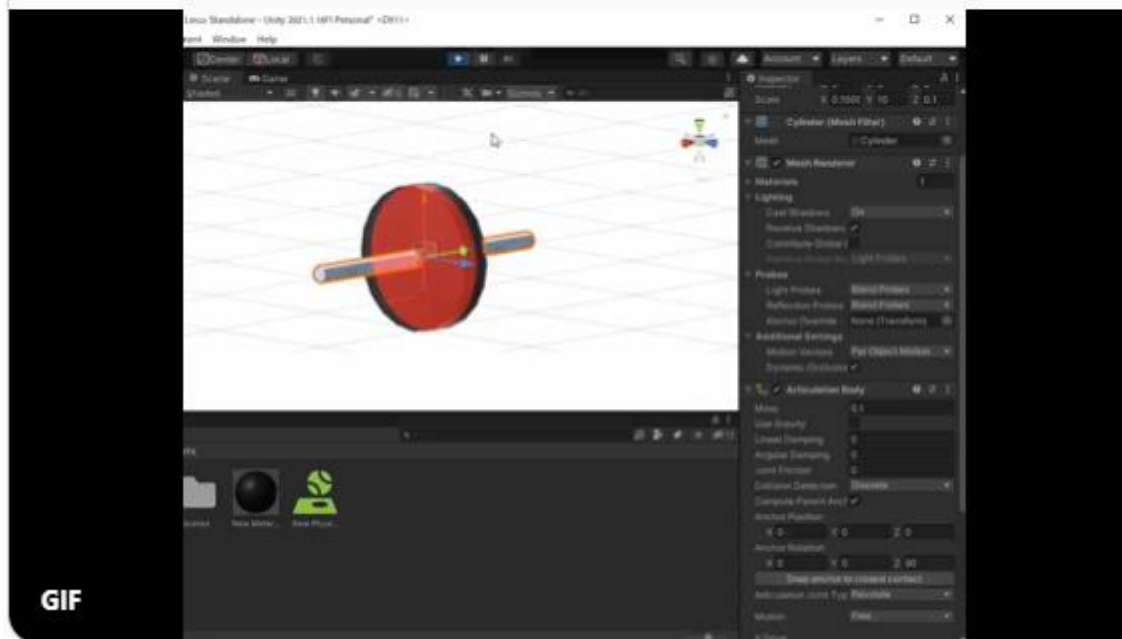


2021年11月：
Unityの新しい物理モデルに挑戦！

森 崇 @kanetugu2020 · 2021年11月6日

Unity articulationbody 試してみた。
物理オブジェクトはPhyscsじゃなくて、miscellaneousにあった。あとは、オブジェクトを階層構造にしないとジョイント関係を作れないっぽい。

でも、動いた動いた。

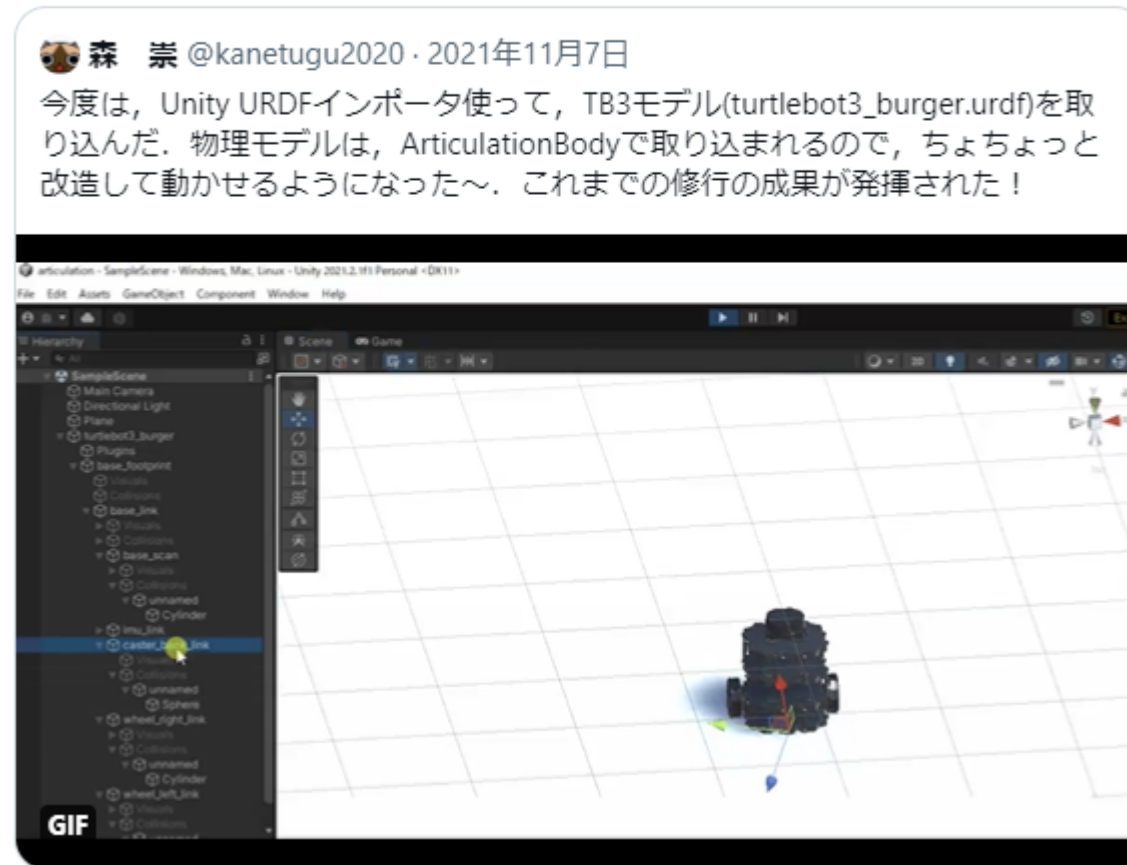


<https://twitter.com/kanetugu2020/status/1456768658349256705>

URDF食わせてUnityで動かせるようになった



2021年11月：
URDF食わせてUnityで動かせるようになった



<https://twitter.com/kanetugu2020/status/1457143055228624897>

TB3のカメラ対応できた！箱庭仲間に感謝.



2021年11月：
TB3のカメラ対応できた！箱庭仲間に感謝.



<https://twitter.com/kanetugu2020/status/1462335768945254410>

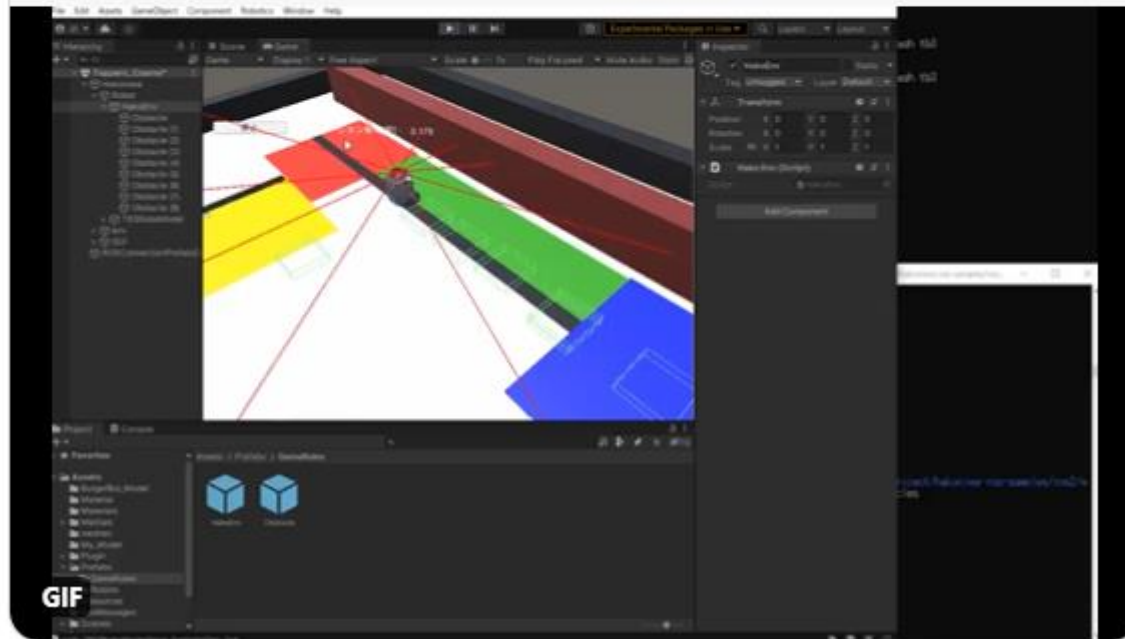
ROS対応版箱庭には，実はこんな隠し機能があったりします



2021年11月：
ROS対応版箱庭には，実はこんな隠し機能があったり
します．

森 崇 @kanetugu2020 · 2021年11月25日

TB3の移動軌跡を観察できるように，たくさん監視センサにおいて，ROSトピック
で見えるようになった．



<https://twitter.com/kanetugu2020/status/1463784652066353155>

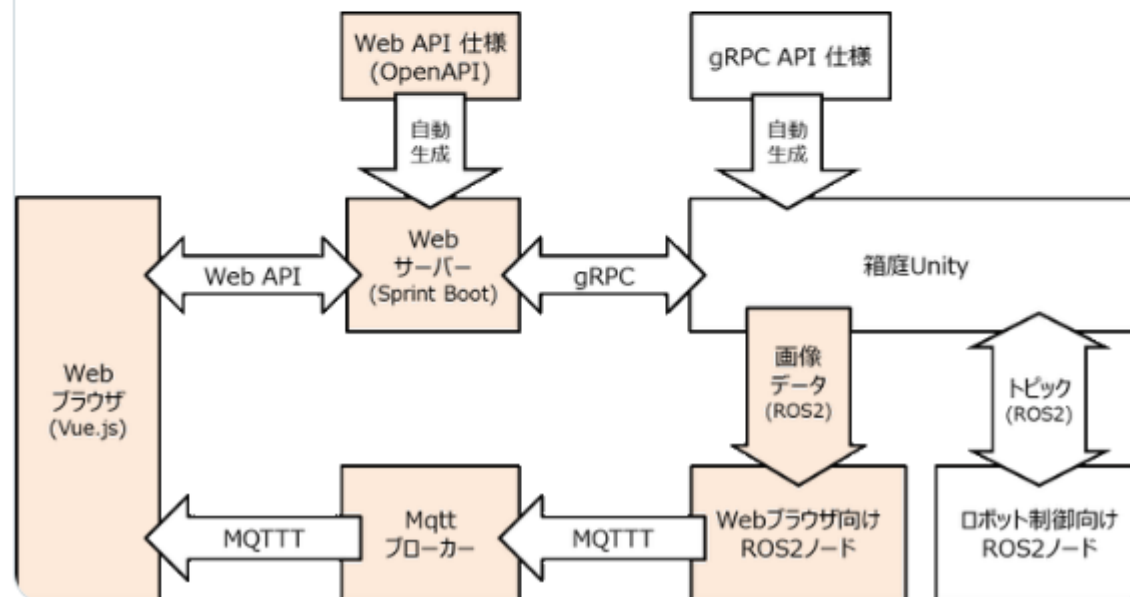
箱庭のWebサービスのアーキテクチャを描く

2021年年末：

箱庭のWebサービスのアーキテクチャを描く．実装されているがこれもまだ目の目を見てないことに気づく．

森 崇 @kanetugu2020 · 2021年12月31日

年末追い上げて作ったTOPPERS/箱庭のWebシステムのアーキテクチャのまとめです．技術面でMQTTで動画配信できないかな？と思いましたが，案外いけました．そのうち記事気にしたいと思います．今年も一年ありがとうございました！来年もがんばって箱庭開発してきますー．



<https://twitter.com/kanetugu2020/status/1476769685412913152>