



ROSCon JP 2025

宇宙アプリケーションへのROS適用に向けた取り組み RACS2によるcFS-ROS2連携のISS軌道上実証

宇宙航空研究開発機構（JAXA）
研究開発部門

西下 敦青，池田 勇輝，加藤 裕基





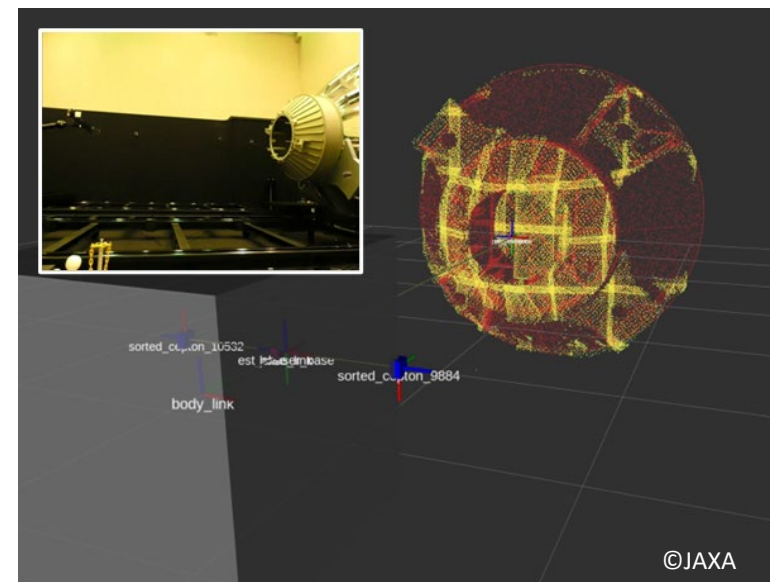
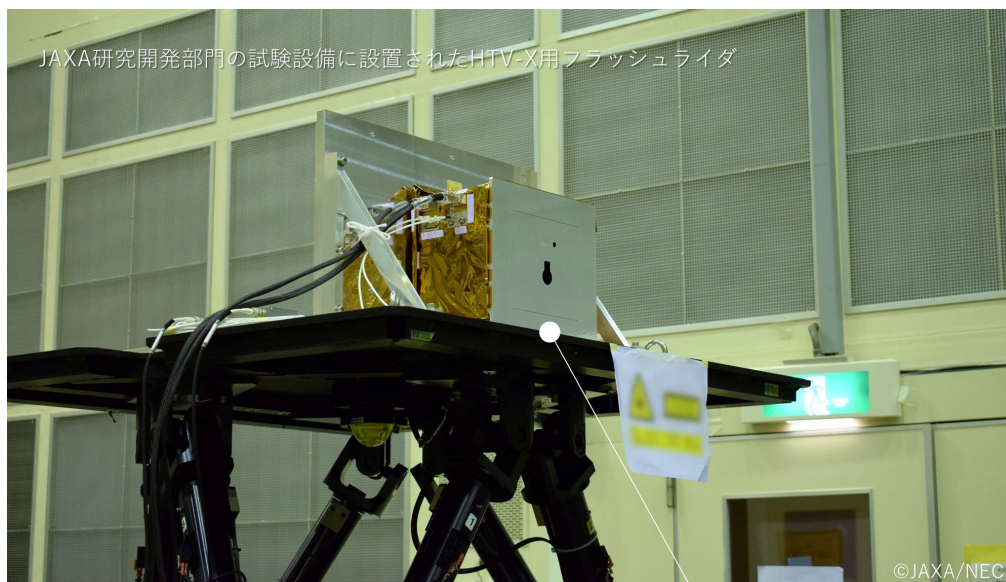
西下 敦青（にしした たいせい）

所属：JAXA 研究開発部門 第一研究ユニット

経歴：

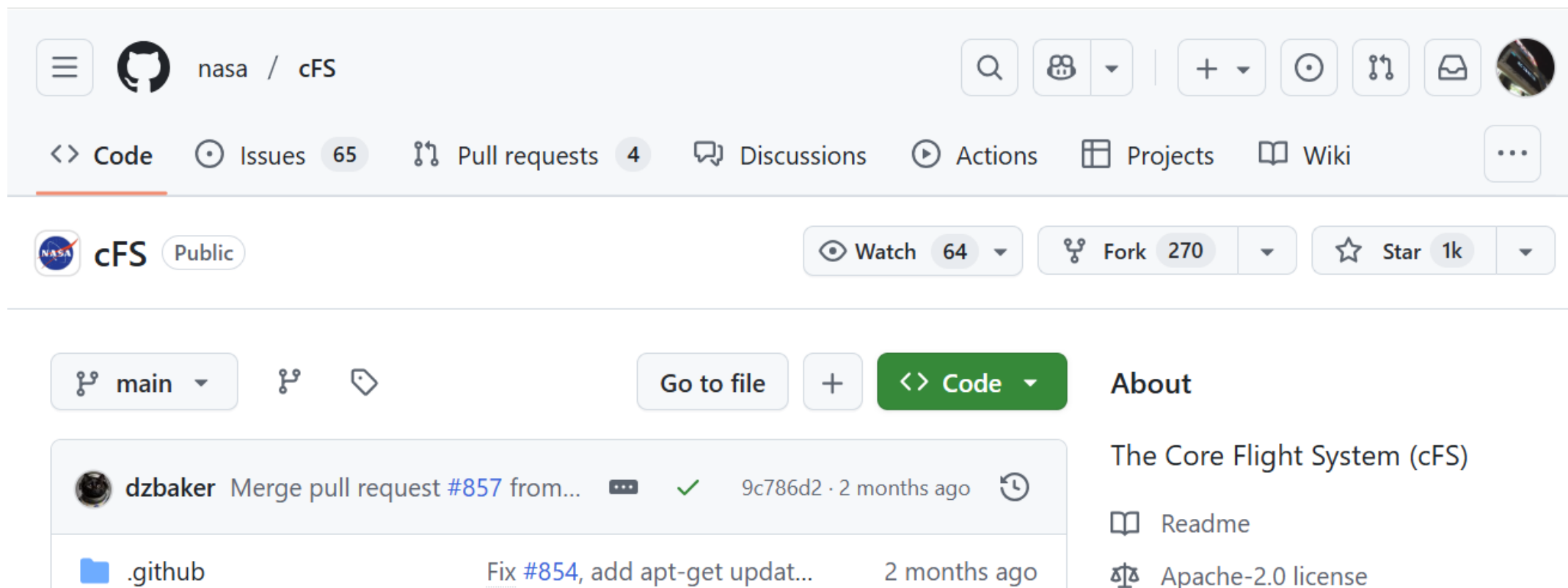
2020年にJAXAに入社（6年目）。

HTV-X用Flash LIDAR開発，JEM船内カメラロボットInt-Ball2の誘導制御開発，デブリ除去衛星用LiDAR航法アルゴリズム研究などに従事。



ソフトウェア開発フレームワークが...

オープンソースで提供されているって知っていましたか？



The screenshot shows the GitHub repository page for NASA's Core Flight System (cFS). The repository is public and has 65 issues, 4 pull requests, 64 watchers, 270 forks, and 1k stars. The main branch is 'main'. The repository description is 'The Core Flight System (cFS)'. The license is Apache-2.0. The repository was created 2 months ago. The repository is located at `https://github.com/nasa/cFS`.

Repository: nasa / cFS

Code Issues 65 Pull requests 4 Discussions Actions Projects Wiki

cFS Public Watch 64 Fork 270 Star 1k

main Go to file + <> Code About

The Core Flight System (cFS)

Readme Apache-2.0 license

dz baker Merge pull request #857 from... 9c786d2 · 2 months ago

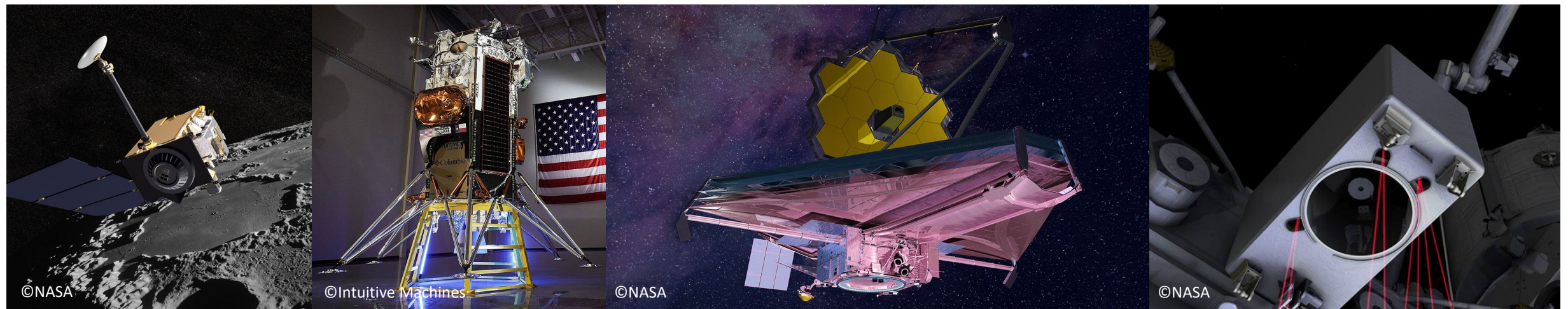
.github Fix #854, add apt-get updat... 2 months ago

cFS (Core Flight System) って??

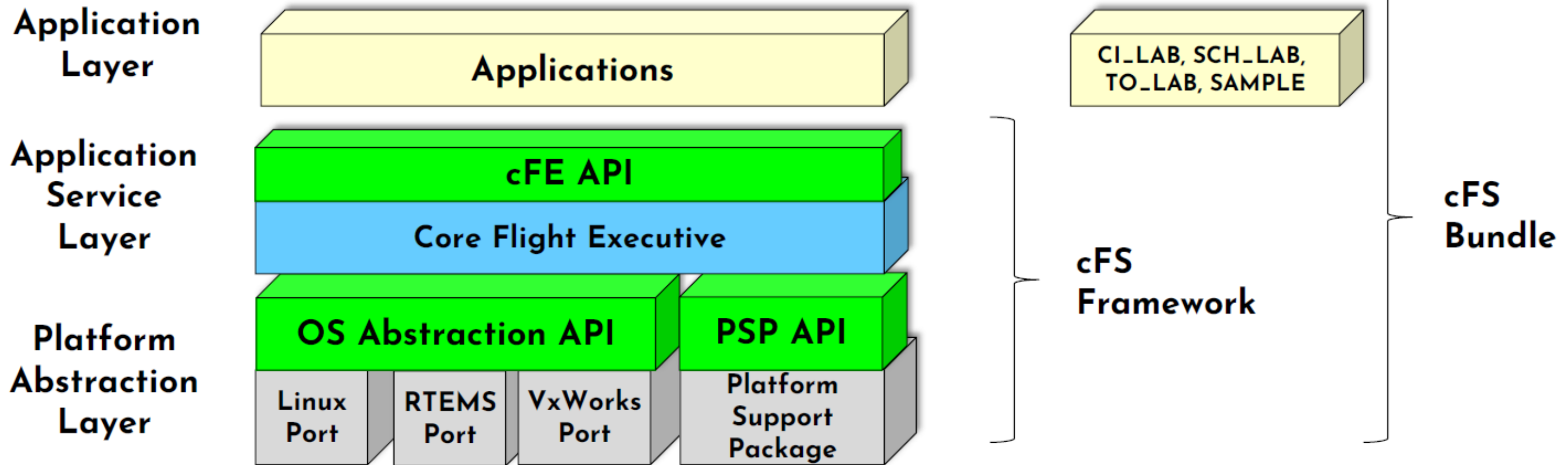
NASAで開発された、宇宙機フライトソフトウェア用のオープンソース基盤フレームワーク

主な特徴

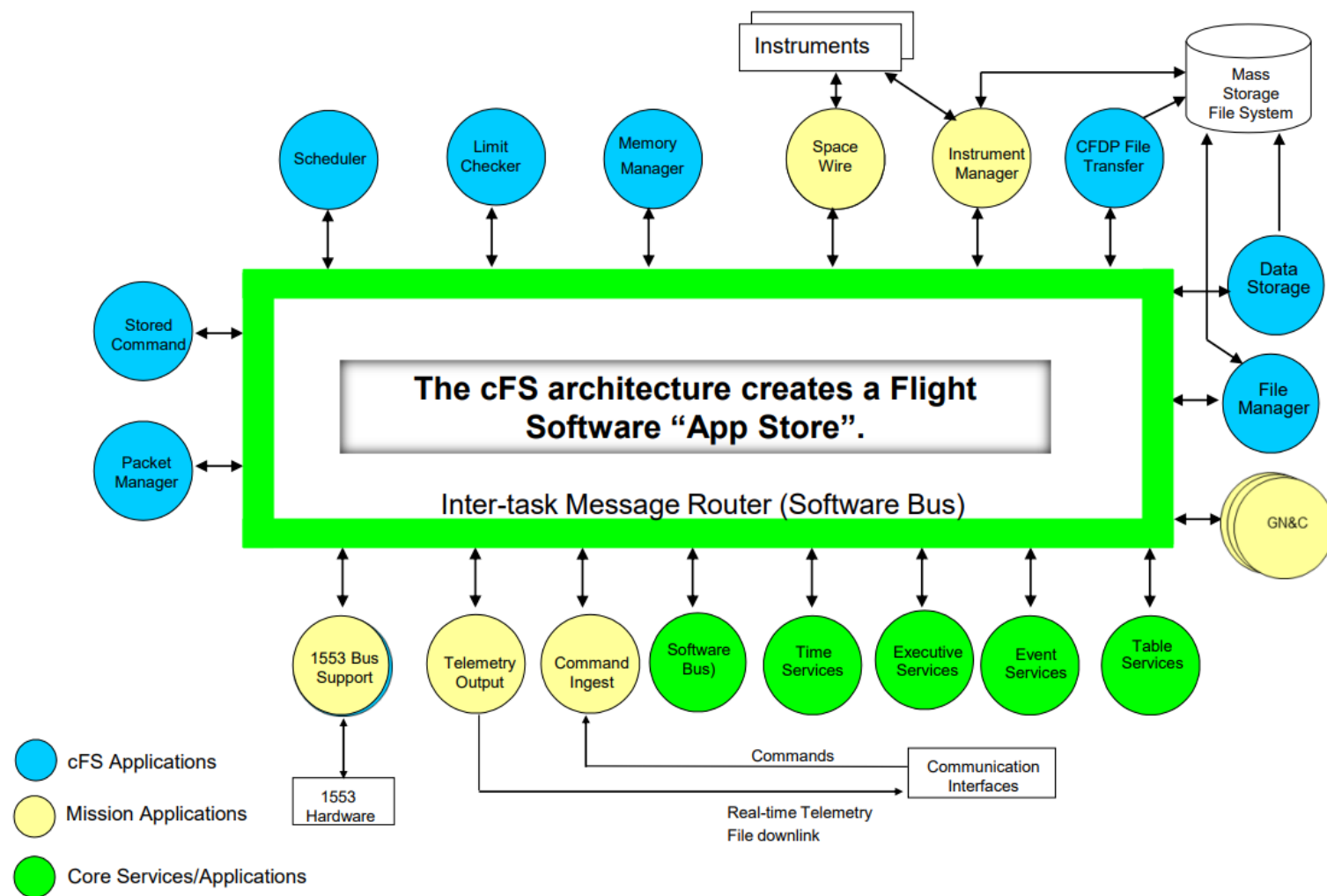
- ① 過去に開発されたソフトウェア資源を利活用 (**再利用性**)
- ② 多数のミッションで実績を積んだバグの少ないシステム (**信頼性**)
- ③ OSやハードに依存しない設計 (**移植性**)



cFS Architecture (1)



cFS Architecture (2)



- Software busを介して各アプリが相互通信する仕組み
- Pub/Sub型の通信モデル
- ランタイムに実行するコンポーネントを切り替えられる
- イベントやログの管理, 可視化ツールが整備されている, シミュレータ連携, etc...



ROSと多くの共通点がある！

<https://etd.gsfc.nasa.gov/capabilities/capabilities-listing/cfs/>



幅広いアプリケーションに適用可能な柔軟な設計

宇宙機フライトソフトウェア開発に特化

地上品で多数の導入実績あり

宇宙用組み込みCPUでFlight proven

Space gradeの検証が難しい
(安全規格対応は各プロジェクト側の責務)

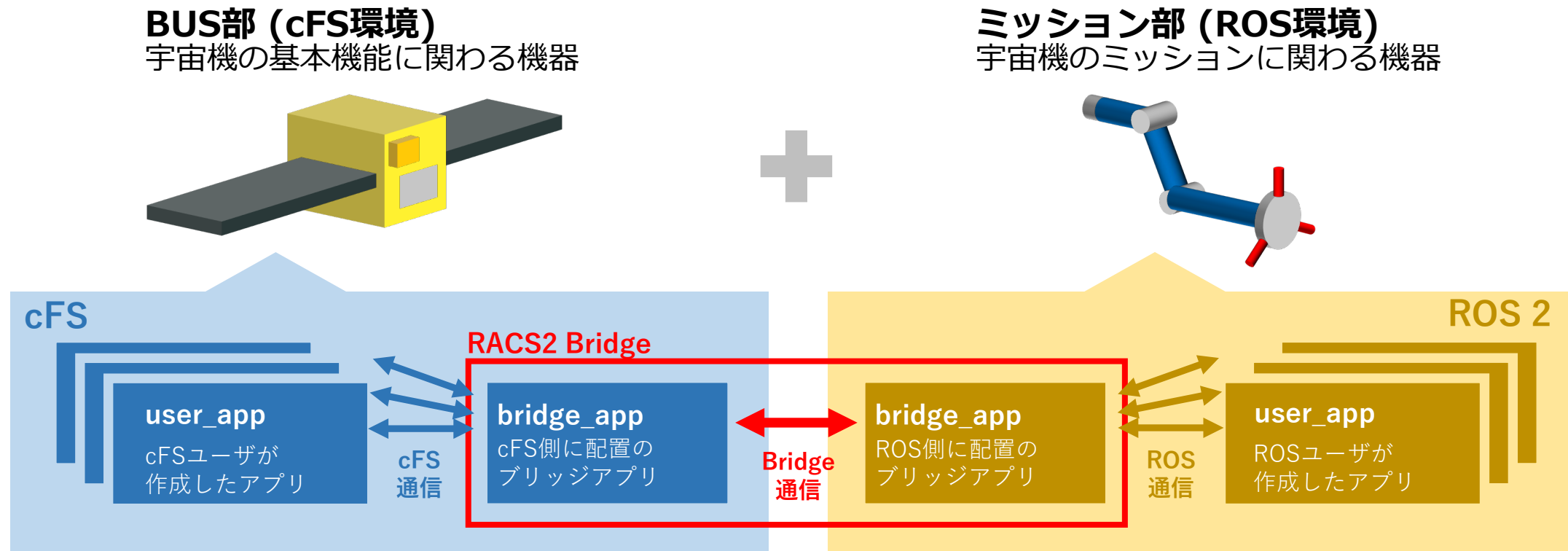
NASA規格 (NPR 7150.2など) 準拠検証ツールあり

非常に豊富なソフトウェア資源

宇宙分野では豊富と言えるがROSと比較してしまうと…

ROSとcFSを掛け合わせることで、より高度で複雑な宇宙ロボットシステムが開発できるのでは？

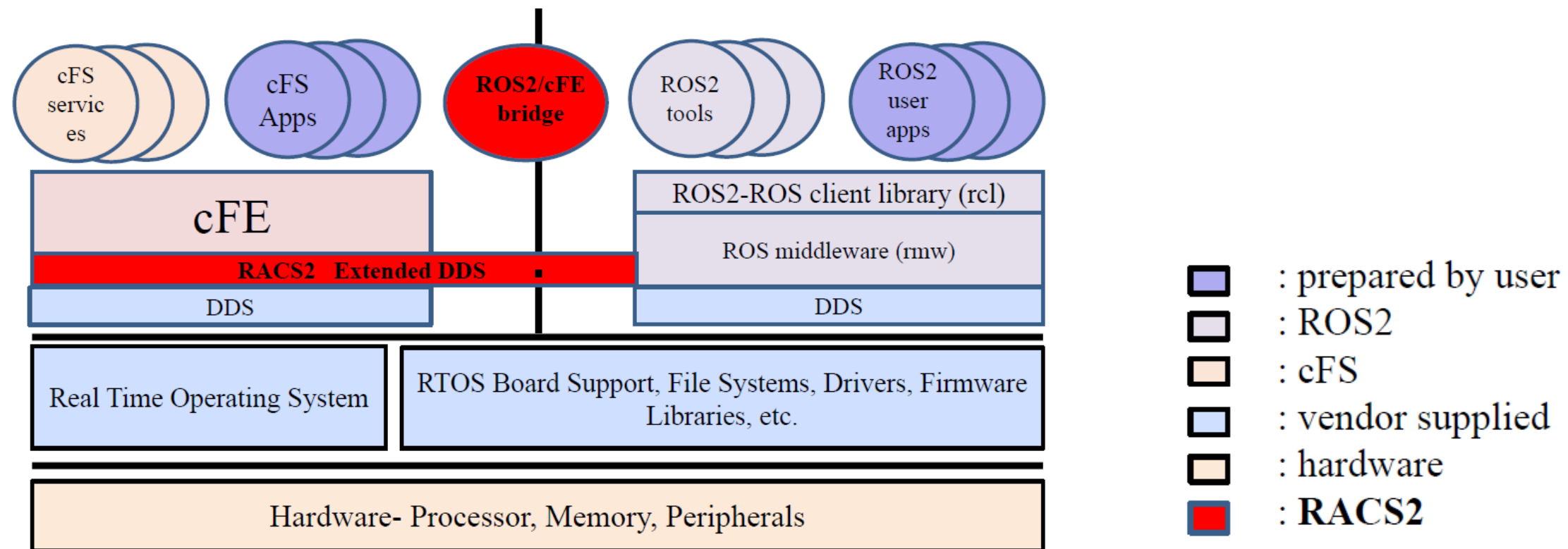
- ミッションクリティカルな機能 → Flight provenな**cFS**で実装
- 非クリティカルかつ複雑な処理 → 豊富なソフトウェア資源を活用して**ROS**で実装



■ 2種類のブリッジ機能

ROS2/cFE bridge : cFSのソフトウェアバスとROS2のメッセージを橋渡しする（アプリケーション層）

extended DDS : DDS（Data Distribution Service）で相互通信する（ミドルウェア層）



https://github.com/jaxa/racs2_bridge

GitHubでソースコード公開中！

RACS2が今後活用されていくには、運用実績を積み重ねていくことが不可欠
いずれは大規模な宇宙システムにRACS2を導入できるよう、段階的に技術実証を進めていく計画



ステージ1

ISS軌道上での概念実証

- Int-Ball2を活用した軌道上実証
- RACS2を介してROS2とcFSが相互通信できることを検証

👉 本発表の位置づけ

ステージ2

実応用を想定した機能実証

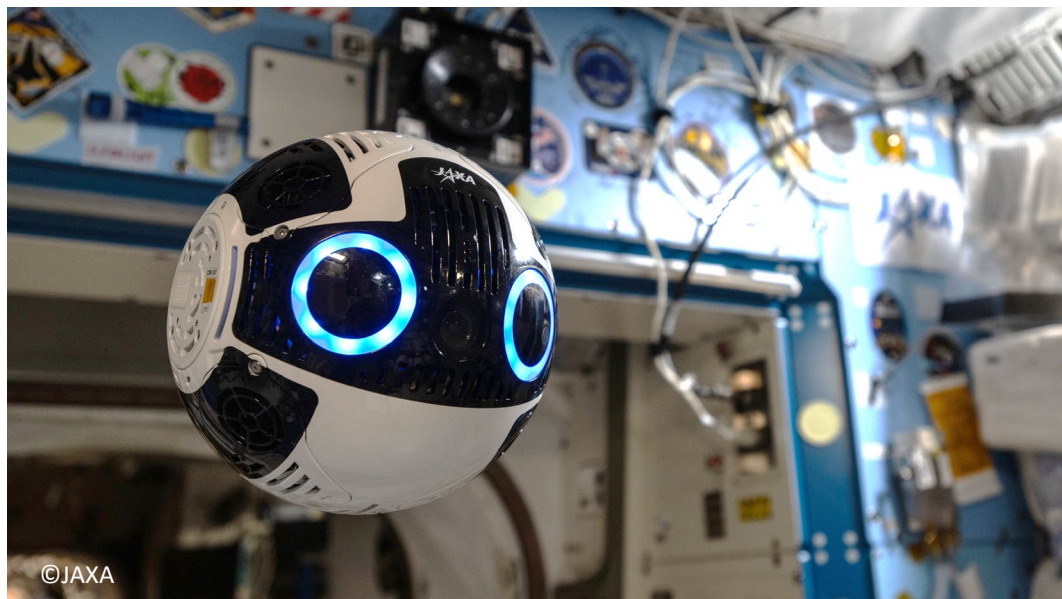
- 宇宙システムを想定したHW/SWコンフィグでの軌道上実証
- cFSとROSを別々の宇宙用OBCに搭載

ステージ3

実ミッションへの適用

- 宇宙ロボティクスミッションにRACS2を適用

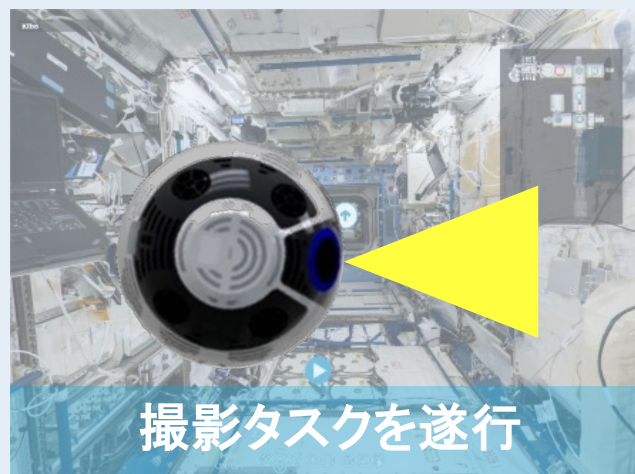
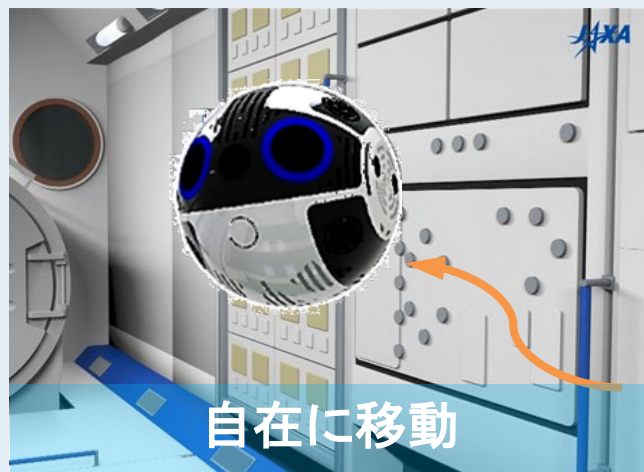
自律飛行型カメラロボット「Int-Ball2」



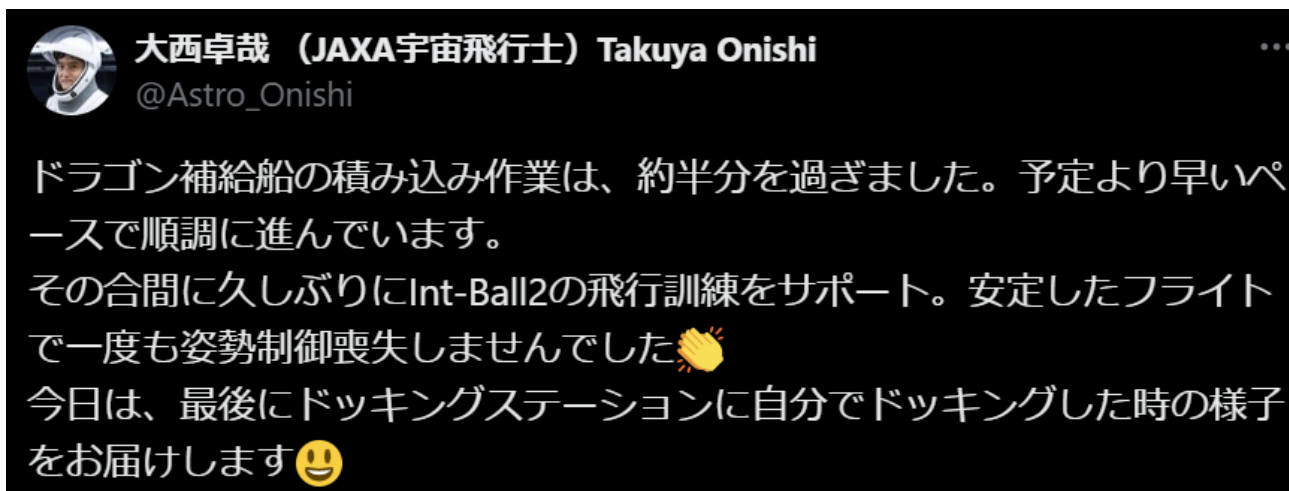
The JEM Internal Ball Camera 2 (Int-Ball2)

- 自律飛行が可能なカメラドローン
- 地上からのコマンド操作によって撮影が可能

クルーの手を介さずに撮影タスクを遂行することができる
JEM船内で現在活躍中！



大西宇宙飛行士のX（旧Twitter）投稿

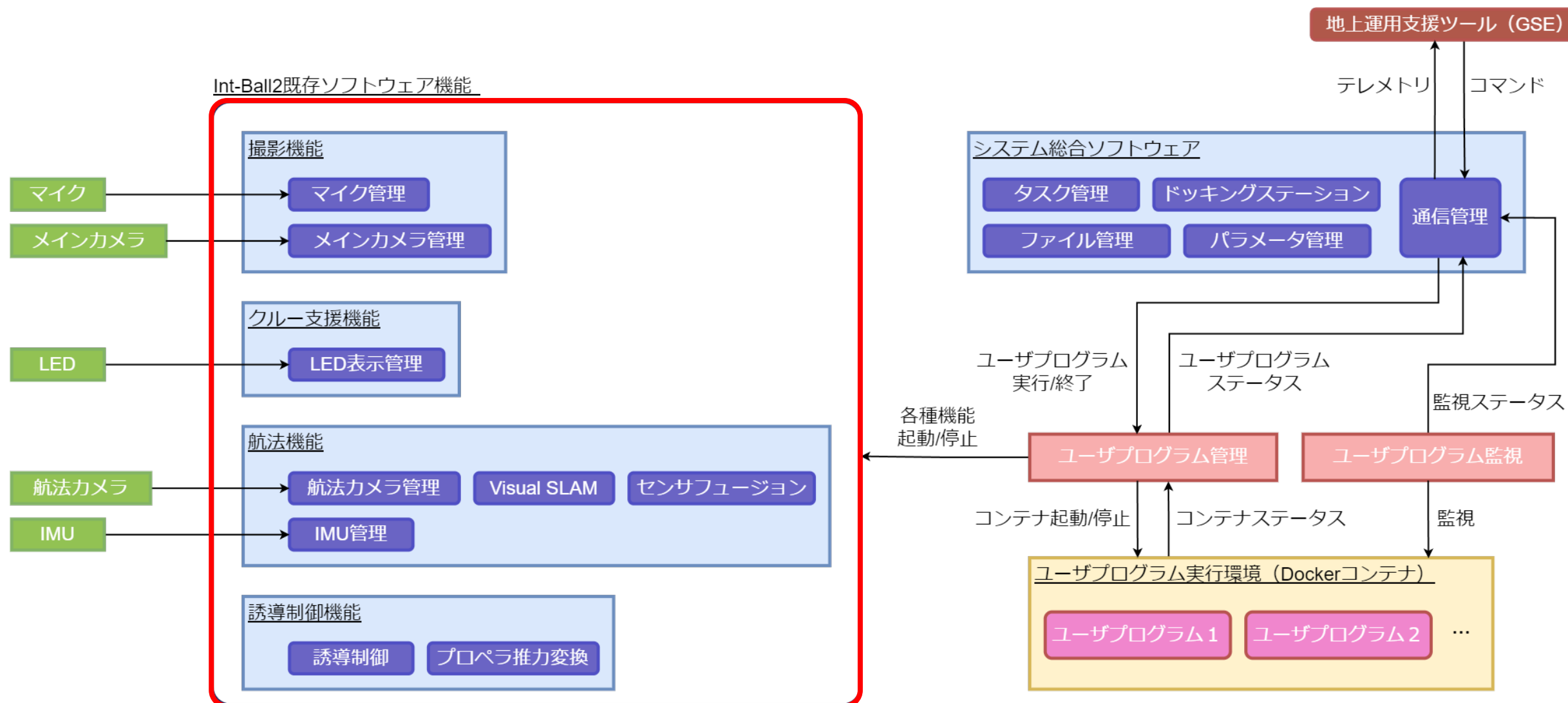


https://x.com/Astro_Onishi/status/1922776713542381715



Int-Ball2のユーザプログラミング機能

Int-Ball2は宇宙技術実証用のロボットプラットフォームとしての役割もある



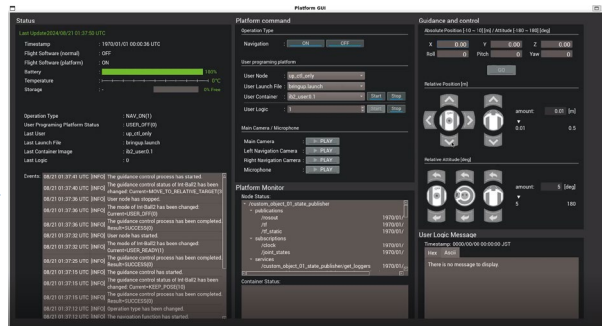
ほぼすべてのSW機能を独自のプログラムに置換可能

軌道上実証時のソフトウェア構成

Int-Ball2は組み込みコンピュータ（Ubuntu OS）1台しか搭載していない

→ 機能用途に応じた「cFSとROSの使い分け」は将来検討とし，今回はRACS2の概念実証を目的とした

地上局側のInt-Ball2操作GUI



Int-Ball2 (IB2)



地上で動作検証を進めるなかで、処理負荷の観点で幾つかの課題があった

1. 画像処理が重い（アプリ側の課題）

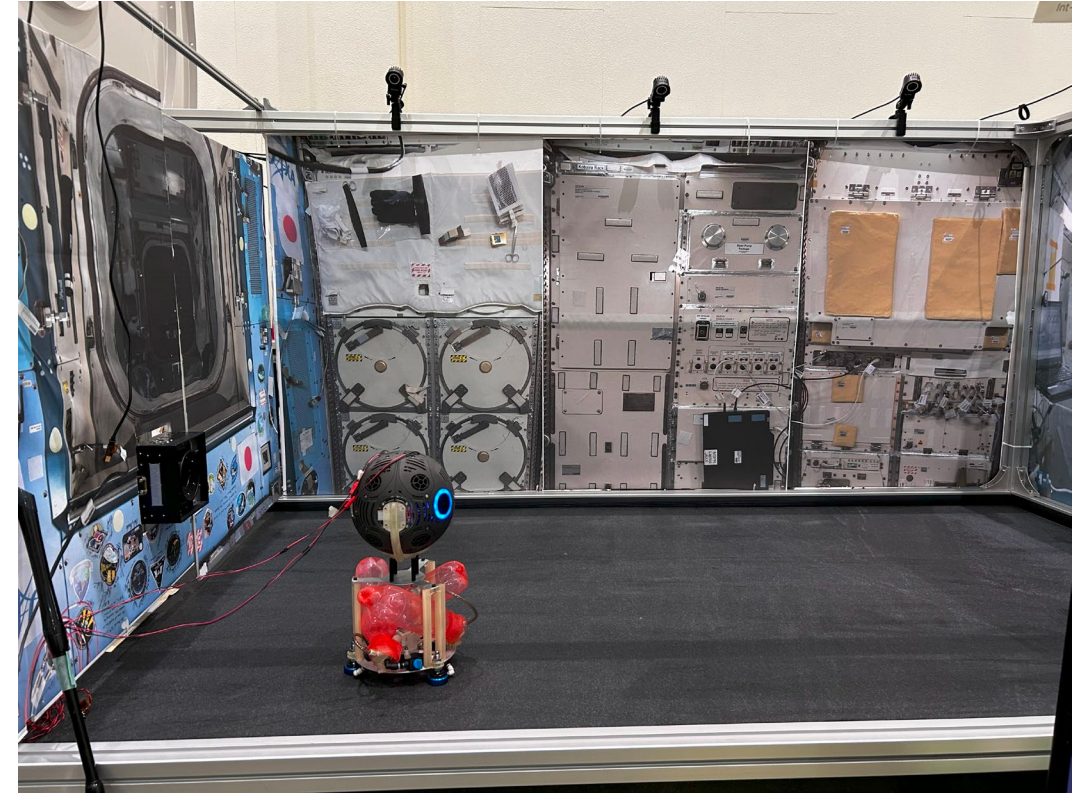
Int-Ball2がGPU並列処理に非対応の環境

使用した画像認識モデルが十分に軽量ではない

2. カメラ映像のPublish周期が遅い（ロボット固有の課題）

3. RACS2が重い（ブリッジ側の課題）

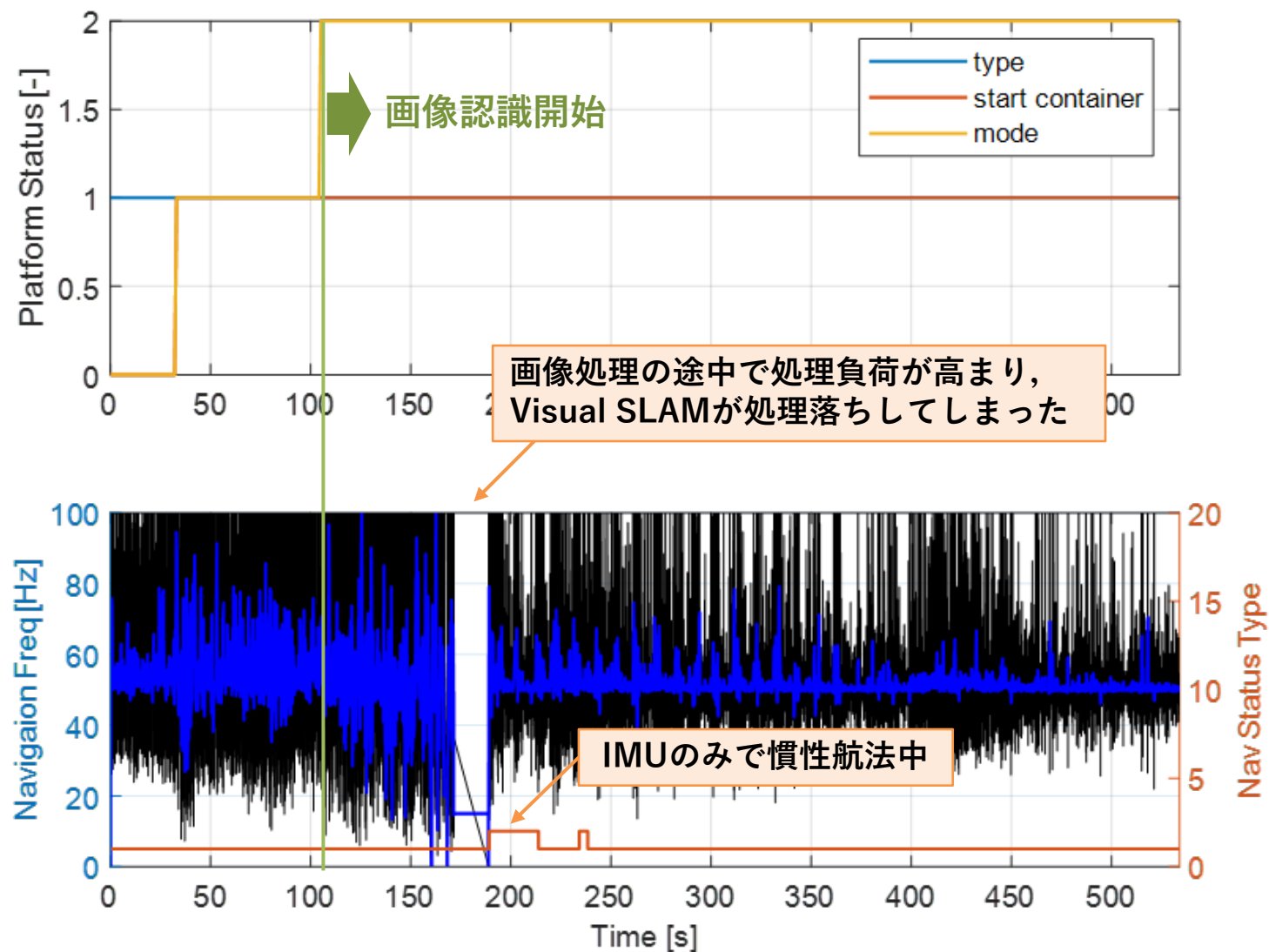
- 1, 2 はプラットフォーム側でオフノミナルな挙動を想定して対処できる仕組みが必要
- 3 はプラットフォーム自体の問題であり、本来発生してはならず、根本的に解決すべき！



Int-Ball2地上モデルと空気浮上盤を使った機能確認試験

ハードウェア制約がある中での実装

機体ステータス（上段）と航法値のPublish周波数（下段）



Platform Status

コンテナ 停止	= 0
コンテナ 起動（アプリは停止）	= 1
画像認識アプリ 開始	= 2

Nav. Status Type

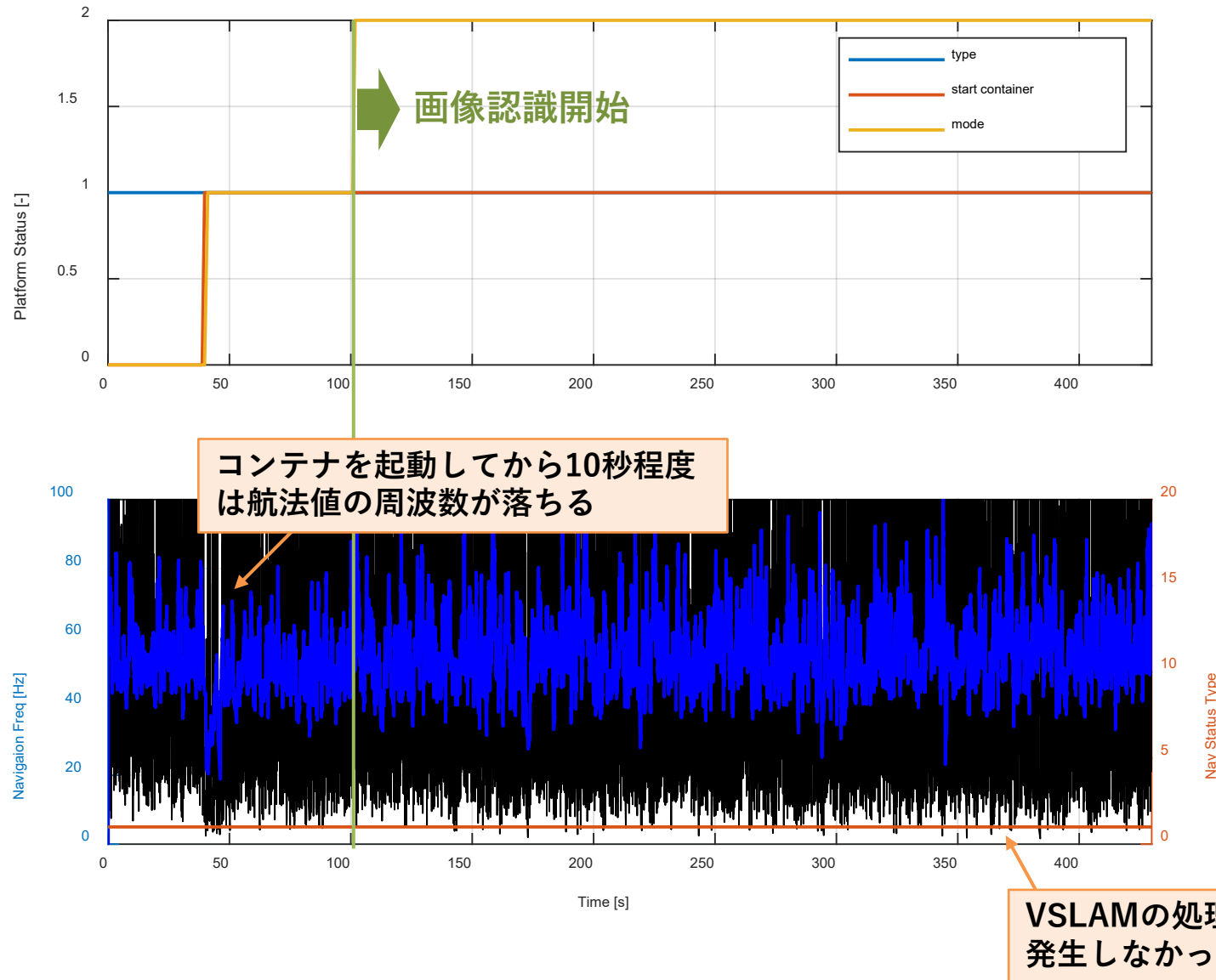
Visual SLAM 正常起動	= 1
Visual SLAM 停止	= 2

- 航法値のPublish周波数が極端に低下する
- 誘導制御系が固まる
（プロペラファンの駆動信号が一定値で固定）

といった飛行性能への悪影響がでていた

ハードウェア制約がある中での実装

機体ステータス（上段）と航法値のPublish周波数（下段）



RACS2の実装を修正して処理負荷低減を行った結果、飛行中に航法誘導制御系が固まる事象は発生しなくなった

一方でDockerコンテナ起動時に航法値のPublish周波数が減少した

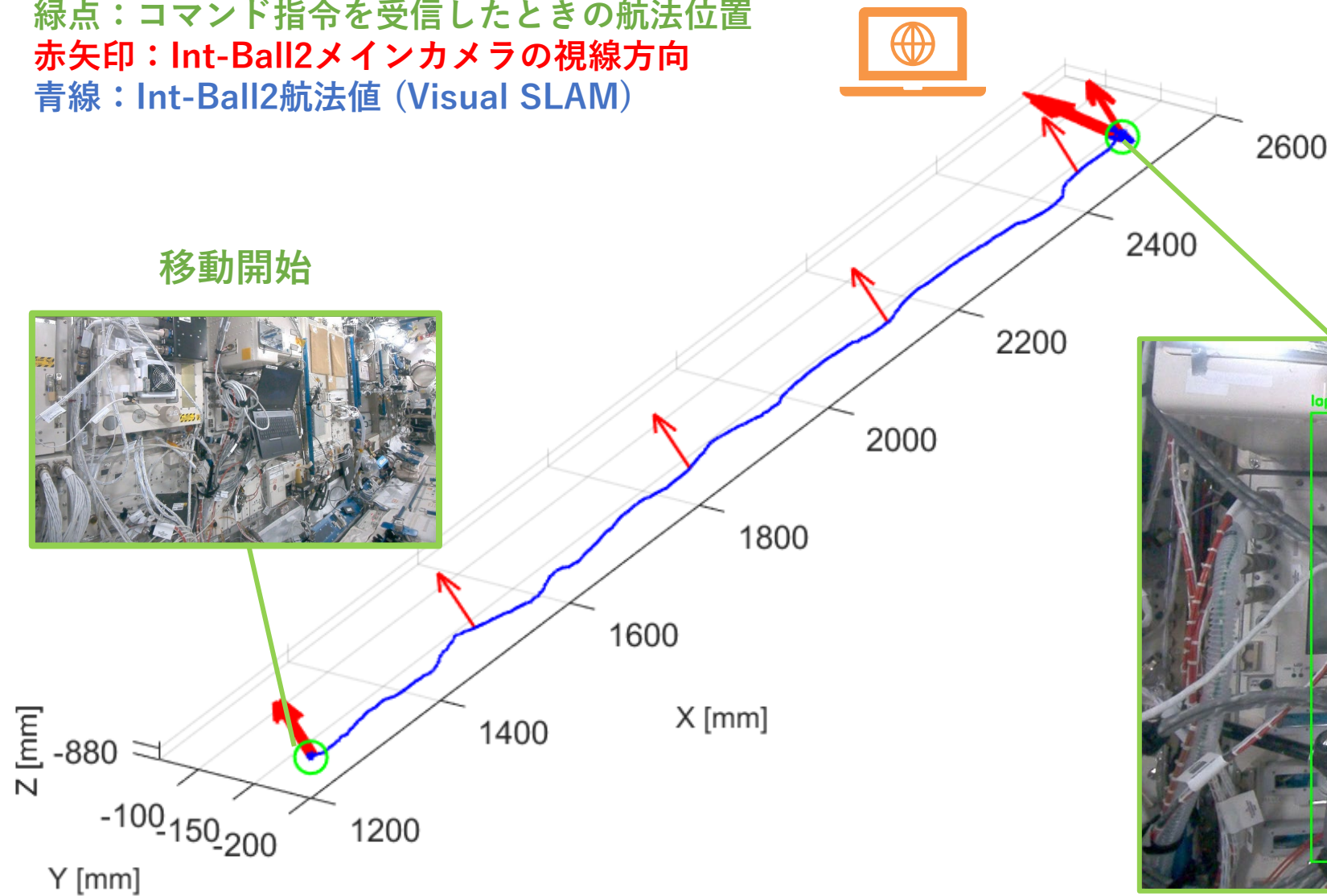
➡RACS2を活用してROSとcFSで機能分担させればこのような事象は発生しなくなる

軌道上実証結果 1：船内ラップトップPC撮影タスク

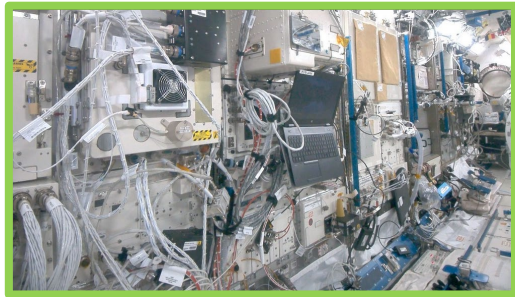


軌道上実証結果 1：船内ラップトップPC撮影タスク

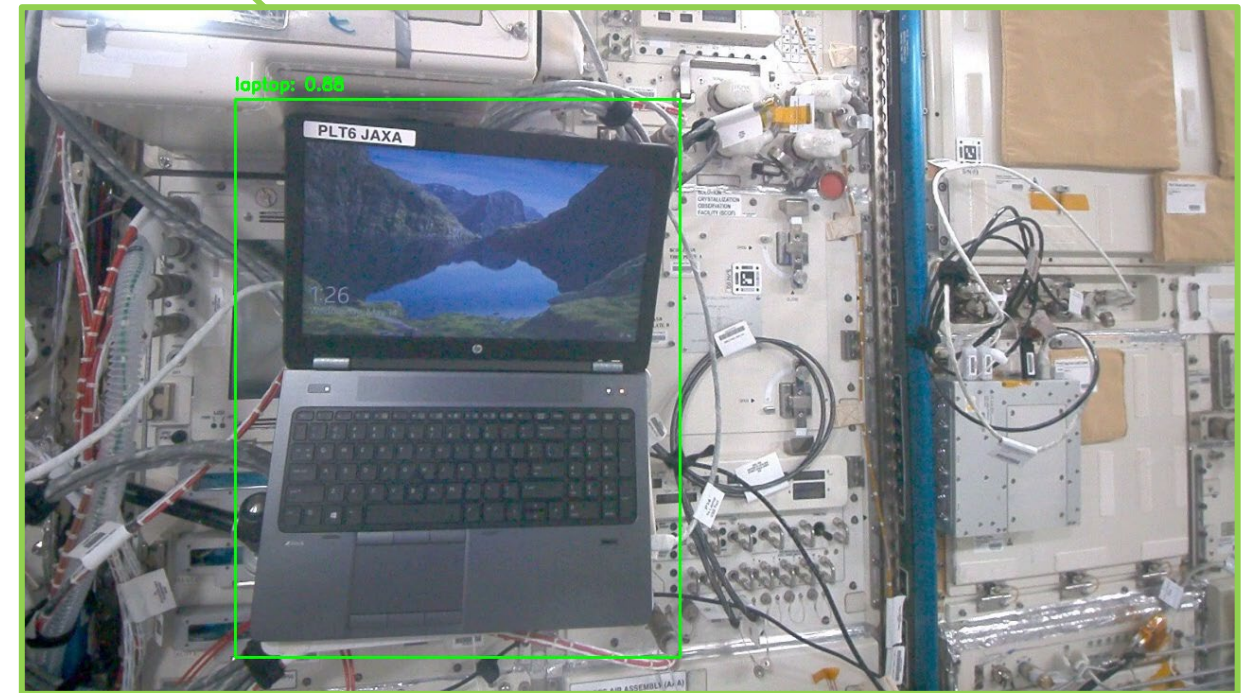
緑点：コマンド指令を受信したときの航法位置
赤矢印：Int-Ball2メインカメラの視線方向
青線：Int-Ball2航法値 (Visual SLAM)



移動開始



ラップトップPC認識



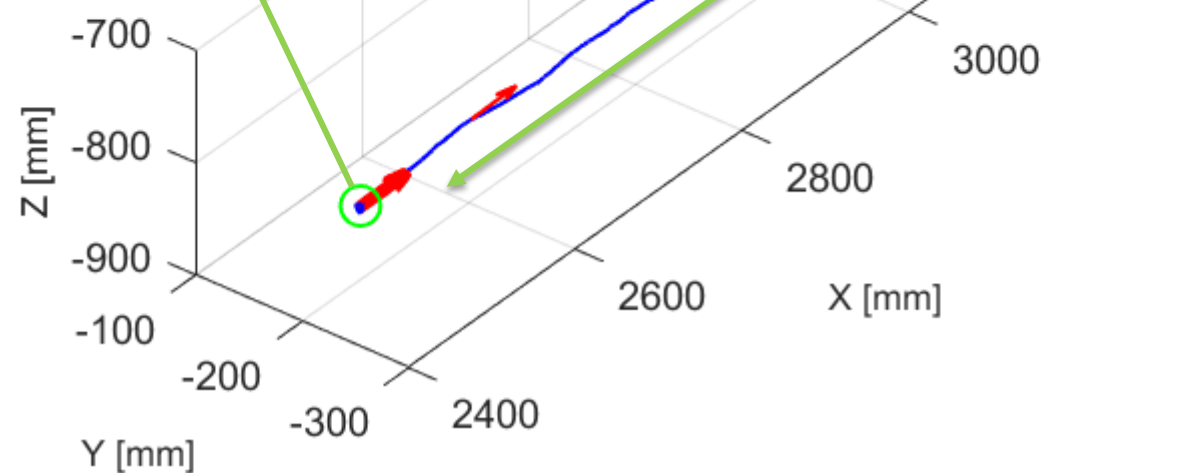
軌道上実証結果 2：大西宇宙飛行士撮影タスク



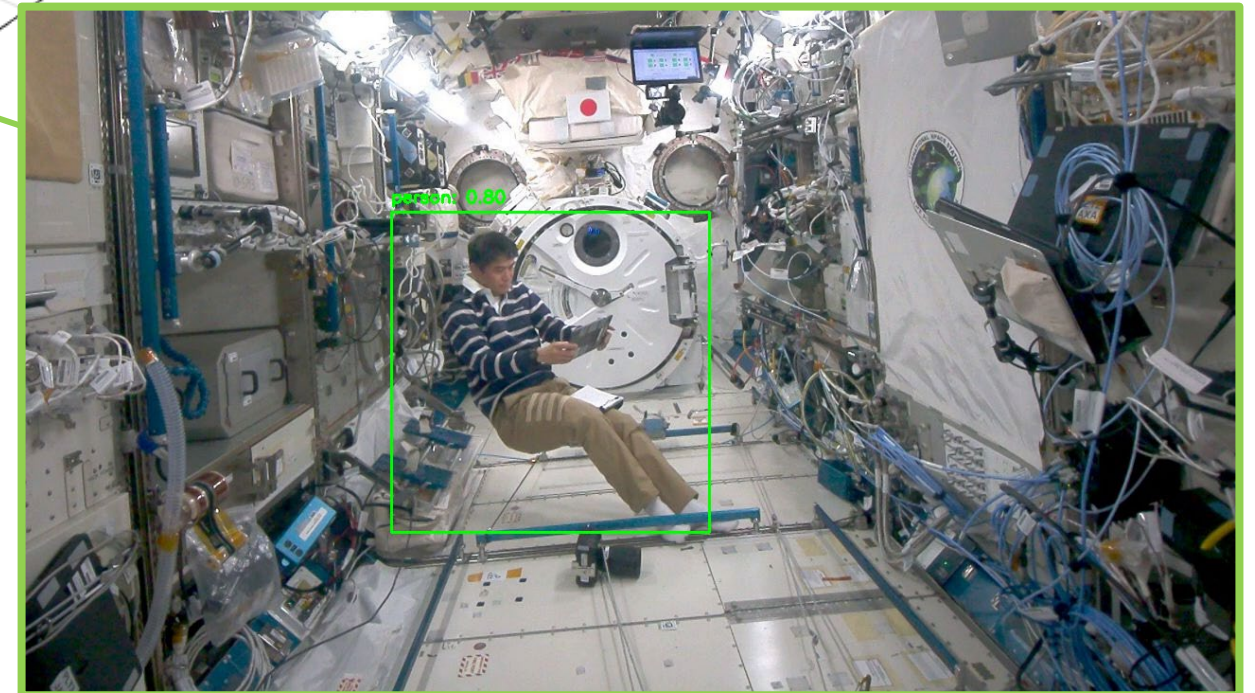
軌道上実証結果 2 : 大西宇宙飛行士撮影タスク

緑点：コマンド指令を受信したときの航法位置
赤矢印：Int-Ball2メインカメラの視線方向
青線：Int-Ball2航法値 (Visual SLAM)

移動開始



クルー認識



RACS2は単に通信をブリッジするのではなく**ROSとcFSの間の様々なギャップを解決**しなければならない

安全・信頼性

cFSはFlight provenであったとして、ROSとRACS2の宇宙機向けの認証・実績をどう評価するか？

➡ちゃんとクリティカルでない部分にROSが使われていて、それがクリティカルな機能に波及しないことをインターフェースであるRACS2側で保証しないといけないのでは？

➡宇宙機用ソフトウェアの開発規格（NPR 7150.2など）にROS部分をどう適合させるか（どうテーラリングさせるのか）

インターフェース (I/F) 標準化

要件が異なるので、ROS側とcFS側で違う宇宙用OBCを用いることが想定される

➡通信I/F、ハードウェアI/Fの標準化をブリッジ（RACS2）側で進められるか

➡例えばセンサがROS側とcFS側で別々にあるケース等で、データの時系列整合性をどう確保するか

フォールトトレランス

ROS実装側のOBCが誤動作した場合など、オフノミナル挙動に対するRACS2側での処理について検討しておく必要がある

➡障害検出・アイソレーション・復帰動作（FDIR）をどうブリッジ層で担保するか。

cFSとROSを繋ぐブリッジソフトウェア「RACS2」

ROSとcFSを掛け合わせることで、信頼性を維持しつつもより高度で複雑な宇宙ロボットシステムの実現を目指す

RACS2の宇宙技術実証

- RACS2を活用してもらうには運用実績を積み重ねていくことが重要
- 今回の軌道上実証はRACS2の概念実証（PoF）という位置づけ。実利用を見据えて今後も段階的に実証を進めていく計画

RACS2の今後の発展

単なる相互通信機能としての役割だけでなく、ROSとcFSの間の様々なギャップを解決していく



宇宙開発に興味があるROSユーザの皆さん、ぜひ色々意見交換しましょう！

Space ROS Clubは宇宙機・宇宙ロボット向けのROSであるSpace ROSに関する勉強会，開発などの活動を行うコミュニティです

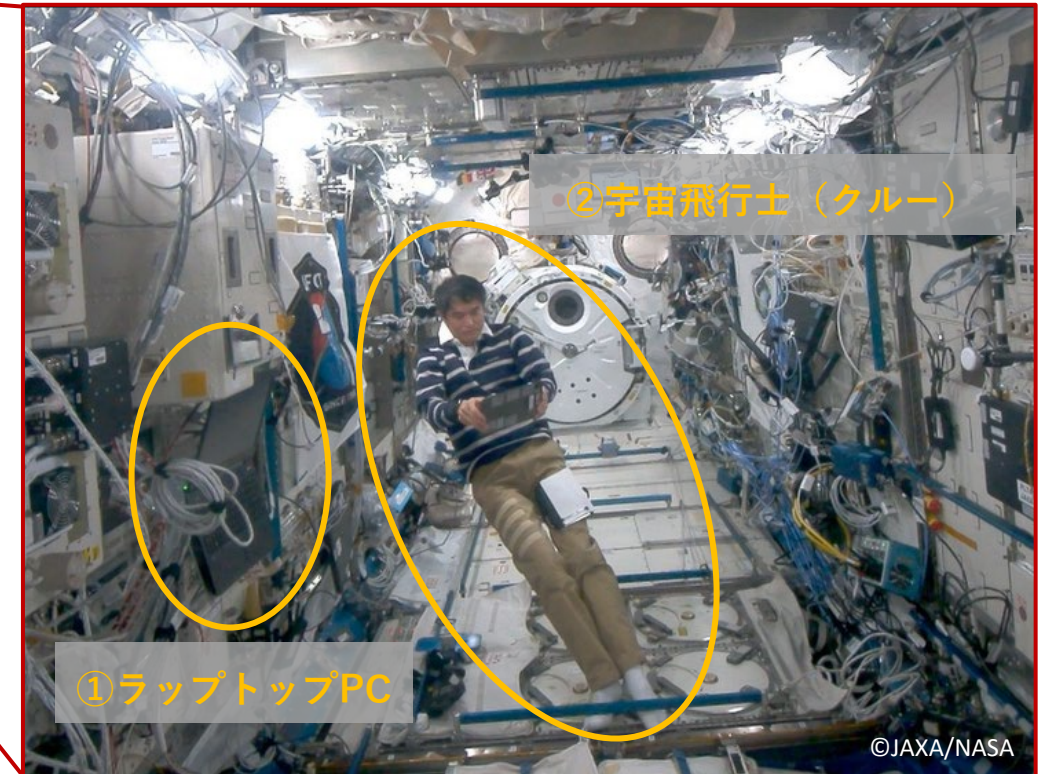
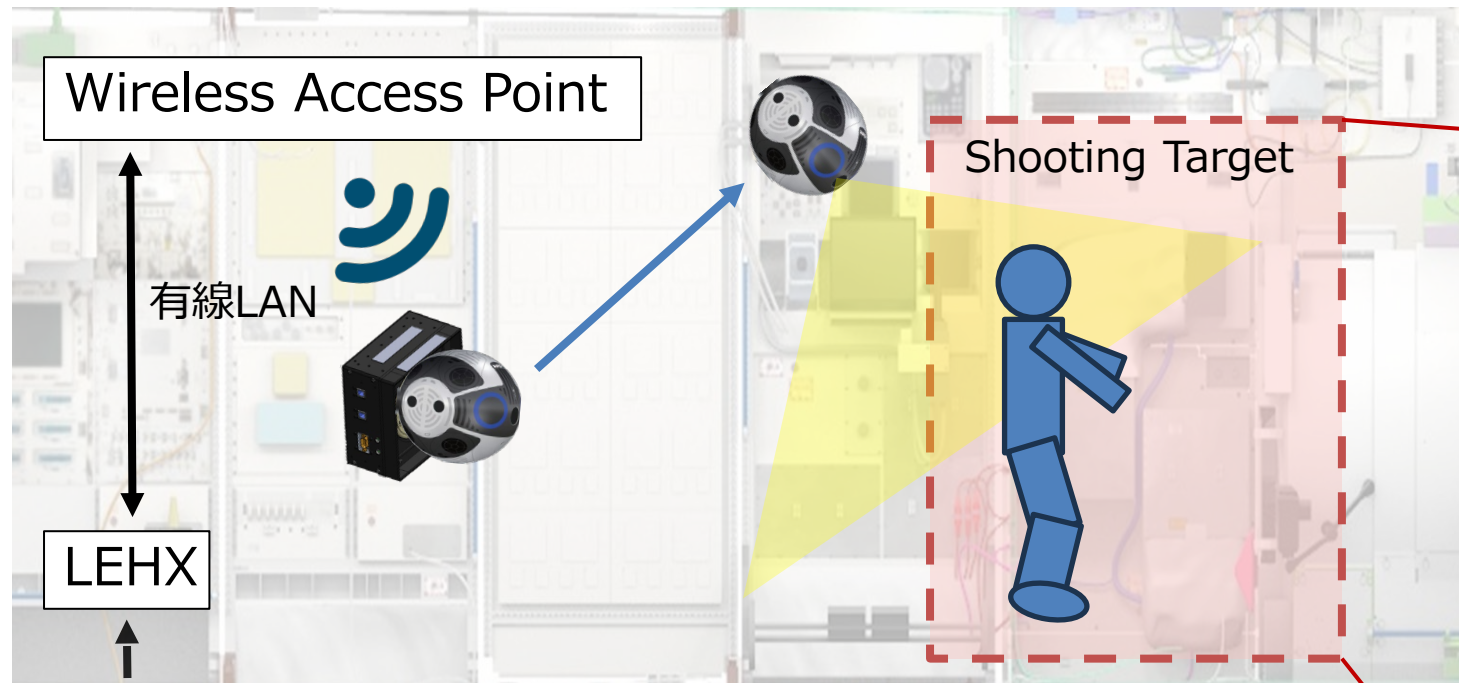
毎週Discordで勉強会を実施していますので，ご興味ある方はぜひ！！



<https://discord.gg/cGXxdcxMWN>

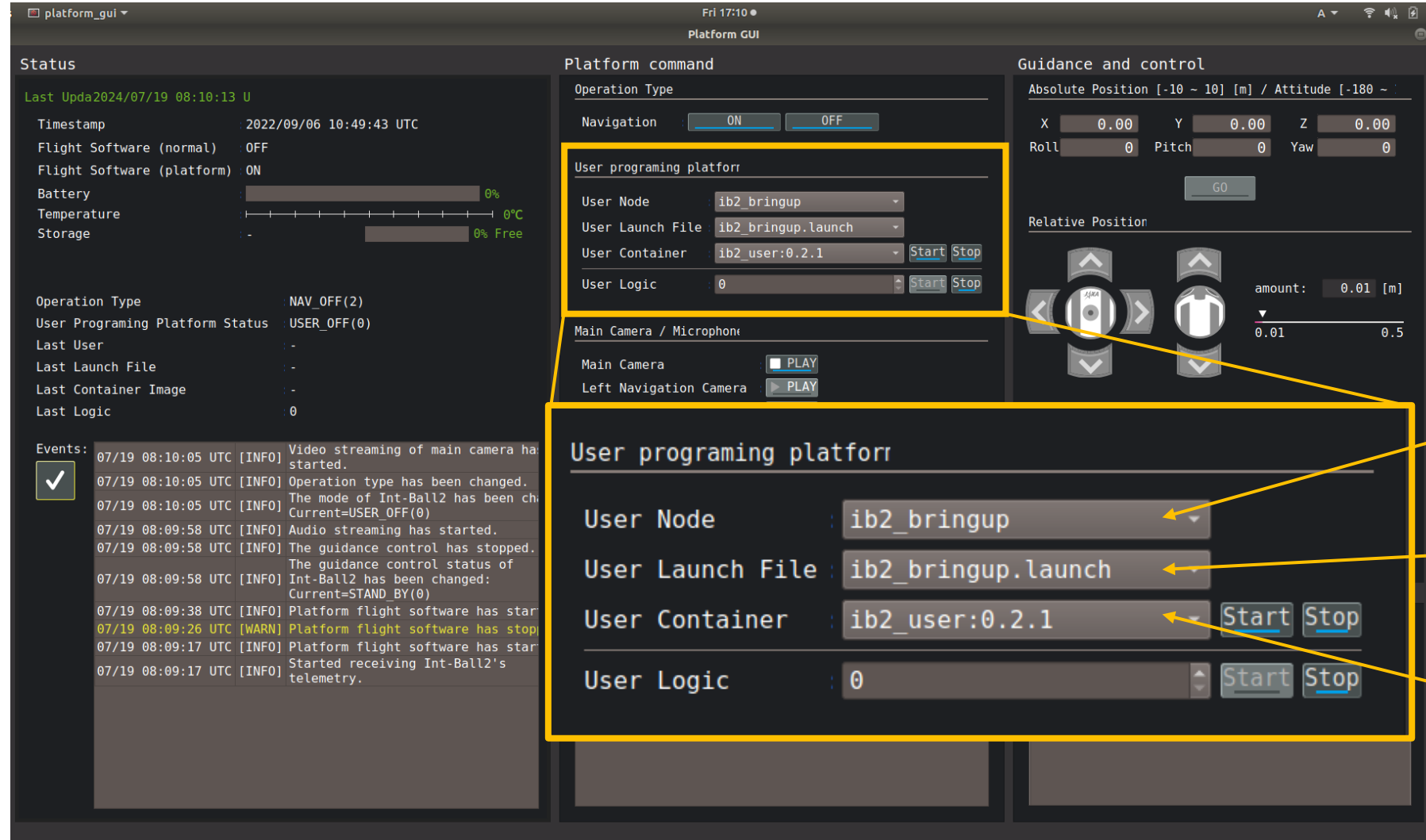


APPENDIX



①ラップトップPC と②宇宙飛行士（クルー）を
飛行中に検知する

Int-Ball2地上操作作用GUI



ISSネットワークを介した大容量データ授受

- ✓ 状況をリアルタイムに確認できる
- ✓ 生のフライトデータを地上に降ろせる



©JAXA

宇宙飛行士による試験支援

- ✓ ロボット単体では難しい試験も実施できる
- ✓ 様子を逐次報告してもらえる



©JAXA/NASA

cFSの導入実績



Project/Program	Operating System (If Reported)	Hardware(If Identified)	Launch (If Known)
Lunar Reconnaissance Orbiter (LRO)	VxWorks	RAD750 (PowerPC 750 family)	2009
Morpheus	VxWorks	AiTech S950 (PPC750FX PowerPC 750 family)	2011
Radiation Belt Storm Probes (RBSP) / Van Allen Probes	VxWorks	RAD750 (PowerPC 750 family)	2012
Lunar Atmosphere and Dust Environment Explorer (LADEE)	VxWorks	Unk	2013
Global Precipitation Measurement (GPM) mission	VxWorks	RAD750 (PowerPC 750 family)	2014
Observatory for Planetary Investigations from the Stratosphere (OPIS)	Xenomai Linux	Intel Duo	2014
Magnetospheric Multiscale Mission (MMS)	RTEMS	Rad Hard Coldfire (5208)	2015
Dellingr	FreeRTOS	Gomspace Nanomind A712d ARM7 RISC processor	2017
Neutron star Interior Composition Explorer (NICER)	VXWorks	BRE440 PowerPC	2017
Simulation-to-Flight 1 (STF-1)	FreeRTOS	Gomspace Nanomind A3200 AVR3200 MCU	2018
Compact Radiation Belt Explorer	Linux	XB1 Bus with Cubesat/Chrec Space Processor Inst	2018
Parker Solar Probe (PSP)	RTEMS	LEON3 UT699	2018
Global Ecosystem Dynamics Investigation (at ISS) (GEDI)	VxWorks	PowerPC 440	2018
Seeker ISS Flight Experiment	Linux	CHREC space processor	2019
Coordinated Applied Capitol Technology University Satellite (CACTUS-1)	Unk	Unk	2019
Kenobi ISS Flight Experiment	Unk	Unk	2019
Orion Ascent Abort 2 (AA-2)	VxWorks	AiTech SP0 1 GHz SBC PowerQUICC III processor	2019
Space Test Program -Houston 6, USAF-NASA Goddard	Unk	Unk	2019
Int-Ball2 at ISS JEM	Linux	JETSON TX2 NVIDIA Pascal	2020
Orion Camera Controller Artemis I (EM-1)	Linux	Intel i5 CPU (NUC)	2021
Lunar IceCube	Linux	P400	2021
MX-1, MX-2, MX-5, MX-9 (CLPS) Moon Express	Unk	Unk	2021
Peregrine Lander - Commercial Lunar Payload Services (CLPS)	Unk	Unk	2021
Astrobotic Technology			
BioSentinel	VxWorks	UT700 LEON 3FT	2021
Orion Vision Processing Unit Artemis I (EM-1)	VxWorks	UT700 LEON 3FT	2021
SkyFire	Unk	Unk	2021
XL-1 Lander (CLPS) Masten Space Systems, Inc.	Unk	Unk	2021
Orion Camera Controller Artemis II (EM-2)	Linux	Intel i5 CPU (NUC)	2022
Plankton, Aerosol, Cloud, ocean, Ecosystem (PACE)	VxWorks	MUSTANG (custom LEON3 Dual core + LEON3-FT i	2022
Orion Vision Processing Unit Artemis II (EM-2)	VxWorks	UT700 LEON 3FT	2022
Lunar Gateway - Minimal Habitation Module (formerly Utilization Module)	VxWorks	Unk	2023
Exploration Extra-vehicular Mobility Unit (xEMU) Caution & Warning System (CWS)	VxWorks	Leon3 SPARC processor	2023
Roman Space Telescope (RST) (previously Wide-Field Infrared Survey Telescope - WFIRST)	RTEMS	Custom LEON4	2025
Mars Ascent Vehicle (preliminary)	Not App.	Sphinx	
Ames Modular Common Spacecraft Bus	VxWorks	RAD750 processor, 1GB TMR NVRAM	
Avionics & Software Platform for Exploration Capabilities & Technologies (ASPECT)	VxWorks	SP0 processor PowerQUICC III processor	
Certification of cFE on VxWorks ARINC-653	VxWorks ARINC 653	SP0 processor PowerQUICC III processor	

John W Bradbury, “Open Source Core Flight System (cFS) Flight Software (FSW) Verification & Validation (V&V) Final Summary”, IV&V Analysis Technical Report, NASA/CR-20205010026

[https://ntrs.nasa.gov/api/citations/20205010026/downloads/CR%2020205010026%20Core%20Flight%20System%20\(cFS\)%20Flight%20Software%20\(FSW\)%20Final%20Report-1.pdf](https://ntrs.nasa.gov/api/citations/20205010026/downloads/CR%2020205010026%20Core%20Flight%20System%20(cFS)%20Flight%20Software%20(FSW)%20Final%20Report-1.pdf)

■ Space ROS

- ROS2をベースに宇宙ロボティクス用途に拡張したプラットフォーム (Open Robotics, NASA, PickNik社が開発を主導)
- コード解析ツール, 要求トレーサビリティ管理ツール, その他宇宙固有のツール群を使った開発ができるようになっている

「ROSを拡張して宇宙ロボット用にする」というモチベーション



■ RACS2

- ROS2とcFSの間のデータ授受を行うブリッジ機能を提供

「ROSを宇宙用プラットフォームと繋げる」というモチベーション

宇宙ロボット開発にROSを活用する点では共通しているが, 異なるアプローチをとっている