

箱庭のプロトタイプ開発と 異種レイヤ間の時間同期機構

2020/06/12

森 崇 (永和システムマネジメント)
高瀬 英希 (京都大学/JSTさきがけ)

目次

1. 箱庭プロトタイプ・モデルの紹介

- 単体ロボット(ETロボコン)向けシミュレータ
- ROS・マルチECU向けシミュレータ
- ロボット間協調動作向けシミュレータ(RDBOX連携)
- 箱庭プロトタイプ・モデルの開発状況

2. シミュレーション時間同期機構の紹介

- シミュレーション時間同期の仕組み
- 時間同期の実装方式(UDP方式/MMAP方式)
- 適用事例紹介

箱庭のプロト・タイプモデル

箱庭コンセプトの実現/技術研鑽するために、
以下の3つのプロトタイプモデルを構築する予定です。

仮想化 レベル	プロト・タイプモデル	目的
1,2	A : 単体ロボット向けシミュレータ	<ul style="list-style-type: none"> ・時間同期の仕組み ・広報活動
2	B : ROS・マルチECU向けシミュレータ	<ul style="list-style-type: none"> ・ヘテロ環境の机上構築 ・通信可視化 ・箱庭アセット仕組み
3	C : ロボット間協調動作向けシミュレータ	<ul style="list-style-type: none"> ・クラウド連携 ・ロボット連携

A : 単体ロボット(ETロボコン)向けシミュレータ

マイコン・シミュレータ

制御処理(C/C++)

EV3RT

ASP3/ASP

athrill



ETロボコンを題材として構築

技術研鑽視点での狙い：

- ・物理シミュレータとマイコンシミュレータ間の連携方法の検討
- ・異なるシミュレータ間の時間同期の検討

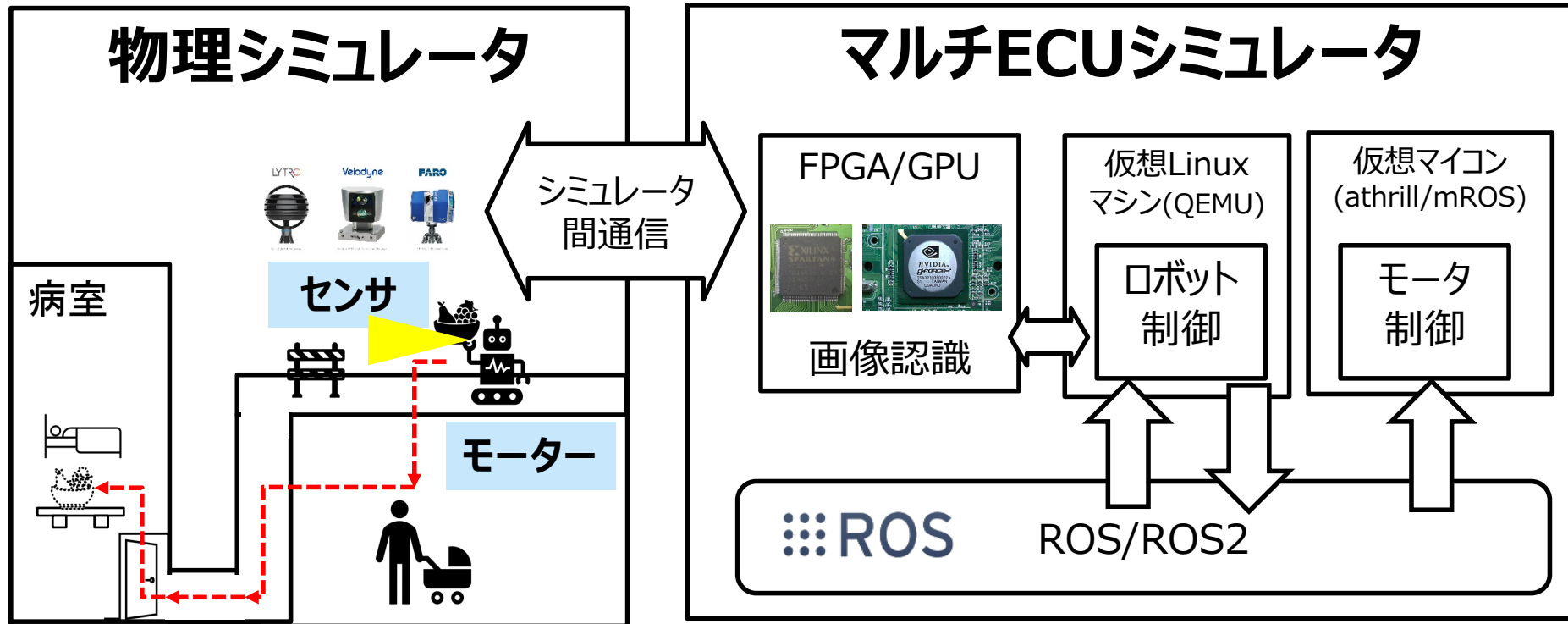
その他の狙い：

- ・ETロボコンユーザ層に箱庭を広める（広報活動）

Unityパッケージの設計と作成にあたっては、宝塚大学 東京メディア芸術学部 吉岡章夫准教授および学部生の杉崎涼志さん、木村明美さん、千葉純平さんにご協力いただきました。

HackEVのUnityアセットは、ETロボコン実行委員会より提供いただいたデータを基に作成しています。実行委員会の皆さまに深く感謝いたします。ただし本アセットはETロボコンの本番環境とは異なりますので、大会に参加予定の方はご注意ください。また、本アセットは、個人利用または教育利用に限定してご利用ください。

B : ROS・マルチECU向けシミュレータ



技術研鑽視点での狙い :

- ・マルチECU/FPGA/GPU間の連携方法検討(シミュレーション時間同期等)
- ・箱庭アセット間の通信可視化方法の検討(ROS/ROS2連携含む)
- ・箱庭アセットの仕組み検討

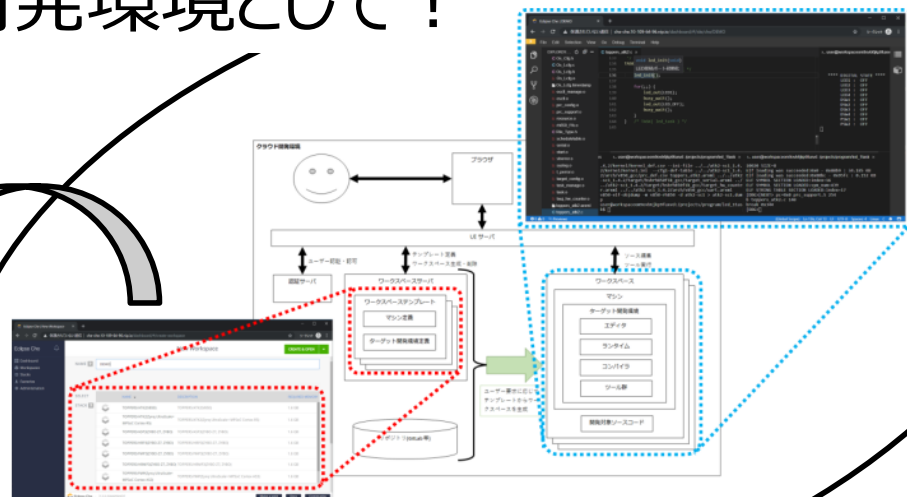
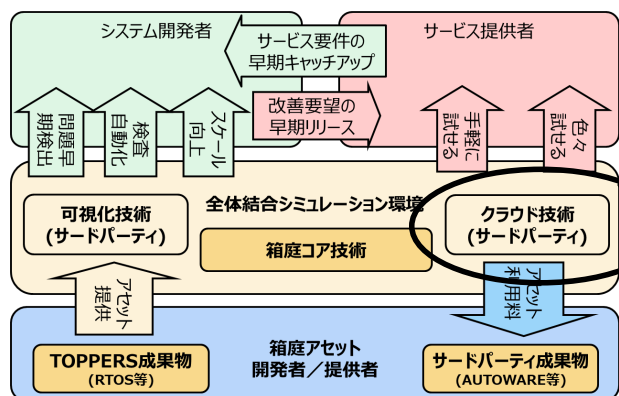
その他の狙い :

- ・ROSユーザ層に箱庭を広める (広報活動)
- ・その他チャレンジ(つくばチャレンジ/FPGAデザインコンテスト)

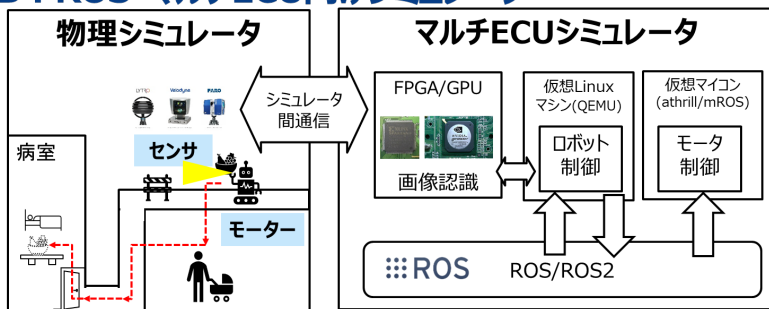
トピック

- クラウド IDE とエミュレータを利用した実機レス開発環境とも連携します！
 - 2019年/TOPPERS 活用アイデア・アプリケーション開発コンテスト受賞作品(金賞)

★クラウドIDEを箱庭の開発環境として！

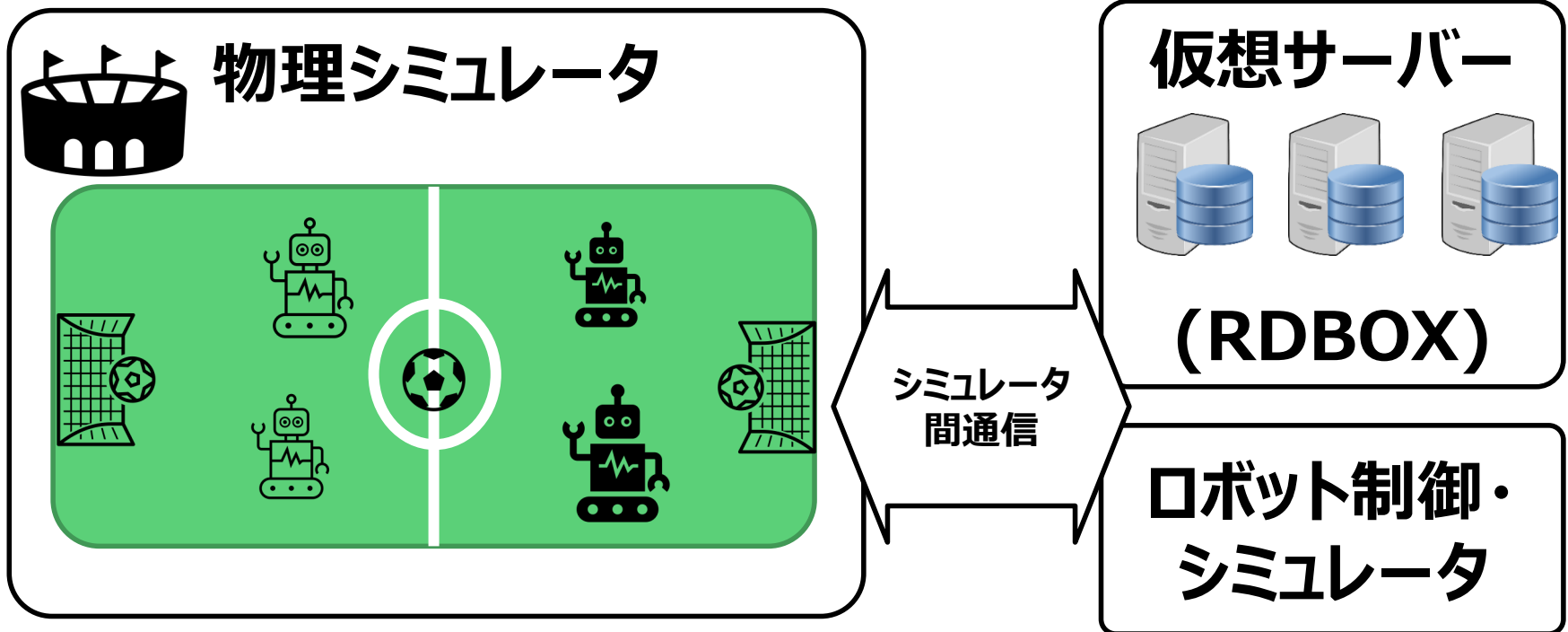


B : ROS・マルチECU向けシミュレータ



まずは、箱庭プロトタイプの開発・検証環境として利用させて頂きます！！
★庭野さんに感謝★

C : ロボット間協調動作向けシミュレータ



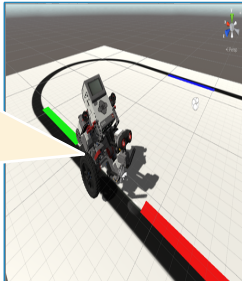
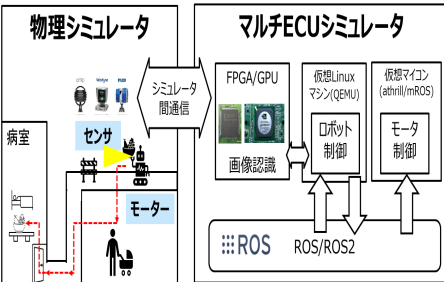
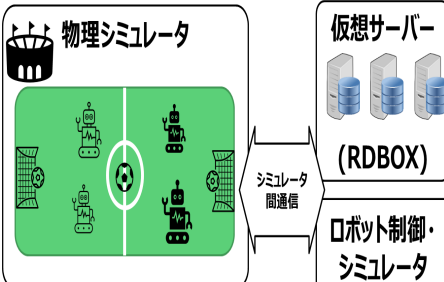
技術研鑽視点での狙い :

- ・クラウド連携方法検討
- ・ロボット間の連携方法検討(より複雑なロボットの動き/干渉に挑戦)
- ・箱庭アセットを増やす仕組みの検討

その他の狙い :

- ・RDBOX連携(開発支援仮想環境としての箱庭の実績作り)
- ・RDBOXユーザ層に箱庭を広める(広報活動/ROSCon JP 参加)

箱庭プロトタイプ・モデルの開発状況

プロトタイプモデル	技術調査	実装	動作確認	配布
<p>マイコン・シミュレータ</p> <p>制御処理(C/C++)</p> <p>EV3RT</p> <p>ASP3/ASP</p> <p>athrill</p> 	○	○	○	×
EV3RTの教育向け活用に興味のある方, ETロボコンへの参加を検討される方向けに配布を準備中				
<p>物理シミュレータ</p> <p>マルチECUシミュレータ</p> 	△	△	△	×
仮想マイコン(docker/athrill/mROS)とUnity間連携完了 FPG/GPUおよびLinux(QEMU)との連携方法を検討予定				
<p>物理シミュレータ</p> <p>仮想サーバー</p> <p>(RDBOX)</p> <p>ロボット制御・シミュレータ</p> 	△	×	×	×
RDBOXとの連携方法/デモ内容等について構想検討中				

目次

1. 箱庭プロトタイプ・モデルの紹介

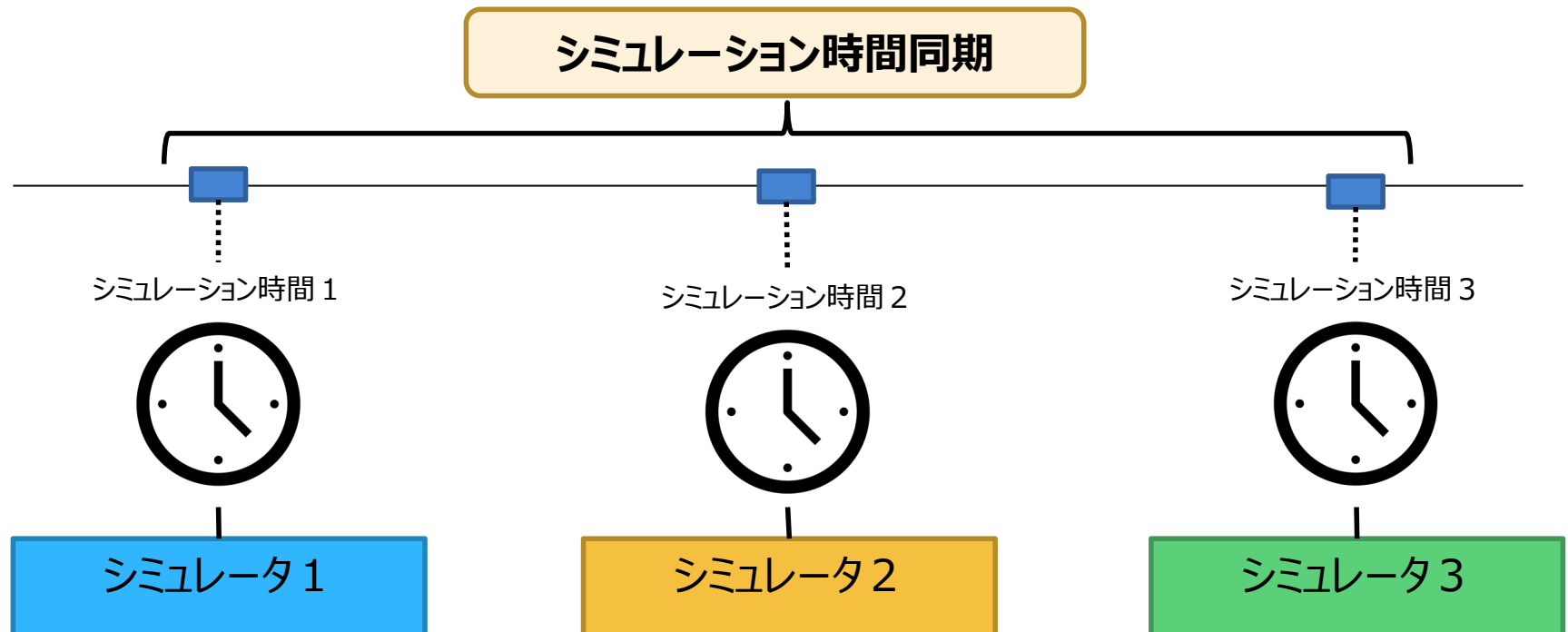
- ETロボコン向けシミュレータ
- ROS・マルチECU向けシミュレータ
- ロボット間協調動作向けシミュレータ(RDBOX連携)
- 箱庭プロトタイプ・モデルの開発状況

2. シミュレーション時間同期機構の紹介

- シミュレーション時間同期の仕組み
- 時間同期の実装方式(UDP方式/MMAP方式)
- 適用事例紹介

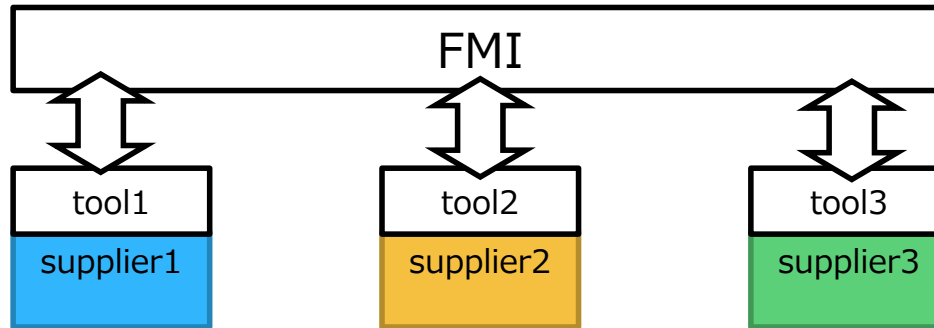
シミュレーション時間と時間同期の背景

- **箱庭の構成要素**
 - 様々なシミュレータが混在を許す環境です
- **シミュレーション時間**
 - 各シミュレータはそれぞれ固有のシミュレーション時間を持ちます
- **シミュレーション時間同期**
 - 各シミュレータが独立して動作するとシミュレーション実行タイミングがズレます
 - 箱庭環境では, これらのシミュレーション時間を同期させる方法を確認します



既存のシミュレータ時間同期方式

- FMI (Functional Mock-up Interface)
 - 欧州の公的プロジェクトが規格化
 - シミュレーションツールに依存しないモデル接続のための共通インターフェース



- シミュレーション時間同期方式
 - ME(Model Exchange)
 - 完全に時間同期が可能
 - CS(Co-Simulation)
 - シミュレータ間の遅延時間(固定)を許容
 - 適切な遅延時間をユーザが選択する
 - 考察
 - いずれも中央制御方式であるため、精度調整は容易であるがシステム構成要素が増えると処理オーバーヘッドが高くなると考えられる。

箱庭WGで新規に検討した時間同期方式

箱庭の時間同期方式

- 中央制御方式ではなく、並列化が容易な分散制御方式でのシミュレーション時間同期方式を検討

仕組み(ハーモニー)

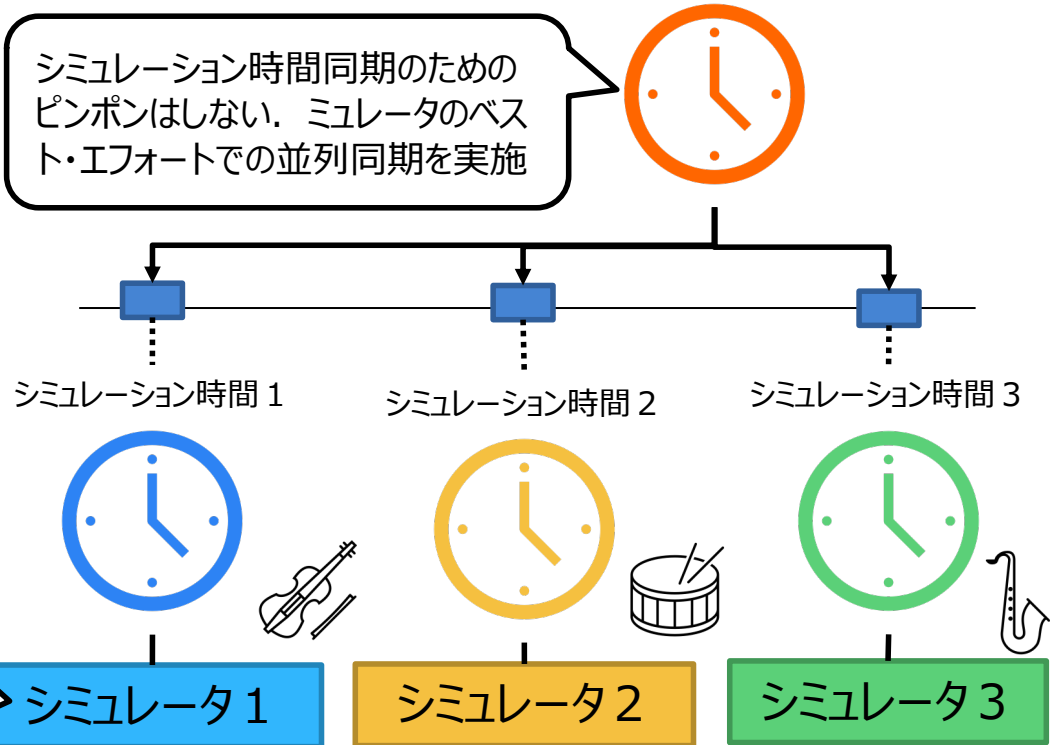
- 各シミュレータは 箱庭時間(マエストロ)をみながら、シミュレーション時間調整し時間同期する(ハーモニー)



箱庭時間

シミュレーション時間同期のためのピンポンはしない。ミュータのベスト・エフォートでの並列同期を実施

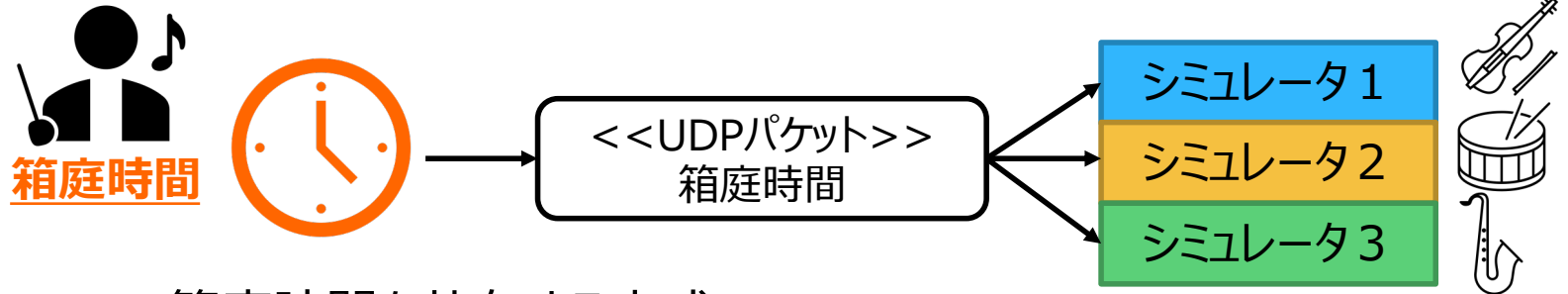
- 箱庭時間より早い場合
 - シミュレーション時間を遅くする
- 箱庭時間より遅い場合
 - シミュレーション時間を早くする
- 各シミュレータ時間を可視化
 - 時間同期の程度を定量化
 - 環境スペックの妥当性を評価・調整



各シミュレータは箱庭時間に対して自分のシミュレーション時間の進め方を調整する

シミュレータ間の通信方式(UDP/MMAP)

- 箱庭時間の通信方式として以下の2つの方式があります
 - UDPで箱庭時間を共有する方式
 - メリット：各シミュレータは別PCに配置可能であり，負荷分散可能
 - デメリット：遅延あり

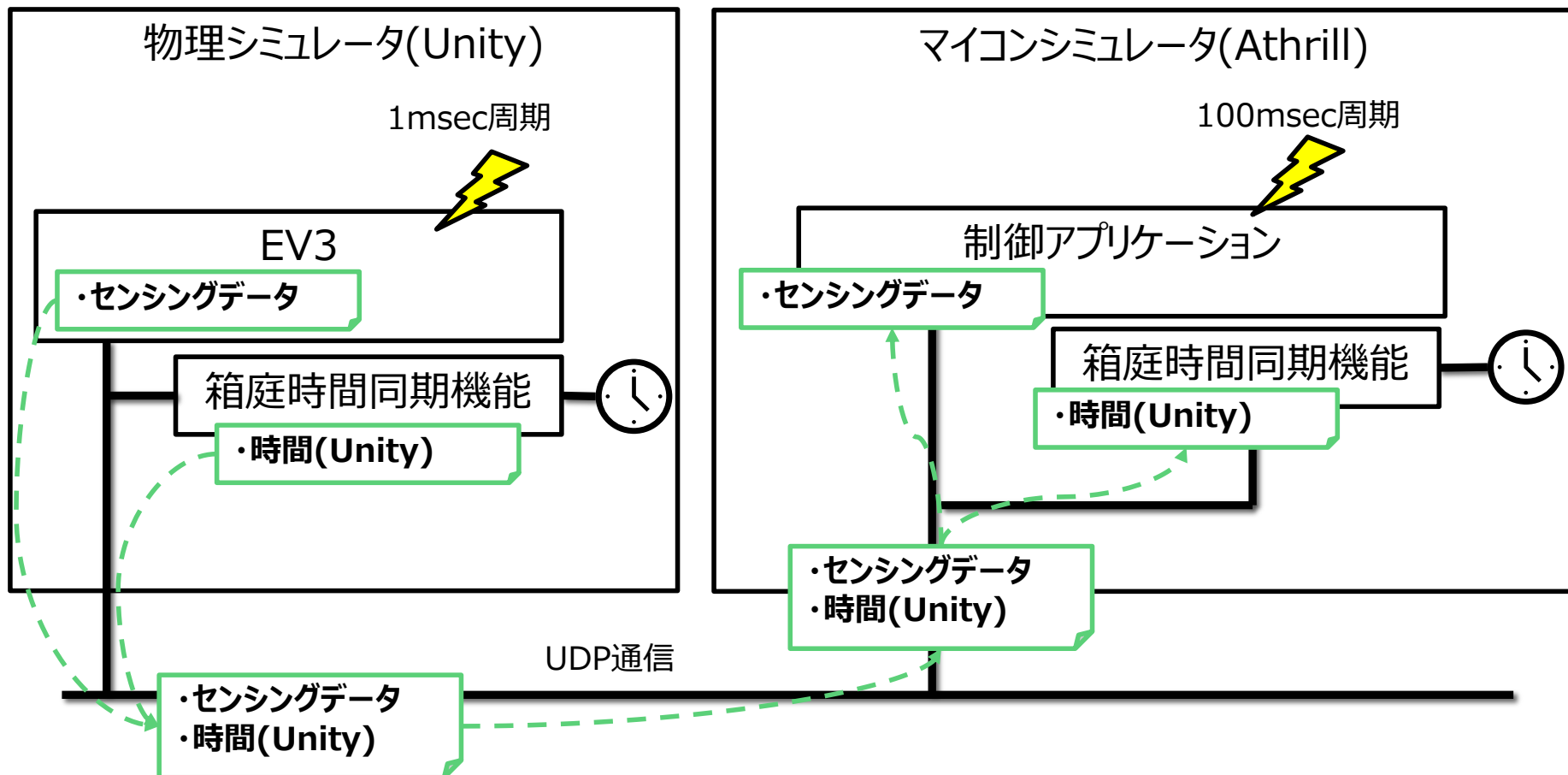


- MMAPで箱庭時間を共有する方式
 - メリット：遅延なし
 - デメリット：箱庭時間管理機構と隠しシミュレータは同一PC内に配置する必要あり



適用事例紹介(単体ロボット向けの場合の構成)

- UDP方式(MMAP方式も可能)
 - 物理シミュレータ(Unity)のセンシングデータ送信タイミングで箱庭時間通知する
 - マイコンシミュレータ(athrill)が箱庭時間にあわせて時間調整する



適用事例紹介(デモ動画)

[デモ内容]

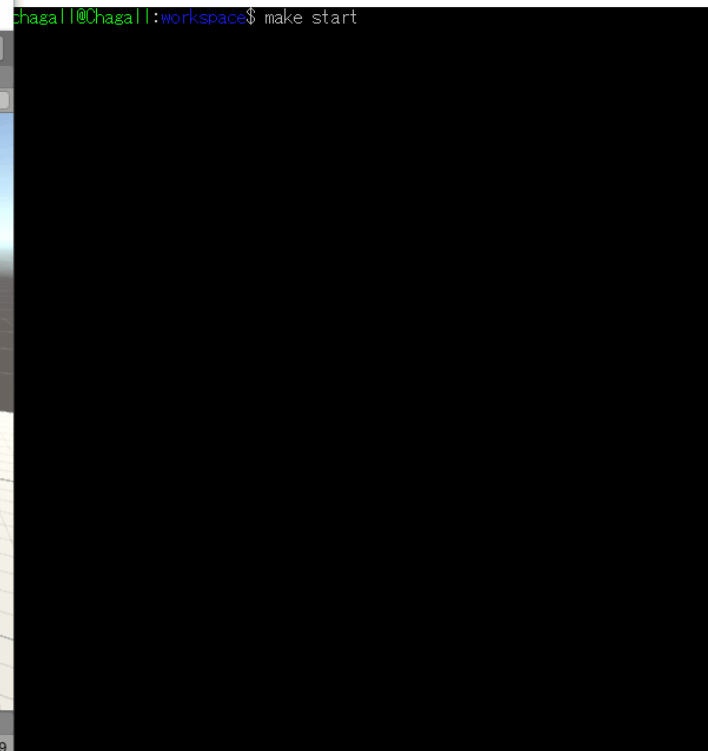
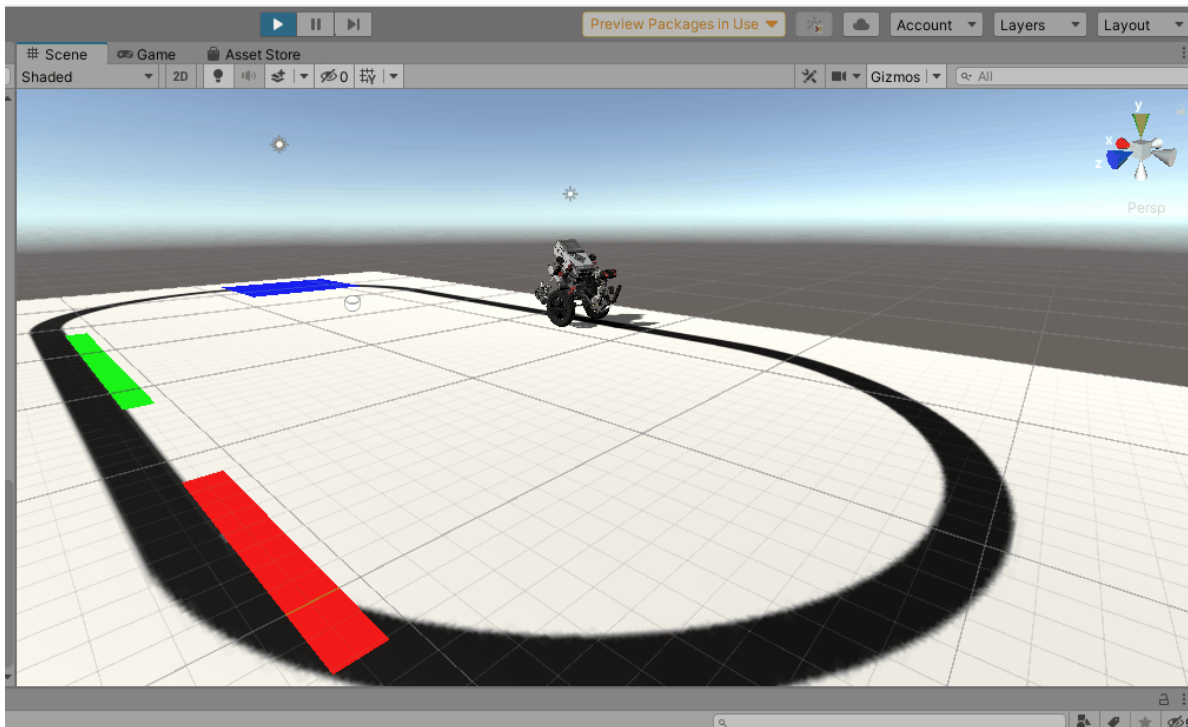
制御アプリケーションにより, HackEVがライントレースします.

走行路にはカラーコードを配置しており, 制御アプリケーションの色認識により, 加速/減速/停止します.

赤 : 加速, **緑 : 減速**, **青 : 停止**

[Unity]

[Athrill]



[条件]

Unityのシミュレーション精度 : 5msec
制御アプリケーション周期 : 10msec

HackEVのUnityアセットは, ETロボコン実行委員会より提供いただいたデータを基に作成しています. 実行委員会の皆さまに深く感謝いたします.

ただし本アセットはETロボコンの本番環境とは異なりますので, 大会に参加予定の方はご注意ください. また, 本アセットは, 個人利用または教育利用に限定してご利用ください.

謝辞・特記事項

- Unityパッケージの設計と作成にあたっては、宝塚大学 東京メディア芸術学部 吉岡章夫准教授および学部生の杉崎涼志さん、木村明美さん、千葉純平さんにご協力いただきました。
- HackEVのUnityアセットは、ETロボコン実行委員会より提供いただいたデータを基に作成しています。実行委員会の皆さまに深く感謝いたします。
ただし本アセットはETロボコンの本番環境とは異なりますので、大会に参加予定の方はご注意ください。また、本アセットは、個人利用または教育利用に限定してご利用ください。
- 本資料は、ユニティ テクノロジーズまたはその関連会社がスポンサーとなったり、ユニティ テクノロジーズまたはその関連会社と提携しているものではありません。
本資料に掲載された [Unity の登録商標一覧](#)に含まれる Unity の登録商標はすべて、ユニティ テクノロジーズまたはその米国や他の国々に所在する関連会社の登録商標または商標です。