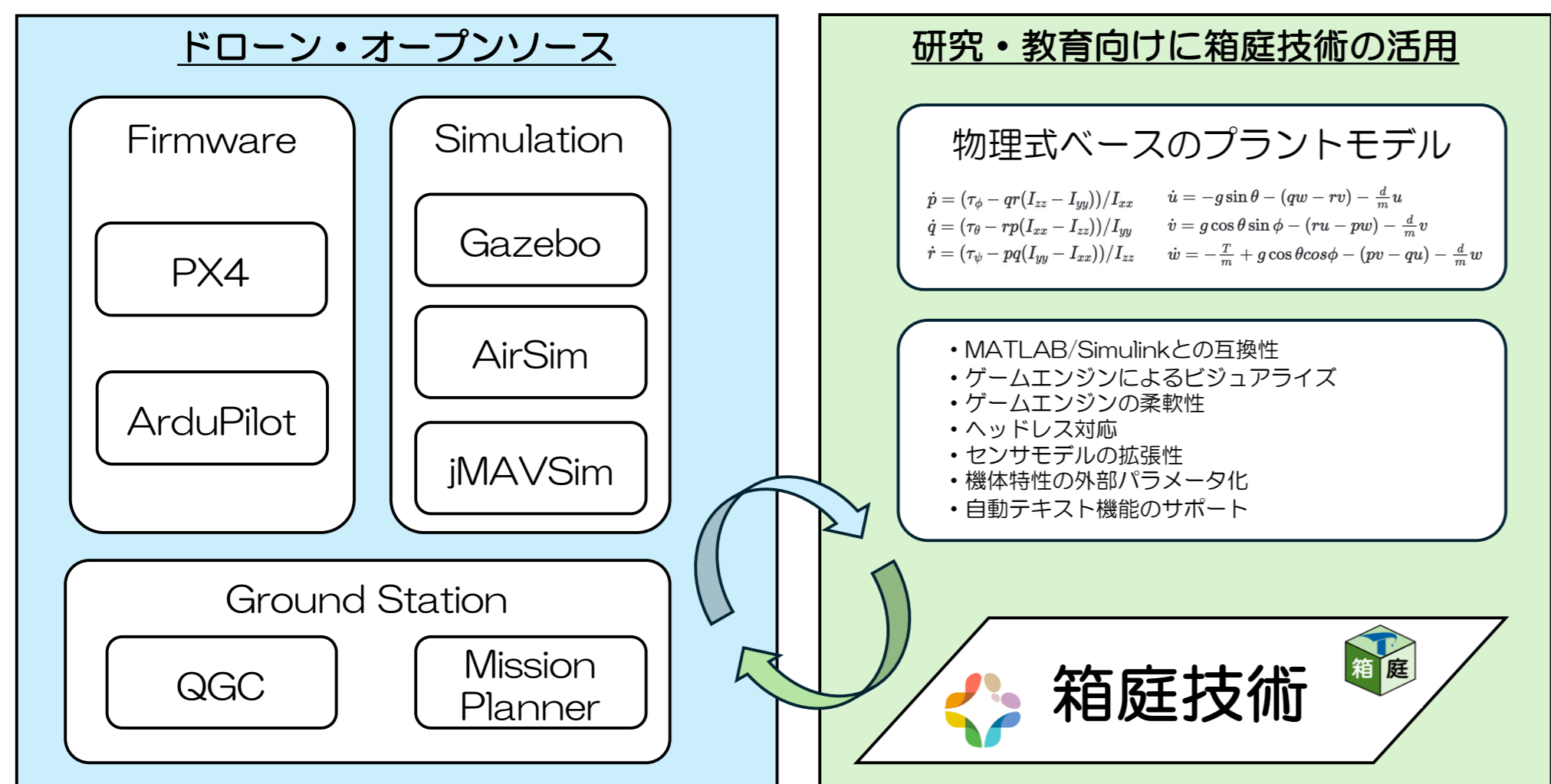


箱庭技術を活用したPX4 向けドローンシミュレータの検討

森 崇 (箱庭ラボ)、平鍋 健児 (永和システムマネジメント)、
高田 光隆 (名古屋大学)、久保秋 真 (チェンジビジョン)、
細合 晋太郎 (東京大学)、○高瀬 英希 (東京大学)

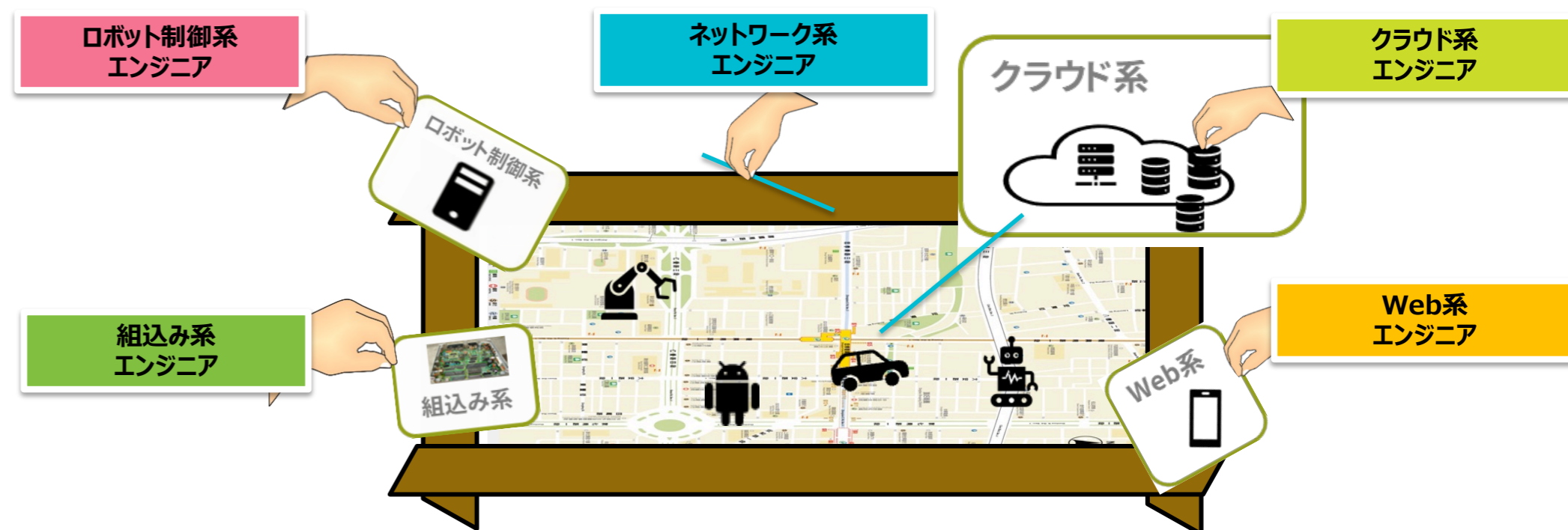
①背景と目的

オープンソース&シミュレーション技術でドローン開発を加速する！



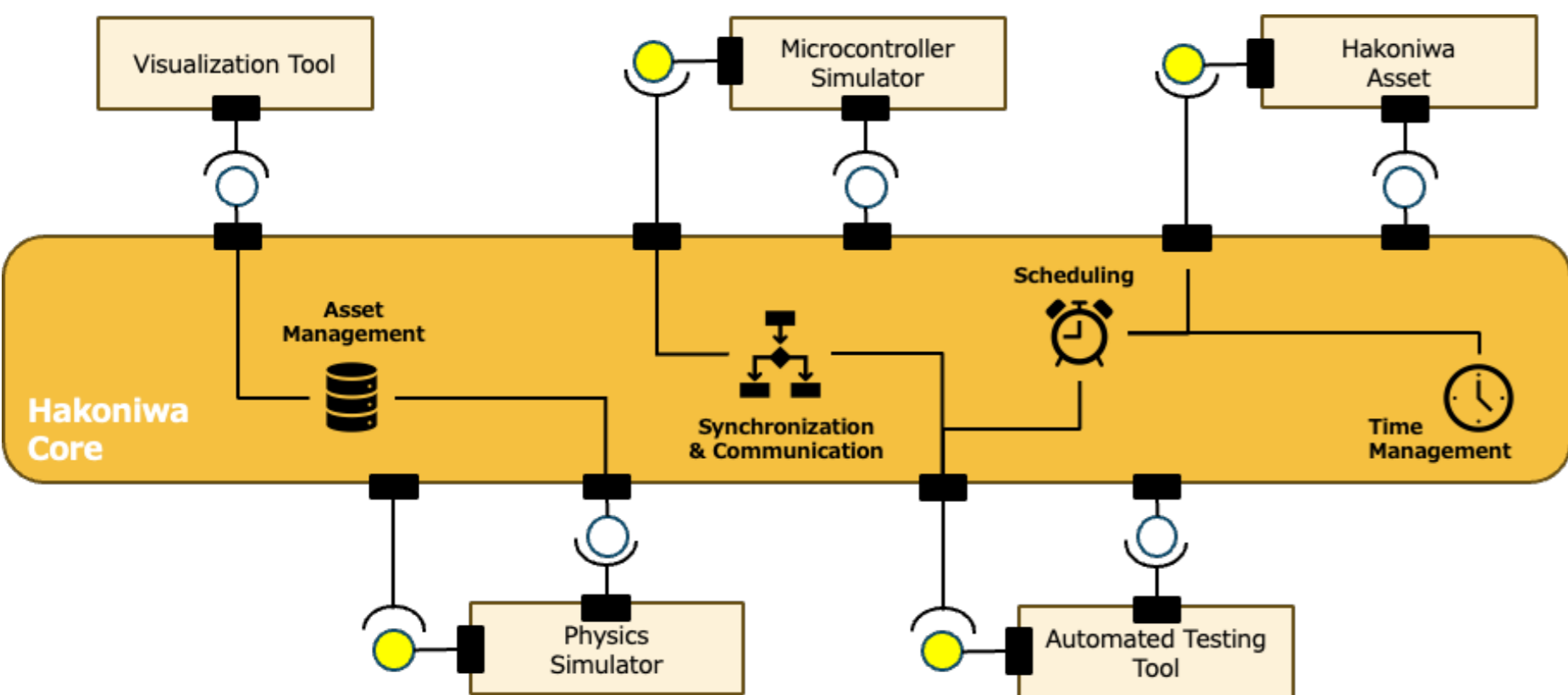
②箱庭のコンセプト

箱の中に、様々なモノをみんなの好みに配置して、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！



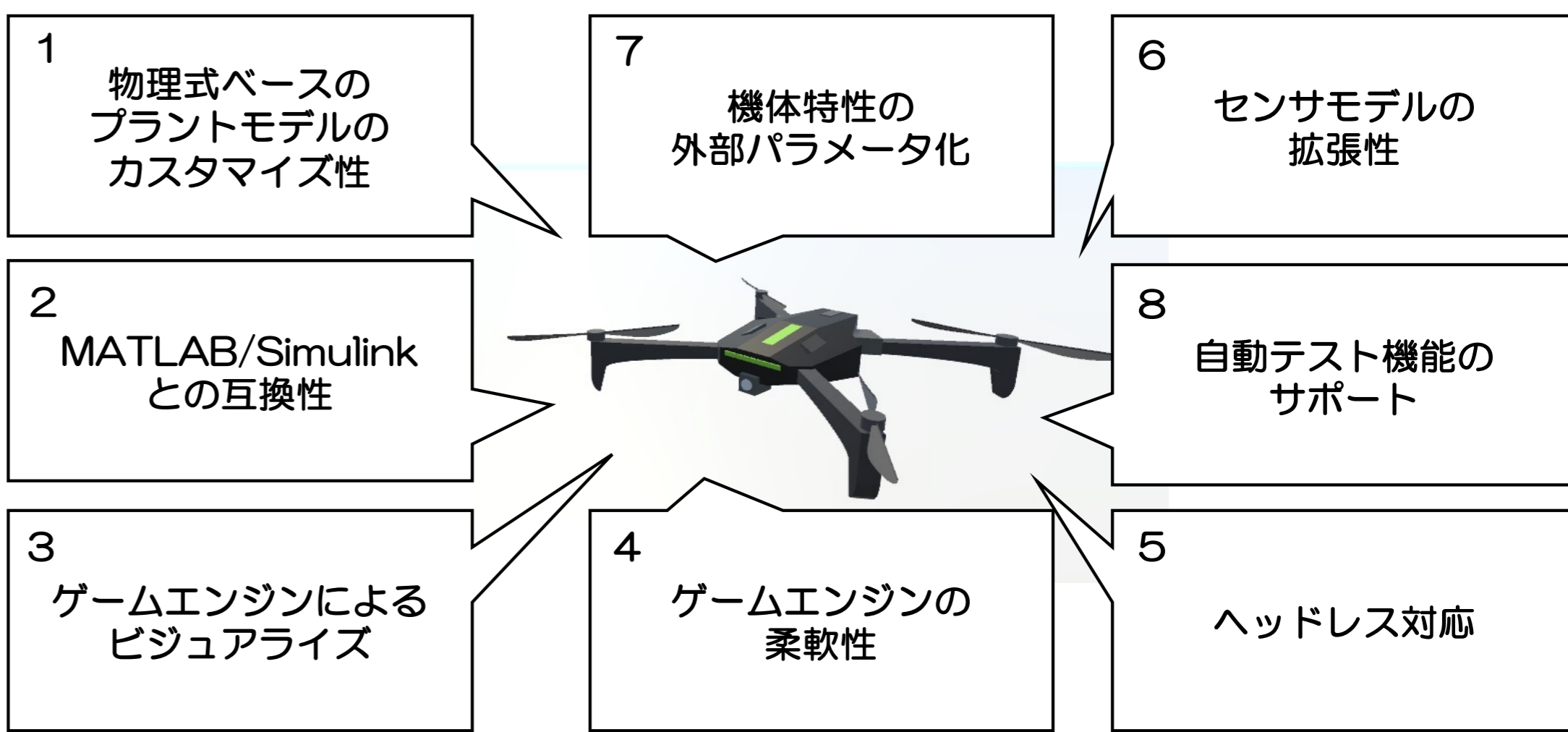
③箱庭の主な機能

- 時間管理 : シミュレーション時間の管理・調歩
- 同期・通信 : アセット間のデータ交換、イベント通知
- スケジューリング : アセットの実行タイミングの管理
- アセット管理 : アセットのライフサイクル管理等

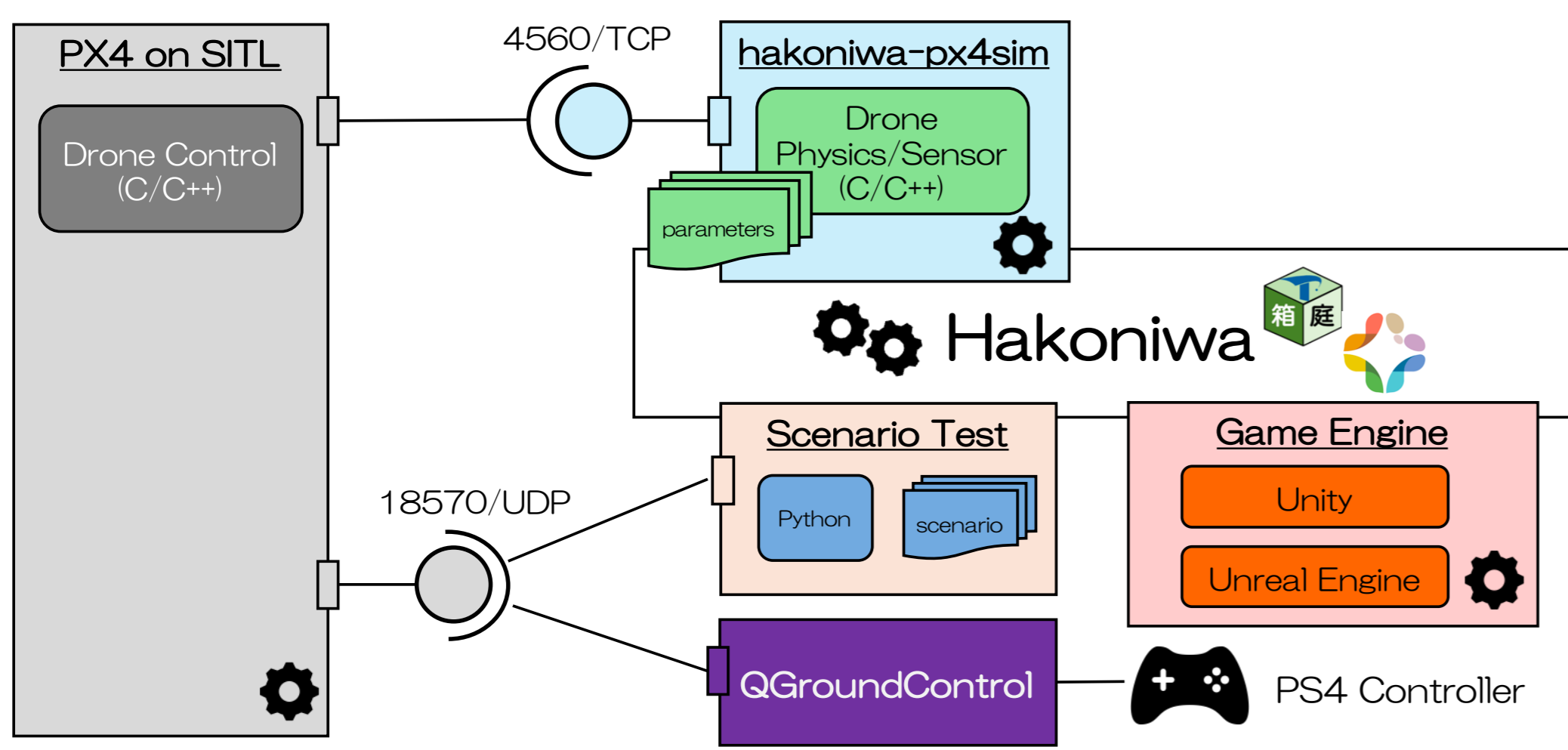


④ドローンシミュレータ要件

教育・研究開発向けドローン開発で重要と考える要件



⑤ドローンシミュレータのアーキテクチャ



箱庭は様々なソフトウェアを繋げるハブです！

⑥実装機能および既存環境との比較

教育・研究開発向けドローン開発で重要と考える要件での比較結果

要件	Gazebo	jMAVSim	AirSim	本研究
プラントモデルのカスタマイズ性	○	×	○	○
MATLAB/Simulink との互換性	○	×	○	△
ゲームエンジンによるビジュアライズ	△	△	○	○
ゲームエンジンの柔軟性	×	×	△	△
ヘッドレス対応	○	○	○	△
センサモデル拡張性	○	×	○	○
機体特性の外部パラメータ化	×	×	×	○
自動テストの機能サポート	○	○	○	○

[凡例] ○ : サポート、△ : 部分サポート、× : 未サポート

⑦プラントモデル

「ドローン工学入門」のクアドコプタの物理式をC/C++言語で実装し、数値シミュレーションを実現

運動方程式 Body Frame

速度, 加速度(並進)

$$\begin{aligned} \dot{p} &= (v_0 - qv(I_{xx} - I_{yy}))/I_{xx} & \dot{u} &= -g \sin \theta - (qv - ru) - \frac{d}{m} u \\ \dot{q} &= (v_0 - rv(I_{xx} - I_{zz}))/I_{yy} & \dot{v} &= g \cos \theta \sin \phi - (ru - pv) - \frac{d}{m} v \\ \dot{r} &= (v_0 - pq(I_{yy} - I_{zz}))/I_{zz} & \dot{w} &= -\frac{x}{m} + g \cos \theta \cos \phi - (pv - qu) - \frac{d}{m} w \end{aligned}$$

ニュートンの運動方程式

関数名は、acceleration_in_body_frame

角速度, 角加速度(回転)

$$\begin{aligned} \dot{p} &= (v_0 - qv(I_{xx} - I_{yy}))/I_{xx} \\ \dot{q} &= (v_0 - rv(I_{xx} - I_{zz}))/I_{yy} \\ \dot{r} &= (v_0 - pq(I_{yy} - I_{zz}))/I_{zz} \end{aligned}$$

関数名は、angular_acceleration_in_body_frame

詳細はこちらを参照ください。
<https://speakerdeck.com/hiranabe/math-physics-and-dynamics-of-drone-in-hakoniwa>

⑧実装および動作検証結果

提案アーキテクチャでのシミュレーション基本構成を実現できた

開発画面, Unity, QGC, 箱庭起動端末, PX4起動端末

実装コード一式は、GitHubで一般公開中です。
<https://github.com/toppers/hakoniwa-px4sim>

