Space Technology Mission Directorate

Game Changing Development Program

Trey Smith, ISAAC | Annual Program Review | September 10-12, 2024

ISAAC Overview

Timer

Needs Addressed by Project

Lunar Logistics Container	ISAAC Technology Thrusts: Integrated Data, Coordinated Execution. ISAAC Capability Areas: Autonomous State Assessment, Autonomous Logistics Management, Integrated Fault Management	STARPort: Advanced Robotics 686, 892; Autonomous Systems 795, 1134
Mars Long-Duration Surface Hab	ISAAC Technology Thrusts: Integrated Data, Integrated Control Interface, Coordinated Execution. ISAAC Capability Areas: Autonomous State Assessment, Integrated Fault Management	STARPort: Autonomous Systems 969, 1032, 795; Advanced Robotics 1070, 686

Project Goals

Goal #1	Develop/enhance key technologies for the autonomous operation of complex, space-based infrastructure (e.g., Gateway) that provide the capability to support operations where human intervention is limited.
Goal #2	Provide new capabilities for spacecraft that incorporate both autonomy and robotics to enable a new paradigm for in-space operations and adaptive vehicle caretaking.

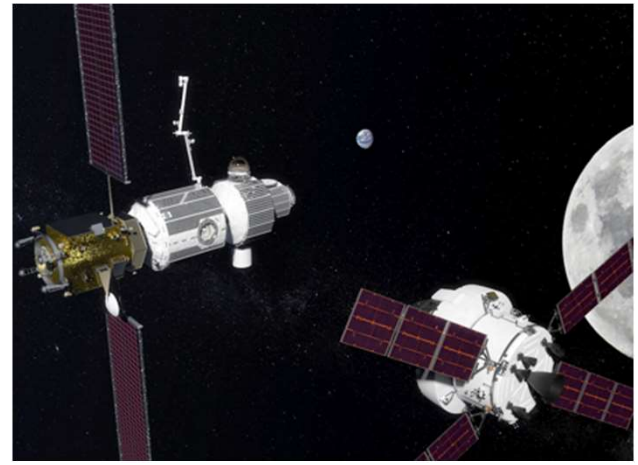
Project Objectives

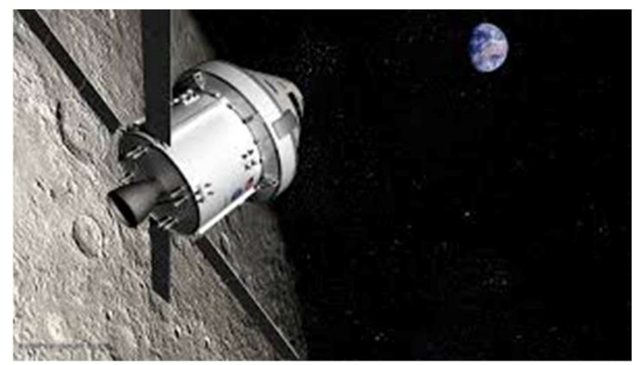
Objective #1 **Integrated data**	Current state of the art in human spaceflight systems is that data from sensors associated with different spacecraft subsystems (Structures, GN&C, and ECLSS, etc.) remain siloed. ISAAC will unify spacecraft data and models with autonomous robotics, linking data streams from spacecraft subsystems, sensor networks, and robots, as well as linking 3D geometry and sensor data map layers, and detecting changes and anomalies. ISAAC will test autonomous state assessment with a mobile inspection robot, ideally on ISS.
Objective #2 **Integrated control interface**	Current state of the art manages spacecraft with largely independent interface tools for different subsystems. Interfaces have different heritage, design assumptions and operator interface styles. Subsystem interactions are hard to analyze due to poor connectivity between separate tools. ISAAC will develop an integrated control interface that combines interface tools for spacecraft subsystems and autonomous robots; improves system-level situation awareness, situation understanding, and operator productivity; enables linking and embedding between tools to improve subsystem interaction analysis; and supports the entire activity life cycle from planning through execution and analysis. ISAAC will test logistics management with mobile inspection and mobile manipulation capabilities, and test operator productivity in a logistics management scenario, both ideally on ISS.
Objective #3 **Coordinated execution**	Current state of the art for executing spacecraft activities that require coordination between subsystems is either manual commanding (operator tracks dependencies between subsystems) or simple sequencing that can be brittle to even minor contingencies during execution. ISAAC will develop coordinated execution technology that models dependencies, uses automated planning to translate a high-level task definition to a plan that can work given the current system state (e.g. include actions to open hatches so that a robot can move where needed), and leverages ISAAC's integrated data technology to watch for execution contingencies and trigger replanning as needed. This technology will reduce the time to effect changes on the system during critical faults and emergencies. ISAAC will test leak detection, isolation, and repair with mobile inspection and mobile manipulation capabilities, ideally on ISS.

ISAAC Overview

Timer

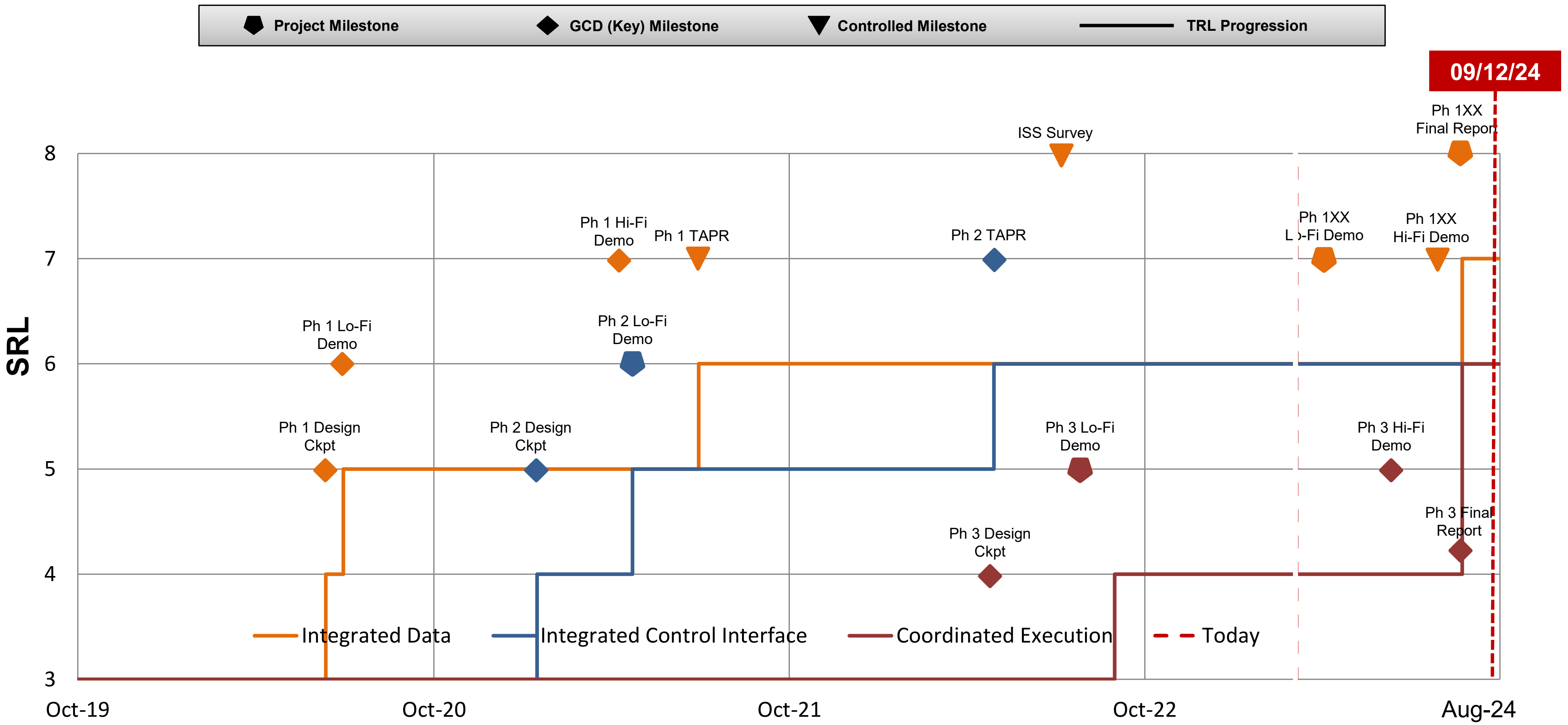
- Research project to develop technology for **autonomous caretaking** of spacecraft primarily during uncrewed mission phases
- Led by NASA Ames Research Center with collaboration from Johnson Space Center
- Integrate autonomous **intra-vehicular robots** (IVR) with spacecraft infrastructure (power, life support, etc.) and ground control
- Focus on capabilities required for the Gateway that also apply to human missions to Mars and beyond
- Test with **existing IVR** on the ISS (Astrobee, Robonaut) as an analog for **future IVR** on Gateway
- Due to delays in Gateway program phasing of IVR development, ISAAC pivoted its near-term infusion effort to target operational use of IVR on ISS (Astrobee)





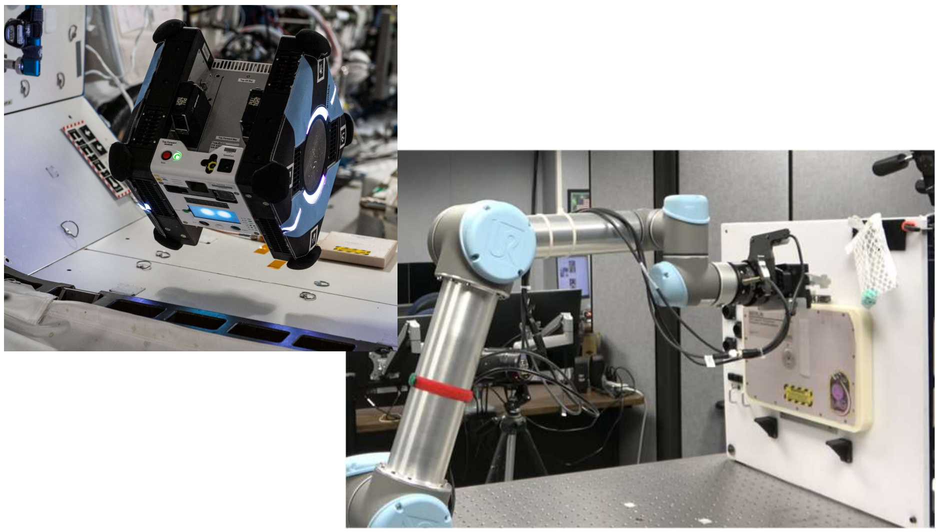
ISAAC Lifecycle Milestone/Maturity Schedule

Timer



ISAAC FY24 Focus Areas

Timer



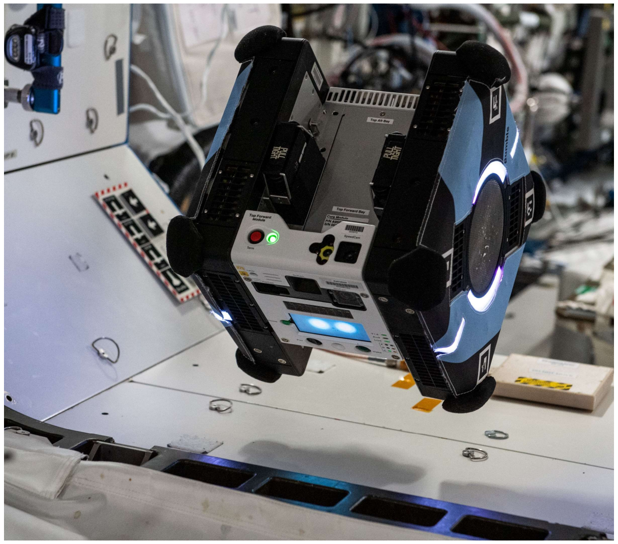
Phase 1XX
Technical thrust: Integrated Data
Capability area: Autonomous State Assessment
Hifi demo scenario:
- ISS demo: Astrobee close-up targeted inspection of hatch seals

Phase 3
Technical thrust: Coordinated Execution
Capability area: Integrated Fault Management
Hifi demo scenario:
- ISS demo: Coordinated ISS interior survey with multiple Astrobees

Phase 1XX: Technical Development

Timer

Task
Targeted inspection
Precision targeting: Improve Astrobee's capability to image a precise target location through improvements to Astrobee localization.
Precision focus: Improve Astrobee's capability to precisely focus on a specific target, adjusting standoff distance using its HazCam depth sensor.
Target selection: Develop an interface for ISS ops staff to select targets for close-up inspection using panoramic imagery from a previous ISAAC-style Astrobee survey
Context imagery: Develop an interface for ISS ops staff to view the resulting high-resolution image "postage stamps" located in context relative to prior survey imagery to aid interpretation.
Survey data analysis
Develop a flexible framework that enables an analyst to define and detect events of interest in Astrobee survey data, including change & anomaly detection

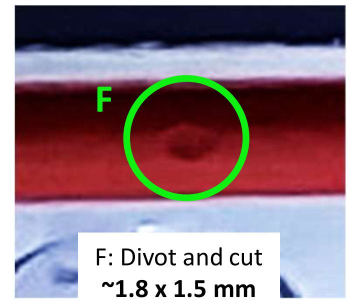


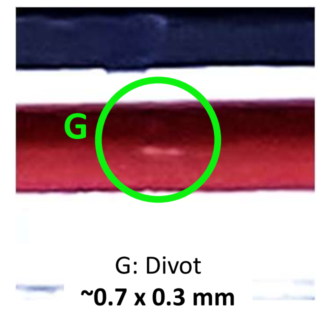
Bumble performing a close-up inspection of a hatch seal in USL during ISAAC-11 (2022/07/11)

Targeted Inspection: Scenario

Timer

- Close-up targeted inspection capability
  - Designate a specific target site (e.g., in imagery from a previous survey)
  - Astrobee autonomously maneuvers its SciCam as close as possible with the target centered
  - Astrobee fine-tunes the standoff distance (~20-25 cm) to properly focus and maximize effective image resolution on the target
- Broad applicability
  - This capability is relevant for any task where maximum image resolution is important
  - An obvious example would be inspection related to ISAAC's original motivating scenario of a micrometeoroid-induced cabin air leak. (The leak site would likely be quite small.)
- Specific application for production ISS operations:
  - The ISS OSO console team reached out to the Astrobee Facility to ask whether Astrobee could perform hatch seal inspections to save crew time.
  - The hatch seals are exposed to the crew cabin and could be damaged whenever the hatches are open (as they usually are)
  - Since hatch closure would be used as an emergency hazard mitigation for some catastrophic hazards, hatch seal damage significant enough to cause a leak would be taken very seriously
  - Hatch seals are presently inspected by crew about twice a year
- Pre-FY24 progress
  - As a late-breaking change to ISAAC Phase 1X, secondary hatch seal inspection tasks were added to mapping activities ISAAC9, 10, and 11. ISAAC/Astrobee worked with OSO to iterate, improving targeting and image quality.
  - OSO's most recent evaluation is that the Astrobee hatch seal inspection demonstrated to date is not yet ready to be deployed for production use, but they are interested in further evaluation if the inspection process can be improved (potential improvements to image quality and analyst user interface for image review)



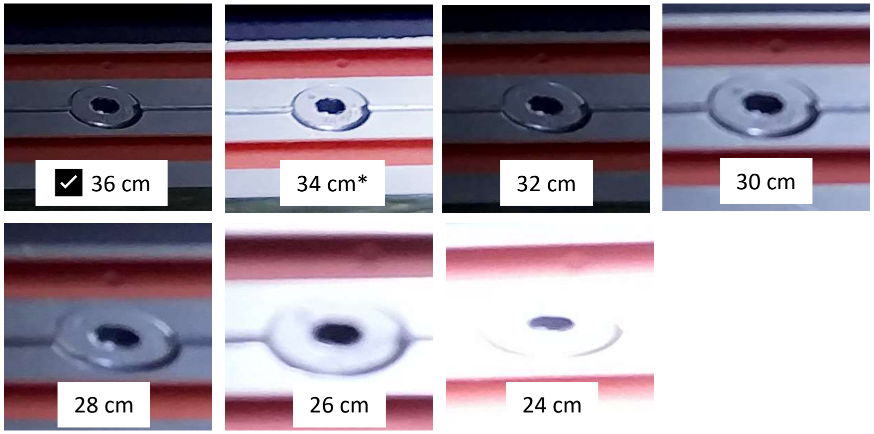


These Astrobee photos confirmed previously known hatch seal damage sites (ISAAC11, 2022/06/16). A new "focus stack" ops approach was used to get improved focus on the targets, yielding Astrobee's highest-resolution ISS images to date (~0.1-0.2 mm).

Targeted Inspection: Precision Targeting and Focus

Timer

- **Precision targeting**
  - We found that we can reliably get the target within the camera view using general Astrobee navigation
  - ISAAC improvements to Astrobee localization help! (more on this later)
- **Precision focus**
  - We can estimate the SciCam-to-target distance using Astrobee's HazCam depth imager (pick the target location from the point cloud)
  - We can set the SciCam focus distance to focus specifically on the target location.
  - This requires calibrating the relationship between the focus index values in the camera API and the actual SciCam-to-target distance. Because the calibration is not uniform across all Astrobees, focus calibration data had to be collected on-orbit.
  - This work can't improve the quality of the very best images, but it can make acquiring them more reliable. This year we were able to get high-quality hatch seal images on our first try, vs. requiring three attempts in 2022.

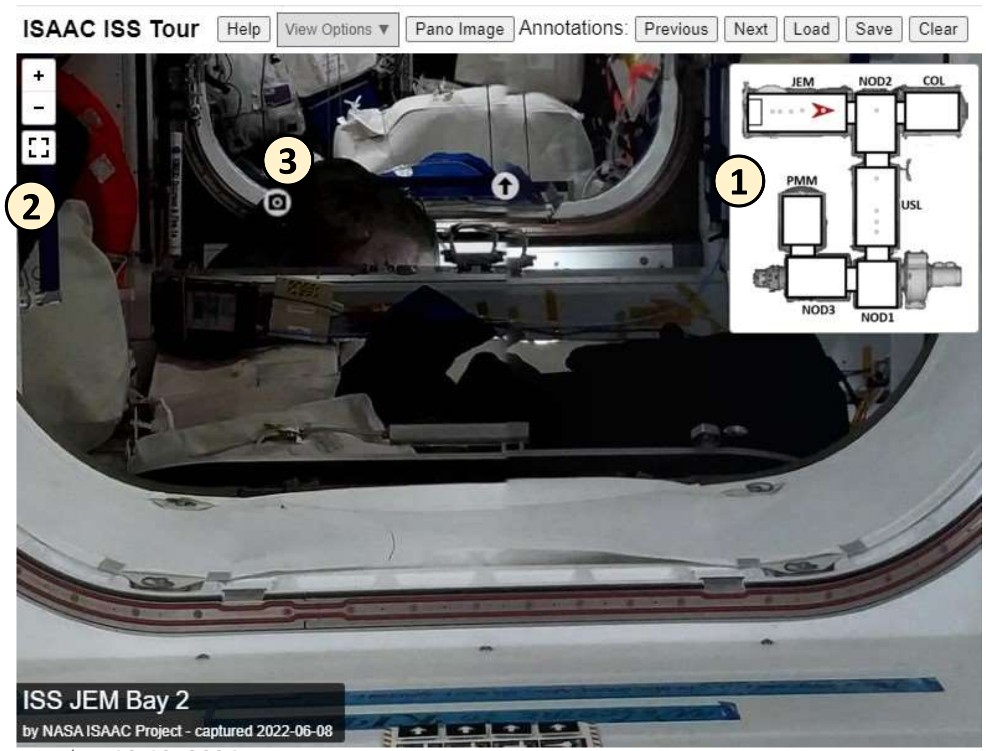


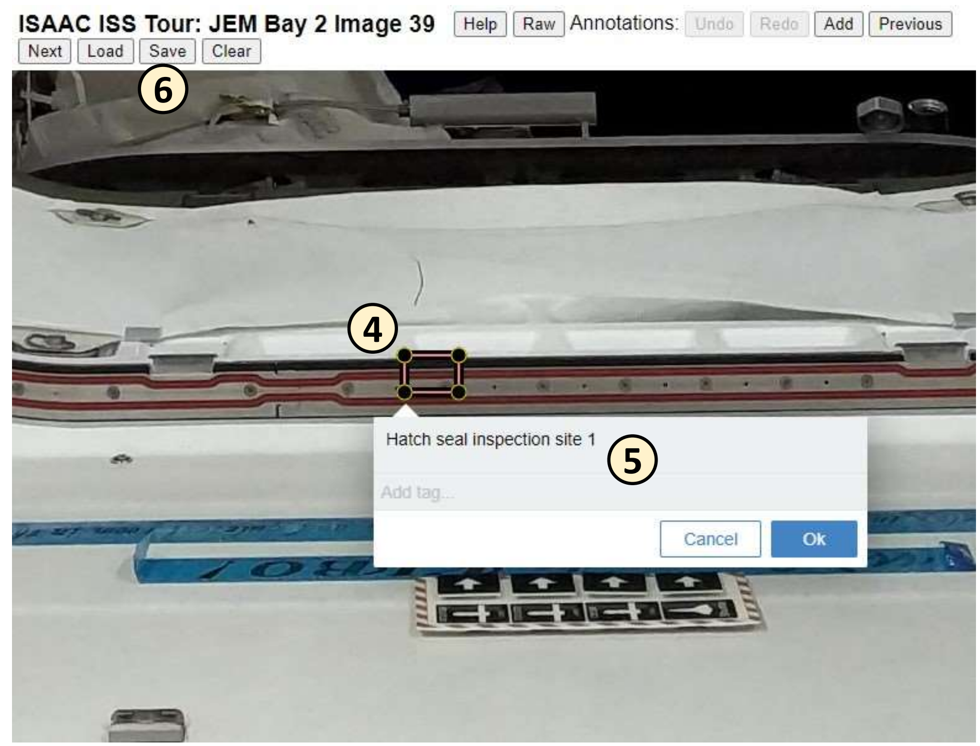
Example focus stack showing how sensitive focal adjustment is at close range (fixed camera focus, standoff distance varies)

Targeted Inspection: Selecting a Target

Timer

1. Select a panorama near the target of interest using the ISS overview map.
2. Use intuitive pan/zoom controls to navigate within the selected panorama.
3. Camera icons indicate the centers of the individual Astrobee SciCam image frames that were stitched to form the panorama. Select the image closest to your target.
4. Annotate a region of the image to target for inspection. The pixel coordinates will be converted to 3D positions for Astrobee to target.
5. Add details about the target (e.g., motivation, priority, what you want the analyst to look for in the resulting imagery).
6. Save your annotations and submit to the Astrobee Facility to convert to an Astrobee ISS targeted inspection plan.

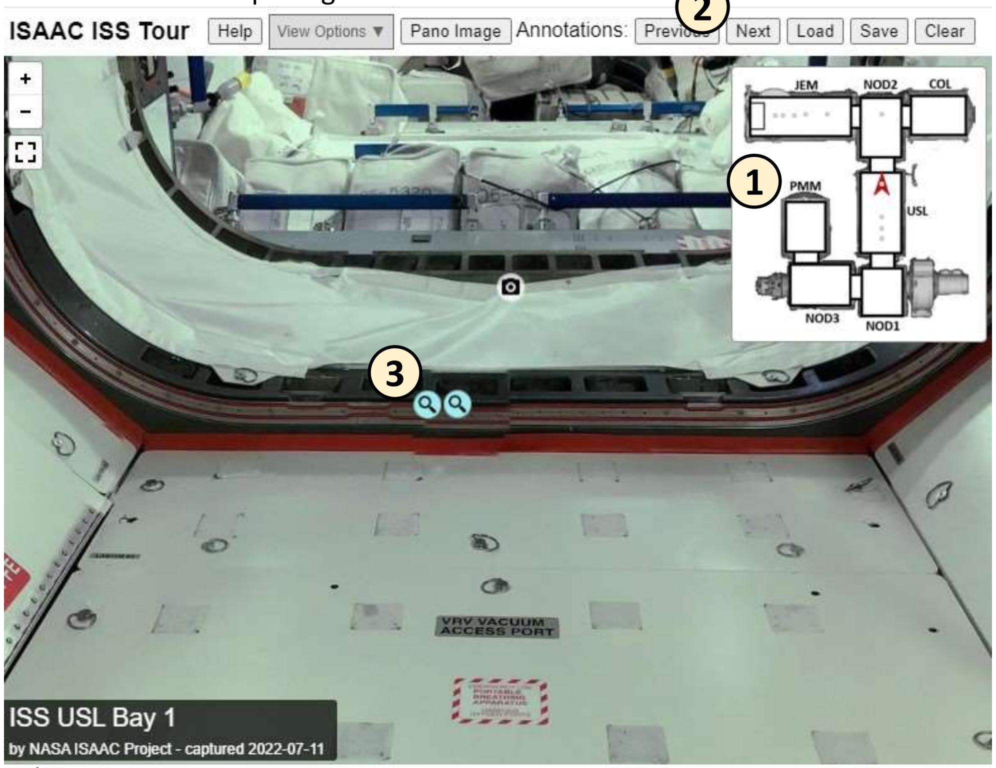


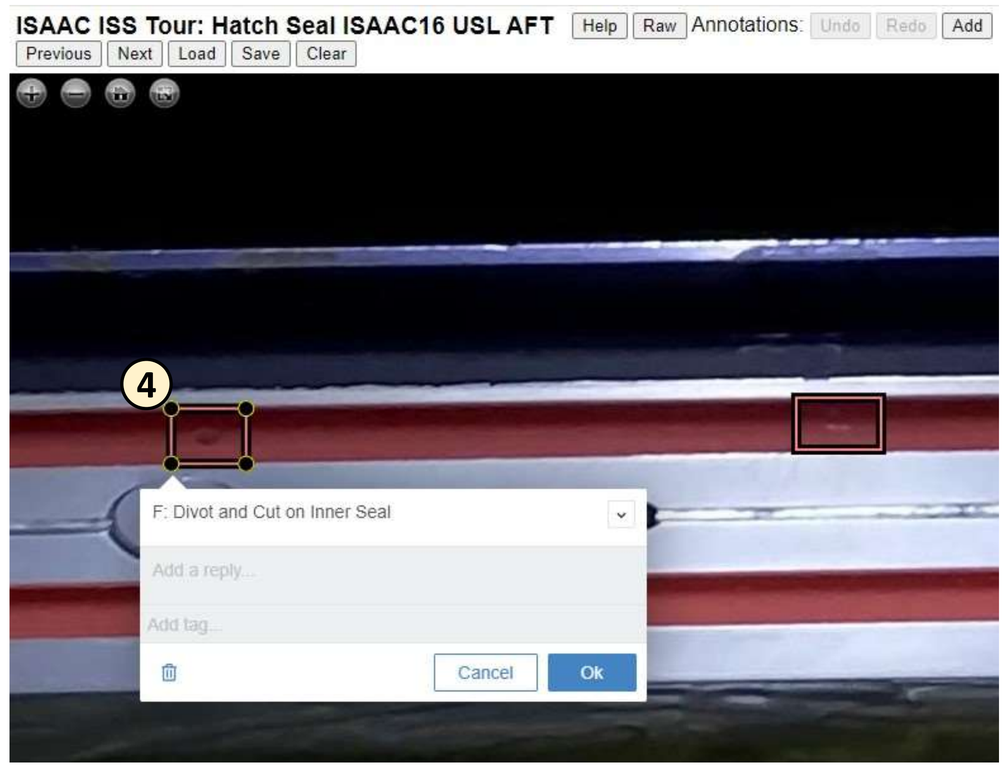


Targeted Inspection: Viewing Results in Context

Timer

1. In the panoramic tour, navigate to the location of your close-up targeted inspection image using the overview map and intuitive pan/zoom controls
2. Or scan through the available locations by repeatedly clicking the “Next” button.
3. When you find your location of interest, click the magnifying glass icon to view the close-up image.
4. You can pan/zoom to explore the image, and add annotations (here, noting the location of visible hatch seal damage).





Panoramic Tour

Timer

Visit Panoramic Tour

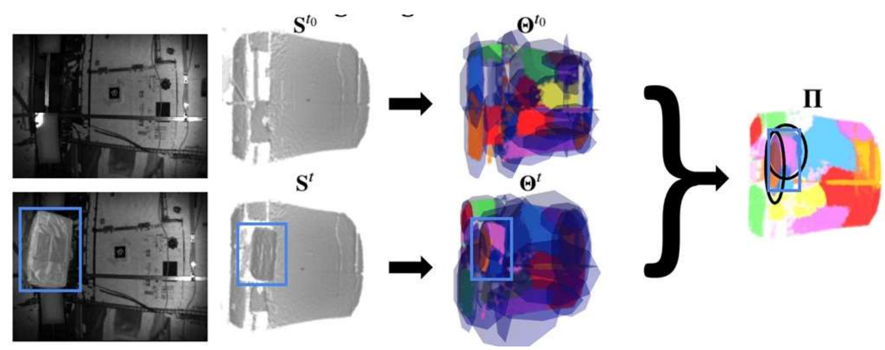
(must be on NASA network)

Survey Data Analysis

Timer

- The **ISAAC Analyst Notebook** is a flexible framework for implementing different types of survey data analysis. Uses have included:
- Label text search: In ISS survey imagery that includes labels, perform image-to-text conversion and indexing, enabling search by label text to find 3D locations with matching imagery.
- Switch state identification: In ISS survey imagery that includes control panels, repeatably identify specific switches on a control panel and classify their switch state.
- Changed object detection: In ISS-analog Astrobee lab imagery that includes before-and-after with objects added, removed, or moved, detect the location of changes.
  - Dinkel, et al. AstrobeeCD: Change detection in microgravity with free-flying robots. *Acta Astronautica* 223:98-107, Oct. 2024. https://doi.org/10.1016/j.actaastro.2024.06.037
  - Santos, et al. Unsupervised change detection for space habitats using 3D point clouds. *Proc. AIAA SciTech Forum*, 2024. https://arxiv.org/abs/2312.02396

Examples of switch patches extracted from larger images and classified on/off. Credit: Tim Rehnstroem.



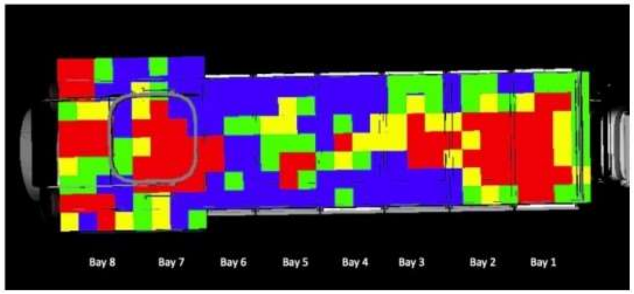
Before-and-after change detection successfully finds the location of a cargo bag added to the scene. Credit: Santos, et al.

Phase 3: Technical Development

Timer

Task	Status
Coordinated Execution	
Onboard planning and execution: Demonstrate coordinated multi-robot automated task planning and execution with two Astrobees on ISS	Demonstrated in software sim, granite lab, and on-orbit in ISAAC13 and ISAAC16.
Automated multi-robot planning will be available for future Astrobee platform users as part of ISAAC open-source release	Happens automatically with ISAAC rolling open-source release
Astrobee Robot Software	
Space ROS: Begin migration to ROS 2 framework with better quality assurance, potentially enabling future software infusion into robot platforms with more mission-critical capabilities	Well under way. Core modules and software sim migrated.
Robust Navigation: Improve overall Astrobee navigation, especially including sparse map localization reliability	Major upgrade (“split localizer”) tested using ISS imagery playback and granite lab. Performed well during ISAAC16 on ISS!
Inter-robot Communication: Make it easier to send new message types without needing an expert to write custom code.	Tested during ISAAC13 + ISAAC16 and now part of baseline Astrobee flight software.

Astronaut Chris Cassidy poses with Bumble and Honey Astrobees on ISS



Astrobee localization is complicated by areas of the prior map that have fewer features (red)

Coordinated Execution: Multi-Robot Mapping Scenario

Timer

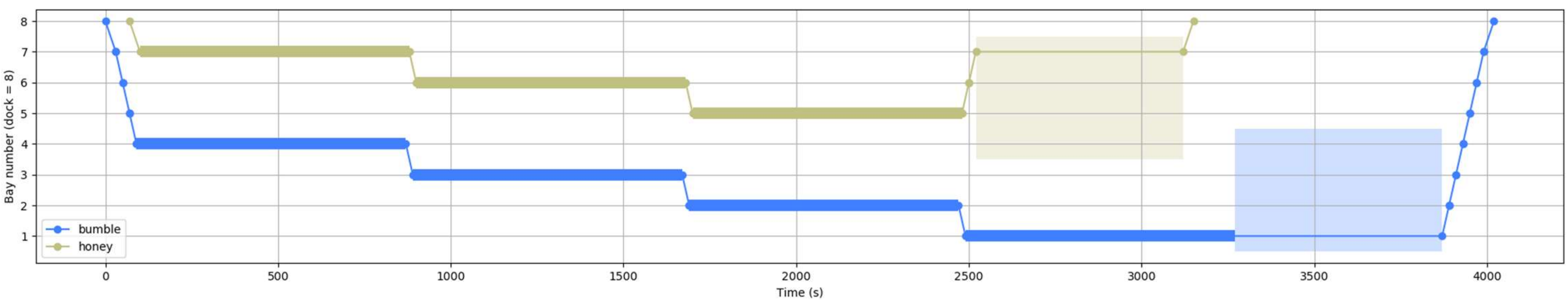
- Aim for near-term infusion of ISAAC multi-sensor mapping to **operational use with Astrobee on ISS** in addition to medium-term use on a future Gateway IVR platform
- Autonomous robotic multi-sensor mapping can provide several operational benefits to ISS:
  - **Save crew time.** Autonomous robotic data collection could partially replace the need for crew safety videos (although Astrobee can't cover all parts of the ISS)
  - **Provide more accurate and timely information**. The ISS interior is a very dynamic environment, and the robot could collect map updates more frequently than the current tempo of crew safety videos (once every few months).
  - **Enhance usability.** Enable users to navigate directly to the area of interest in a 3D model of the ISS, rather than downloading a massive video file and scrubbing through it to find the area of interest
  - **Provide new "superpower" senses**. In the near term, we propose to enhance ISAAC maps with sound source information from the SoundSee microphone array payload for Astrobee. Going forward, the ISAAC mapping framework can easily accommodate new sensor payloads added to Astrobee.
  - **Enable automated change and anomaly detection.** Automated detection of many types of anomaly (e.g., hardware intruding into a keepout zone) requires understanding the scene geometry and naturally builds on ISAAC's 3D map building. Without automation, labor-intensive manual review of safety videos is presently slow enough that by the time the review is done, its recommendations are often obsolete.
- Initial candidate ISS use cases for map data:
  - **Safety:** Check keepout zone violations, hatch seals, Emergency Egress Guidance System visibility, translation paths, and other items.
  - **Logistics:** Spot check the locations of cargo. Review available capacity of stowage locations. Correlate data from RFID Recon payload and update Inventory Management System database.
- The ISS program is currently considering a proposal from the Astrobee Facility team to perform Astrobee surveys of the USOS every six months, starting in January 2023
  - Further ISAAC development is needed to streamline operations and post-processing so the Astrobee Facility can support this service going forward

Bumble and Queen operate side-by-side to map USL during ISAAC-11 (2022/07/11)

Coordinated Execution: Onboard Planning and Execution

Timer

- ISAAC used onboard planning and execution to manage multi-robot mapping during the ISAAC13 and ISAAC15 activities on ISS.
- Motivation:
  - Prior ISAAC multi-robot survey activities really needed two ground teams to operate two robots. This staffing level is unsustainable for production ops.
  - With coordinated execution and increased automation overall, one team can operate both robots with less workload than either team had using the previous operational approach.
  - Automation can also improve reliability (e.g., previous activities have silently failed to record required robot data due to subtle mistakes made by an expert operator under very heavy workload)
- Prior to the activity:
  - An operator specifies a list of survey goals together with hints about goal-to-robot allocation and goal ordering.
- During the activity:
  - A planner running on one of the Astrobees generates a multi-robot plan to achieve the goals. The planner orders and links the plan actions to enforce a collision-avoidance constraint between robots.
  - A multi-robot executive running on one of the Astrobees issues commands to both Astrobees. The executive can command parallel actions on the robots, but it will make one robot wait on the other when commands are linked to enforce constraints.



Summary plot of two-Astrobee survey plan enforcing collision avoidance– Robot position (y axis) vs. time (x axis)

Astrobee Robot Software: Space ROS

Timer

ROS1
Widely Used for Research
EOL in 2025

→

ROS2
Terrestrial Industry Standard

→

Space ROS
Space Cert. S/W

Table 1. Summary of ROS 2 features compared with ROS 1.

Category	ROS 1	ROS 2
Network transport	Bespoke protocol built on TCP/UDP	Existing standard (DDS), with abstraction supporting addition of others
Network architecture	Central name server (roscore)	Peer-to-peer discovery
Platform support	Linux	Linux, Windows, and macOS
Client libraries	Written independently in each language	Sharing a common underlying C library (rcl)
Node versus process	Single node per process	Multiple nodes per process
Threading model	Callback queues and handlers	Swappable executor

S. Macenski, T. Foote, B. Gerkey, C. Lalancette, W. Woodall, “Robot Operating System 2: Design, architecture, and uses in the wild,” Science Robotics vol. 7, May 2022.

- ROS = Robot Operating System
- ROS1 will be deprecated in 2025
- New tech developments from industry players moving to ROS2
- Space ROS (kicked off via Blue Origin ACO, now managed by an industry technical consortium) is an open-source extension of ROS2 designed to support robot flight software

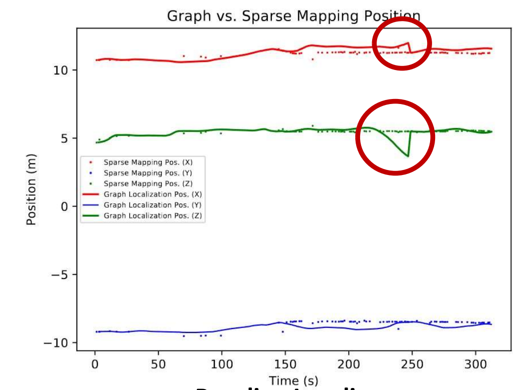
Features	ROS1	ROS2	Space ROS
Swappable middleware		✓	✓
Multi-platform		✓	✓
Built-in network security, authentication, quality of service		✓	✓
Additional aerospace-focused V&V			✓
Artifacts for NPR 7150.2 certification / analysis			✓

Robust navigation

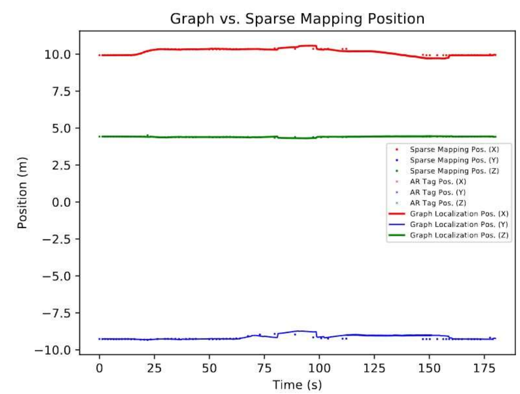
- ISAAC developed AstroLoc2, a "split localizer" version of Astrobee's previous localization system that provides a greater degree of separation between the visual/inertial odometry (VIO) and map-matching components, along with other improvements.
- The split localizer mitigates known Astrobee navigation problems:
  - Extended map-matching outages. With a looser coupling between components, AstroLoc2 can keep VIO running fast as before while allowing map-matching to use a slower but more robust algorithm that is less likely to lose its map-matching fix.
  - Overshooting the end of motion. This can be caused by scale ambiguity in a one-camera system. AstroLoc2 incorporates depth information from Astrobee's HazCam to help resolve this ambiguity.
  - Abruptly flying off course into a wall, even when trying to station keep. This can be caused by a rare bad interaction between localization components when the robot gets a new map-matching fix after an extended outage. AstroLoc2 mitigates the bad interaction by loosening the coupling between components.
- AstroLoc2 performed well during ISAAC's final ISS activity, overcoming navigation challenges previously observed in the ISS US Lab module, and has become the standard localization algorithm for future Astrobee activities.

Inter-robot communication

- New ISAAC-developed bridge nodes are now available in Astrobee flight software to relay ROS messages between multiple Astrobees on ISS using the DDS network protocol. The nodes were used to implement ISAAC's coordinated execution system. They are available for any guest scientist.
- Compared to a previous bridge implementation, the main benefit is reduced developer overhead to relay new message types. Essentially, you just need to add the relayed message topics to a single configuration file without requiring expert knowledge to write new code for interconverting ROS and DDS data structures.



Baseline Localizer



Split Localizer

The new split localizer shows better agreement between position estimates, especially avoiding spurious spikes (circled in the plot above) that can cause Astrobee to abruptly fly off course.

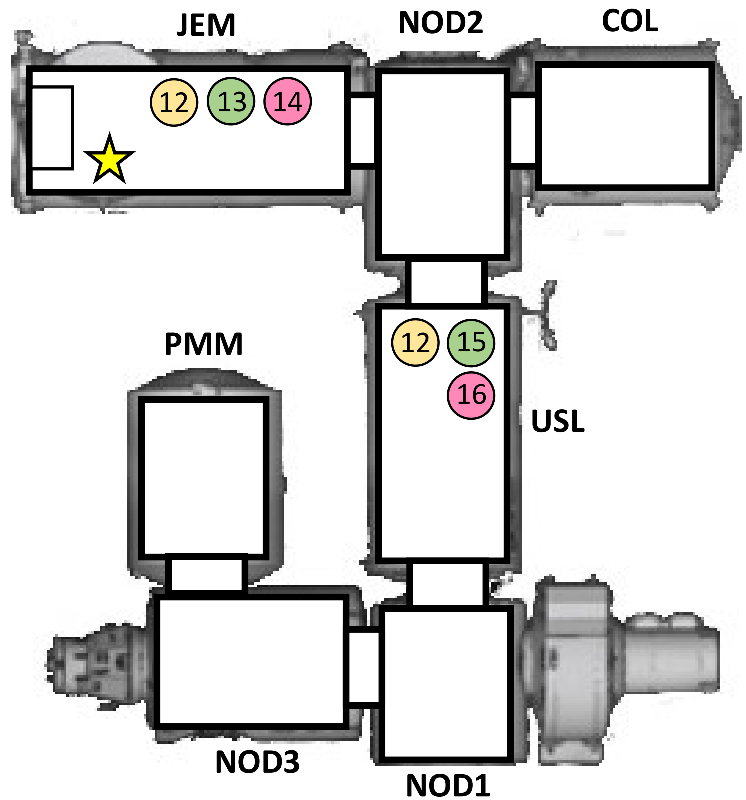
ISAAC FY24 Astrobee ISS Campaign

Timer

Phase 1XX: 2 of 5 activities

Act #	Ph.	Objective	Module	Date
12		Nav mapping & calibration	JEM	2023/12/29
14	1XX	Targeted inspection	JEM	2024/02/02
13	3	Multi-sensor mapping coordinated execution	JEM	2024/04/17
15	3	Multi-sensor mapping coordinated execution	USL	2024/05/31
16	1XX	Multi-sensor mapping coordinated execution / Targeted inspection	USL	2024/07/05

- We reordered to execute ISAAC14 before ISAAC13 based on software readiness concerns.
- Coordinated execution objectives were later added to ISAAC16 due to problems encountered during ISAAC15.

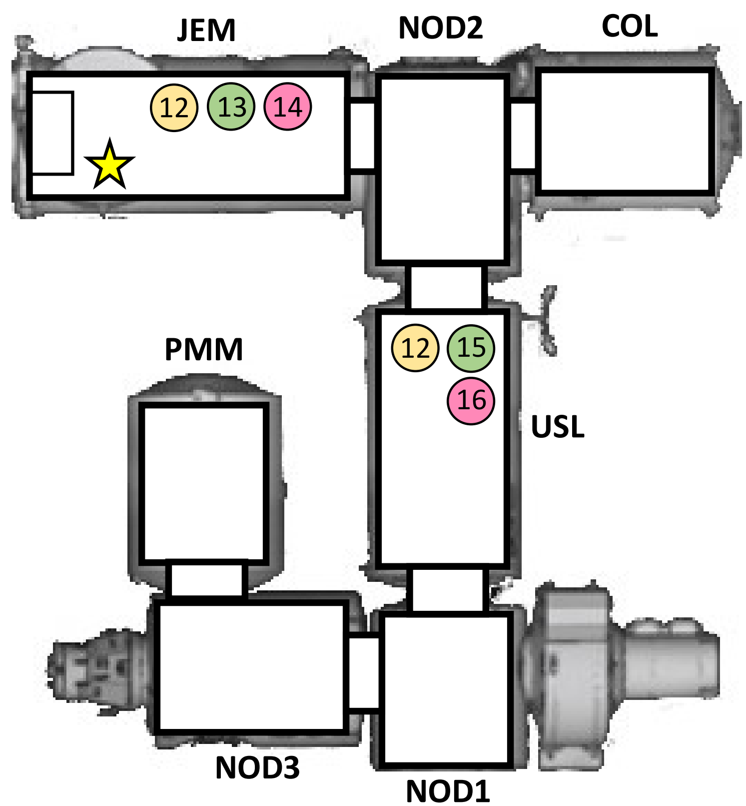


Timer

Phase 3: 2† of 5 activities

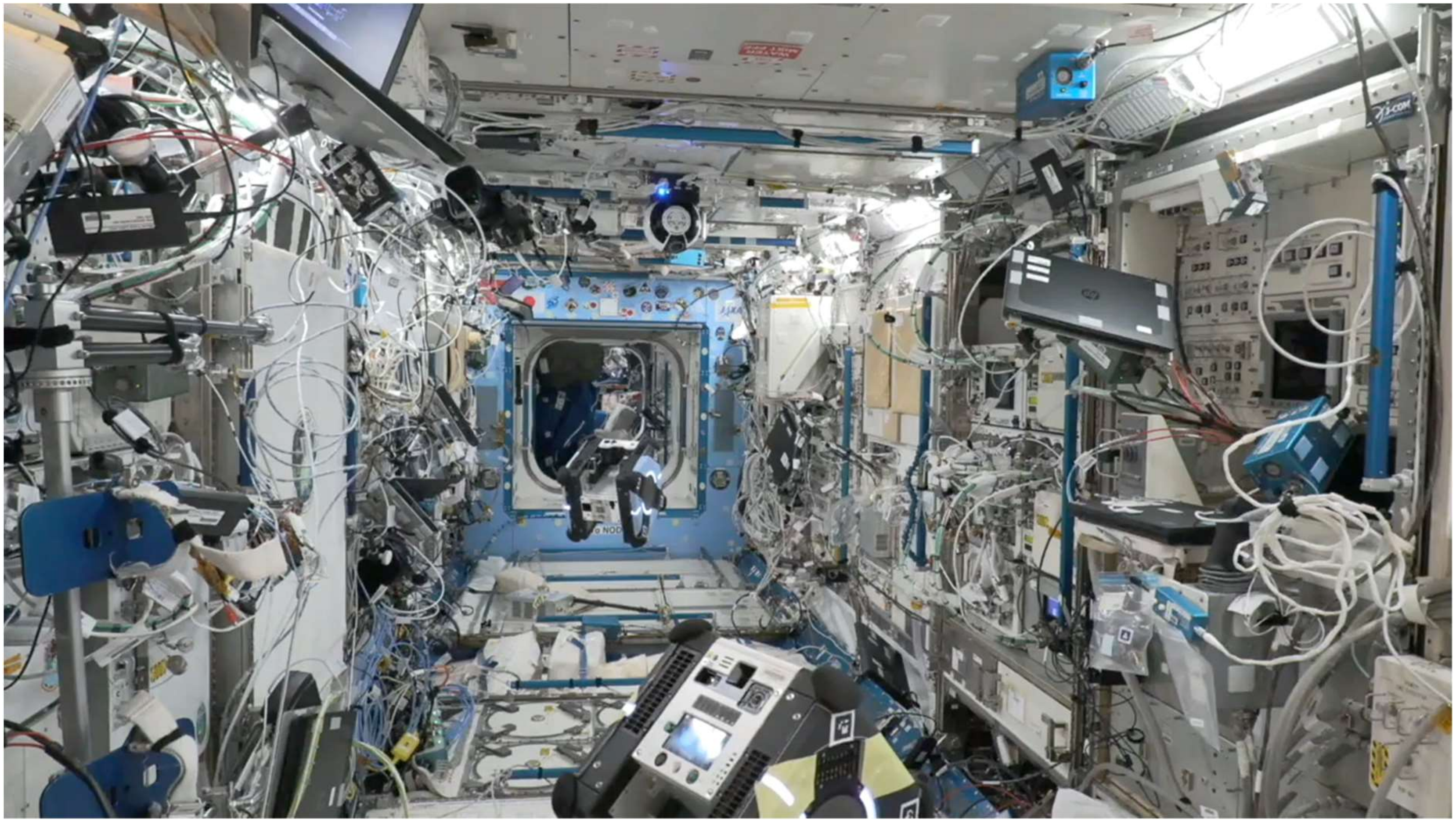
Act #	Ph.	Objective	Module	Date
12		Nav mapping & calibration	JEM	2023/12/29
14	1XX	Targeted inspection	JEM	2024/02/02
13	3	Multi-sensor mapping coordinated execution	JEM	2024/04/17
15	3	Multi-sensor mapping coordinated execution	USL	2024/05/31
16	1XX	Multi-sensor mapping coordinated execution / Targeted inspection	USL	2024/07/05

- We reordered to execute ISAAC14 before ISAAC13 based on software readiness concerns.
- †Coordinated execution objectives have been added to ISAAC16 due to incomplete objectives in ISAAC15.



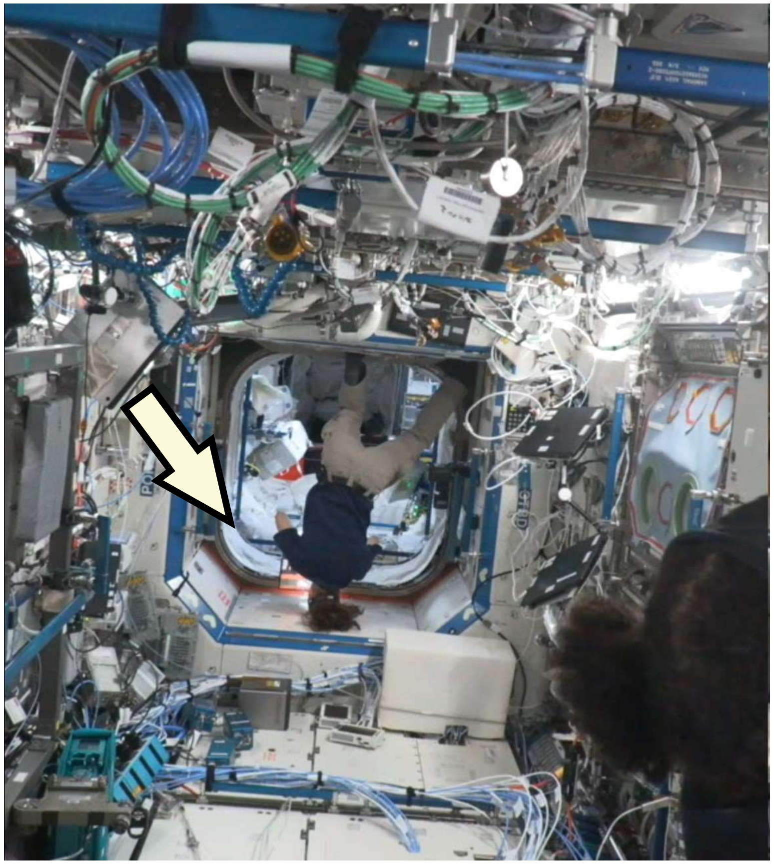
ISS: ISAAC 13 – Coordinated Execution

Timer

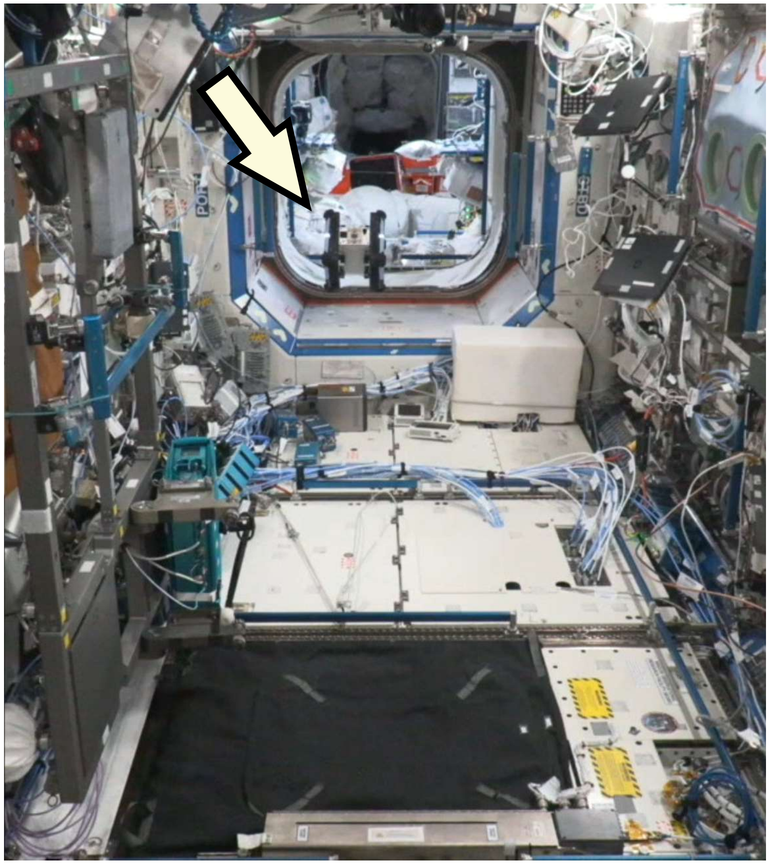


ISS: ISAAC 16 – Hatch Seal Inspection

Timer



NASA astronaut Tracy Dyson performs a manual hatch seal inspection on the US Lab module aft hatchway using an iPad

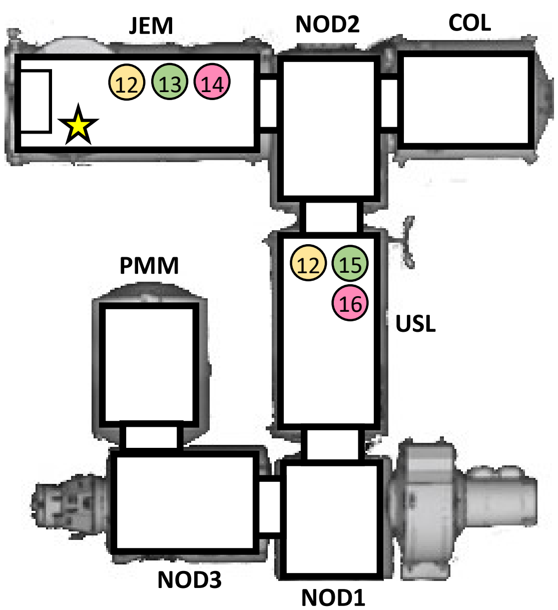


Bumble Astrobee performs an automated robotic inspection of the same hatch seal on the same day (2024/07/05)

ISS: Overcoming Ops Challenges

Timer

- The ISAAC team responded gracefully to many challenges during our FY24 ISS campaign
- ISAAC13:
  - Operating around crew: This activity really pushed Astrobee's ability to operate in a confined space: at one point, there were three crew members and two Astrobees together in the JEM, and Astrobees continued operating through getting bumped by loose cargo bags and crew members multiple times
  - Despite this problem, we successfully demonstrated coordinated execution on ISS for the first time.
- ISAAC15:
  - USL navigation: This activity was a bust because the Astrobees were unable to navigate in the USL using the new navigation map built using ISAAC12 imagery.
  - After realizing the problem, we used the remaining time in the activity to collect more imagery to build a better map. The ISAAC15 objectives had to be slipped to ISAAC16.
- ISAAC16:
  - Honey hardware failure: One of Honey's onboard processors failed to power up for reasons that remain unknown (it later resumed working, but not that day). We lost activity time debugging Honey and ended up only being able to use Bumble.
  - Despite the failure, we still managed to complete the ISAAC15 *and* ISAAC16 primary objectives:
    - The USL navigation problems from ISAAC15 were solved by a combination of an improved navigation map with more imagery and ISAAC's first use of our more robust localization algorithm (AstroLoc2)
    - We responded to the loss of Honey by editing the survey specification on the fly to allocate all goals to Bumble and ran coordinated execution successfully.



ISAAC Assessment Summary

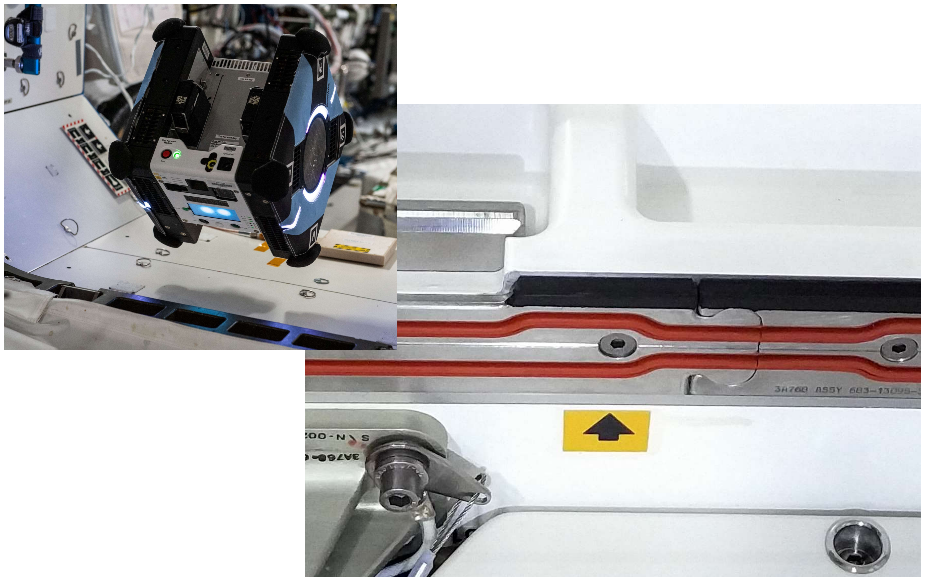
Timer

Project Name	Performance				Comments
	C	S	T	P	
Annual					Project complete!

	Oct	Nov	Dec	Jan	Feb	MYR	Apr	May	Jun	Jul	Aug	APR
C												
S												
T												
P												

This has been an unusually drama-free year for ISAAC – past years had lots of yellow or red due to factors like mid-year budget cuts, a key robotic platform not available, delays getting approved to run ISS activities, the whole project team getting pre-empted to do urgent mission work, etc.

ISAAC FY24 Summary

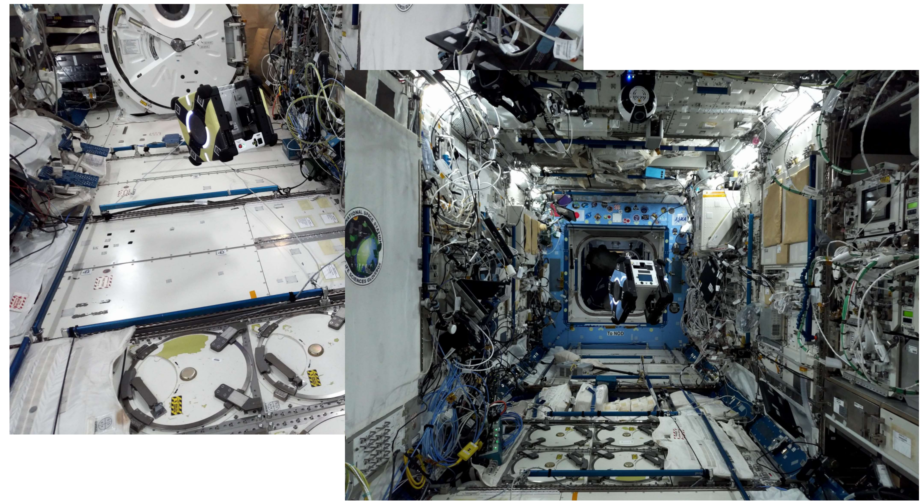


Phase 1XX

- ISS demo: Astrobee close-up targeted inspection of hatch seals
    - Demonstrated in the JEM in ISAAC14
    - Final demonstration in USL in ISAAC16

Technology

- Close-up targeted inspection
- Change & anomaly detection



Bumble and Honey image each other in ISAAC13

Phase 3

- ISS demo: Coordinated ISS interior survey with multiple Astrobees
    - Demonstrated in the JEM in ISAAC13
    - Unable to execute in ISAAC15
    - Will retry in the USL in ISAAC16

Technology

- Multi-robot coordination
- Begin migrating to ROS 2
- Map-building and navigation

ISAAC Project Summary

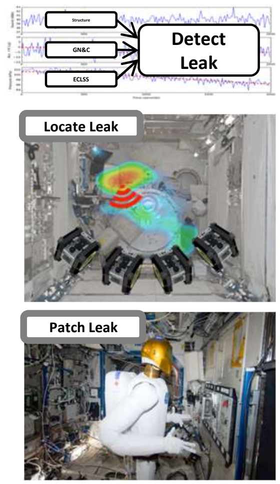
Timer

- ISAAC has been a very broad study of IVR integration
  - Just looking at our FY24 capstone demonstrations covered here won't give you a full picture of the topics we've studied over the years.
  - Examples of pre-FY24 topics include logistics management and integrated 3D user interfaces.
- Over the years, ISAAC accomplished many firsts:
  - First time a robot navigated between modules of a space station
  - First time a robot located a sound source in space (collaboration with the Bosch USA SoundSee payload team)
  - First time a robot mapped the interior of a space station for human visualization
- ISAAC's lasting contributions include research publications, developing future NASA talent through intern mentorship, and open-source software contributed both to the Astrobee Robot Software and in ISAAC's own repository.
- Infusion prospects:
  - Gateway IVR subsystem: ISAAC's original infusion target is on hold until development of the IVR subsystem itself is funded.
  - Production use on ISS: ISAAC has successfully matured an ops approach for performing interior surveys of the ISS using Astrobee. We've heard enthusiastic interest from ISS operators in using the resulting panorama data sets, and automating surveys could save crew time. Unfortunately, given the budget-constrained ISS situation, the program is currently unable to invest in sustaining periodic Astrobee survey ops.
  - Other opportunities: We are continuing to seek other infusion opportunities, for example in the Commercial LEO Development Program (CLDP) and the Earth Independent Operations (EIO) section of the Mars Campaign Office

Autonomous Logistics Management

Robotic cargo transfer

Integrated Fault Management



QUESTIONS?