Fly Foundational Robots
Guest Roboticist Capabilities Brief

Mission Background

The Fly Foundational Robots (FFR) mission is a flight demonstration intended to validate autonomous robotic capabilities in microgravity. Led by Motiv Space Systems in partnership with NASA, FFR supports NASA's ISAM strategy by:

- Demonstrating advanced robotic mobility, manipulation, and autonomy in LEO
- Testing foundational behaviors necessary for future servicing and assembly operations
- Enabling collaboration across government, industry, and academia through an open experimentation framework

FFR is positioned as a key step toward scalable orbital infrastructure and provides a rare opportunity to flight-test robotic systems in an operational space environment.

Figure 1: Notional picture of the FFR payload in-orbit.

Guest Roboticist Program Objectives

The Guest Roboticist program invites external researchers and technologists to engage directly with the FFR mission team by:

- Contributing algorithms, modules or experimental payloads for robotic inspection, manipulation, or navigation
- Supporting testbed integration, simulation, and pre-flight verification
- Participating in mission operations planning or post-mission data analysis
- Demonstrating approved technologies in orbit as part of the hosted flight experiment set

This program is intended to foster collaborative innovation and reduce the barrier to orbital demonstration for promising robotic technologies.

System Overview

The robotics platform provides a controlled environment where guest roboticists can safely deploy, test, and benchmark their experiment software on a shared physical system. Users interact with the platform via well-defined command interfaces, with development and deployment happening first through simulation and ground tests. Once the experiment's development has been finalized, and safety verified, it can be deployed on orbit.

- **Primary objective:** Enable reproducible, robotic experimentation remotely through simulation packages, Motiv's on site testing setup, and on orbit.
- **Access model:** Time-slotted reservations with remote connectivity over secure channels.
- **Supported stacks:** ROS 2 / custom C++ / Python processes; containerized deployments (Docker).

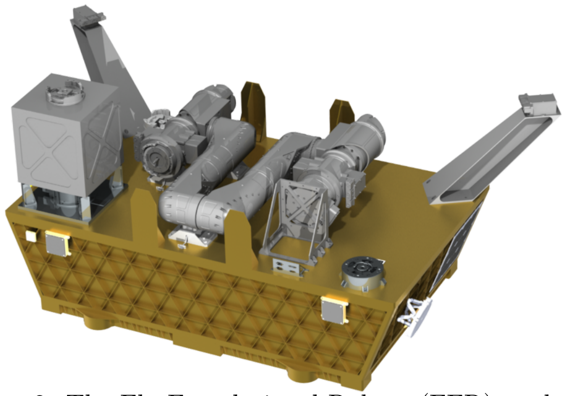

Figure 2: The Fly Foundational Robots (FFR) payload shown with the CrossLink robotic arm, separable interfaces, and an ORU Module.

Robotic Arm Technical Specification	
Joint Configuration	7-DoF articulated arm
Control Rate	100 Hz
Communication Protocol	RS422
Maximum Reach	1.0 m (tool center point) TBR
Payload Capacity	5 kg nominal TBR
Repeatability	±0.05 mm TBR
Joint Sensing	Absolute encoders, current sensing, strain gauges, PRT temperature sensing
Force-Torque Sensor	6-axis FTS located at each end effector
Safety Features	33 configurable telemetry errors, E-stop

Payload Compute Technical Specification	
CPU	16-core Cortex A72
GPU	NVIDIA Jetson Orin-class compute
Memory	Up to 16 GB DDR4
Storage	Up to 128 GB eMMC
Networking	1 Gbit/s Ethernet

Vision System Technical Specification	
Distributed Compute	Image sensors paired with dedicated processor boards
End Effector Cameras	Vision subsystems located at each end effector
Context Cameras	Two vision subsystems offset laterally via fixed booms
High Resolution Capture	4208x3120 image capture and 1920x1080 @ 30 FPS video capture TBR
Stereo Pair	Dual 640x480 monochrome image sensors per subsystem
CPU	8-core Cortex A77
GPU	Adreno 650 - 1024 ALU
Memory	8 GB LPDDR5

Guest Roboticist Supported Interfaces

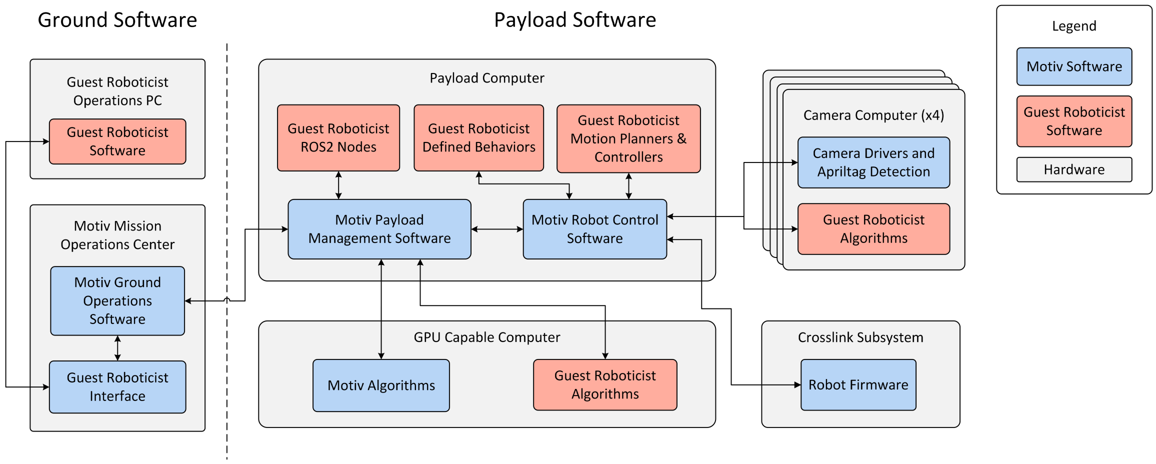

Figure 3: Notional guest roboticist code on the ground and in orbit.

Guest roboticists will have multiple options for deploying and executing software on the FFR payload. The overall software architecture and the notional "entrypoints" for guest-developed code are illustrated in Figure 3, where Motiv software is shown in blue and potential guest roboticist software is shown in red.

The main payload computer hosts both the payload management software and the core robot control software. Most components in this primary software stack are integrated through a ROS2 network, with the ROS2 DDS messaging layer serving as the preferred mechanism for inter-process communication. Guest roboticists may deploy their own ROS2 nodes directly onto the payload computer and subscribe to the existing ROS2 network in order to access payload telemetry and other system data for their experiments.

For computationally intensive algorithms, platform research partners will also have the opportunity to deploy ROS2 nodes on a dedicated GPU-capable computer that resides on the same DDS network. In addition, each onboard camera bar is supported by a dedicated embedded computer intended primarily for vision processing. These vision computers host camera-specific processing pipelines, and guest roboticists may deploy their own vision system code directly on these platforms as well. There are four total vision subsystems, each with their own dedicated vision computer: one at each end effector, and two laterally displaced from the payload via a boom for visual context.

At the robot control level, the system leverages behavior trees to provide a clear, modular framework for orchestrating robotic behaviors. Complex tasks are decomposed into a hierarchy of reusable behavior nodes. Guest roboticists will be able to design and execute their own custom behavior trees, building on a foundational library of existing robot behaviors. In addition to the behavior tree framework, the system employs MoveIt2 paired with ros2_control as the primary motion planning and control infrastructure. This architecture enables guest roboticists to integrate and evaluate custom motion planners and controllers as a part of their experimental workflows.

From a ground operator's perspective, all command and telemetry flows through a defined guest roboticist interface to Motiv's ground operations software. Guest roboticists and their ground operations software will be able to remotely interface with the Motiv Mission Operations Center (MMOC) via a VPN connection, or connect on-site via LAN. For developing and testing algorithms before they are deployed on orbit, simulation tools and a testing environment will be provided.

Contact

Principal Investigator: Michael Hagman
Email: `michael.hagman@motivss.com`

Project Manager: Brett Lindenfeld
Email: `brett.lindenfeld@motivss.com`

Revision	Pages Affected	Issue Date
0	Init. Release	12/30/2025