

# OS<sup>4</sup>

## Open Source Space Systems Simulator

五十里 哲 (いかり さとし)

東京大学

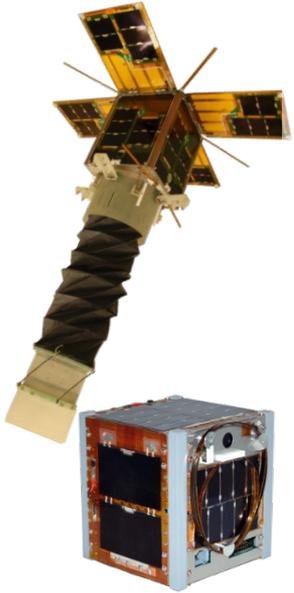
UNISECワークショップ, 2<sup>nd</sup> Dec. 2018

# 中須賀・船瀬研での超小型衛星開発の歴史

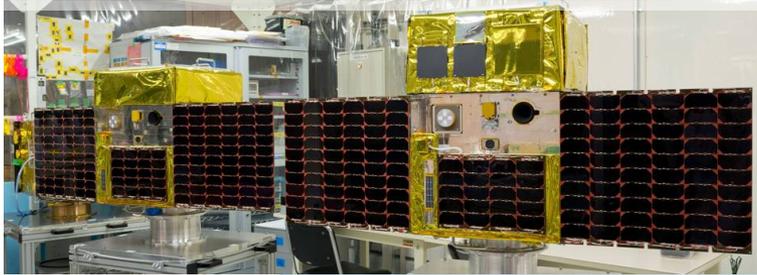
PRISM (2009): 8kg

Remote sensing

(30m GSD)

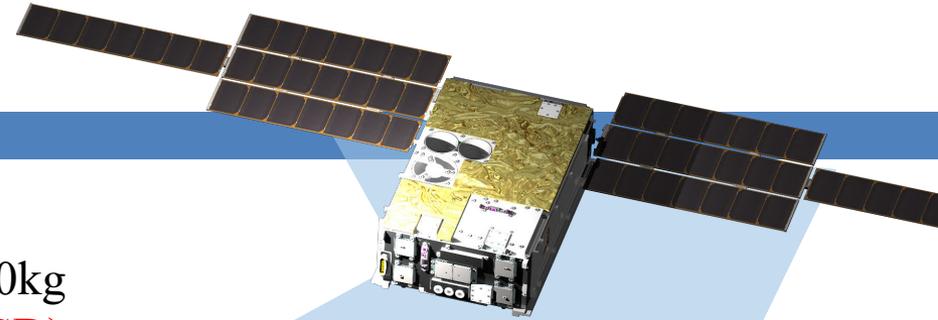


Hodoyoshi-3 and  
Hodoyoshi-4 (2014): 60kg  
Remote sensing (6m GSD)



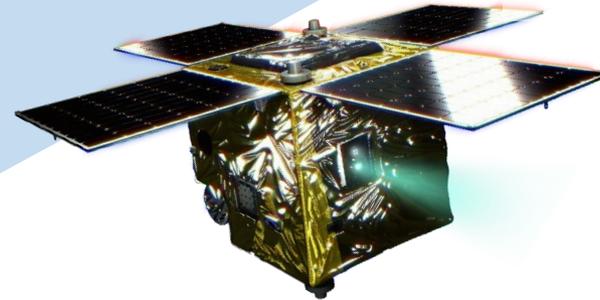
EQUULEUS(2019): 12kg

EML2 orbiter



TriCom-1R(2017):3kg

Data gathering



XI-V (2005): 1kg

Tech. demo.



PROCYON(2014): 65kg

The first interplanetary  
micro-spacecraft



XI-IV (2003): 1kg

The first CubeSat

Nano-JASMINE: 33kg

Astrometry

# 発表の要点

3

## 課題

- 超小型衛星用におけるS/W開発の低信頼性
- 宇宙開発を学ぶことへの敷居はいまだに高い

## 提案

- S/Wに主眼を置いた衛星開発
- S/Wの高効率化・高信頼性化・教育プラットフォーム

OS<sup>4</sup>:Open Source Space Systems Simulator

## 超小型衛星開発における問題

- H/W重視の開発
  - ▣ H/W開発スケジュールに押され、S/W開発はないがしろに...
  - ▣ H/W試験は定義されているが、**S/W試験**は曖昧
  - ▣ H/Wコンポーネントは**標準化・モジュール化**が進んでいるが、S/Wでは不十分
  
- S/Wの低信頼性
  - ▣ S/Wバグ多数
  - ▣ (大学で)超小型衛星開発を行っている人(学生・スタッフ)は必ずしもS/Wの専門家でない

# 宇宙開発への敷居はいまだに高い

5

- “超小型衛星”で宇宙開発への敷居は確かに下がった
- しかし、今のままで十分か？
  - UNISEC内でも超小型衛星開発ができる団体は少ない
- より、実践的な宇宙開発教育・トレーニングは？
  - CanSatから超小型衛星へ至るまでのギャップ
  - CubeSatからより、発展的なミッションへどうつなげる？
- より敷居を下げるためには？
  - 多くの人々が、宇宙開発に携わるためには？
    - 大学院生、大学生、高校生、中学生、小学生...
    - 別業種の人達：理学者、教育者、S/Wエンジニア...
  - 多くの国が、宇宙開発に携わるためには？

# 課題1、2を解決するには？

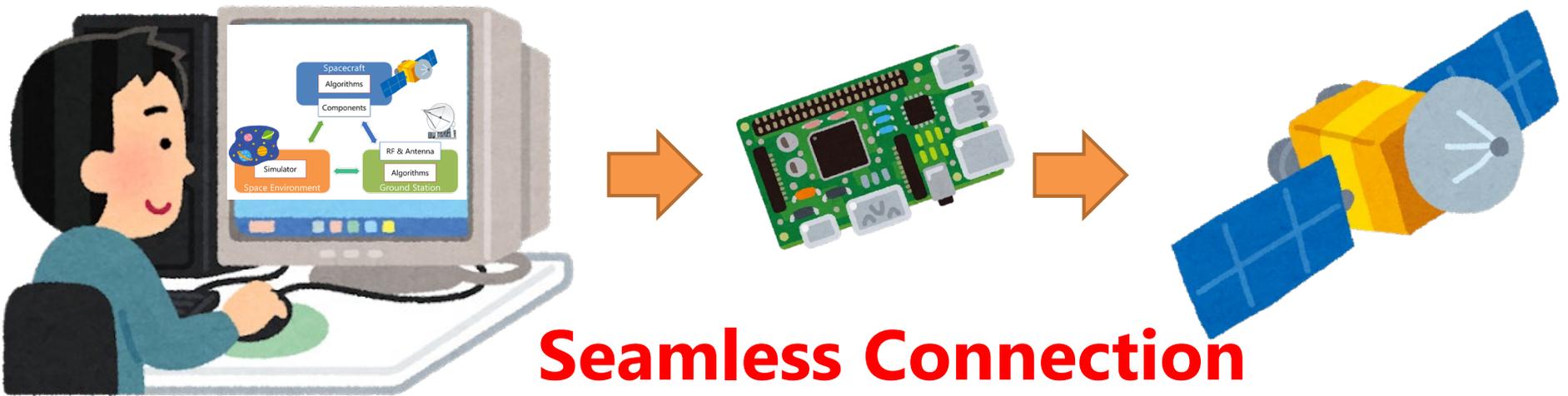
6

- “安価な”**バーチャル空間**をうまく活用する！
- 課題1：宇宙開発への敷居はいまだに高い
  - S/Wに主眼を置いた宇宙開発教育
  - バーチャル空間上での宇宙機運用体験
- 課題2：超小型衛星用におけるS/W開発の低信頼性
  - バーチャル空間上での宇宙機S/Wの動作検証
  - S/Wの標準化、モジュール化
  - テストシステムの自動化、標準化

# 五十里の提案（理想、夢）

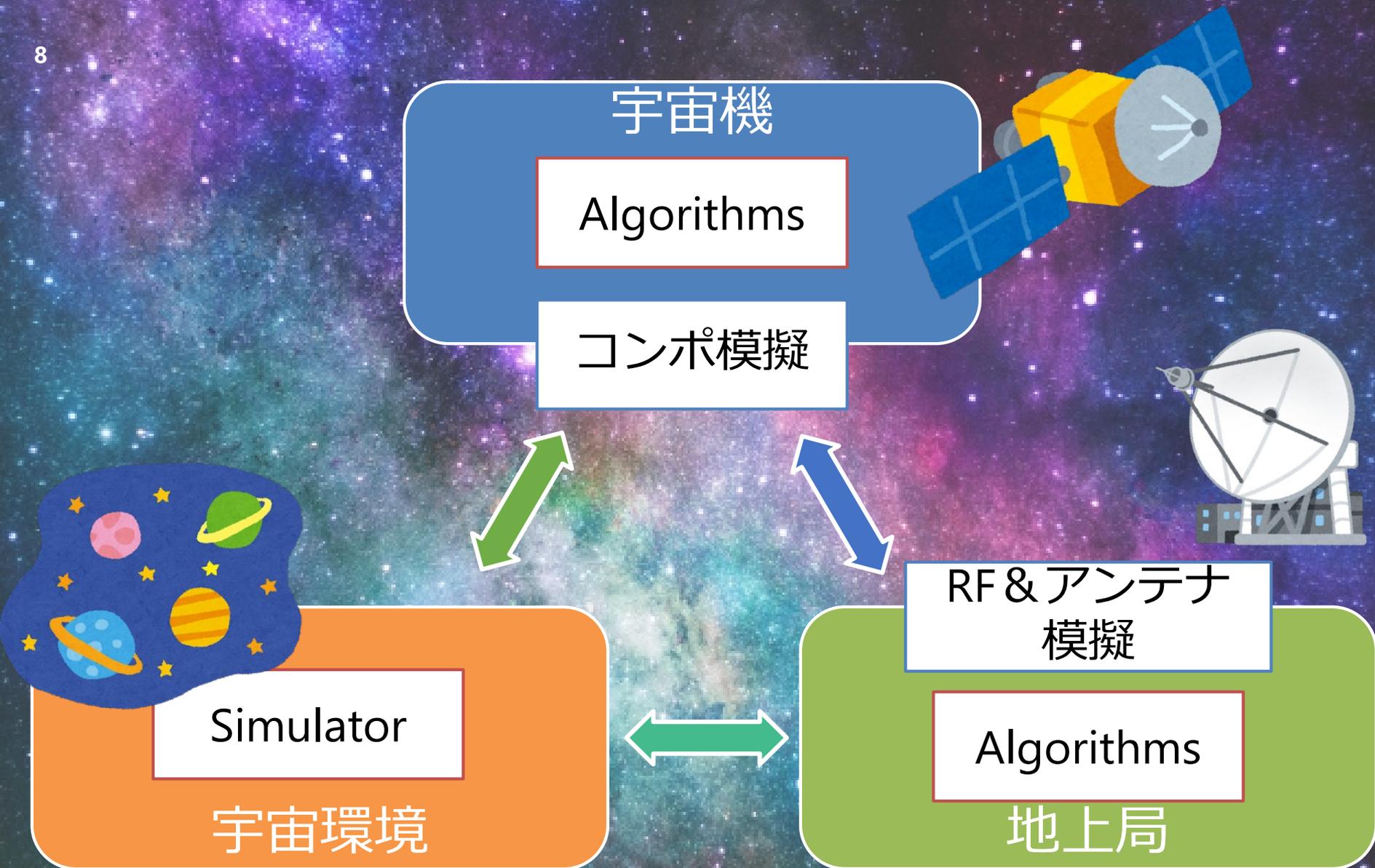
7

- PC内の**バーチャル空間**内で、宇宙開発に関わる**全てのS/W**の、**開発・評価・検証**を行いたい
  - ▣ 全てのS/W：搭載 C&DH, ADCS, 熱制御, 電源 etc...  
地上局 C&DH, 画像処理, 自動運用, etc...
- バーチャル空間は、**ただのバーチャル空間ではなく**実際のハードウェアコンポーネントとシームレスにつながらなければならない
- **Open Source** プロジェクト



# Virtual Satellite

8



# Open Source Virtual Satellite

効率的で効果的な研究開発

S/W標準化

再利用

S/Wモジュール化

宇宙開発教育

Spacecraft

S/Wの信頼性向上

Algorithms

仮想衛星の運用体験

S/Wテストの標準化・自動化

Components

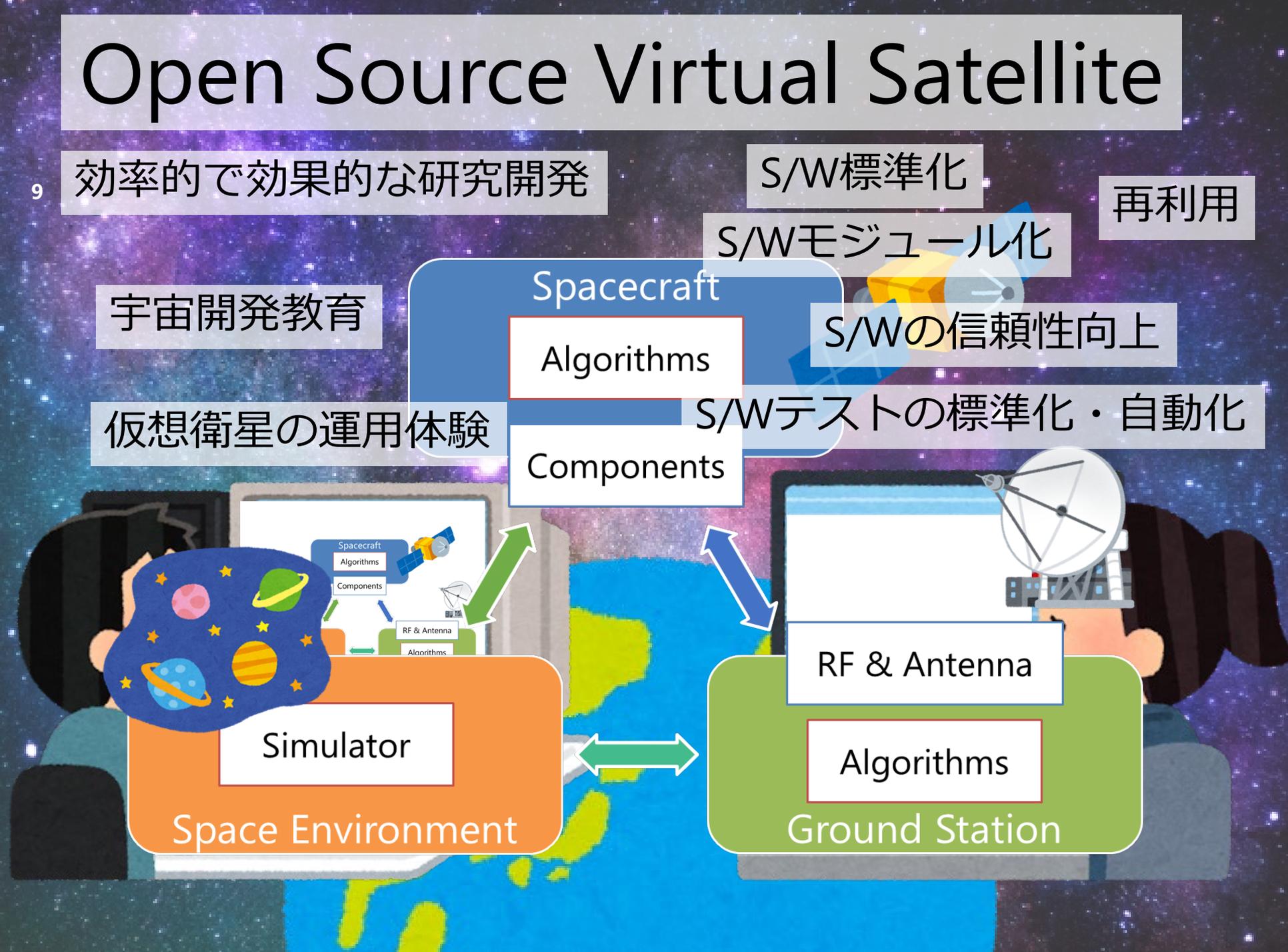
Simulator

Space Environment

RF & Antenna

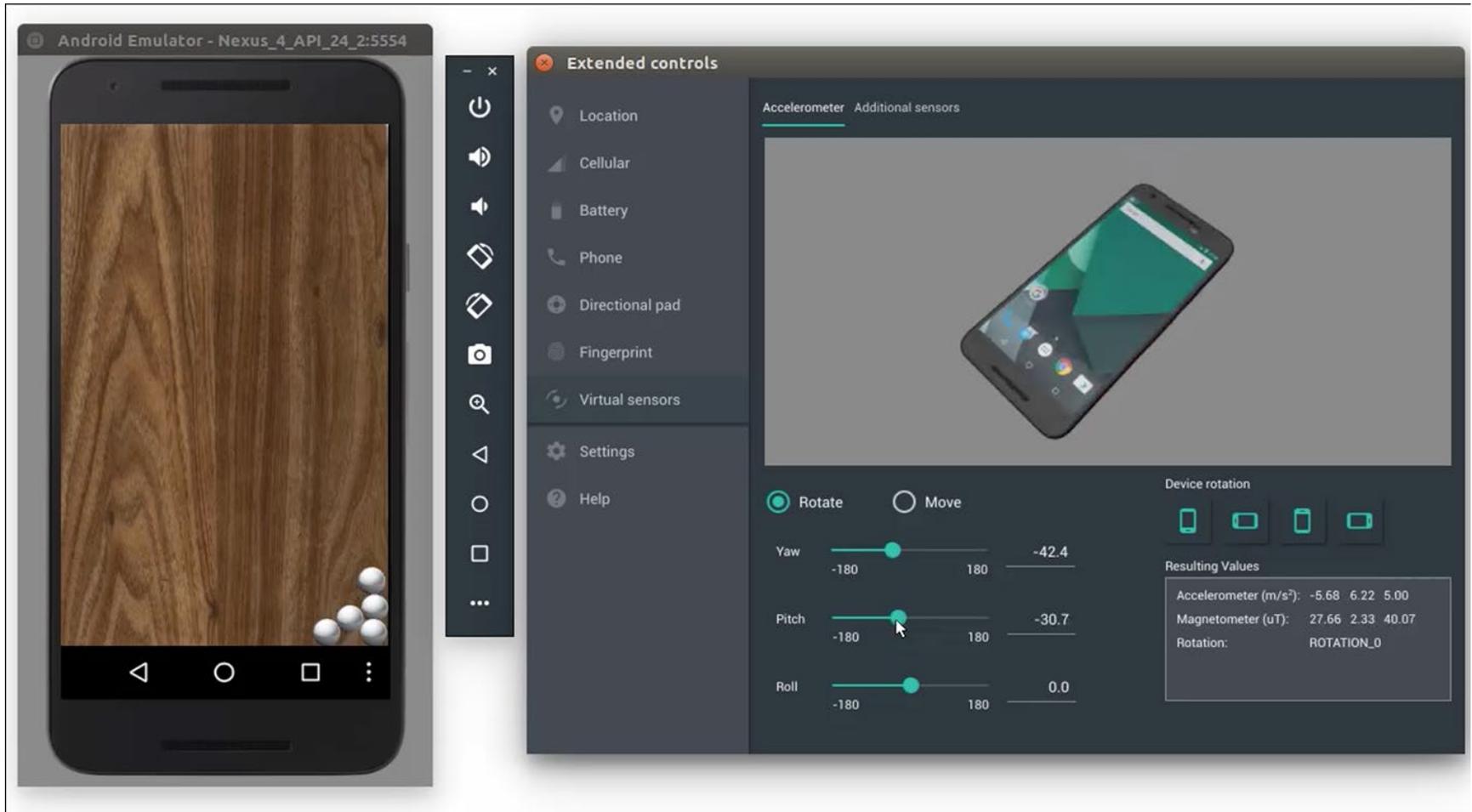
Algorithms

Ground Station



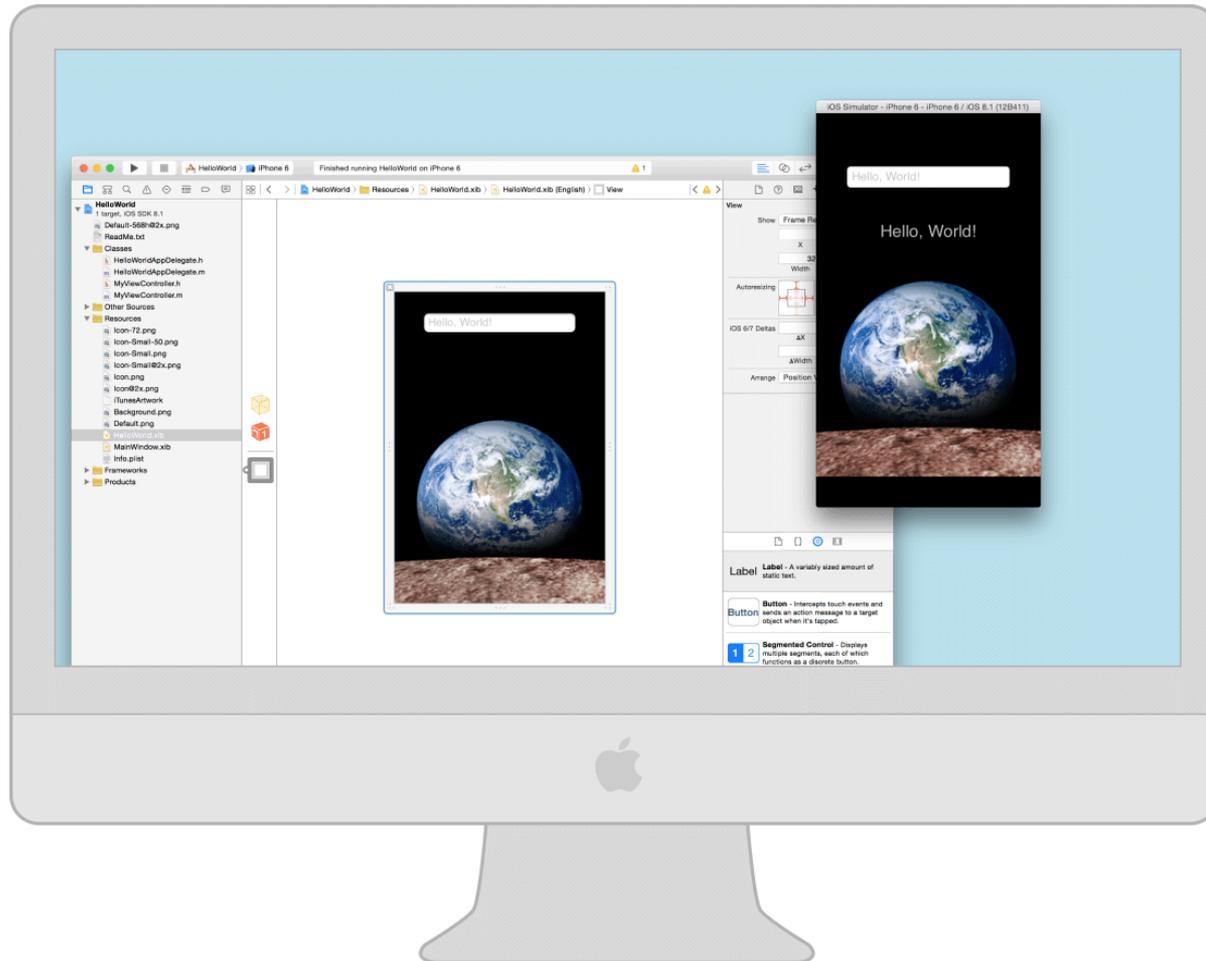
# 参考：Android Emulator

10



# 参考 : iOS simulator

11



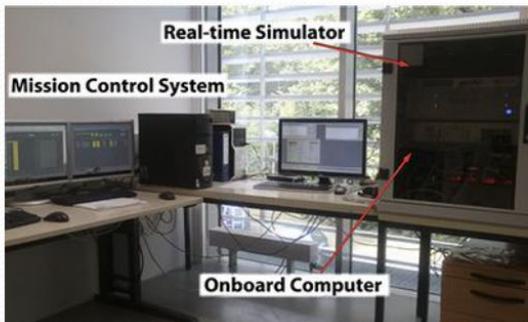
# 類似コンセプト

## 国外の例

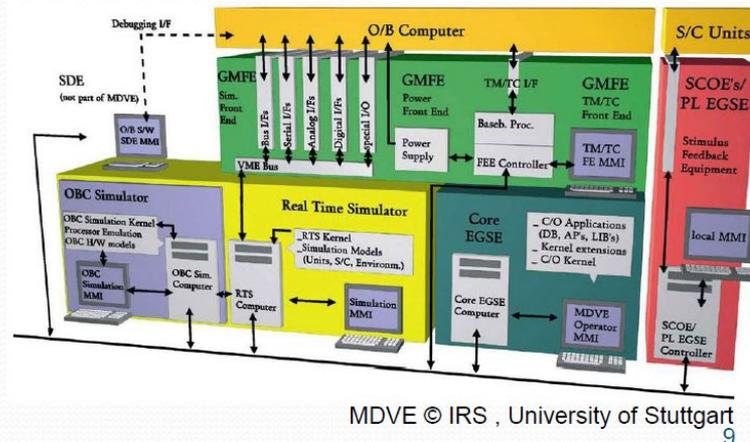
- ドイツ Stuttgart大学  
超小型衛星 FLPプロジェクト
- Model-based Development and Verification Environment
- 統合開発評価環境の利用



FLP衛星 © IRS, University of Stuttgart



© IRS, University of Stuttgart



MDVE © IRS, University of Stuttgart

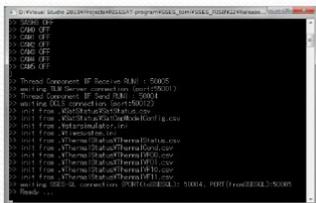
# 東北大 MEVI $\mu$ S

## 東北大での事例

- 超小型衛星統合開発評価運用環境 … “MEVI $\mu$ S”

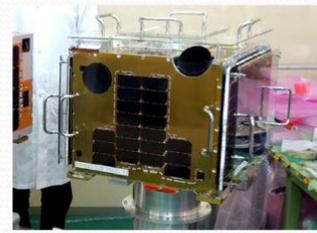
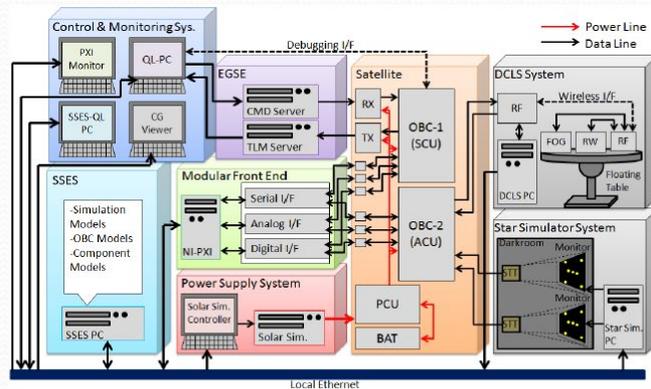


シミュレーション制御卓

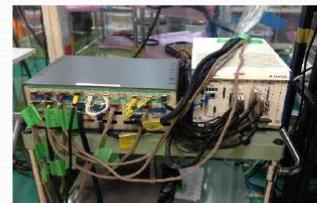


数学モデル

**MEVI $\mu$ S :**  
 Model-based Environment for Verification  
 and Integration of  $\mu$ -Satellite



衛星実機ハードウェア

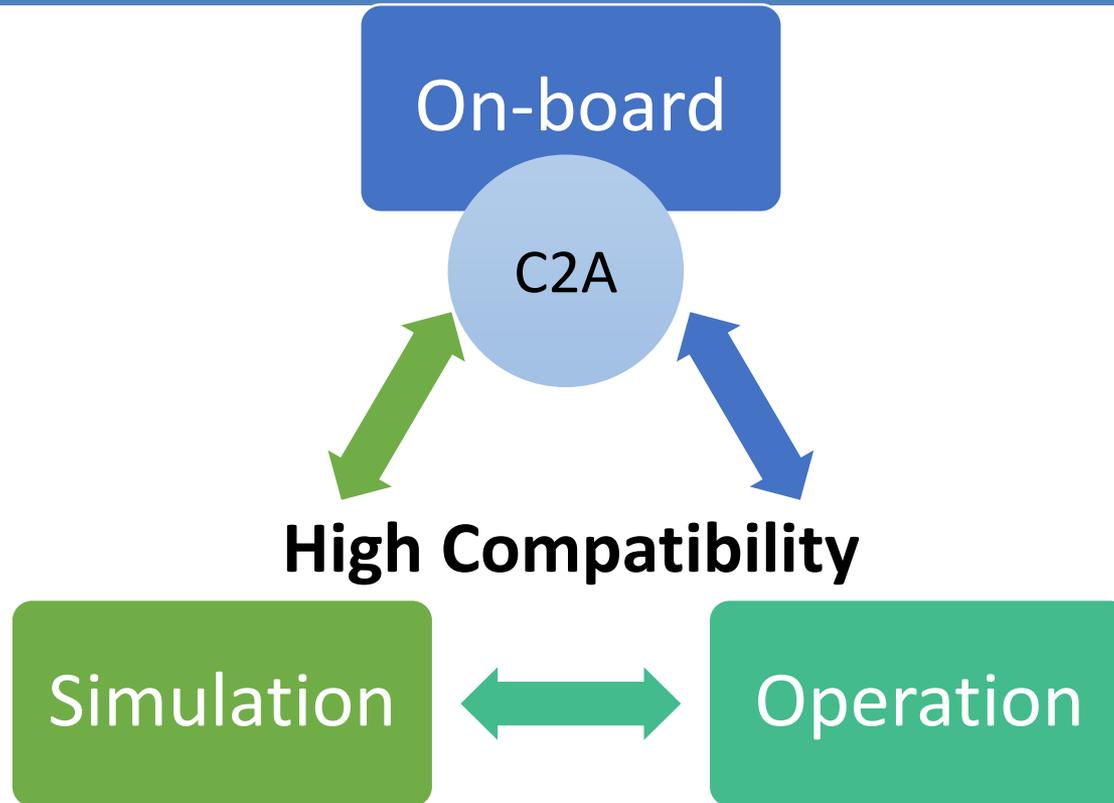


通信インターフェース

- 姿勢制御に限らず、地上局機器やソーラーシミュレータを利用した**衛星全体の試験可**
- **軌道上運用結果と地上試験データ**からセンサモデルを構築
- ⇒ センサノイズ、通信遅延を再現した現実的な環境で評価可能

# 東大SWシミュレータ(名称未定)

14



- Effective E2E development
- Effective Education

# Open Source?

15

- このプロジェクトは一つの研究者、研究室、国で行うには大きすぎる！

If you want to go further, go together!

- Virtual Satelliteのほぼ全てのコードを**無償公開**
  - コードの相互レビューによる信頼性アップ
  - 標準化・モジュール化を促進
  - 教育効果
  - 多人数・多様性による発展(**得意分野の持ち寄り**)

UNISECの活動にフィットしている！

# 参考：ROS(Robot Operating System)

16

The Construct  
Just Simulate

ROS Development Studio

Ruben Alves

Menu | Python 2

sgs = SmartGrasper()

**Pick the ball**

Try to pick the ball using the pick function.

In [ ]:

```
sgs.pick()
```

In [ ]:

```
sgs.reset_world()
```

Explore

- user\_ws
  - gazebo2rviz
  - gazebo\_ros\_pkgs
  - model\_gazebo
  - pysdf
  - ros\_controllers
  - smart\_grasping\_sandbox
  - universal\_robot

```
user src $ pwd
/home/user/catkin_ws/src
user src $ ls
CMakeLists.txt
gazebo2rviz
gazebo_ros_pkgs
model_gazebo
pysdf
ros_controllers
smart_grasping_sandbox
universal_robot
user src $
```

shadow\_robot\_smart\_grasping\_sandbox

# OSCW : Open Source Cubesat Workshop

17



2017年12月にESOCで開催されたOSCWの様子

# 6<sup>th</sup> UNISEC Global @ ISU

## Group Discussion

18

- 5か国8人のグループで議論
- 具体的なプロジェクトとして進めていくことに
  - プロジェクト名：Open Source Virtual Satellite

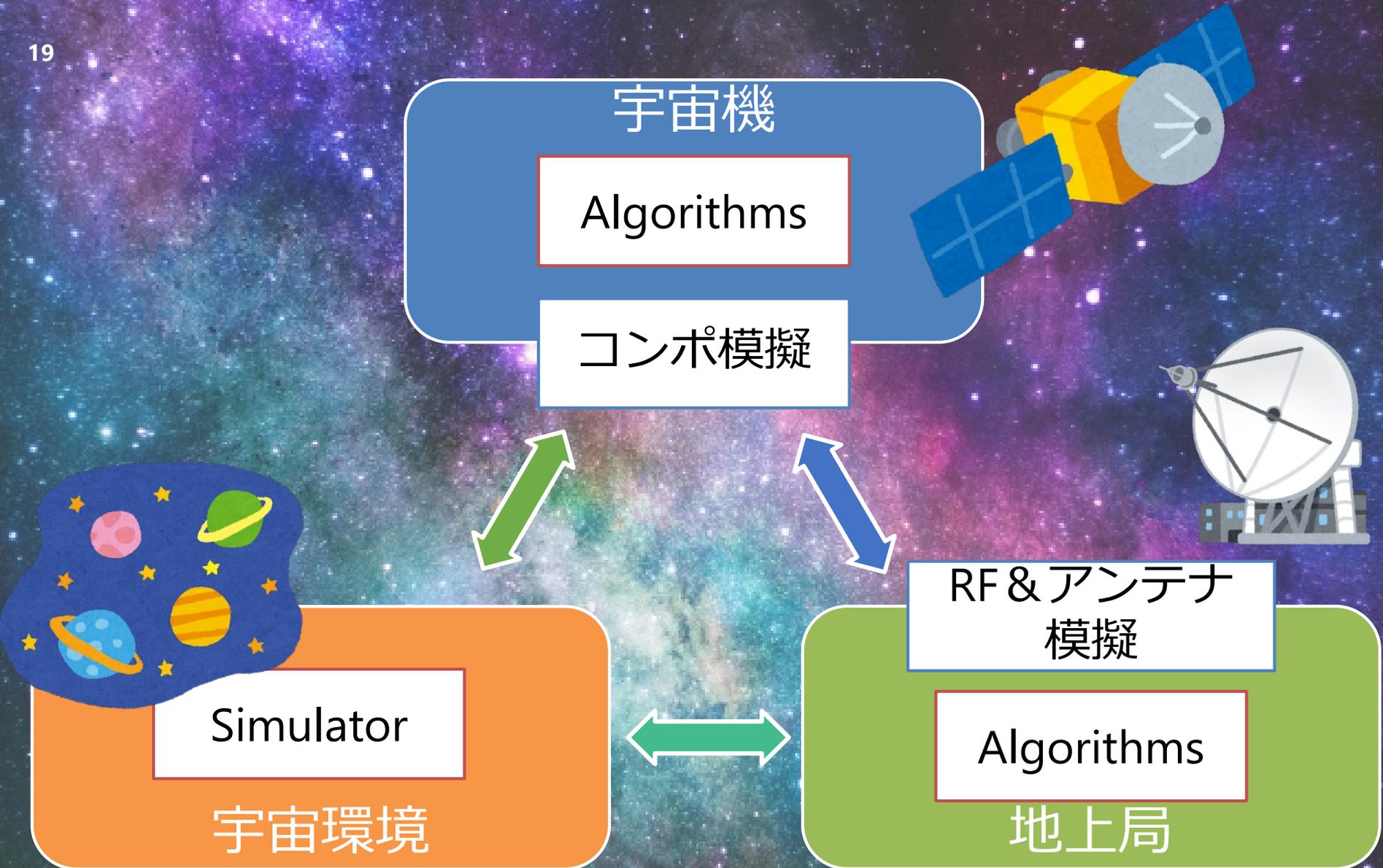
## OS<sup>4</sup>: Open Source Space Systems Simulator

- 次回MTG  
12/10 19時(JST)から(skype)
- 日本国内メンバー募集中！



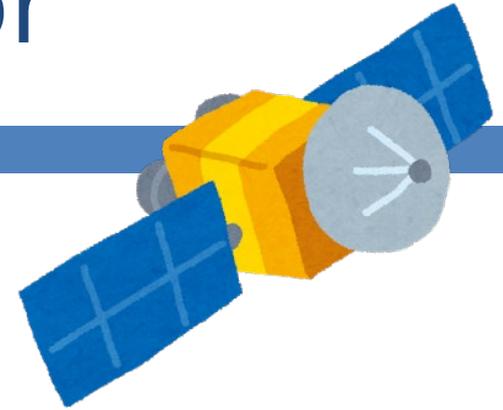
# Virtual Satellite

19



# 宇宙機 Emulator

20



- コンポの挙動を模擬
  - ▣ 電源：電源 On/Off, 電圧, 電流
  - ▣ 通信：接続状態, データプロトコル
  - ▣ 物理情報：熱, 姿勢, 軌道運動、展開機構などの入出力
  - ▣ 主にOS<sup>4</sup>プロジェクト内メンバーで開発
    - この活動が有名になってきたら、コンポ開発メーカー側から、この模擬S/Wを提供してもらえることになるかも
- OBC挙動を模擬
  - ▣ 通信：各機器との接続状態、データプロトコル
  - ▣ アルゴリズム：検証したい姿勢・軌道制御則など
  - ▣ 主にユーザーが開発

# 宇宙環境 Simulator

21

- 軌道ダイナミクス
- 姿勢ダイナミクス
- 座標変換
- 天体ダイナミクス
- 熱ダイナミクス
- 故障シミュレーション
- 主にOS<sup>4</sup>プロジェクト内メンバーで開発



# Ground Station Emulator

22

- アンテナ挙動の模擬
  - 主にOS<sup>4</sup>プロジェクト内メンバーで開発
- 地上解析S/Wの模擬
  - 衛星とのテレメトリ/コマンド処理
    - 主にOS<sup>4</sup>プロジェクト内メンバーで開発
    - 天候なども模擬する？
  - (発展的な)地上データ処理
    - 画像処理
    - 自動異常検知
    - 自動運用・プランニング
    - 主にユーザーが開発



# その他サイド機能

23

- UI/GUI
  - コンポーネント性能や宇宙機特性を決めるFile format
  - 軌道、姿勢、電源情報などのGUI
  - 他のS/WアプリケーションへのExport and import 機能
    - STK, CAD, Thermal Desktop, ESA TAN, Matlab/Simulink, ROS etc...

# 開発体制：チームメンバー

24

- Project management
- Experts of spacecraft, space environment, ground station
  - ▣ スペック、要求、テスト手法などを定義するチーム
- Core SW architects
  - ▣ 要求などからS/W構成を設計するチーム
  - ▣ S/Wエキスパートと宇宙技術エキスパートからなる小グループ
- Senior developers
  - ▣ 詳細なコーディング方針決めやコードレビューを行う
- Developers
  - ▣ コードを書く人
  - ▣ 学生大歓迎！

# やるべきこと

25

- 各大学が持っている宇宙関連S/Wの共有
  - ▣ 同一機能の抽出、標準化のラインを定める
- プロジェクト全体をPh.D.論文として進めるよう、学生・予算を確保する

# OS<sup>4</sup> : Open Source Space Systems Simulator

26

効率的で効果的な研究開発

宇宙開発教育

仮想衛星の運用体験

S/W標準化

再利用

S/Wモジュール化

S/Wの信頼性向上

S/Wテストの標準化・自動化



# まとめ

27

## 課題

- 超小型衛星用におけるS/W開発の低信頼性
- 宇宙開発を学ぶことへの敷居はいまだに高い

## 提案

- S/Wに主眼を置いた衛星開発
- S/Wの高効率化・高信頼性化・教育プラットフォーム

OS<sup>4</sup>:Open Source Space Systems Simulator

UNISEC-Japan内でも、議論を進めたい。

協力者（先生も学生も）募集中！

[ikari@space.t.u-tokyo.ac.jp](mailto:ikari@space.t.u-tokyo.ac.jp) まで気軽にご連絡ください