Integrated System for Autonomous and Adaptive Caretaking (ISAAC): Phase 1 Low-Fidelity Demo

ISAAC team, edited by Trey Smith

30 Jun 2020

Outline

- **Context**
- **Software architecture**
- **Lab demo**
  - Mapping checkout activity
- **Software simulation demo**
  - Survey activity
  - Fault management activity
- **Other progress**
- **Conclusion**

ISAAC team

Staff

- Abiola Akanni (planning)
- Oleg Alexandrov (former mapping lead)
- Julia Badger (former JSC project manager)
- Laura Barron (JSC technical lead)
- Maria Bualat (deputy project manager, ops lead)
- Chuck Claunch (former JSC technical lead)
- Brian Coltin (Ames technical lead)
- Matt Deans (former project manager)
- Lorenzo Fluckiger (Astrobee architect)
- Jeremy Frank (planning architect)
- Janette Garcia (spatially linked models)
- Katie Hamilton (former Ames technical lead)
- Lewis Hill (operator interface)
- Patrick Knauth (former member, MAST)
- David Lees (operator interface)
- Marina Moreira (Astrobee integration)
- Ted Morse (Astrobee integration)
- Nicole Ortega (spatially linked models)
- Jonathan Rogers (JSC project manager)
- Khaled Sharif (operator interface lead)
- Trey Smith (project manager)
- Ryan Soussan (localization)

Interns

- Holly Dinkel (affordance templates)
- Varsha Kumar (localization)
- Ian Miller (NSTRF fellow, survey planning)
- KJ Newman (localization)

Many thanks to all!

ISAAC phase 1 low-fidelity demo context

- **ISAAC is a three-year project split into three phases of roughly one-year length**
- **ISAAC phase 1 includes two demos:**

			Demo environment		
Demo iteration	**Date**	**Phase 1 tech maturity**	**Software simulation**	**Lab hardware**	**ISS**
Low-fidelity	2020/06	Early version	x	x	
High-fidelity	2021/01 (est.)	Complete	x	x	x*

* Subject to availability of ISS Astrobee facility resources

Our topic today

- **The demo satisfies the milestone from Gateway-ISAAC MoU to "Demonstrate spatial and logical data registration between robotics and spacecraft"**

ISAAC phase 1 development

- **Spatially linked model**
  - A database of information about vehicle components that links **semantic information** with **spatial information**
  - Why: A common vocabulary of vehicle components is critical for enabling robust communication between vehicle and robot autonomy
    - Semantic information: determine what task is needed
    - Spatial information: send a robot to the right place to help
  - For example, an ECLSS return vent can be modeled at multiple levels:
    - Functional role: what duct it connects to, nominal flow rates, etc.
    - 3D geometry and 6-DOF pose – in a common coordinate frame usable by robots
    - Robotic task information: such as how to detect or clear a vent blockage
- **Multi-sensor mapping**
  - Fusing data from multiple sensors carried on an autonomous mobile inspection robot, to form a co-registered map of the vehicle interior
  - Why: Map registration in the common coordinate frame lets us **leverage the spatially linked model** and enables **multi-sensor analysis**
  - For example, both a hissing sound and a cold spot could indicate a leak.
    - Co-located anomalies from both acoustic and thermal IR sensors would provide a stronger indicator of leak location than either sensor alone.

ISAAC phase 1 development

- **Integrated data interface**
  - A web-based operator interface that displays our multi-sensor maps, spatially linked model data, time series telemetry, and robot state.
  - Why: Unified interface provides better situation awareness
  - Demo videos were recorded using integrated data interface screen capture
- **Anomaly detection**
  - Automated time series and imagery anomaly detection are already used in the fault management scenario
  - In future work, we will expand this into a flexible framework for managing multiple detection algorithms for different tasks
- **Software simulation infrastructure**
  - To test time series anomaly detection in a relevant scenario, we developed a low-fidelity simulator for a small subset of the ISS ECLSS subsystem
  - Multiple new sensors were added to the Astrobee software sim to enable testing of multi-sensor mapping
- **Survey planner**
  - Coarse module interior geometry is used to generate an Astrobee survey trajectory for generating a high-quality map of the interior

Demo activities

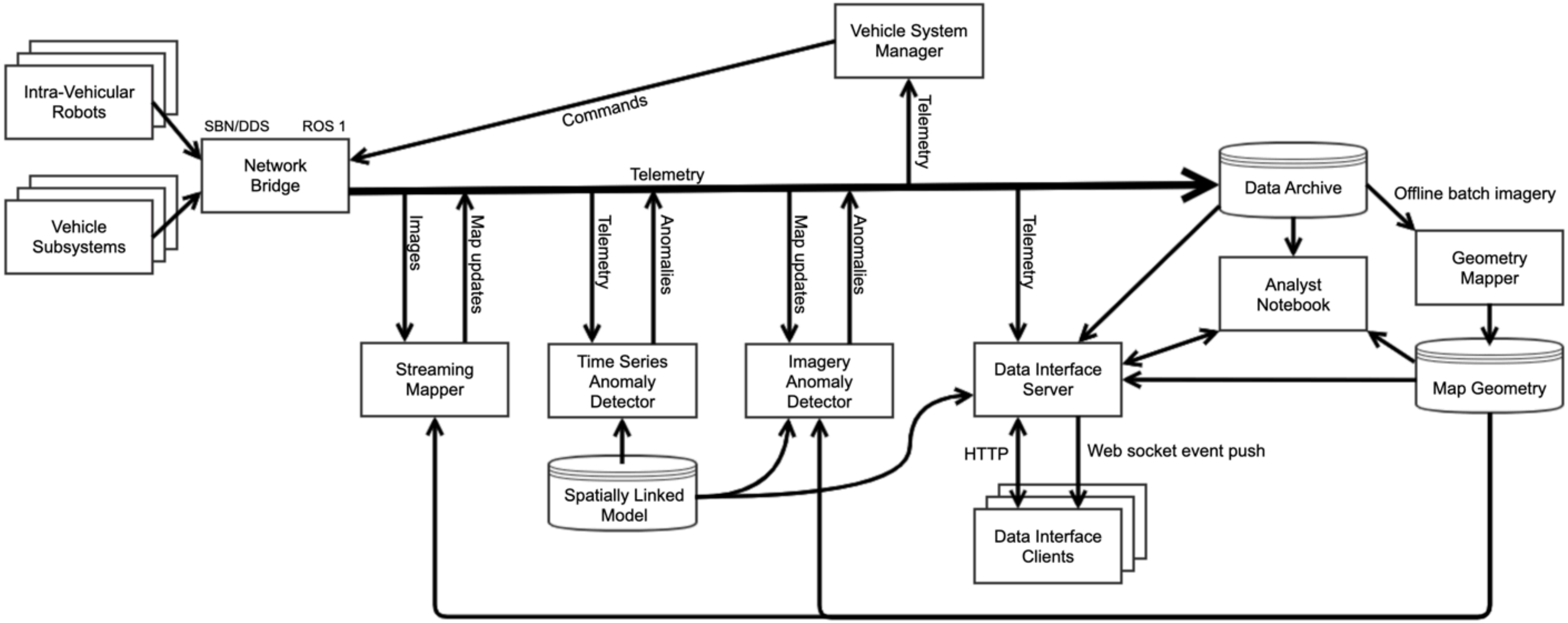
		Environment	
Activity	**Objective**	**Low-fidelity**	**High-fidelity**
Calibration and mapping checkout	Calibrate sensors and validate mapping approach with a small survey	Lab demo	ISS session 1*
Baseline survey	Cover full survey area, build baseline 3D geometry and high-res surface texture	N/A	ISS session 2*
Follow-up survey	Cover full survey area again, stream live updates to multiple sensor layers. (Checking for changes and anomalies not implemented yet.)	Software sim	ISS session 3*
Fault management	Respond to a simulated high-CO_2 anomaly in the JEM by sending Astrobee to autonomously check whether a vent is blocked	Software sim	ISS session 3*

Our topic today

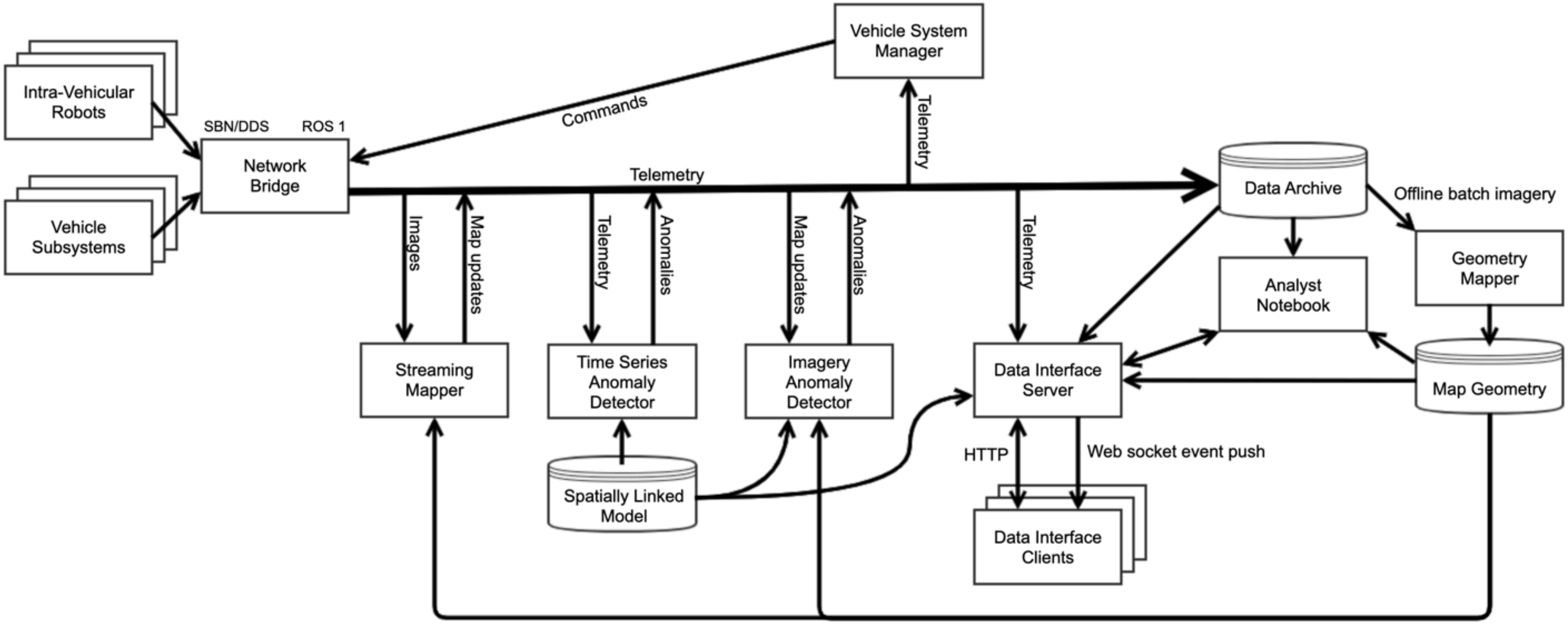
* Fall back to lab demo if ISS Astrobee facility resources not available

SOFTWARE ARCHITECTURE

Phase 1 software architecture

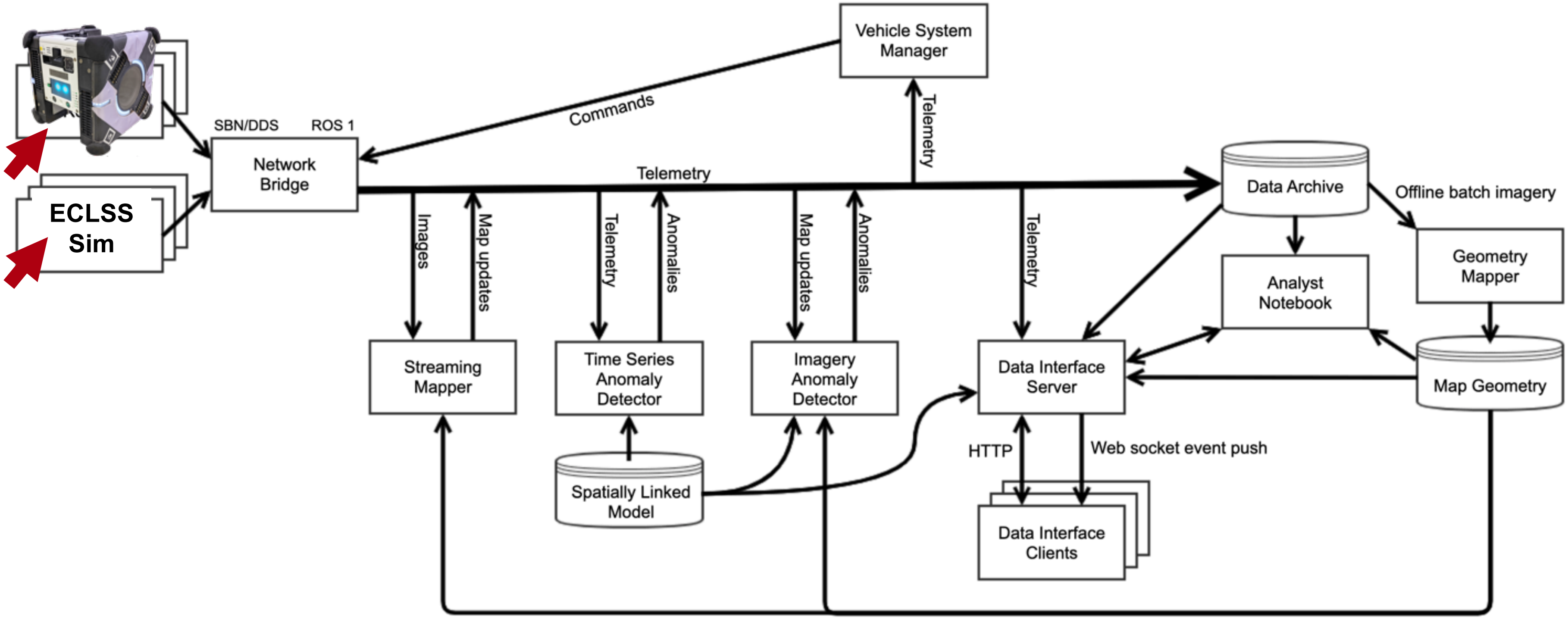


Software hosting



- For eventual Gateway deployment, all these modules (possibly excepting operator interface) would be **hosted onboard** the Gateway spacecraft, to support autonomy goals
- For ISAAC phase 1 demos, as much as possible, modules will be **hosted on the ground** – computing onboard ISS is not relevant to the goals of the demonstration and would incur major overhead.

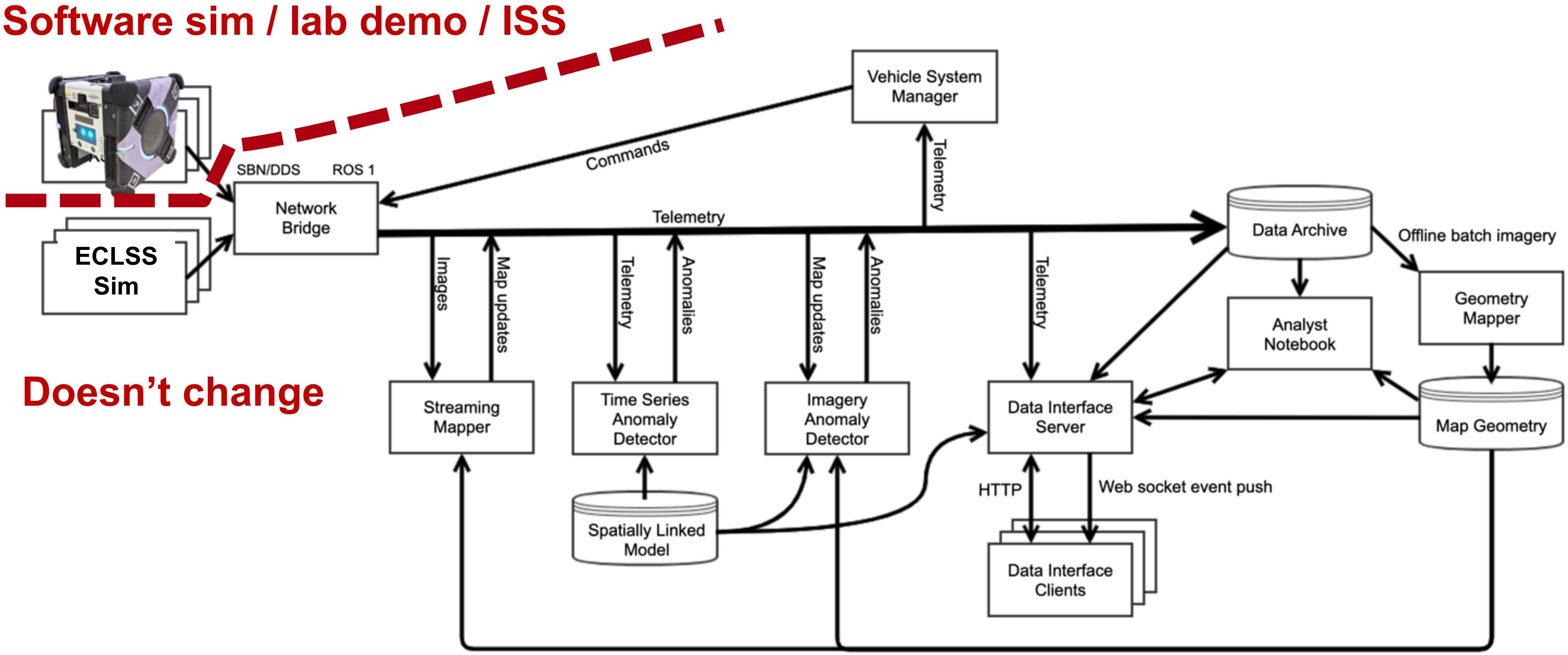
Robots and vehicle subsystems



During phase 1:

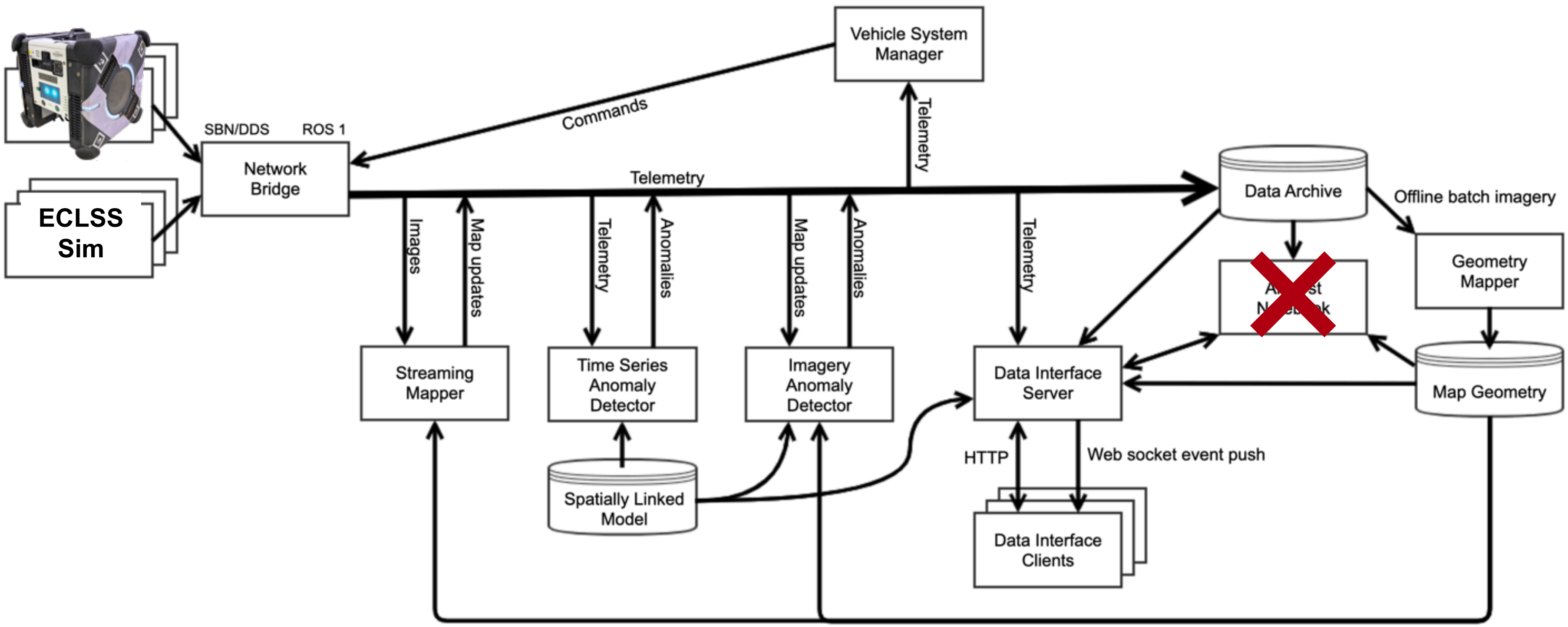
- **Astrobee** is the only robot involved. (Robonaut not available on ISS yet.)
- To demonstrate vehicle subsystem integration, we implemented our own **low-fidelity ISS ECLSS sim** that produces simulated telemetry. It includes a fault injection feature to enable our fault scenario. (Accessing actual ISS telemetry or high-fidelity sim would incur major overhead and is not critical for our technical objectives.)

Environment variations



- Phase 1 demo environment variations only affect the robot section of this architecture
- The rest of the architecture runs on ground computing and doesn't change
  - The robot commanding and telemetry interface is the same regardless of whether the robot is a software simulation, an Astrobee ground unit in the Granite lab, or running on the ISS

Not included yet

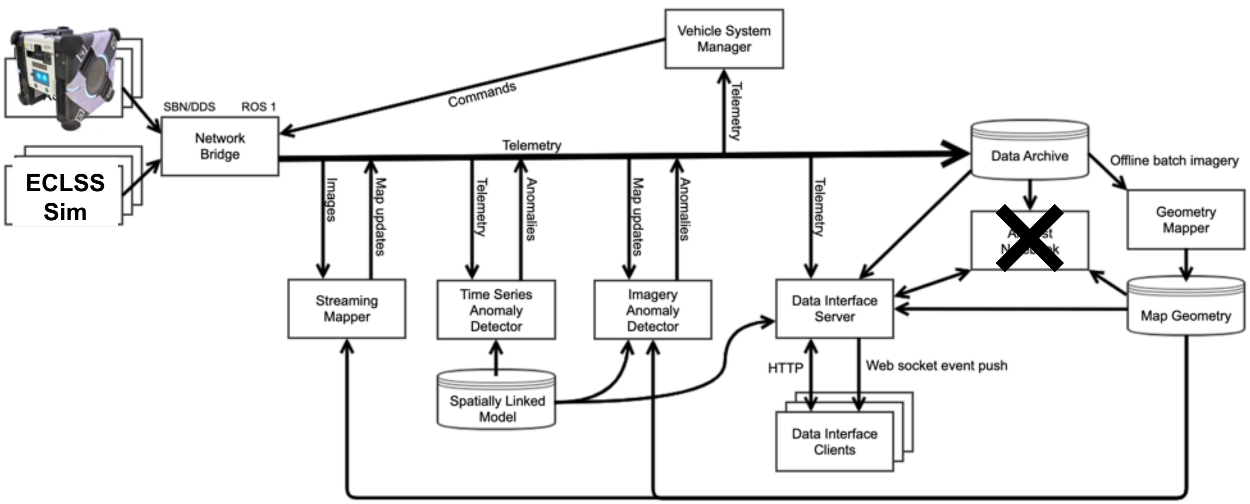


For the low-fidelity demo:

- The **Analyst Notebook** is the only software module in this architecture that hasn't been implemented yet
  - This module will provide an analyst with a flexible interactive interface for running new change and anomaly queries against vehicle and robot telemetry.
- The other modules exist but are not yet fully mature

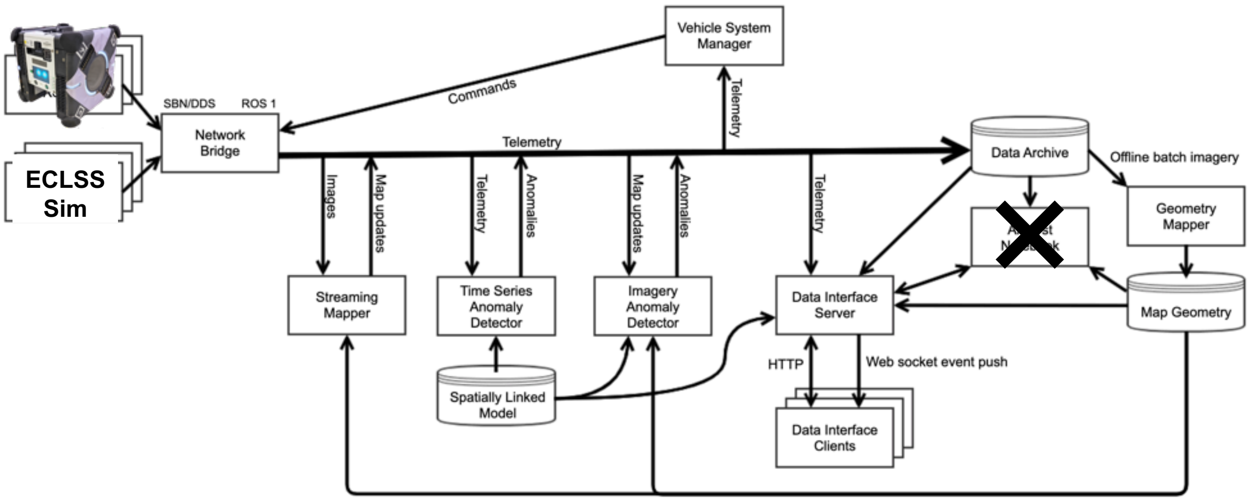
MAPPING CHECKOUT ACTIVITY

Mapping checkout activity



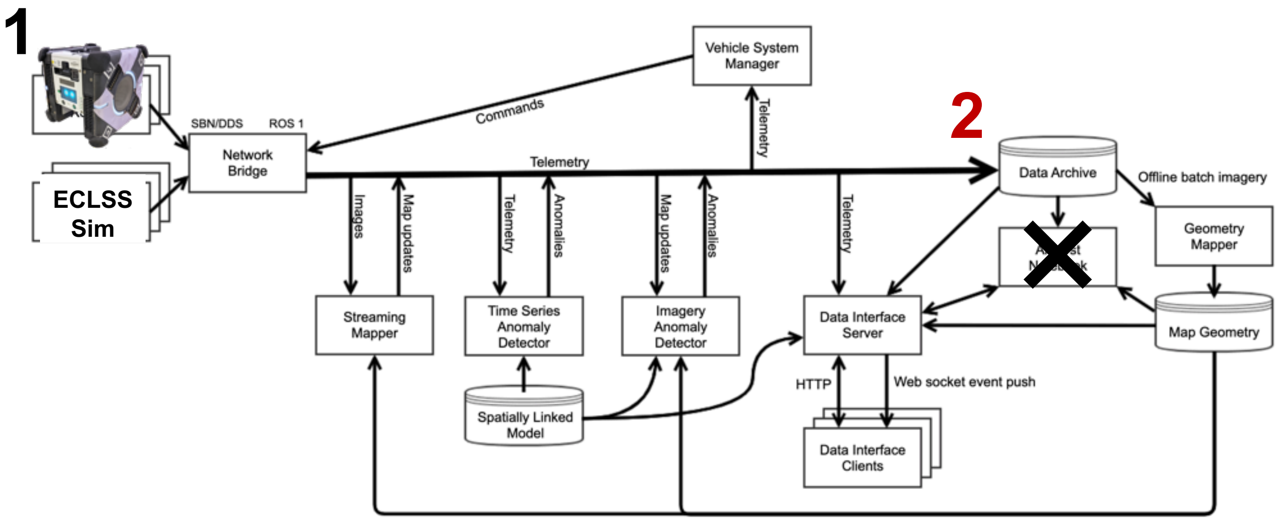
Mapping checkout activity

1



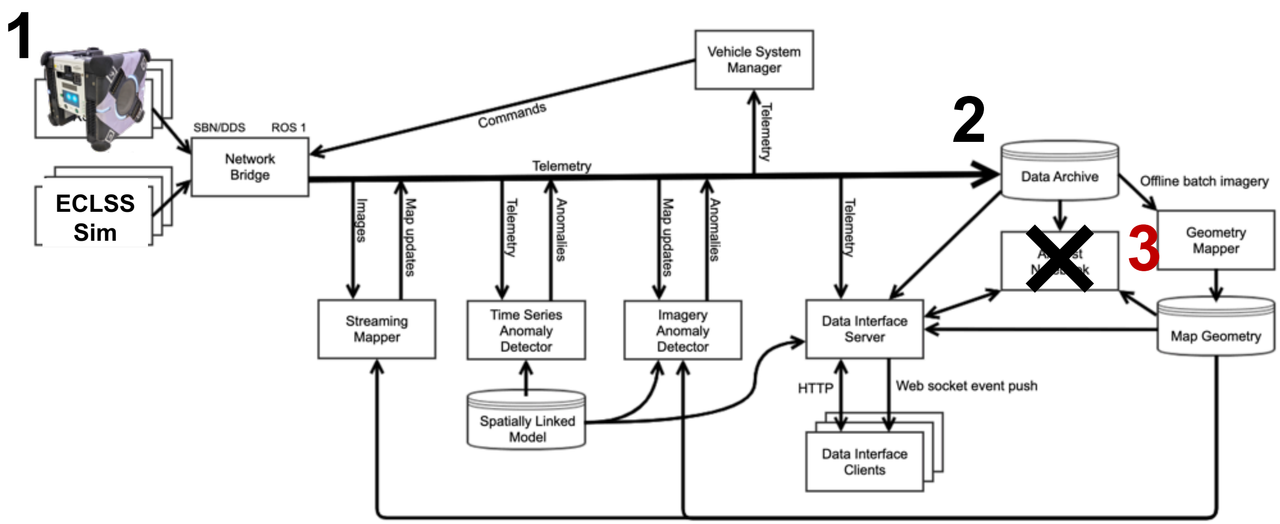
1. **Astrobee collects multi-sensor imagery and logs to onboard storage**

Mapping checkout activity



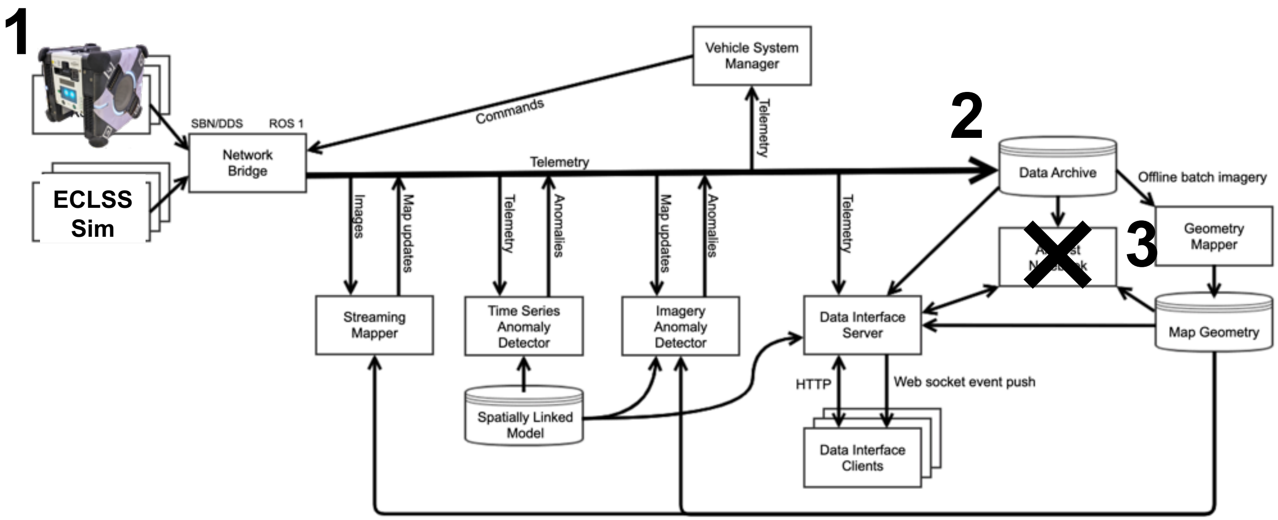
1. Astrobee collects multi-sensor imagery and logs to onboard storage
2. **After activity, imagery is batch downlinked to data archive**

Mapping checkout activity



1. Astrobee collects multi-sensor imagery and logs to onboard storage
2. After activity, imagery is batch downlinked to data archive
3. **A 3D geometry model with high-resolution surface texture is constructed**

Mapping checkout activity



1. Astrobee collects multi-sensor imagery and logs to onboard storage
2. After activity, imagery is batch downlinked to data archive
3. A 3D geometry model with high-resolution surface texture is constructed

ISAAC 3D Mapping Objectives

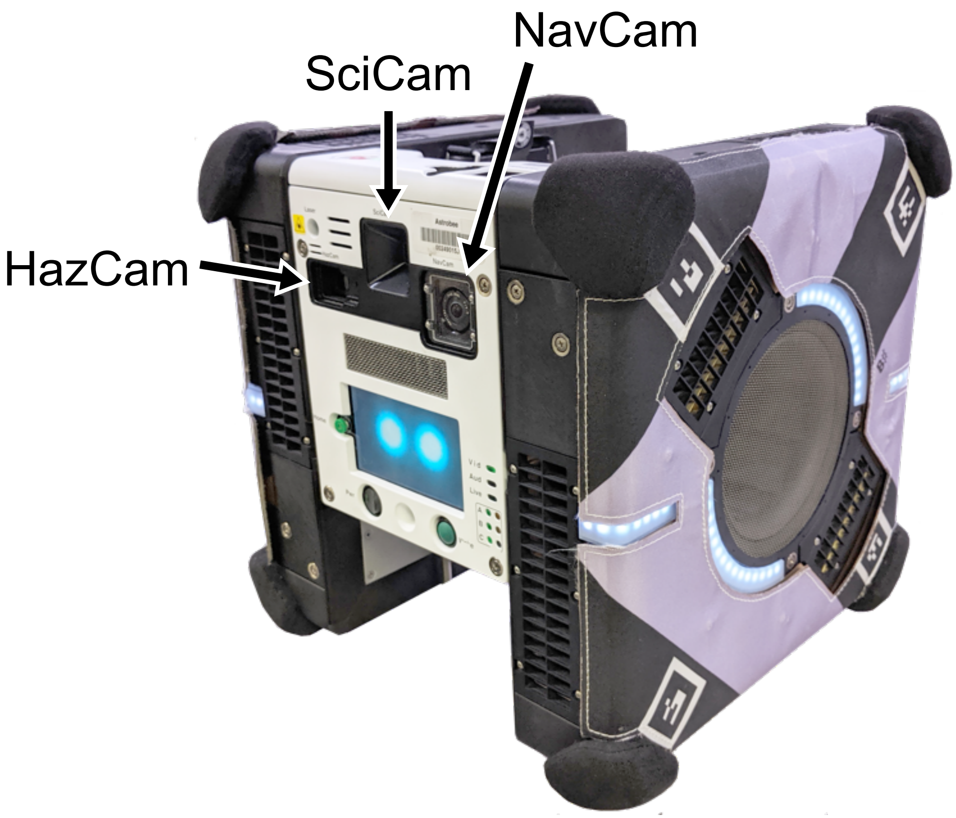
- **ISAAC is developing technology for using robotic data collection to build 3D maps of module interiors that support:**
  - Human usability: Astrobee's existing capability builds a "sparse map", which is a sparse 3D cloud of points that mark features used by its vision-based localization system, hard for a human to understand or use. In contrast, ISAAC technology will build a dense 3D mesh that we can texture imagery onto and visualize for operator situation awareness.
  - Detailed inspection: High-res visual imagery (~1 mm) enhances the usefulness of the 3D map for inspection tasks.
  - Multi-sensor analysis: Co-registering data from multiple sensor types enables new kinds of joint analysis. For example, when localizing a leak by listening for a hissing sound, detecting a co-located cold spot in thermal IR imagery could provide a useful cross-check.
  - Registered position: Objects seen in map have known position (< ~10 cm error) in a repeatable coordinate frame useful for robot commanding. For example, a mobile inspection robot could find an object of interest and communicate the location to a mobile manipulator robot for further interaction.

ISAAC 3D Mapping Features

- Generates a map layer for each sensor, given multiple sensors with different modalities
- Stitches together each map layer from many images with overlapping views (when feasible, improves alignment using feature matching between overlapping images)
- Registers all map layers to a common coordinate frame and textures them onto a common 3D mesh
- High-quality 3D mesh and surface texture are generated by batch processing after robotic data collection is complete
- Once a 3D mesh has been generated, during subsequent activities, new imagery can be textured onto the 3D mesh and displayed to an operator during robotic data collection in real time
- ISAAC mapping software currently runs on the ground, but analogous software could run onboard Gateway fixed computing to support autonomy objectives

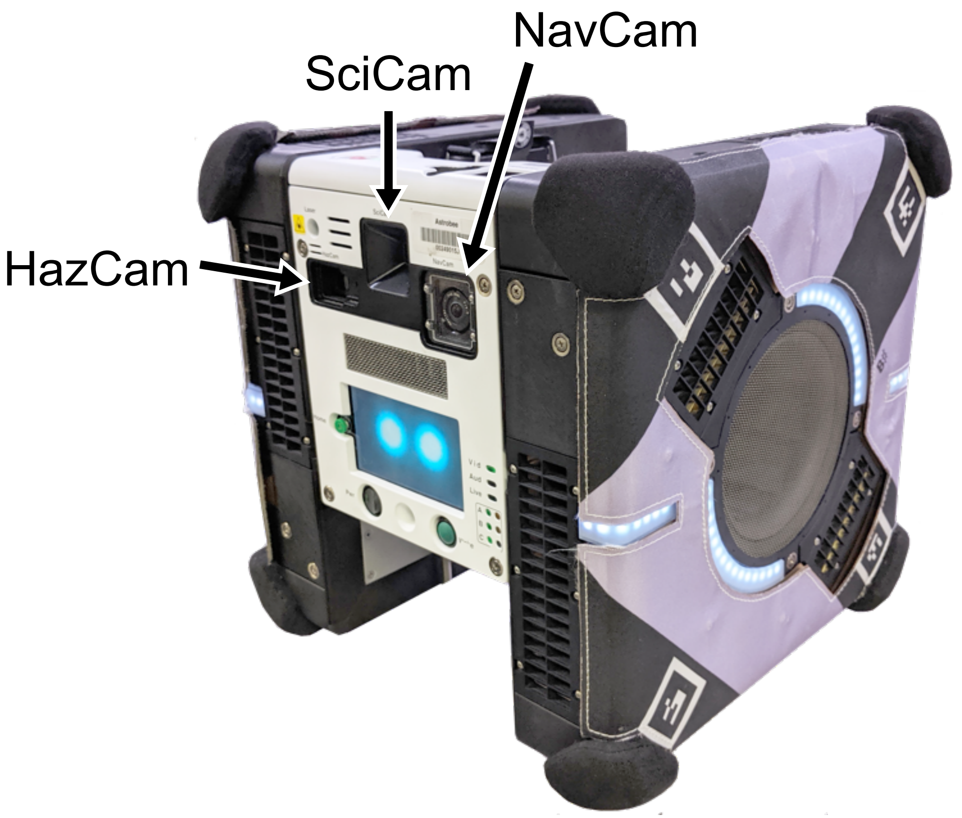
3D Mapping Astrobee Sensors

- **ISAAC maps currently incorporate data from the following sensor modalities:**
  - Depth imagery – HazCam LIDAR
  - RGB imagery – SciCam
- **Other sensors play an important supporting role:**
  - NavCam and IMU: These sensors are used by Astrobee's baseline vision-based localization system to estimate sensor poses that seed the multi-sensor map registration.
- **Candidate future sensor modalities to include in the mapping:**
  - WAP signal strength
  - RFID tag signal strength
  - Acoustic image
  - Thermal IR image
  - Gas concentration



3D Mapping Astrobee Hardware Challenges

- Astrobee sensors were not optimized for dense 3D mapping; future free flyers may want to prioritize this use case
- HazCam miniaturized LIDAR produces significant noise and distortion
- Manually controlling SciCam focus was initially problematic due to driver limitations (key for calibration)
- Sensors are not synced, which can cause motion misalignment artifacts
- For various reasons, custom calibration code was required to properly align images from HazCam, SciCam, and NavCam



ISAAC 3D Mapping Astrobee Limitations

- **What we can't get (with Astrobee as currently configured)**
  - Fine geometry: For example, handrails or wires below the reliable resolution of the HazCam may be missing or poorly represented
  - CAD-perfect geometry: For example, 3D geometry of flat panels may appear distorted when inspected closely
  - High-accuracy dimension measurements: Dimensions are subject to HazCam limitations and other sources of error
  - Perfect alignment: Registration between multiple images and multiple sensors will always introduce local artifacts.
  - Complete coverage: Astrobee can't view objects from all angles, e.g. when getting a view would require flying into a tight confined space

3D Mapping Demo Source Data

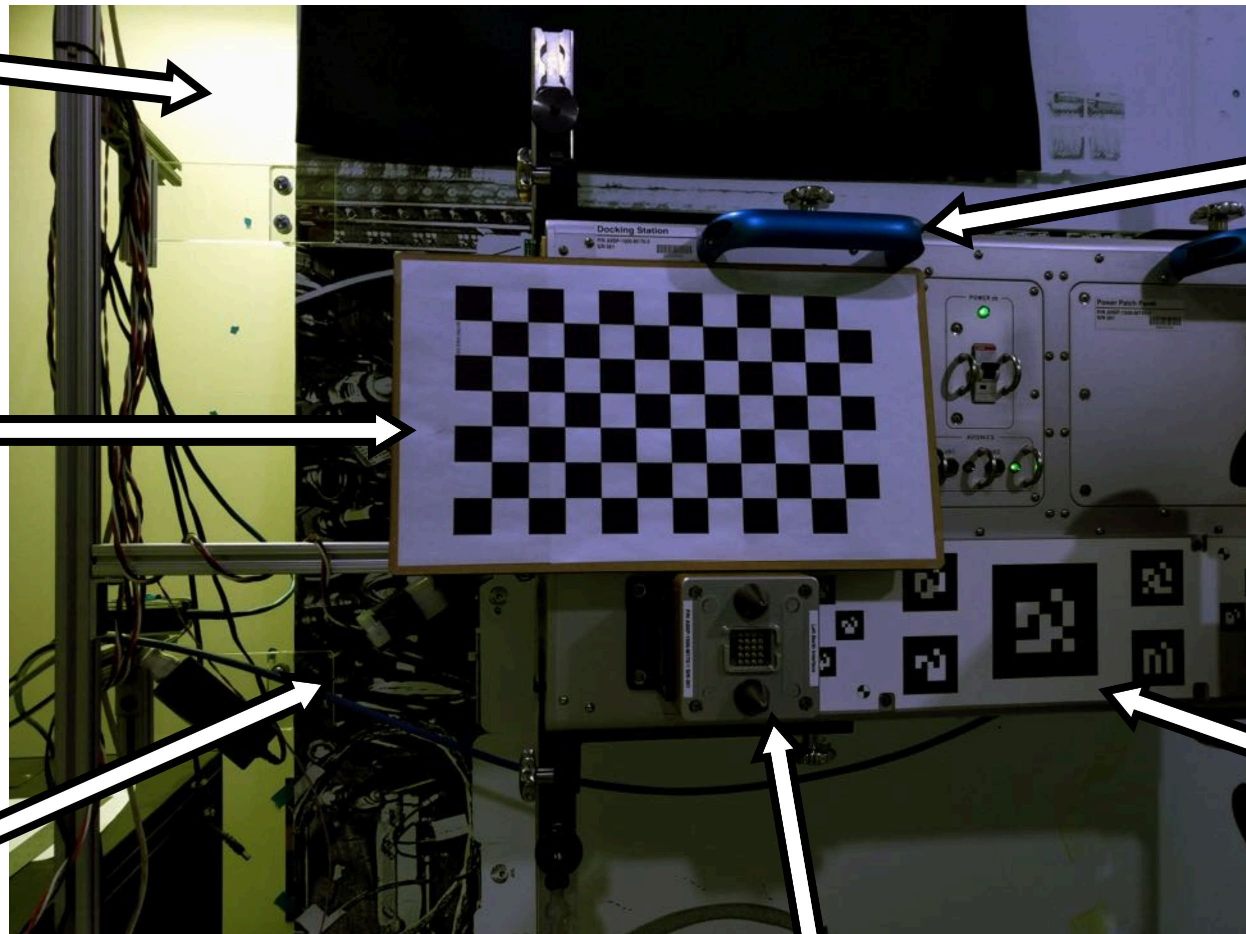
- **Source data**
  - Collected on the Ames granite table using the Astrobee ground unit "Wannabee"
  - Astrobee sensors included in this data set:
    - SciCam (source of texture) – Small set of 4 overlapping images used for this demo
    - HazCam (source of 3D geometry)
    - NavCam and IMU (used to seed registration)
- **We would prefer a better demo data set, but this is the best we had collected for mapping prototype development prior to Ames pandemic closure**
  - As Ames pandemic posture relaxes over the next few weeks, we will start evaluating when we can access the lab to collect an improved data set.

3D Mapping Demo Scene

Scene foreground is unfortunately somewhat underexposed because auto-gain adjusted for bright lighting on background wall

Checkerboard calibration target, paper print mounted on cardboard, resting on dock berth and tilted against dock chassis

"ISS module wall" printed photo of ISS interior mounted on posterboard backdrop



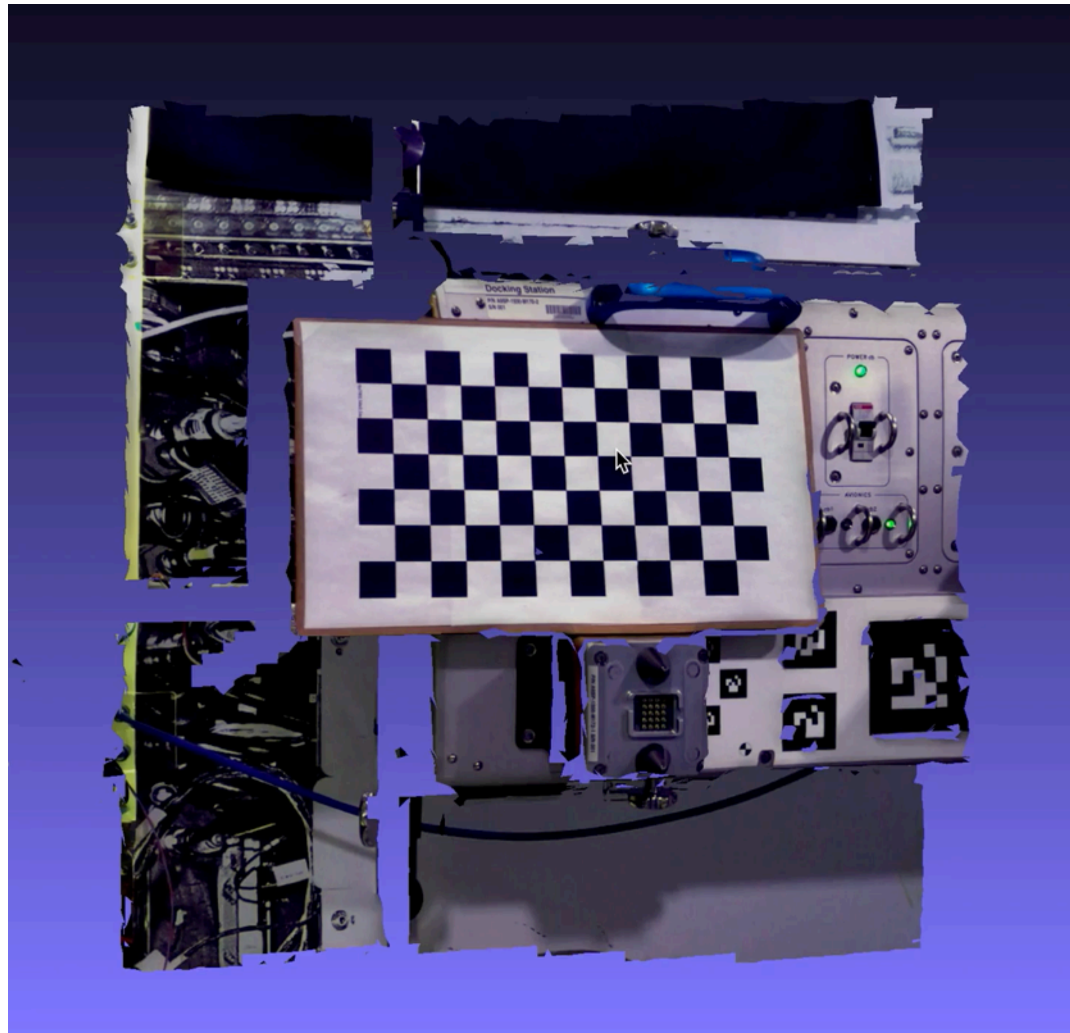
Blue handle projects out from dock chassis

Dock AR target decal is applied to dock chassis

Dock berth projects out from dock chassis

3D Mapping Demo Result

- **[PLAY VIDEO]**
- Gross geometry is accurate
- Effective surface texture resolution ~0.6 mm (enough to read 14-pt font labels)
- Alignment between HazCam geometry and SciCam texture is accurate
- No obvious stitching artifacts between multiple SciCam frames in this small example



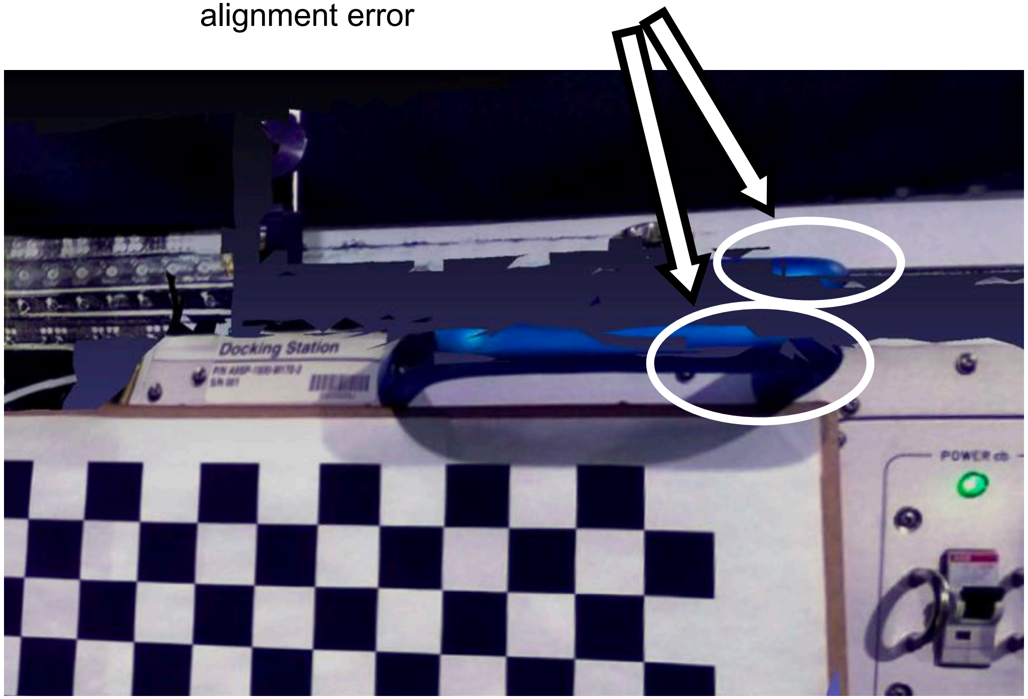
3D Mapping Demo Issues

Issue	Possible mitigations
Texture tearing and other alignment artifacts	Severity has already been reduced by improving image-to-image and sensor-to-sensor registration. Can't be eliminated completely.
Distorted geometry	Post-processing to selectively replace near-planar geometry with plane fit. Fundamental limitation of HazCam sensor, can't be eliminated completely.
Scattered 3D points	Filter out 3D mesh patches too small to texture accurately; in some cases these aren't real points anyway.
Holes in 3D geometry	Interpolate mesh across small holes.

3D Mapping Demo Issues

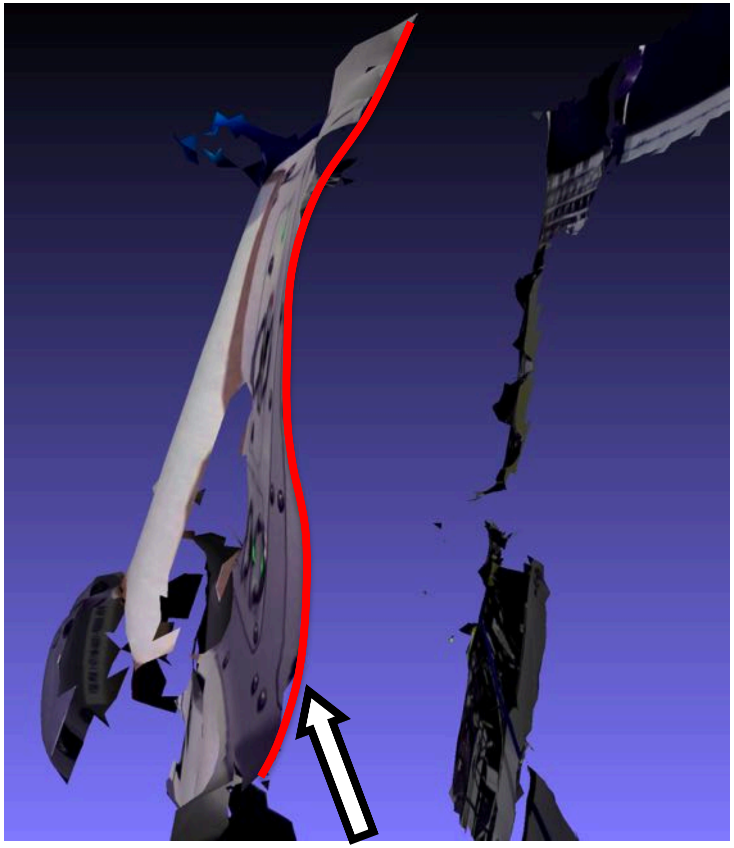
Issue	Possible mitigations
Texture tearing and other alignment artifacts	**Severity has already been reduced by improving image-to-image and sensor-to-sensor registration. Can't be eliminated completely.**
Distorted geometry	Post-processing to selectively replace near-planar geometry with plane fit. Fundamental limitation of HazCam sensor, can't be eliminated completely.
Scattered 3D points	Filter out 3D mesh patches too small to texture accurately; in some cases these aren't real points anyway.
Holes in 3D geometry	Interpolate mesh across small holes.

Texture tearing: some blue handle foreground texture incorrectly applied to backdrop geometry due to small alignment error



3D Mapping Demo Issues

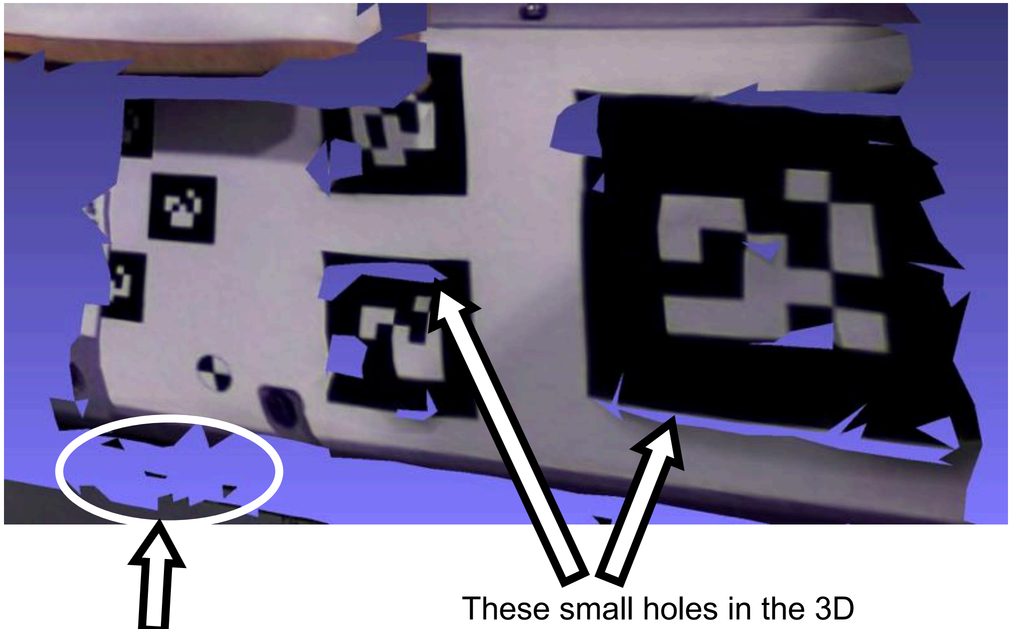
Issue	Possible mitigations
Texture tearing and other alignment artifacts	Severity has already been reduced by improving image-to-image and sensor-to-sensor registration. Can't be eliminated completely.
Distorted geometry	**Post-processing to selectively replace near-planar geometry with plane fit. Fundamental limitation of HazCam sensor, can't be eliminated completely.**
Scattered 3D points	Filter out 3D mesh patches too small to texture accurately; in some cases these aren't real points anyway.
Holes in 3D geometry	Interpolate mesh across small holes.



When the dock chassis panel is viewed in profile, distortion is obvious (red line indicating the edge of the flat panel should be straight). Note HazCam LIDAR tends to have more distortion when viewing shiny objects.

3D Mapping Demo Issues

Issue	Possible mitigations
Texture tearing and other alignment artifacts	Severity has already been reduced by improving image-to-image and sensor-to-sensor registration. Can't be eliminated completely.
Distorted geometry	Post-processing to selectively replace near-planar geometry with plane fit. Fundamental limitation of HazCam sensor, can't be eliminated completely.
Scattered 3D points	**Filter out 3D mesh patches too small to texture accurately; in some cases these aren't real points anyway.**
Holes in 3D geometry	**Interpolate mesh across small holes.**



Scattered 3D points can't be textured accurately and sometimes aren't real geometry. *Note: We manually removed some other points like these for clarity.

These small holes in the 3D mesh are probably due to the HazCam LIDAR sometimes getting insufficient return signal from a very dark object, like the black part of the target decal.

3D Mapping Demo Issues

Issue	Possible mitigations
Texture tearing and other alignment artifacts	Severity has already been reduced by improving image-to-image and sensor-to-sensor registration. Can't be eliminated completely.
Distorted geometry	Post-processing to selectively replace near-planar geometry with plane fit. Fundamental limitation of HazCam sensor, can't be eliminated completely.
Scattered 3D points	Filter out 3D mesh patches too small to texture accurately; in some cases these aren't real points anyway.
Holes in 3D geometry	Interpolate mesh across small holes.

- **None of these issues are show-stoppers preventing the map output from providing the key functionality we noted before:**
  - Detailed inspection
  - Registered position
  - Multi-sensor analysis
- **However, time permitting, we may be able to apply some of the mitigations and get:**
  - Better effective coverage
  - Fewer distracting or confusing reconstruction artifacts
  - Cosmetically nicer models
  - Reduced error rate for downstream automated analysis of map data

3D Mapping KPPs

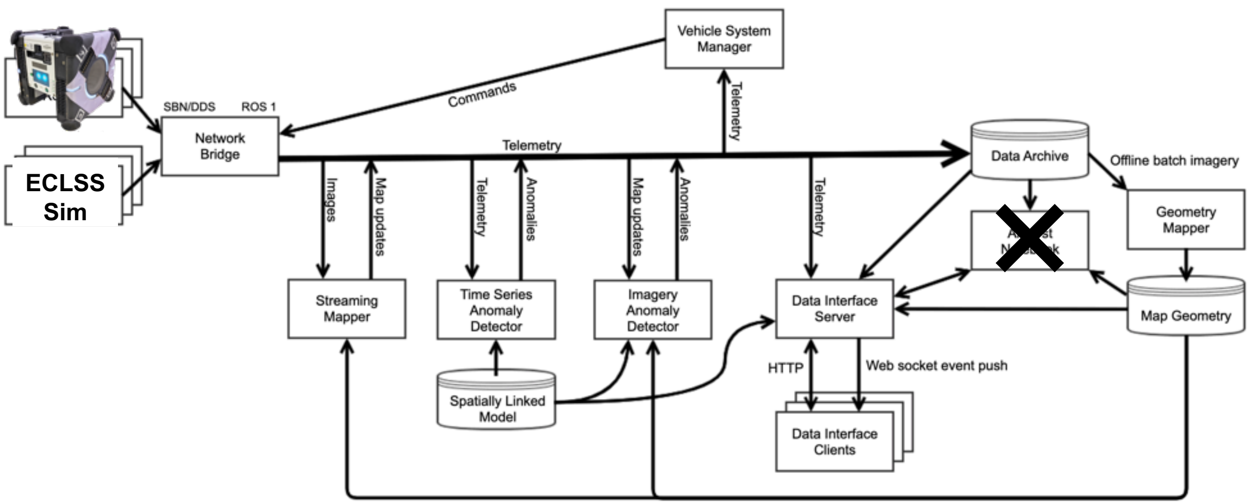
#	Key Performance Parameter (KPP)	Relevance	State of the Art	Threshold Value	Goal Value	Current value
1	Spatial resolution	Resolving smaller features provides higher-fidelity state assessment	Sparse	1 cm	0.2 cm	0.06 cm*
2	# sensing modalities	More co-registered sensors (e.g., RFID, thermal) support more applications	1	4	7	2

1. **Spatial resolution**
   - Visual texture in 3D mapping demo result has effective spatial resolution of ~0.06 cm (judging from ability to read 14-pt font labels), already exceeding the goal value of 0.2 cm

     * Under simplified lab conditions; may not carry over perfectly to on-orbit testing
   - The test data set was acquired at ~50 cm standoff distance from the foreground objects. This is feasible for Astrobee to replicate on-orbit. However, for actual on-orbit testing we may prefer to use a greater standoff distance of ~100 cm in order to more quickly map a designated area.
   - We are presently using 4x down-sampled SciCam images due to poor focus quality in the test data set, but we may be able to improve that with focus tuning in the SciCam driver
2. **Sensing modalities**
   - So far demonstrated 2 modalities, not yet reaching threshold value of 4
   - Modalities used in this hardware demo: Depth image, RGB image
   - Note that we already have preliminary software-sim mapping capabilities with three more modalities: WAP signal strength, thermal IR image, acoustic image
     - Further development and restored lab access will be required to demonstrate these in hardware

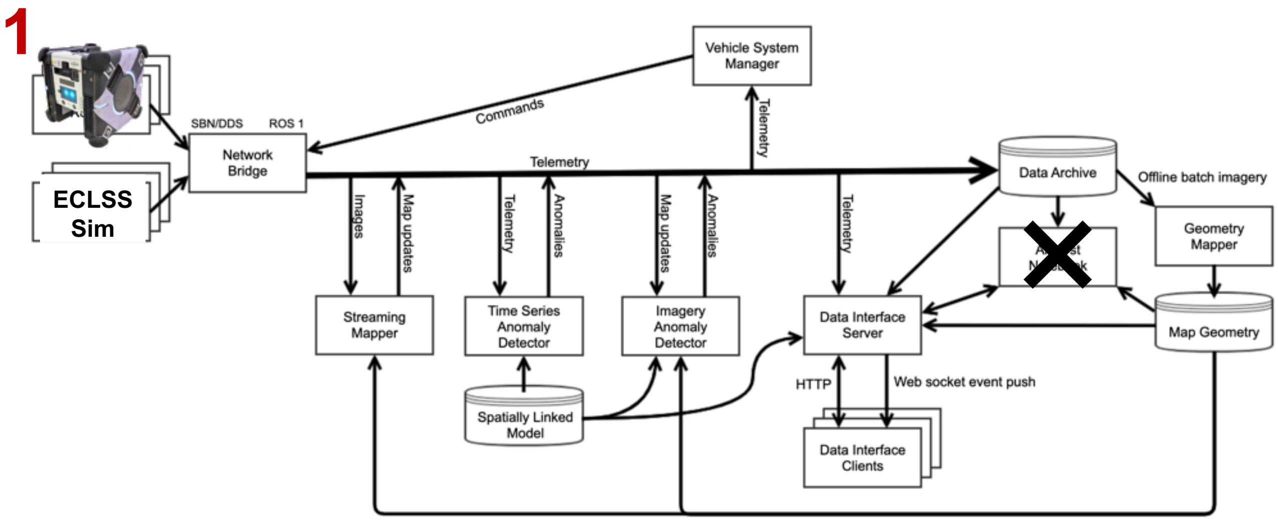
SURVEY ACTIVITY

Survey activity

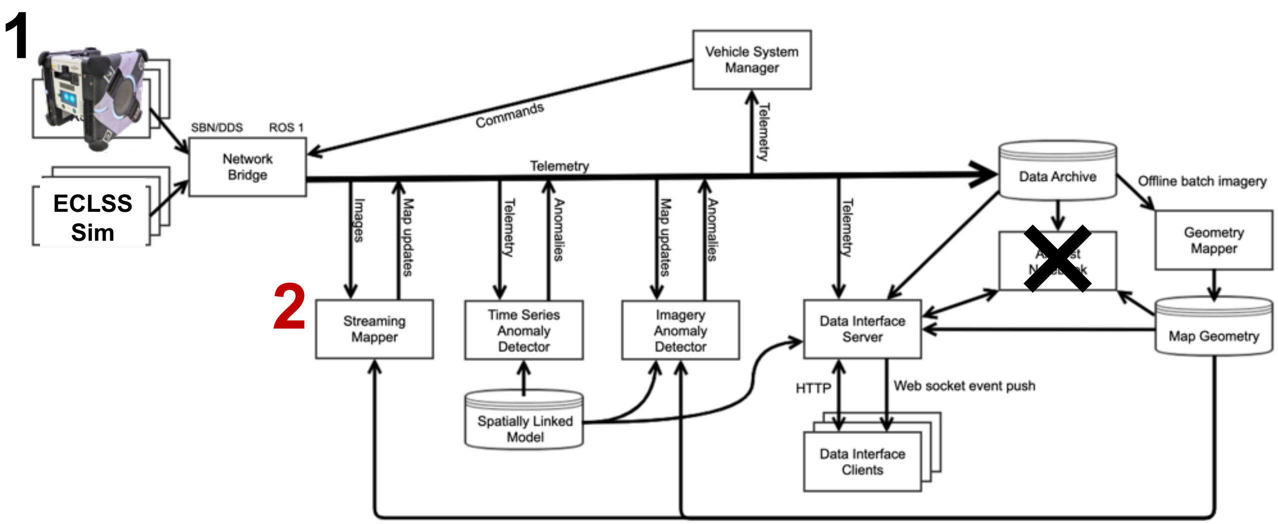


Survey activity

1. **Astrobee executes a survey plan that covers the full survey area**

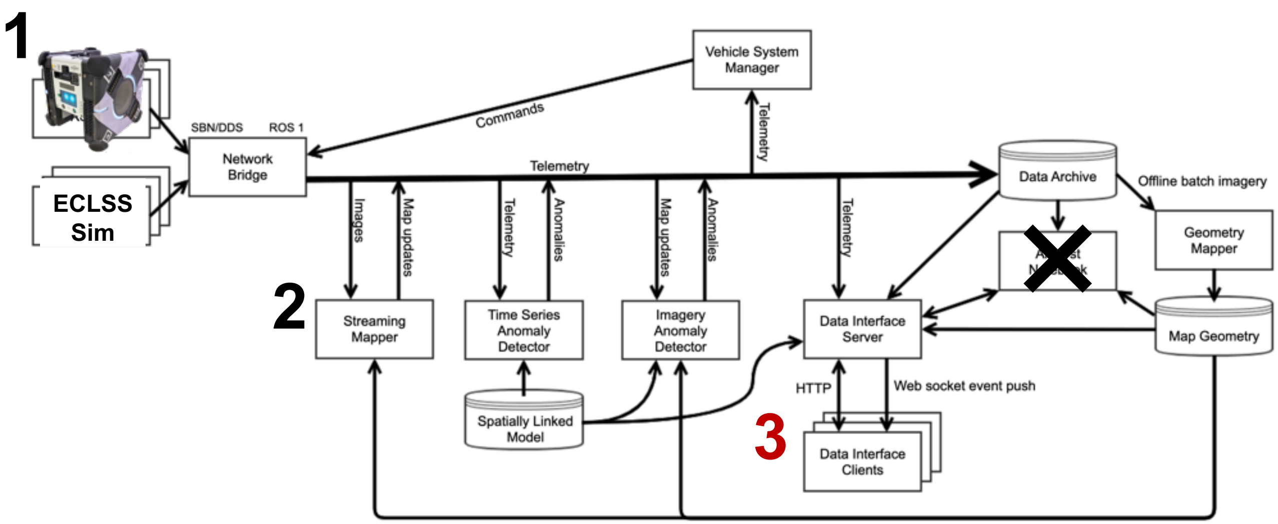


Survey activity



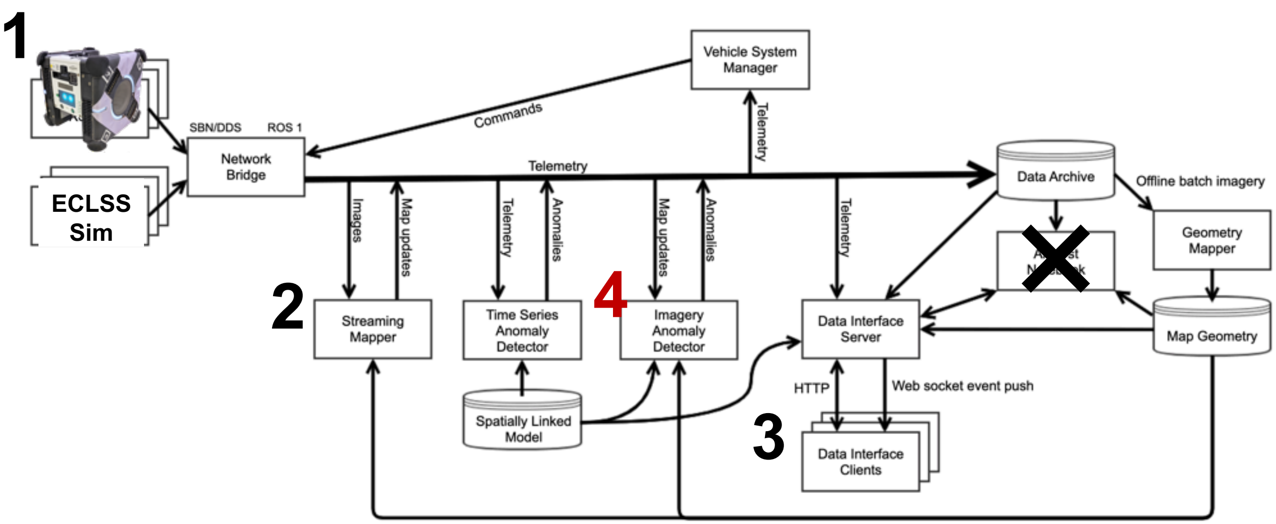
1. Astrobee executes a survey plan that covers the full survey area
2. **During execution, the streaming mapper receives updates from multiple Astrobee sensors and publishes incremental updates to the co-registered map layers**

Survey activity



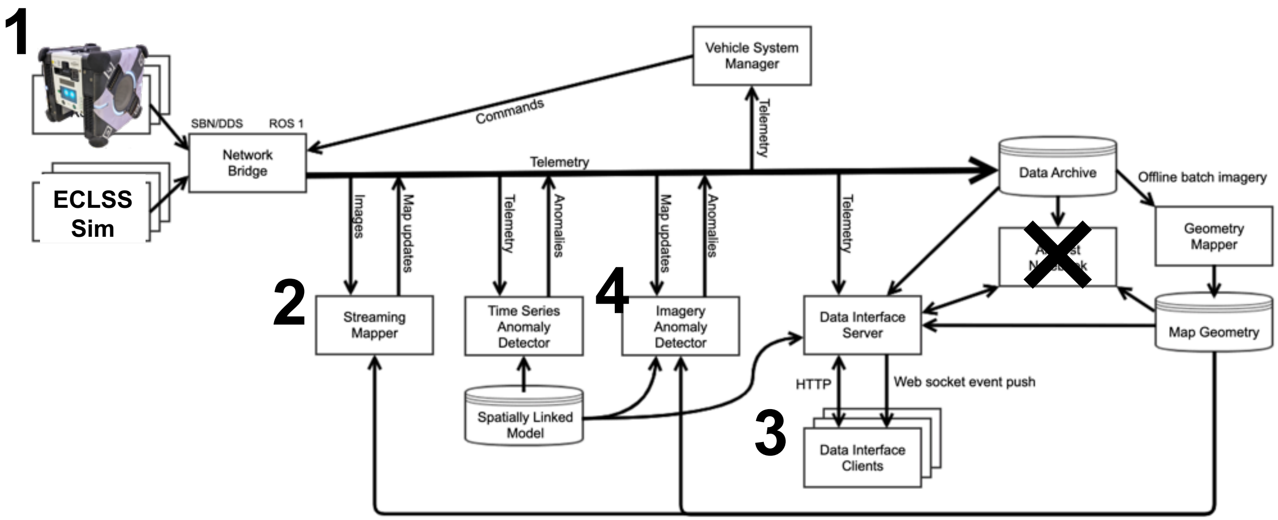
1. Astrobee executes a survey plan that covers the full survey area
2. During execution, the streaming mapper receives updates from multiple Astrobee sensors and publishes incremental updates to the co-registered map layers
3. **The map updates are rendered live in the data interface for operator situation awareness**

Survey activity



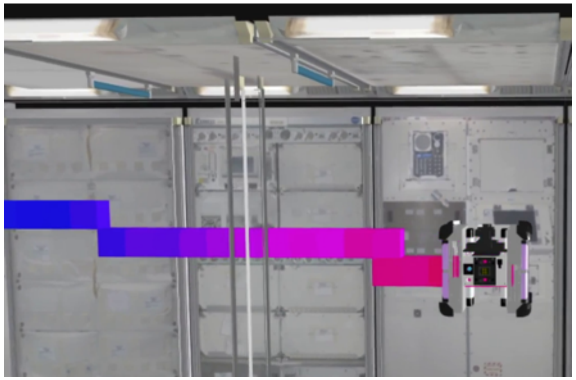
1. Astrobee executes a survey plan that covers the full survey area
2. During execution, the streaming mapper receives updates from multiple Astrobee sensors and publishes incremental updates to the co-registered map layers
3. The map updates are rendered live in the data interface for operator situation awareness
4. **[Future work] Changes and anomalies are detected during survey**

Survey activity



1. Astrobee executes a survey plan that covers the full survey area
2. During execution, the streaming mapper receives updates from multiple Astrobee sensors and publishes incremental updates to the co-registered map layers
3. The map updates are rendered live in the data interface for operator situation awareness
4. [Future work] Changes and anomalies are detected during survey

Survey demo video detail views



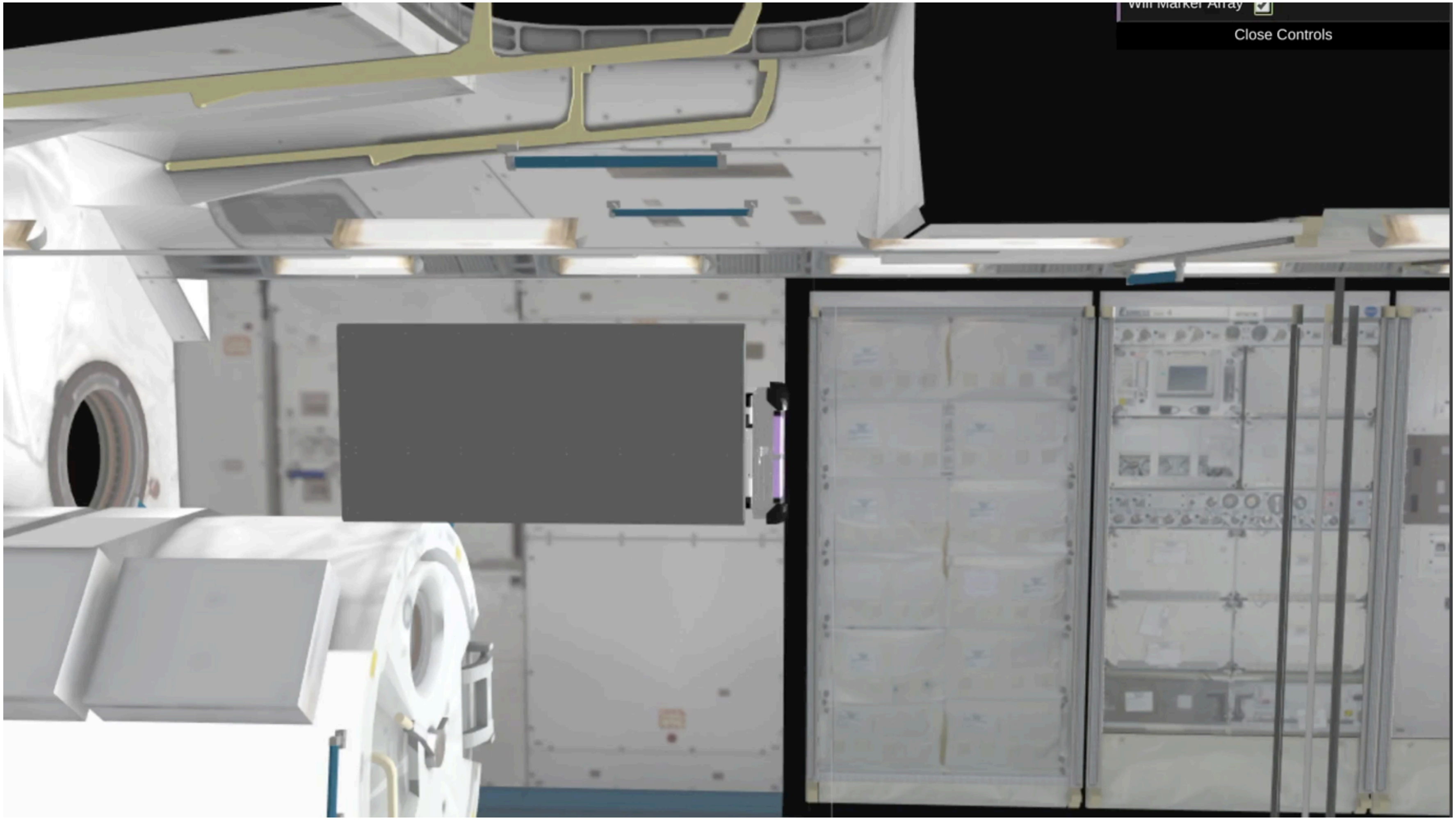
Colored trail is WAP signal strength map (because samples are collected in situ, this is a “volumetric” map layer, as opposed to a surface texture layer)

Prior visual texture layer (displayed with reduced contrast)

Red flash indicates extent of incoming map update

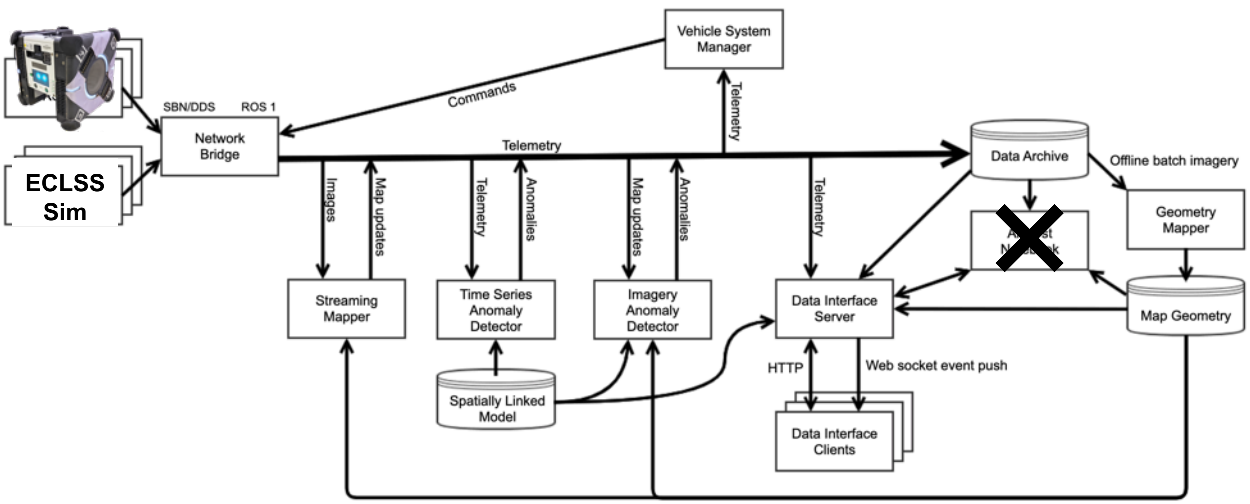
New SciCam imagery replaces prior imagery

Survey demo video



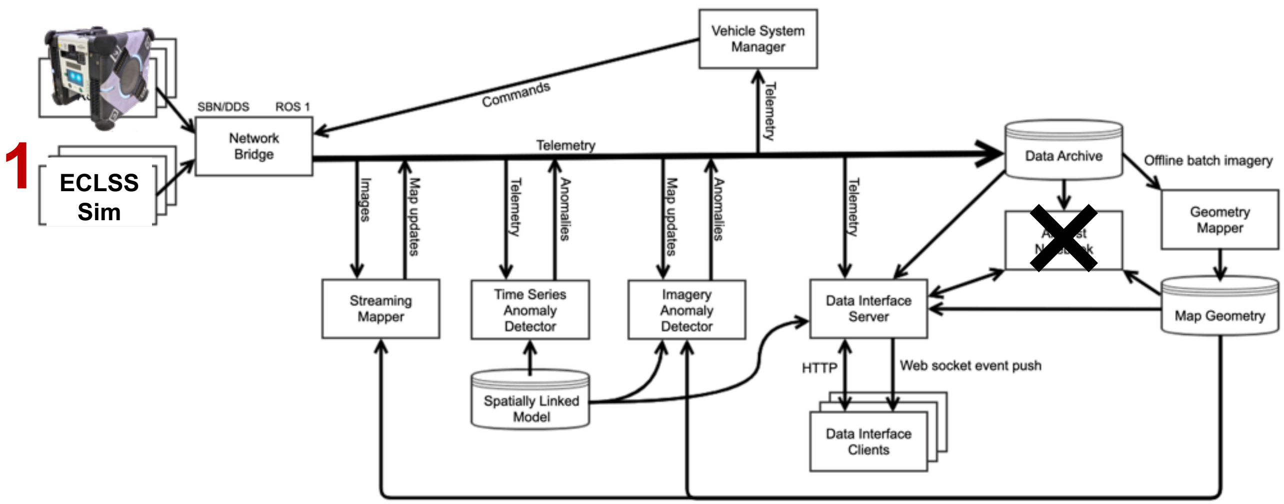
FAULT MANAGEMENT ACTIVITY

Fault management activity

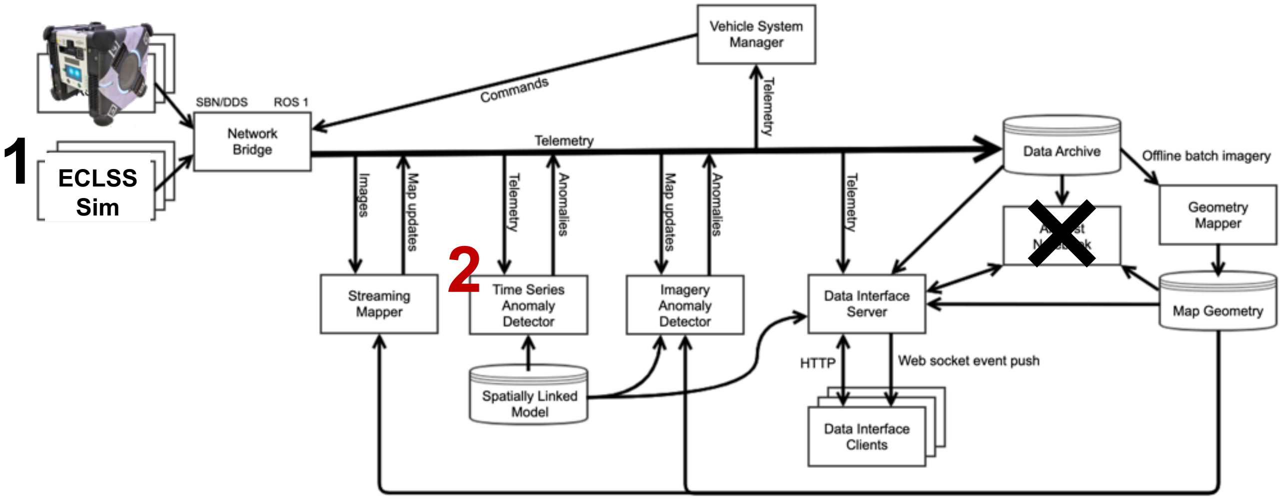


Fault management activity

1. A JEM high-CO_2 anomaly is injected into the ECLSS sim time series telemetry

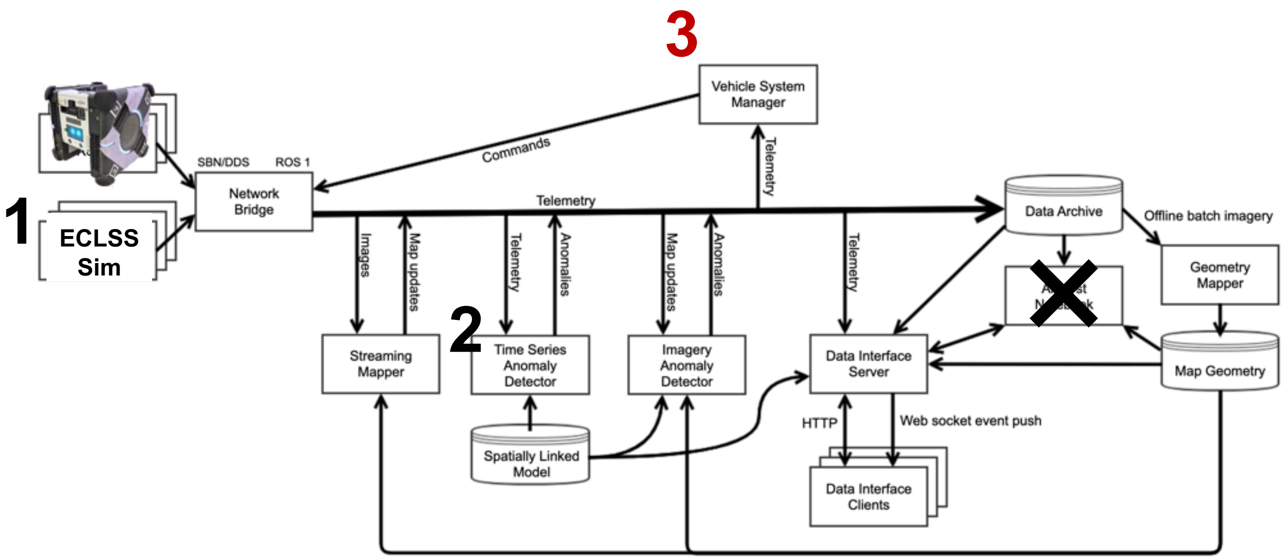


Fault management activity



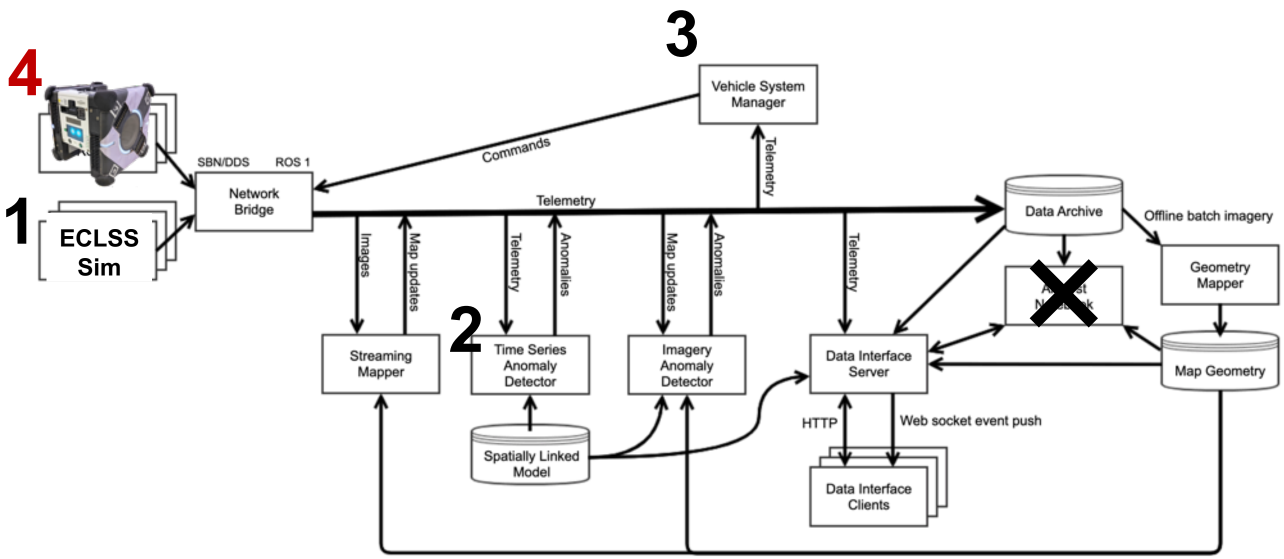
1. A JEM high-CO_2 anomaly is injected into the ECLSS sim time series telemetry
2. **The anomaly is detected**

Fault management activity



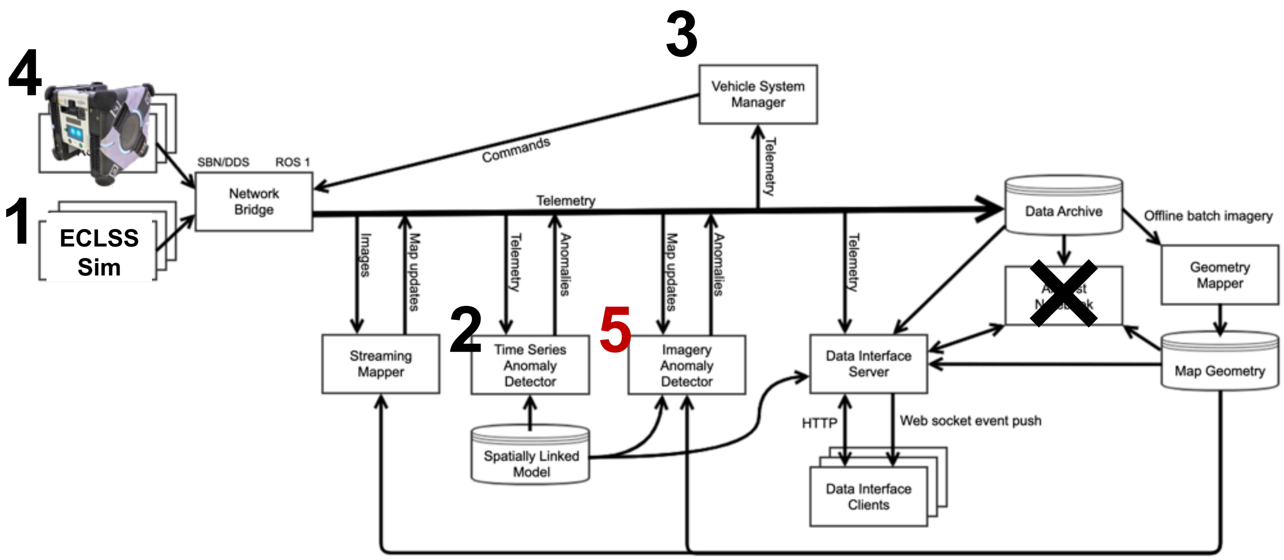
1. A JEM high-CO_2 anomaly is injected into the ECLSS sim time series telemetry
2. The anomaly is detected
3. **The Vehicle System Manager (VSM) determines that a possible cause for the anomaly is blockage of an ECLSS return vent in the JEM, and commands Astrobee to inspect the vent**

Fault management activity



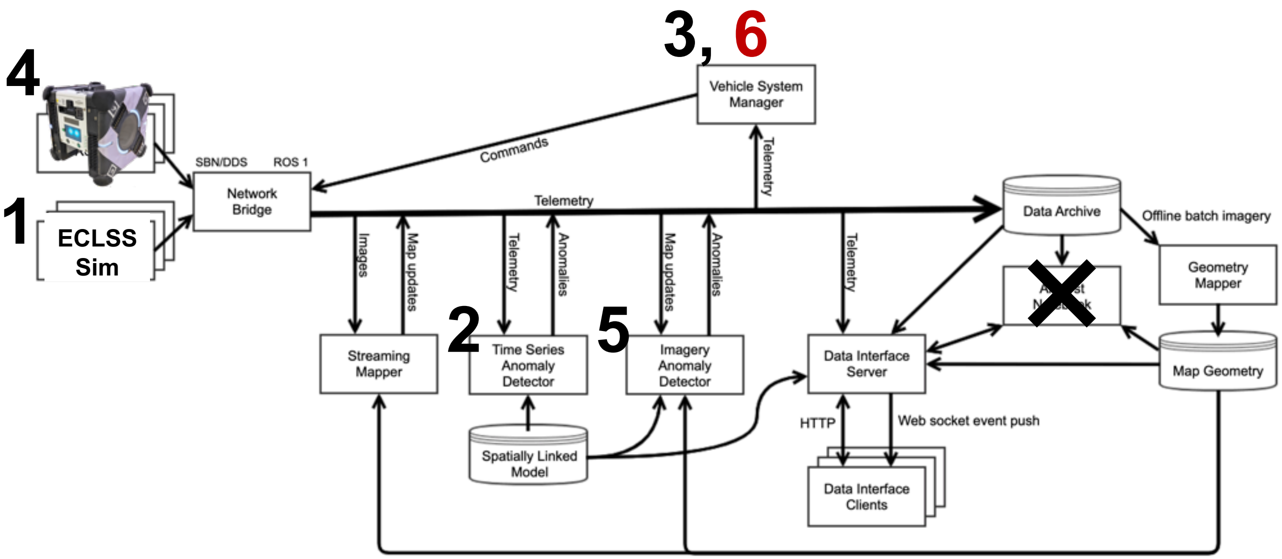
1. A JEM high-CO_2 anomaly is injected into the ECLSS sim time series telemetry
2. The anomaly is detected
3. The Vehicle System Manager (VSM) determines that a possible cause for the anomaly is blockage of an ECLSS return vent in the JEM, and commands Astrobee to inspect the vent
4. **Astrobee performs the targeted inspection**

Fault management activity



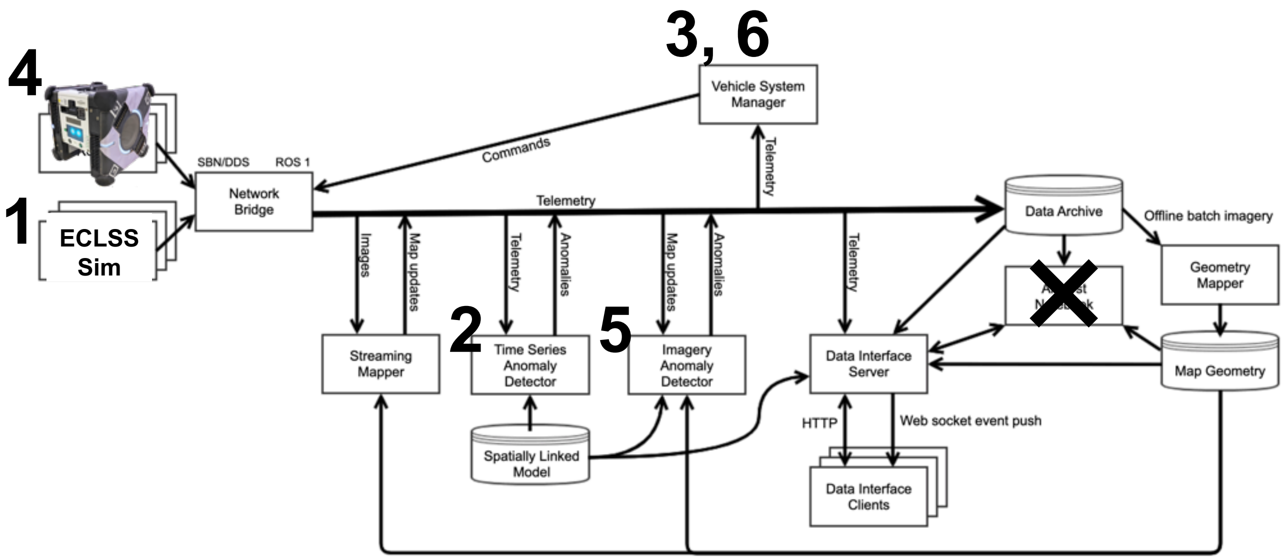
1. A JEM high-CO_2 anomaly is injected into the ECLSS sim time series telemetry
2. The anomaly is detected
3. The Vehicle System Manager (VSM) determines that a possible cause for the anomaly is blockage of an ECLSS return vent in the JEM, and commands Astrobee to inspect the vent
4. Astrobee performs the targeted inspection
5. **A vent blockage is detected**

Fault management activity



1. A JEM high-CO_2 anomaly is injected into the ECLSS sim time series telemetry
2. The anomaly is detected
3. The Vehicle System Manager (VSM) determines that a possible cause for the anomaly is blockage of an ECLSS return vent in the JEM, and commands Astrobee to inspect the vent
4. Astrobee performs the targeted inspection
5. A vent blockage is detected
6. **The VSM sends a dummy command to Robonaut to clear the blockage (but Robonaut is not actually involved in the demo)**

Fault management activity

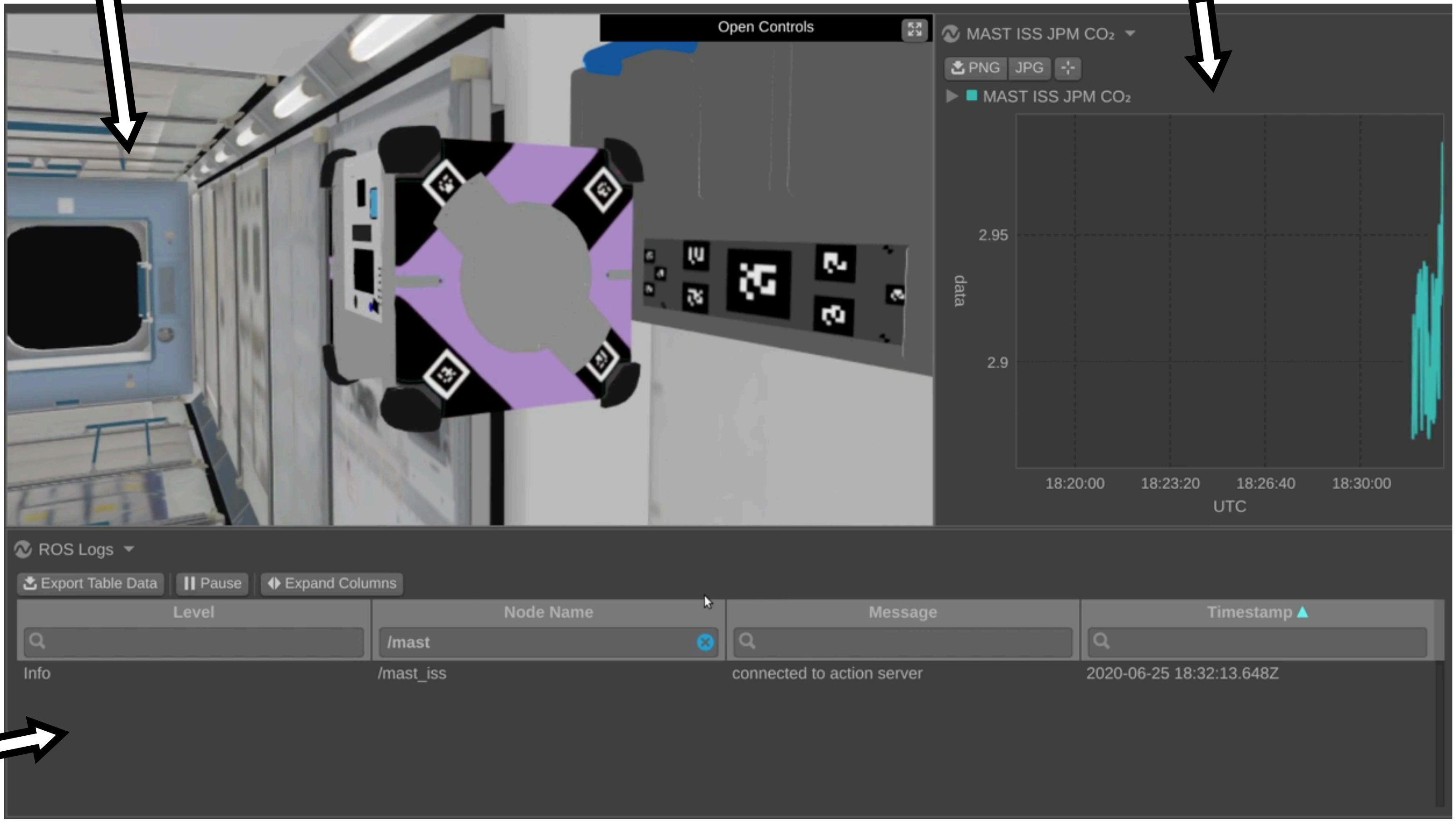


1. A JEM high-CO_2 anomaly is injected into the ECLSS sim time series telemetry
2. The anomaly is detected
3. The Vehicle System Manager (VSM) determines that a possible cause for the anomaly is blockage of an ECLSS return vent in the JEM, and commands Astrobee to inspect the vent
4. Astrobee performs the targeted inspection
5. A vent blockage is detected
6. The VSM sends a dummy command to Robonaut to clear the blockage (but Robonaut is not actually involved in the demo)

Fault management demo video

3D view. Red flash shows extent of incoming imagery update.

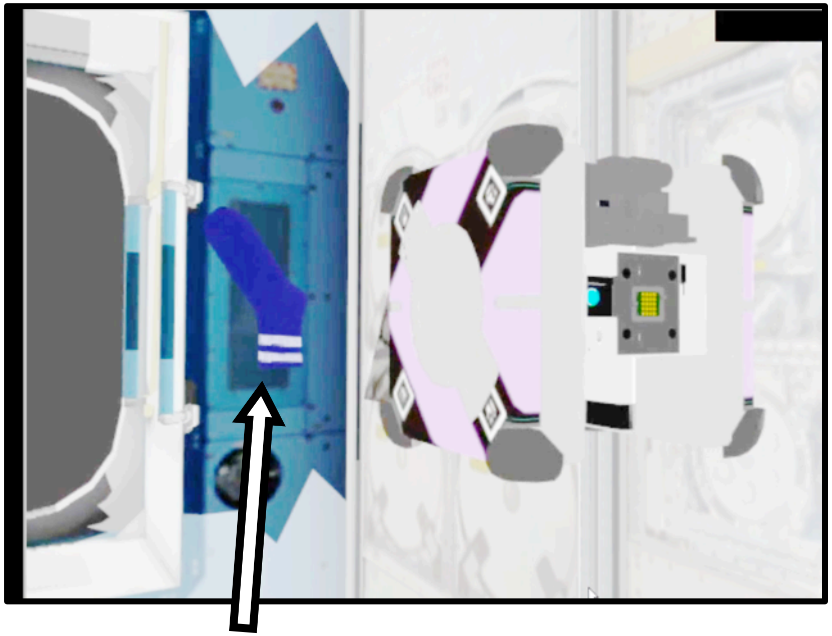
Time series view. Shows CO_2 spike that triggers Astrobee targeted inspection



Event log

Vent blockage detail view

- To enable fault injection, we can spawn virtual objects in the simulator visual environment.
- We extracted this detail image and enhanced the contrast because the map imagery happened to come out under-exposed in the video.
- Note that the sock is present in the simulated environment throughout the run. However, in the live updating map as viewed by the ground operator, it does not appear until Astrobee images it with the SciCam.



Upside-down astronaut sock (a typical item that might cause an ISS vent blockage, no kidding)

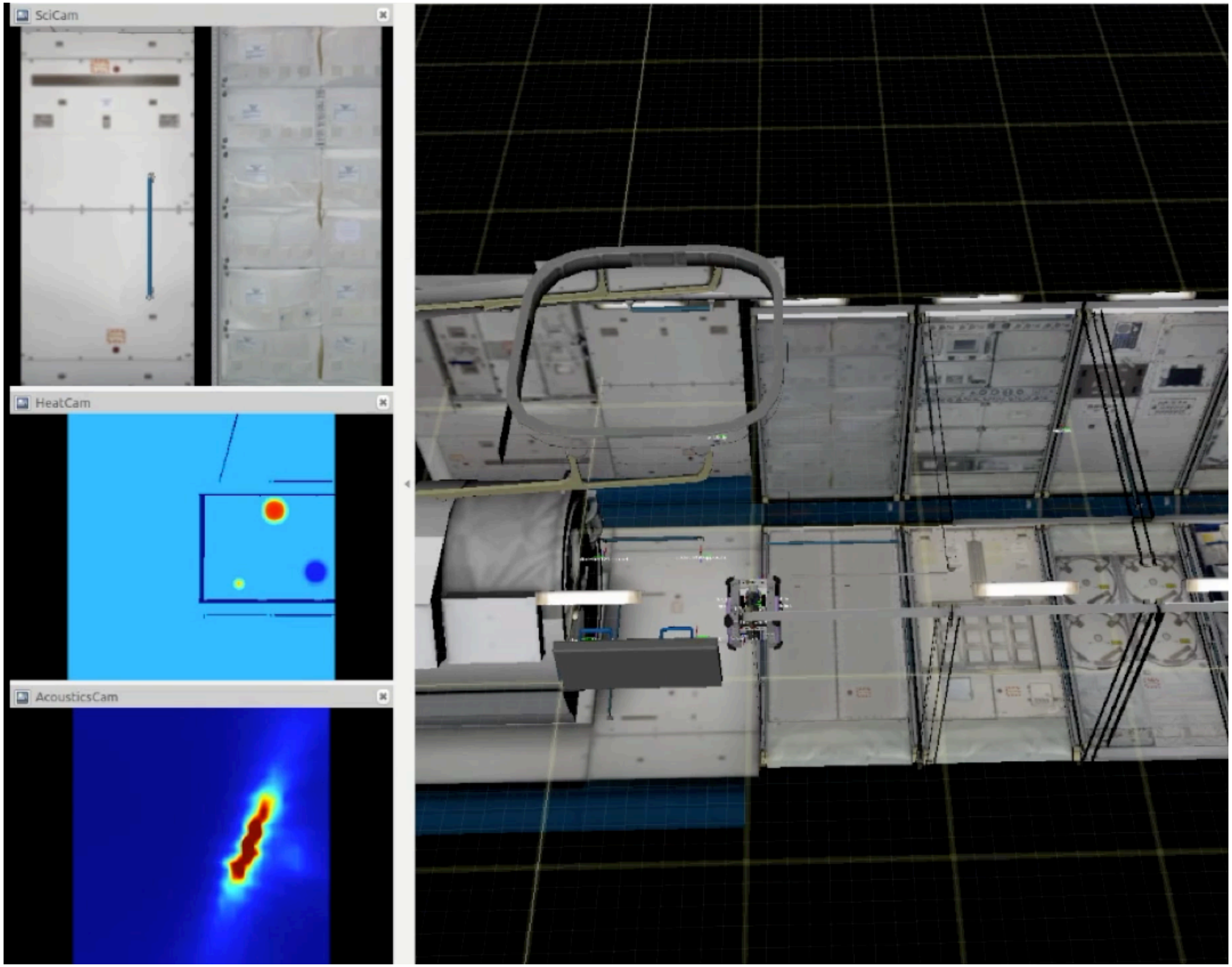
OTHER PROGRESS

[PLAY VIDEO]

SciCam

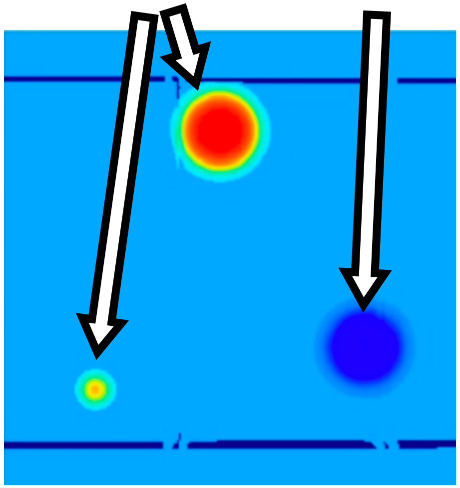
HeatCam

SoundCam
(lags other sensors)



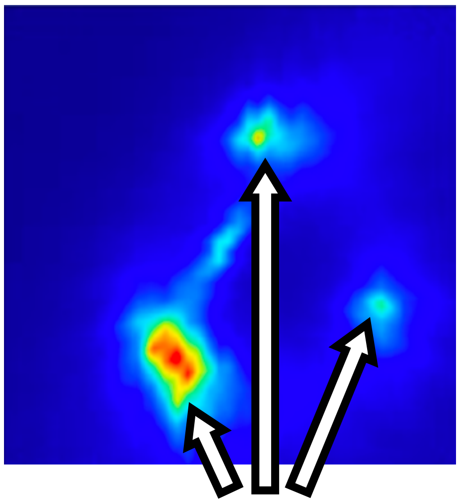
Software simulation infrastructure: New sensors

Hot spots Cold spot



HeatCam

- Configure heat sources: 3D location, size, Δradiance (negative for a cold spot)
- Radiance level is effectively textured onto the simulated module interior geometry, based on proximity to sources
- Simulated HeatCam renders image as if it was viewing the geometry with this radiance texture



Ultrasound sources

SoundCam

- Configure ultrasound sources: 3D location, intensity, ultrasound sample
- Simulate signals (and noise) picked up by individual microphones in an array
- Each pixel of the SoundCam virtual camera image represents estimated ultrasound intensity in that direction, reconstructed from mic array signals using a direction-of-arrival algorithm
  - Produces some realistic reconstruction artifacts. especially when the mic array is not sufficient to resolve the sources

Survey planner

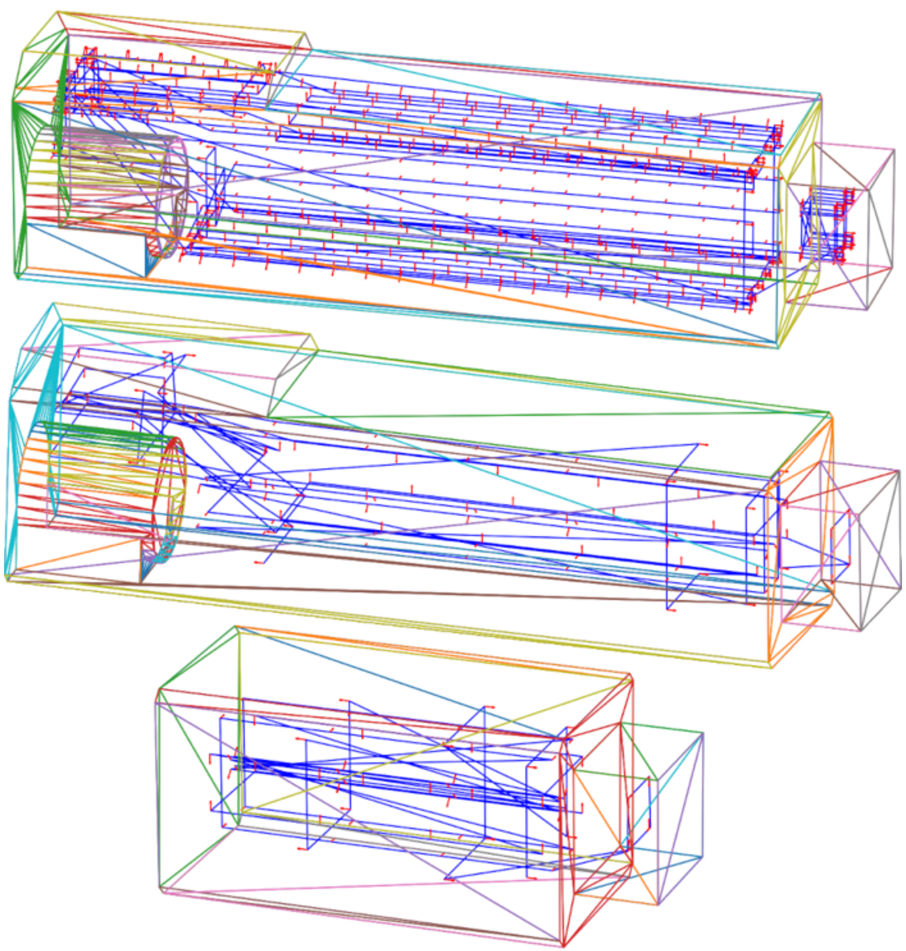


Fig. 2. (Top) JPM high resolution survey, (Middle) JPM low resolution trajectory, (Bottom) Columbus low resolution trajectory.

- **Given coarse prior geometry of module interior, generates an Astrobee survey plan**
- **Survey plan ensures:**
  - High-quality imagery covering all accessible areas
  - Efficient raster pattern trajectory
  - Astrobee pauses motion to capture SciCam high-res still images, mitigating motion blur
  - SciCam viewing direction optimized to be normal to planar surfaces, minimizing distortion
  - Sufficient image overlap for high-quality registration
- **Desired output map resolution can be adjusted depending on how much survey time is available**

Conclusions

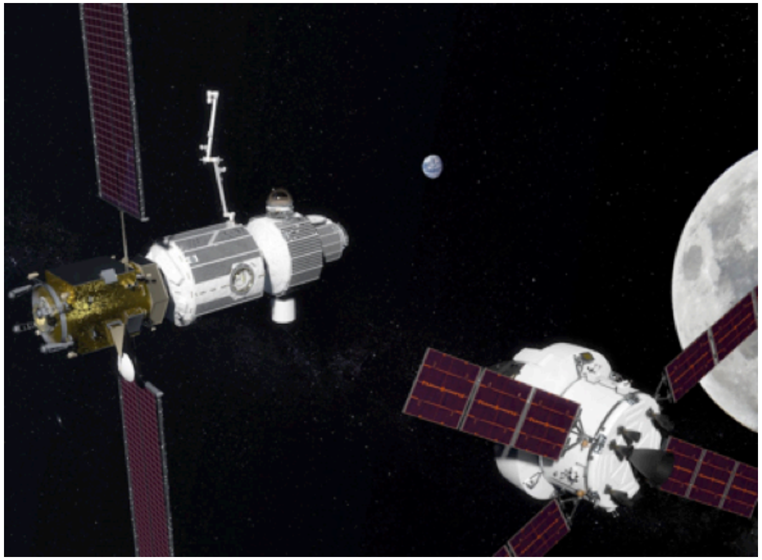
- **Satisfied the milestone from Gateway-ISAAC MoU to "Demonstrate spatial and logical data registration between robotics and spacecraft"**
- **Demonstrated many new ISAAC components, including:**
  - Spatially linked model
  - Multi-sensor mapping
  - Integrated data interface
- **Started to advance ISAAC KPPs related to mapping**
- **Areas for phase 1 forward work:**
  - Improve maturity toward high-fidelity demo on ISS
  - Demonstrate mapping with more sensor modalities
  - Expand initial anomaly detection implementation into flexible framework with multiple detection algorithms for different tasks
  - Begin open source software release process, where feasible

QUESTIONS?

BACKUP SLIDES

ISAAC overview

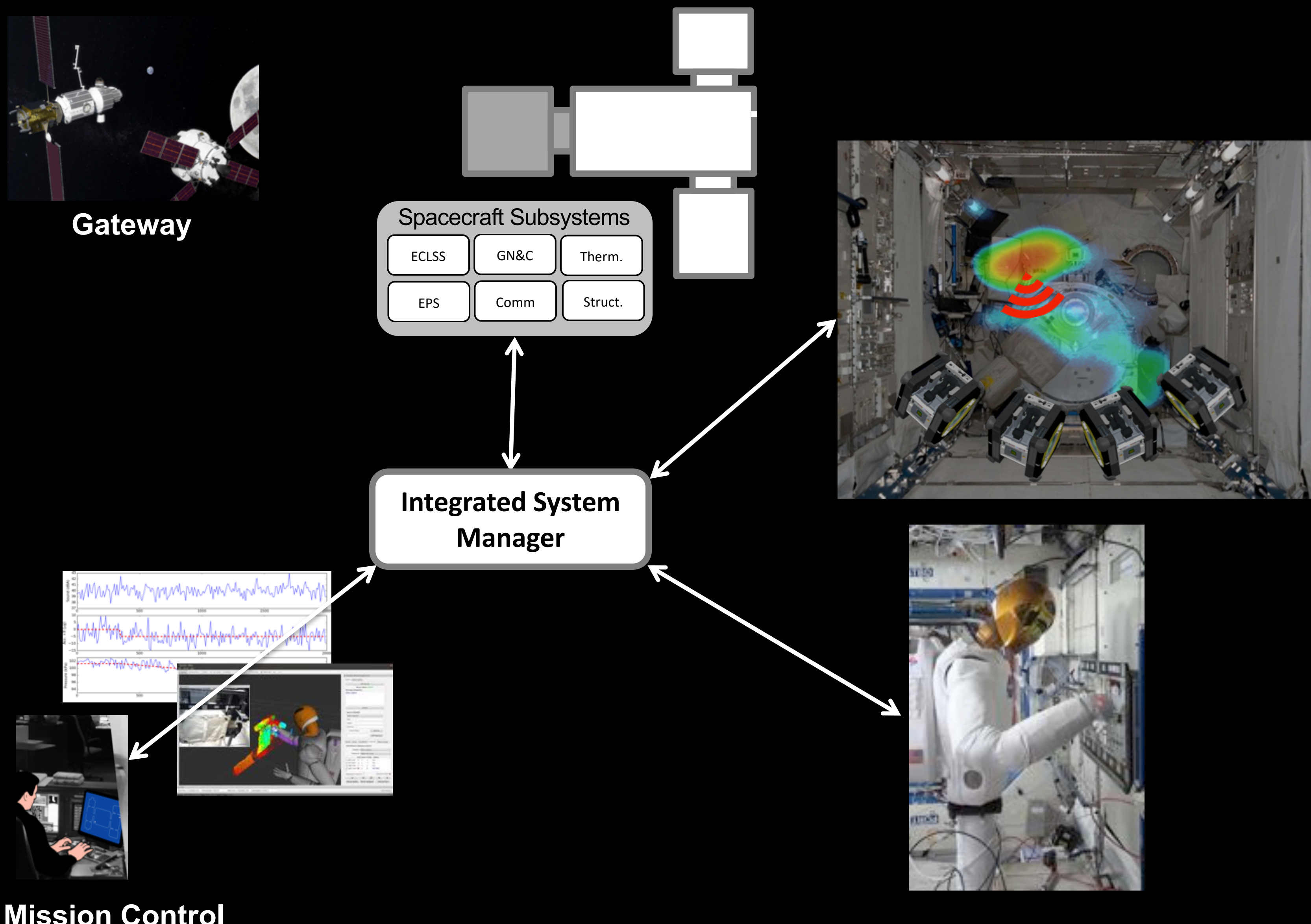
- Research project, 2020-2022, to develop technology for **autonomous caretaking** of spacecraft primarily during **uncrewed** mission phases
- Led by NASA Ames Research Center with collaboration from Johnson Space Center
- Integrate **autonomous intra-vehicular robots** (IVR) with **spacecraft infrastructure** (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**
- Do not:
    - Develop the IVR needed for Gateway
    - Develop Gateway flight software
    - (These tasks are vital but not part of ISAAC.)



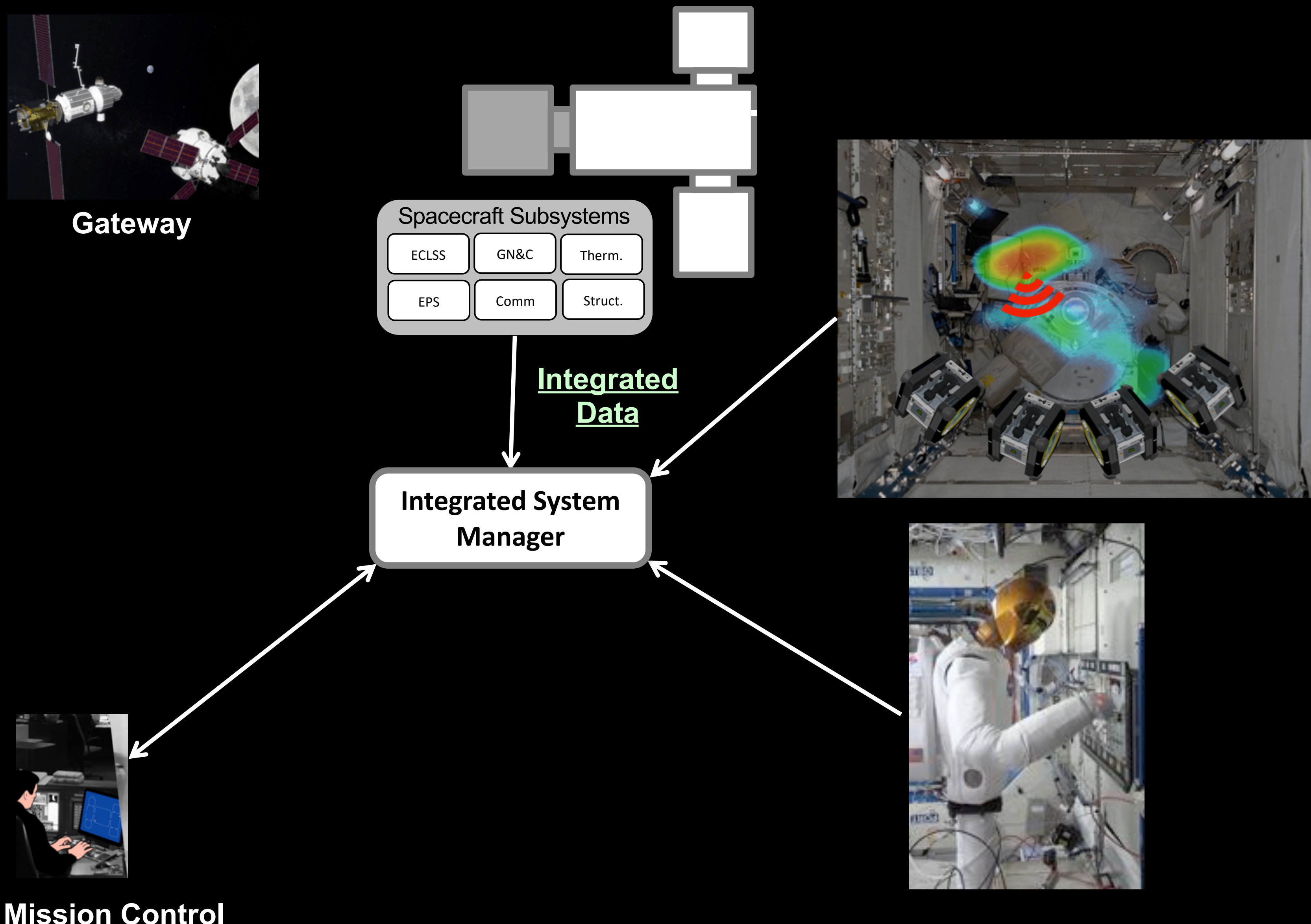
Goals for eventual mission impact

- **Reduce risk** through improved fault recovery during uncrewed phases
- **Reduce cost** by enabling new design options (e.g. one mobile sensor vs. many fixed sensors)
- **Free up crew time** spent on maintenance and logistics
- **Enhance utilization** during uncrewed phases (e.g. enabling sample transfer for experiments that need it)
- (These goals are largely inspired by the IVR WG business case; software for integrating IVR with the vehicle is essential for effective use of IVR.)

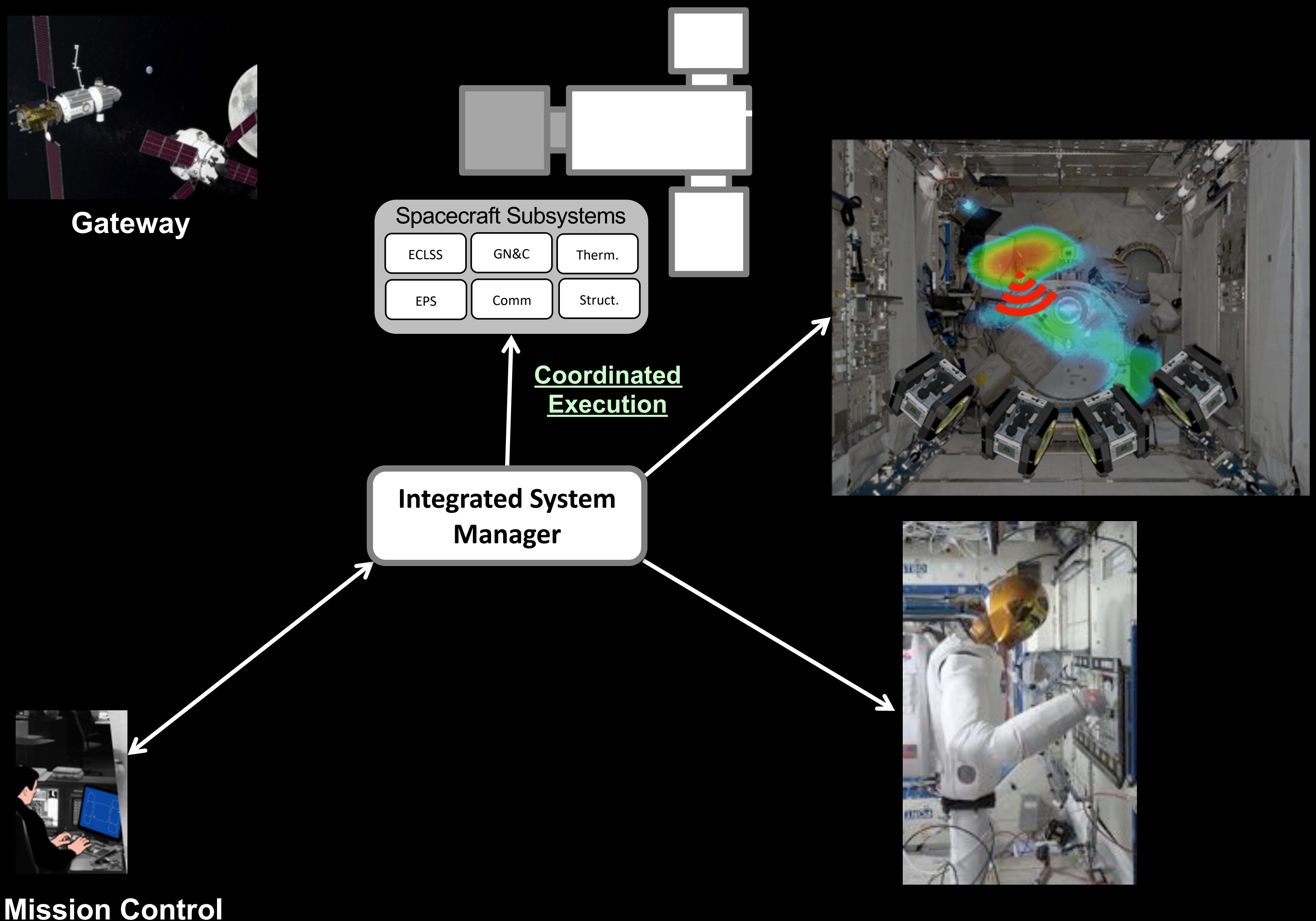
Technical thrusts



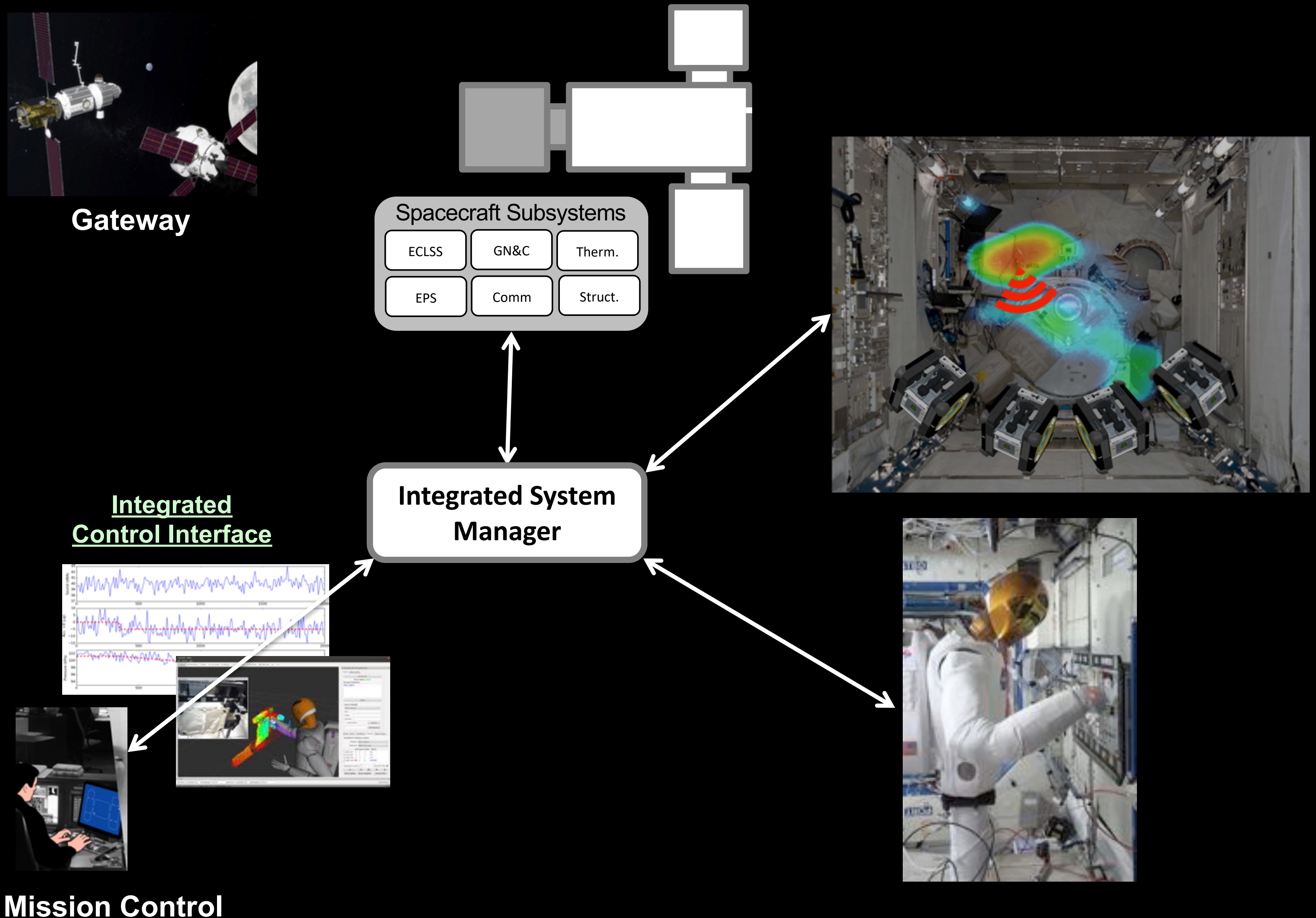
Technology: Integrated Data



Technology: Coordinated Execution

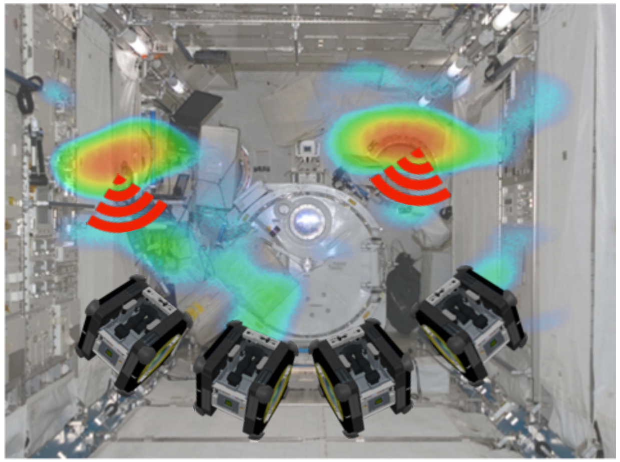


Technology: Integrated Control Interface

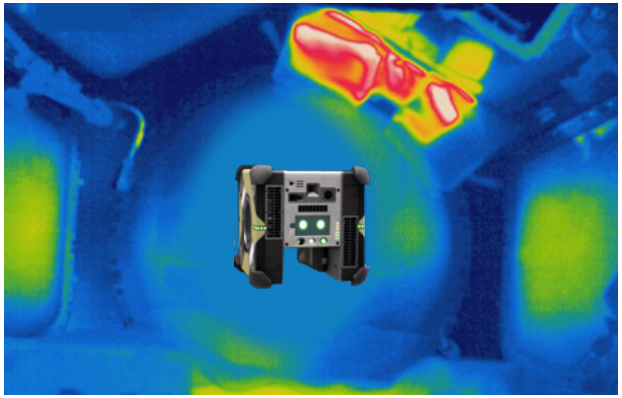


Capability areas

Autonomous State Assessment



Localizing signal sources by analyzing signal strength variation

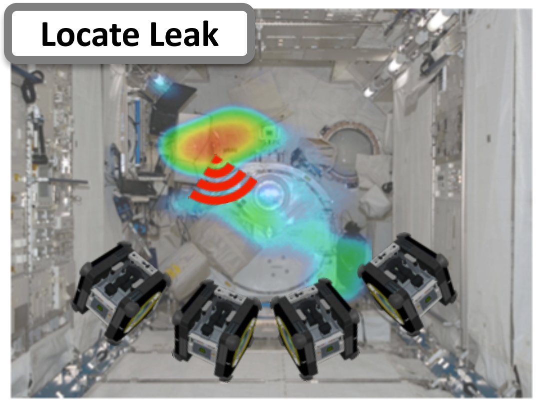


Habitat thermal mapping

Autonomous Logistics Management

Robotic cargo transfer

Integrated Fault Management



Phases

Phase	FY	Technical thrust	Capability area
1	~FY20	Integrated data	Autonomous state assessment
2	~FY21	Integrated control interface	Autonomous logistics management
3	~FY22	Coordinated execution	Integrated fault management

Key Performance Parameters

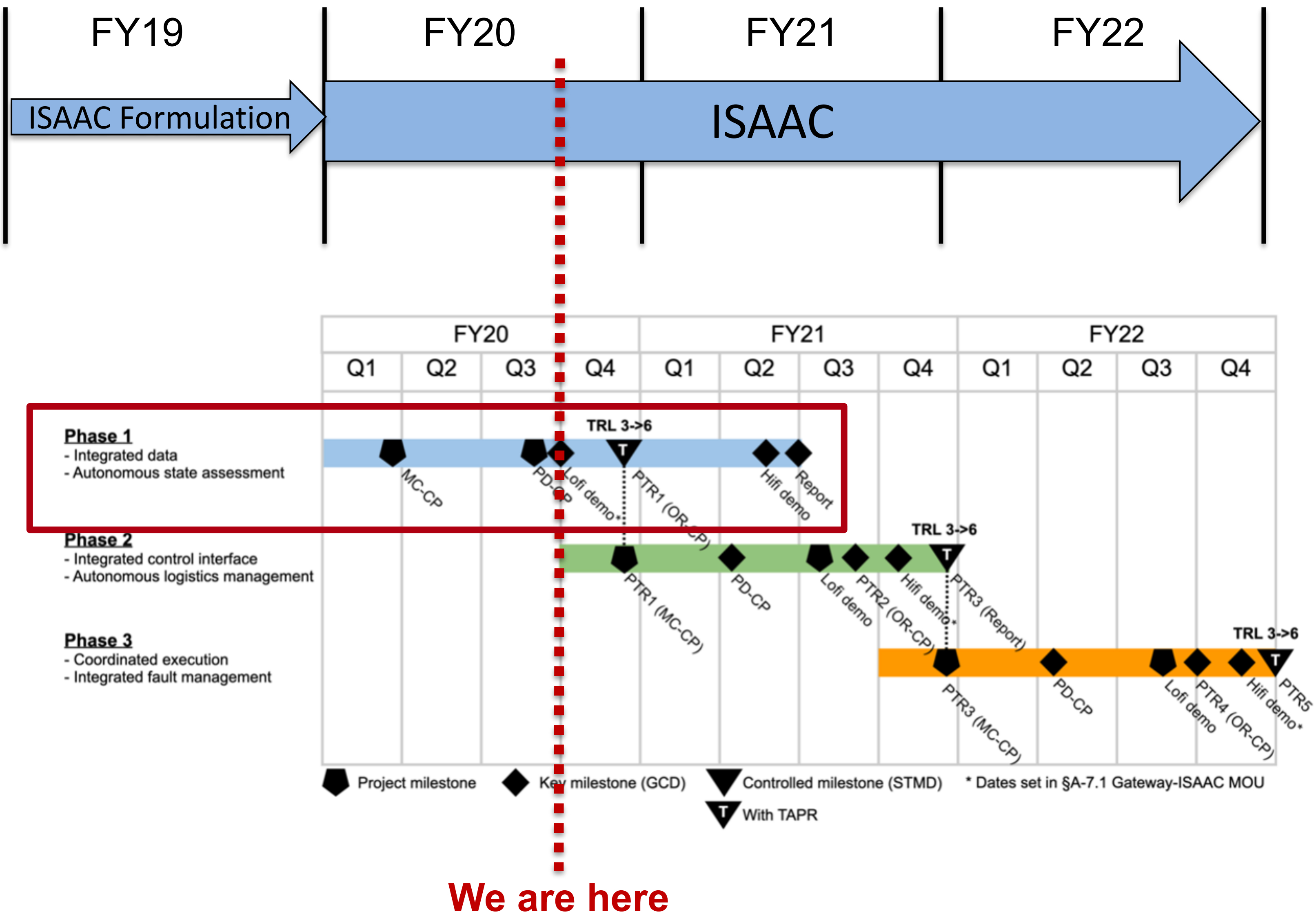
Capability Area	Key Performance Parameter (KPP)	Relevance	State of the Art	Threshold Value	Goal Value
State assessment	Spatial resolution	Resolving smaller features provides higher-fidelity state assessment	Sparse	1 cm	0.2 cm
	# sensing modalities	More co-registered sensors (e.g., RFID, thermal) support more applications	1*	4	7
Logistics	# operator hours per hour of cargo transfer in nominal ops	High operator productivity is key for reducing sustaining ops costs	1.0	0.5	0
	% robot idle time for cargo transfer	Autonomy efficiency constrained by delays due to operator difficulties (e.g. missing info, laborious commanding)	N/A**	60%	20%
Fault management	# subsystems data collected	Demonstrate ability to detect faults that cross subsystems, scale closer to full vehicle system	N/A**	3	5
	Minimum ultrasound SPL for locating leak	Drives percentage of operationally significant leaks that can be effectively located. For MMOD strike, can relate SPL to hole size and shape. For other leak types, can relate SPL to other leak parameters, such as flow rate.	N/A**	50 dB***	40 dB

* Based on Astrobee's existing sparse mapping capability. Other state-of-the-art systems are not for inside a space vehicle.

** "N/A": evaluated with respect to a novel scenario; direct comparison to other state of art is not possible.

*** Needs further study to understand relevant levels.

ISAAC schedule



Gateway-ISAAC Memorandum of Understanding

- Between March and June 2019, ISAAC negotiated an MOU that was signed by the NASA **Gateway** Program and **Advanced Exploration Systems** Program

- **MOU Content:**
  - **Infusion strategy**
  - Key customer requirements
  - Milestones
  - (… and more)

- The MOU documents Gateway's need for ISAAC technologies

Gateway-ISAAC MOU

The purpose of this Agreement is to document the planned infusion path from the technology developer to the technology recipient/user (e.g. the Center, Mission Directorate, or Office that intends to incorporate the technology into a subsystem, system, architecture, or application). This Agreement documents the technology's stakeholders and confirms their interest in the technology development activity to ensure NASA's expenditures are both desired and necessary. This agreement also documents any applicable resources that the receiving mission or project has committed to provide to the technology development effort.

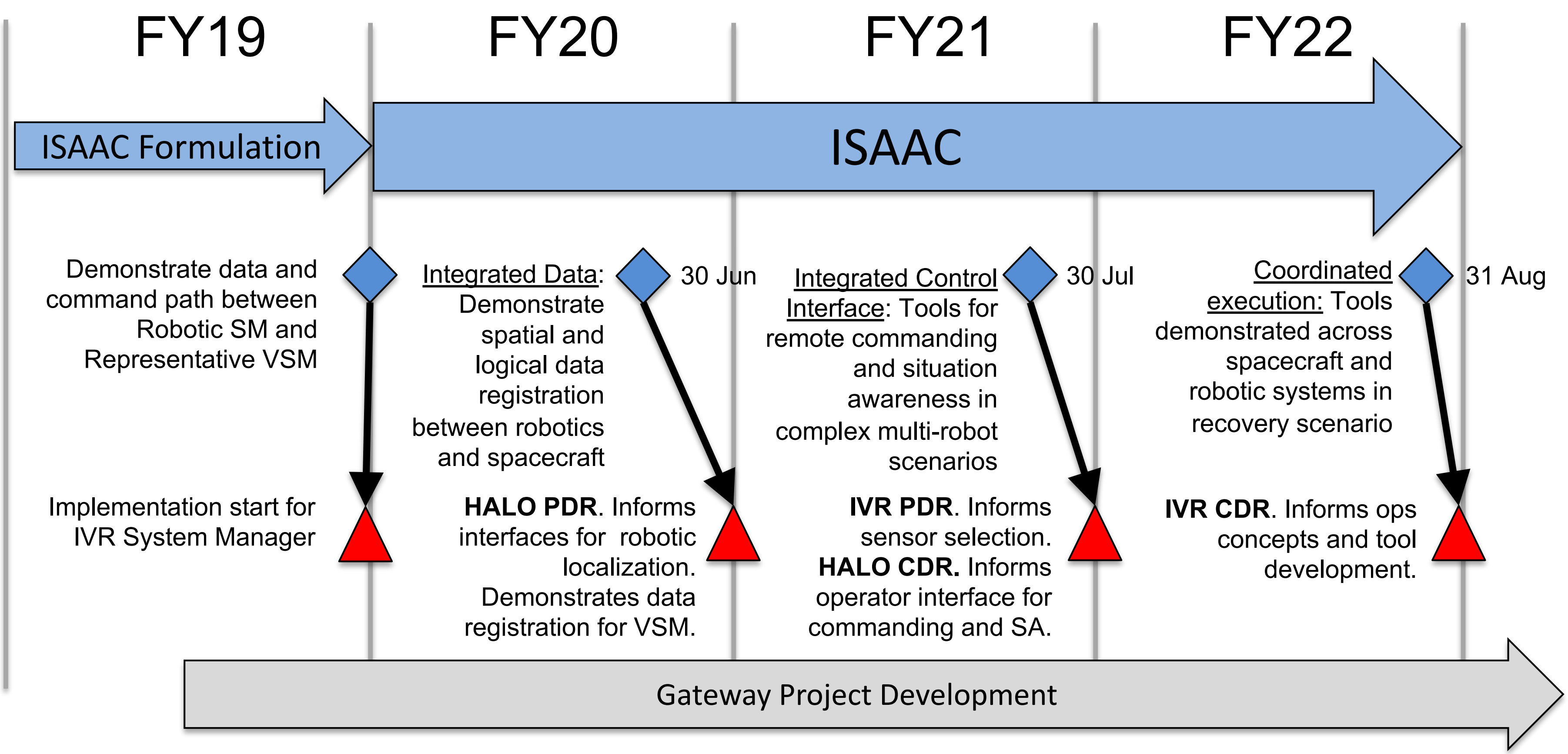
Partnership Summary:

Technology Title: Integrated System for Autonomous and Adaptive Caretaking (ISAAC)	
Technology Development Program: Space Technology Mission Directorate / Game Changing Development Program	**Technology Development Project:** Integrated System for Autonomous and Adaptive Caretaking (ISAAC)
Technology Development Program Manager: Drew Hope	**Technology Development Project Manager:** Matthew Deans
Receiving Program: Gateway	**Receiving Project:** Vehicle Systems Integration
Receiving Program Manager: Dan Hartman	**Receiving Project Manager:** Rod Jones
Is receiving project providing any resources or cost sharing? (Y/N): Yes, Gateway supports iPAS and other elements that ISAAC needs to succeed as cost sharing, Gateway supports the IVR and VSM Working Groups, which provide context, scenarios, specs, and draft requirements (i.e., stakeholder inputs to ISAAC).	
Primary Space Technology Roadmap Category: TA04	

Project Details:

Project Start Date: 2019-10-01	**Project End Date:** 2022-09-30
Project Start TRL: 3. Rationale uses GCD Tech Assessment approach. ISAAC has top-level requirements from Gateway L2s but does not yet have software requirements. Use cases and critical software functions have been identified. Preliminary architecture and design are documented. Interfaces between components are identified. Algorithms for critical software functions are defined. Test environment is identified to use iPAS and cFS. Critical functions exist in iPAS, MAST, and Astrobee simulator for preliminary validation of algorithms and architecture. Relevant target environment of Gateway is defined. Key performance parameters and limit values are drafted.	**Project End TRL:** 6. Rationale uses GCD Tech Assessment approach. By the end of ISAAC, Gateway system requirements will be stable and ISAAC will have demonstrated capabilities in iPAS and/or on ISS. Capabilities will be responsive to stakeholder Gateway working group inputs. Testing will include full-scale prototypes and all of the interfaces that are needed for application on Gateway. Software will be tested and algorithms will be validated in demonstrations in iPAS and or on ISS with realistic scenarios and flight-like data. VSM and system manager software running on flight-like fixed computing will be characterized to establish performance and computing requirements. Robot on-board software is excluded from this assessment (will not be in a flight-like computing environment).

Gateway schedule alignment

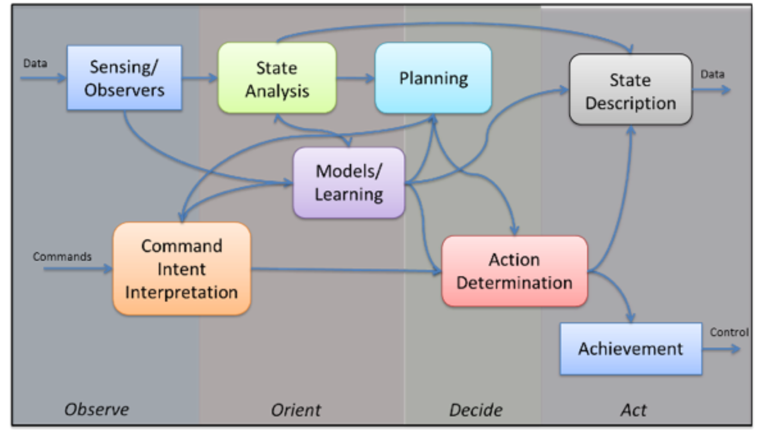


* Gateway milestones are shown with the target dates they had as of when the Gateway-ISAAC MOU was negotiated. We have recently learned that the HALO milestone dates have moved earlier by perhaps 2-3 months and the IVR dates are likely to move later by at least one year. We haven't yet determined how those changes should affect the ISAAC schedule.

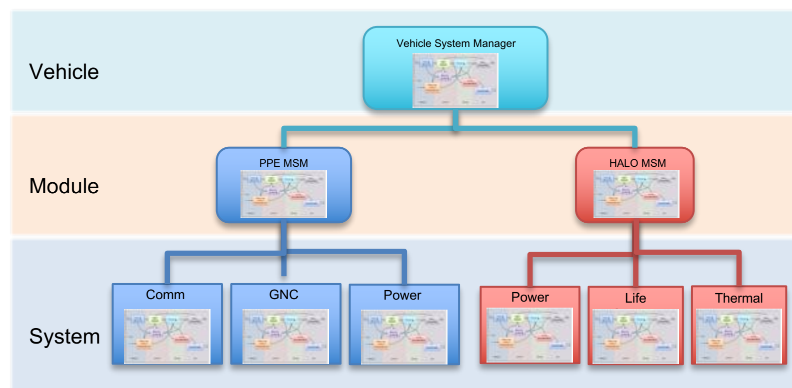
PHASE 1 BACKGROUND

MAST framework

- MAST: Modular Autonomous Systems Technology
- A framework for building distributed, hierarchical autonomous systems
- Intended for the autonomous monitoring and control of spacecraft
- Provides support for variable autonomy, assume-guarantee contracts, and efficient communication between subsystems and a centralized systems manager
- The MAST team has been conducting Gateway-related demonstrations since 2017
- MAST is now being used to architect the Gateway Vehicle System Manager (VSM)
- ISAAC is using MAST as a foundation for integrating vehicle subsystems and IVR



A MAST cluster implements the Observe-Orient-Decide-Act loop



Each node in the hierarchical system contains a MAST cluster

ISS Astrobee Facility

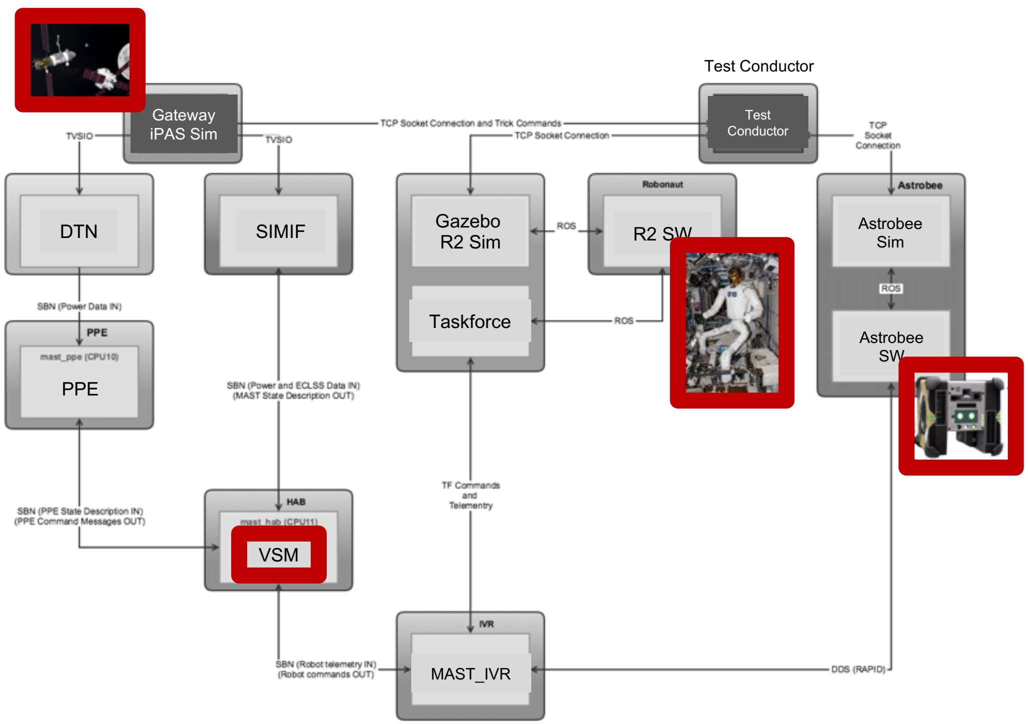
- The Astrobees are free-flying robots that operate inside the ISS
- There are three Astrobee flight units (launched in 2019) and three ground units
- ISAAC will be supported by the ISS Astrobee Facility's Guest Science Program, which:
  - Provides ground testing facilities
  - Facilitates ISS integration process
  - Supports ISS operations
- Per the Gateway IVR WG's IVR Conops document [1], a free flyer like Astrobee is likely to be part of the Gateway IVR architecture

Astrobee initial on-orbit checkout

[1] https://bender.jsc.nasa.gov/confluence/display/GIWG/Section+2.4%3A+Robot+Types

ISAAC formulation demo (Sep 2019)

- During the ISAAC formulation year (FY19), we conducted initial risk reduction technical work
- The work culminated in an end-of-year software simulation demo in Sep 2019
- One objective was to validate our communications architecture, shown at right, for linking the VSM, vehicle subsystems, and robots



Communications architecture

ISAAC formulation demo video

Video link: https://ti.arc.nasa.gov/ti/isaacdemo/
More details: https://ti.arc.nasa.gov/publications/74980/download/

PHASE 1 APPROACH

Need for phase 1

- **Technical thrust: Integrated data**
  - "Integrated data" technology collects data across multiple vehicle subsystems and robots, and takes the necessary steps to make it available for integrated analysis (makes prior data and live telemetry available in common formats, with common coordinate frames, with consistent identifiers, with common units, etc.)
  - **Why it's needed:** When executing cross-cutting tasks that affect multiple subsystems, integrated data is foundational both for enabling autonomy and improving operator situation awareness
  - **Examples:**
    - The VSM can't send a robot to retrieve a cargo bag unless it has a cargo location database with positions expressed in a coordinate frame that is known to the robot navigation system.
      - The same common coordinate frame enables an operator to view the robot and cargo bag together in a 3D visualization
    - To enable automated fault isolation using both vehicle and robot sensing, the same hardware component must be identifiable / cross-linked across (1) its node in the model of nominal system behavior, (2) the associated telemetry topics for its embedded sensors, (3) its geometry and location in a spatial database.
      - The same cross-links enable an operator to click on a component in a 3D visualization and use that to pull up the relevant sensor telemetry strip chart.
- **Capability area: Autonomous status assessment**
  - Autonomous status assessment is the capability for the combined vehicle/robot system to adequately assess its own status without crew, and with no/minimal operator support
  - **Why it's needed:** The information gathered by status assessment is critical for driving interventions that prevent or mitigate faults. The ISS program gets tremendous benefit from relying on crew for status assessment, but that familiar approach will not be available for Gateway during extended uncrewed periods. Even operator support will not be available during LOS.

Phase 1 objectives

- **Description in MOU: "**Demonstrate spatial and logical data registration between robotics and spacecraft"
- **Systems involved**
  - A subset of the vehicle subsystems
  - Free-flying mobile inspection robot: Astrobee
  - NOT mobile manipulator robot: Postpone trying to use Robonaut until phase 2, anticipating availability issues with very new robot still going through commissioning
- **Objectives**
  - 1.1 Build a 3D model of the environment with co-registered data from 4-7 robot sensor modalities ("spatial data registration")
    - **Examples:** RGB imagery, depth imagery, WiFi signal strength, RFID signal strength, thermal IR, etc.
  - 1.2 Detect three types of anomalies using robot sensor data, at least one using integrated data from both robots and vehicle subsystems ("logical data registration")
  - 1.3 Integrated demonstrations
- **Related KPPs:**
  - Spatial resolution
  - # sensing modalities

Phase 1 deliverables

[Same kind of deliverables for each phase]

A. **Preliminary report on feasibility and impact for Gateway**
   - Assess impact of using IVR to achieve focus capability area, for future mission profiles.
   - Study results will help with setting priorities for the rest of the phase.

B. **Software reference implementation**
   - The software developed during the phase. Note that ISAAC's objective is to develop its key technologies to TRL 6 in a Gateway-relevant environment, but ISAAC does not envision delivering flight software to Gateway. ISAAC-developed software may need significant rework to achieve the level of software engineering assurance required for Gateway needs.

C. **Test results report**
   - The results of feasibility demonstrations and tests conducted during the phase. Ideally, some of the tests will be conducted using IVR on the ISS.

D. **Report on performance, lessons learned, requirements for scaling to Gateway**
   - Explanations of the solutions, testing methodologies, and analyses, projections for performance in the target operational environment, definition of the scaling requirements from ISAAC tests to Gateway implementation, and relevant plans for commercialization and technology transfer.

Phase 1 demo concept of operations

- **Step 1: Perform baseline robotic survey**
  - The baseline survey collects robotic sensor data.
  - The data is batch downlinked to the ground
  - Ground software performs offline mapping, creating a 3D model of the environment, together with a set of baseline map (surface texture) layers, one for each sensor channel.
- **Step 2: Perform follow-up robotic survey**
  - The environment is artificially modified, if necessary, between the baseline and follow-up surveys
  - The follow-up survey streams robotic sensor data to the ground during execution
  - Ground software performs online mapping, incrementally updating map layers projected on the existing 3D model, and detecting changes and anomalies in the updated areas
- **Step 3: Perform targeted robotic inspection based on vehicle telemetry**
  - Ground software receives streaming vehicle telemetry and detects an anomaly
  - Ground software commands targeted robotic inspection of locations that may be relevant to the anomaly
  - (Besides the sensor coverage pattern, this step is otherwise the same as the follow-up robotic survey)

Phase 1 demo considerations

- **Robot environment**
  - **Available environments:**
    - **On-orbit (ISS):** Highest fidelity. This test mode can only be used to test sensors already available on Astrobee flight units.
    - **Ground testing:** (Granite lab / MGTF.) Medium fidelity. More sensors become available, assuming we invest in prototype-level Astrobee integration.
    - **Software simulation:** Lowest fidelity. Convenient during development, facilitates easily injecting changes and anomalies.
  - **Approach:** Depending on the maturity of each sensor, demo it either on the ground only, or ground + on-orbit. Either way, software sim can be used for development and debugging.
- **Vehicle telemetry sources**
  - **Available sources:**
    - **Receive real online telemetry:** Highest fidelity. Likely not practical on ISS due to complications with getting data access. Injecting anomalies problematic. At best, a stretch goal.
    - **Play back telemetry logs:** Medium fidelity. May be worthwhile if it's practical to find a log with a relevant anomaly.
    - **Generate telemetry from a software sim:** Lowest fidelity. Much more convenient and flexible than the other modes. Can simulate Gateway in addition to ISS.
  - **Approach:** Probably use software sim only. However, watch for opportunities to improve fidelity.
- **Demo iterations**
  - Iteration 1: Early prototype delivered in time for MOU milestone (by Jun 30)
    - Reduced scope (e.g. fewer sensors, not ready for on-orbit testing yet)
  - Iteration 2: Full demonstration (by Oct 30)

Phase 1 sensors for mapping

Data type	Candidate sensor	Map type	Demo environment	Map layer purpose	Change and anomaly detection	Demo iterations	Conops steps
Depth image	HazCam and/or SciCam SFM	Texture	Ground + on-orbit	Geometric context for all other data sets	[Not in demo scope]	1,2	1 or 2
RGB image	SciCam	Texture	Ground + on-orbit	General situation awareness	Add object obstructing ECLSS return vent. Survey or targeted inspection of vents. Flag obstruction.	1,2	**1-3**
WAP signal strength	MLP and/or HLP WiFi receivers	Volumetric	Ground + on-orbit	Find zones with degraded WiFi coverage. Inform planning for future WiFi upgrades.	[Not in demo scope]	1,2	1 or 2
RFID tag signal strength	RFID Recon	Texture (special)	Ground + on-orbit	Detect and accurately localize RFID-tagged items	[Not in demo scope]	2	1 or 2
Acoustic image	SoundSee*	Texture	Ground only	Detect and accurately localize sound sources (e.g. leak, failing motor)	Add sound source within survey area. Flag the new source.	2	1-2
Thermal IR image	TBD*	Texture	Ground only	Provide temperature spatial knowledge relevant for ECLSS (e.g. insulation performance). Detect and accurately localize heat sources, such as fires.	Add simulated fire thermal source. Flag the new source.	2	1-2
CO_2 concentration	TBD*	Volumetric	Ground only	Detect and accurately localize leak or fire. Note: CO_2 is a good working candidate for a safe and convenient tracer gas; actual ops might detect a different air contaminant (e.g. NH_3, soot particulate).	[Not in demo scope]	2	1 or 2

* Needs further study to confirm feasibility

Phase 1 sensors for mapping

Data type	Candidate sensor	Map type	Demo environment	Map layer purpose	Change and anomaly detection	Demo iterations	Conops steps
Depth image	HazCam and/or SciCam SFM	Texture	Ground + on-orbit	Geometric context for all other data sets	[Not in demo scope]	1,2	1 or 2
RGB image	SciCam	Texture	Ground + on-orbit	General situation awareness	Add object obstructing ECLSS return vent. Survey or targeted inspection of vents. Flag obstruction.	1,2	**1-3**
WAP signal strength	MLP and/or HLP WiFi receivers	Volumetric	Ground + on-or[illegible]	Find zones with degraded WiFi coverage. Inform [illegible]		1,2	1 or 2
RFID tag signal strength	RFID Recon	Texture (special)	Ground + on-orbit	Detect and accurately localize RFID-tagged items	[Not in demo scope]	2	1 or 2
Acoustic image	SoundSee*	Texture	Ground only	Detect and accurately localize sound sources (e.g. leak, failing motor)	Add sound source within survey area. Flag the new source.	2	1-2
Thermal IR image	TBD*	Texture	Ground only	Provide temperature spatial knowledge relevant for ECLSS (e.g. insulation performance). Detect and accurately localize heat sources, such as fires.	Add simulated fire thermal source. Flag the new source.	2	1-2
CO_2 concentration	TBD*	Volumetric	Ground only	Detect and accurately localize leak or fire. Note: CO_2 is a good working candidate for a safe and convenient tracer gas; actual ops might detect a different air contaminant (e.g. NH_3, soot particulate).	[Not in demo scope]	2	1 or 2

Demo iteration 1 – June 30

* Needs further study to confirm feasibility

Phase 1 sensors for mapping

Change & anomaly detection

Data type	Candidate sensor	Map type	Demo environment	Map layer purpose	Change and anomaly detection	Demo iterations	Conops steps
Depth image	HazCam and/or SciCam SFM	Texture	Ground + on-orbit	Geometric context for all other data sets	[Not in demo scope]	1,2	1 or 2
RGB image	SciCam	Texture			Add object obstructing ECLSS return vent. Survey or targeted inspection of vents. Flag obstruction.	1,2	**1-3**
WAP signal strength	MLP and/or HLP WiFi receivers	Volumetric	Ground + on-orbit	Find zones with degraded WiFi coverage. Inform planning for future WiFi upgrades.	[Not in demo scope]	1,2	1 or 2
RFID tag signal strength	RFID Recon	Texture (special)	Ground + on-orbit	Detect and accurately localize RFID-tagged items	[Not in demo scope]	2	1 or 2
Acoustic image	SoundSee*	Texture	Ground only	Detect and accurately localize sound sources (e.g. leak, failing motor)	Add sound source within survey area. Flag the new source.	2	1-2
Thermal IR image	TBD*	Texture	Ground only	Provide temperature spatial knowledge relevant for ECLSS (e.g. insulation performance). Detect and accurately localize heat sources, such as fires.	Add simulated fire thermal source. Flag the new source.	2	1-2
CO_2 concentration	TBD*	Volumetric	Ground only	Detect and accurately localize leak or fire. Note: CO_2 is a good working candidate for a safe and convenient tracer gas; actual ops might detect a different air contaminant (e.g. NH_3, soot particulate).	[Not in demo scope]	2	1 or 2

* Needs further study to confirm feasibility

Phase 1 sensors for mapping

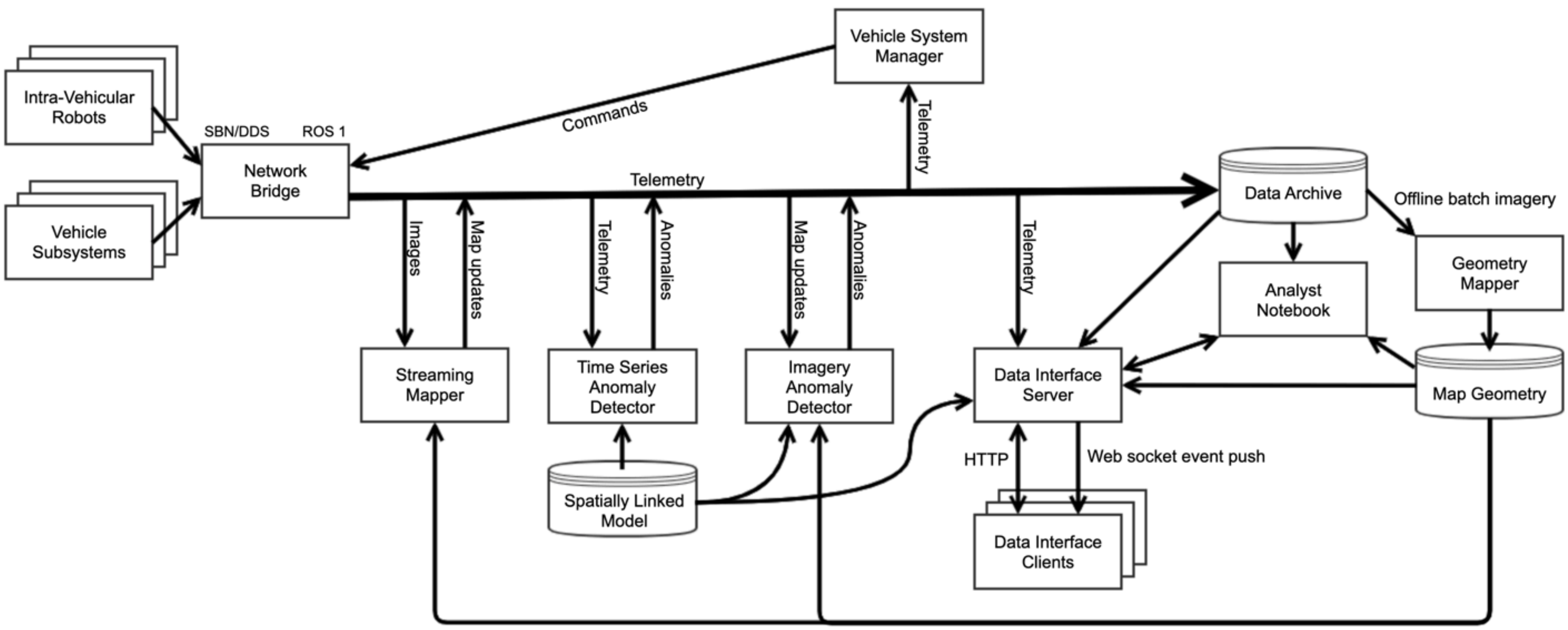
Change & anomaly detection

With linkage to vehicle subsystem data

Data type	Candidate sensor	Map type	Demo environment	Map layer purpose	Change and anomaly detection	Demo iterations	Conops steps
Depth image	HazCam and/or SciCam SFM	Texture	Ground + on-orbit	Geometric context for all other data sets	[Not in demo scope]	1,2	1 or 2
RGB image	SciCam	Texture	[illegible]	[illegible]	Add object obstructing ECLSS return vent. Survey or targeted inspection of vents. Flag obstruction.	[illegible]2	**1-3**
WAP signal strength	MLP and/or HLP WiFi receivers	Volumetric	[illegible]	[illegible] n	[Not in demo scope]	1,2	1 or 2
RFID tag signal strength	RFID Recon	Texture (special)	[illegible]	[illegible] s	[Not in demo scope]	2	1 or 2
Acoustic image	SoundSee*	Texture	Ground only	Detect and accurately localize sound sources (e.g. leak, failing motor)	Add sound source within survey area. Flag the new source.	2	1-2
Thermal IR image	TBD*	Texture	Ground only	Provide temperature spatial knowledge relevant for ECLSS (e.g. insulation performance). Detect and accurately localize heat sources, such as fires.	Add simulated fire thermal source. Flag the new source.	2	1-2
CO_2 concentration	TBD*	Volumetric	Ground only	Detect and accurately localize leak or fire. Note: CO_2 is a good working candidate for a safe and convenient tracer gas; actual ops might detect a different air contaminant (e.g. NH_3, soot particulate).	[Not in demo scope]	2	1 or 2

* Needs further study to confirm feasibility

Phase 1 software architecture



Phase 1 software modules

Module	Function
Network Bridge	Collects live telemetry from multiple vehicle subsystems and robot sensors on different buses. Normalizes data to enable integrated analysis.
Streaming Mapper	Building on prior 3D geometry, generates updated map (surface texture) layers from robot data in real-time.
Spatially Linked Model	A model like a circuit schematic with graphical structure [nodes are hardware components, links are functional connections, like wires, and parts of the graph may be included in a model of nominal subsystem behavior constraints] plus spatial structure [hardware components may have position information and 3D CAD models].
Imagery Anomaly Detector	Enables users to detect changes and anomalies in incoming imagery/map data by subscribing to be notified of new regions of interest that match user queries, including "join" queries that depend on integrated data from multiple sources.
Time Series Anomaly Detector	Same concept as Imagery Anomaly Detector, but applied to time series telemetry, and regions of interest are time windows
Vehicle System Manager	Simplified version of Gateway VSM inherited from MAST and ISAAC formulation year. Main function is to receive an anomaly event and command Astrobee to investigate.
Data Interface	Initial version of the Integrated Control Interface web-based tool that will be a major focus in Phase 2. During Phase 1, the interface will focus on real-time 3D visualization of incoming imagery data, with the hardware components in the 3D model linked to incoming telemetry and SLM data. It will not address commanding.
Analyst Notebook	A component of the Data Interface that provides an interactive sandbox environment for experimenting with and debugging change detection queries.
Geometry Mapper	Generates as-built 3D geometry model with high-quality surface texture from Astrobee sensor data. This can be a manual batch process with expert operator assistance.
(Sensing Enhancements)	Not all sensors we plan to use with Astrobee during Phase 1 are currently integrated with Astrobee. Some need hardware integration (to a prototype level, not intended to launch to ISS); others may need software drivers, calibration, or other integration effort.

Phase 1 relevance to Gateway IVR WG recommendations

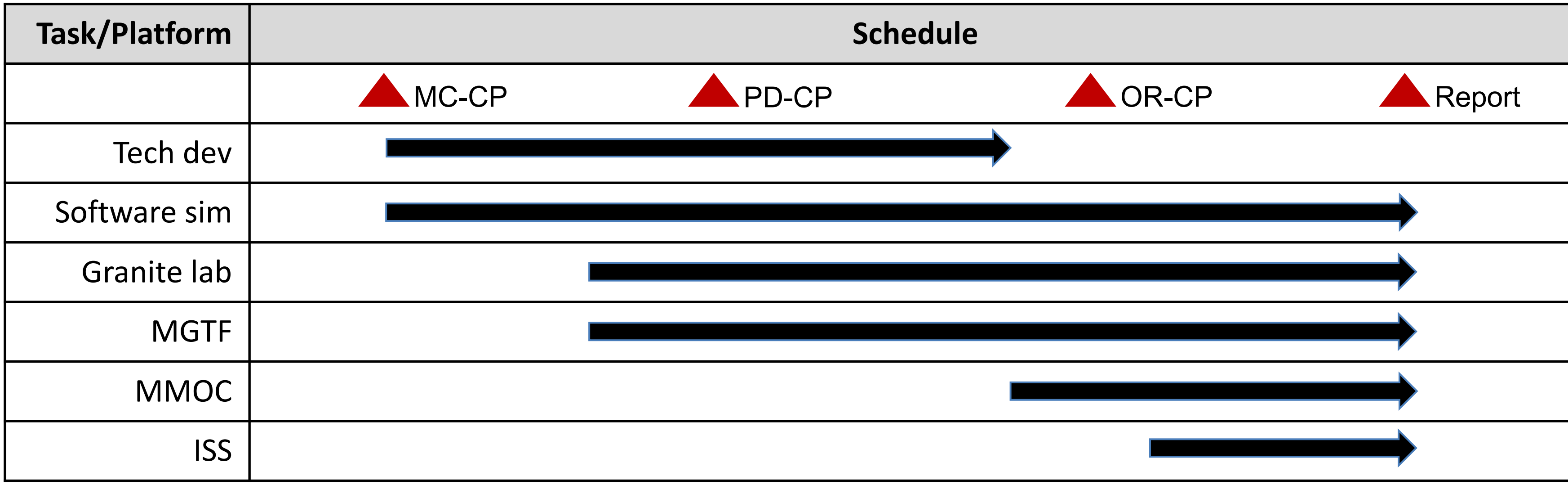
Document	Section/ID	Excerpt	Relevance
Conops use cases [1]	1.3	Due to the comparatively low crew presence time on the Gateway, the IVR system can supplement the Gateway and ECLSS system's needs for inspection tasks.	Motivates "autonomous state assessment" capability. One of the proposed change & anomaly detection demonstrations applies to the ECLSS system (inspect for blocked vents).
Conops use cases [1]	1.3.1	Visual inspection. Can include building a baseline data set and monitoring for unexpected changes.	Motivates change & anomaly detection
IVR subsystem spec requirements [2]		Status Info: The IVR system shall acquire and distribute internal status information indicating position, health, environmental data during IVR operations to the Gateway.	Related to Integrated Data and operation of the Network Bridge module -- concept of collecting telemetry from multiple subsystems including IVR
Gateway human systems integration requirements [3]	10069	Operator control stations for robotic systems shall provide the displays and interfaces needed for [situation awareness] to perform tasks and manage the system.	Motivates Data Interface software module
Gateway human systems integration requirements [3]	10073	Robotic systems designed to have multiple operators shall be able to accept the input from and arbitrate between multiple operators so as to perform safely and without degradation.	Multi-operator collaboration is the main long-term reason that we plan to make the Data Interface a web-based tool (the web server can ensure consistency and enable coordination between multiple operators).
System Interface Requirements [4]	IVR-L3-004 (GTW-L2-0143)	IVR Inspection: The Gateway shall provide a reconfigurable mobile video camera system for inspection and internal robotic operations support.	Motivates "autonomous state assessment" capability.
System Interface Requirements [4]	Under consideration	The Gateway shall provide an integrated robot control station spanning both vehicle subsystems and IVR.	Motivates Data Interface software module

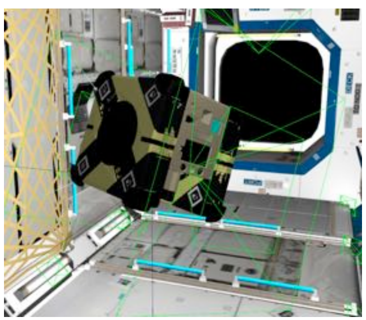
[1] https://bender.jsc.nasa.gov/confluence/display/GIWG/Section+4%3A+Nominal+Use+Cases
[2] https://bender.jsc.nasa.gov/confluence/display/GIWG/IVR+Subsystem+Spec+Requirements
[3] https://bender.jsc.nasa.gov/confluence/display/GIWG/GHSIR+Requirements
[4] https://bender.jsc.nasa.gov/confluence/display/GIWG/Draft+System+Interface+Requirements

Risk mitigation

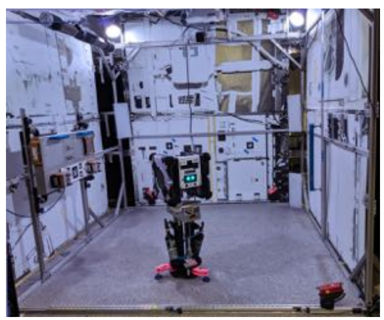
Summary	Description	Imp	L	C	Appr.	Approach notes
Robot availability	Both ground and on-orbit robotic testing require effort from key robot facility staff (Astrobee, Robonaut2) that split their time between maintenance and supporting experiments conducted by multiple robot users. Schedule slips may be caused by unpredictable failures of robots or lab facilities, as well as by other robot users (e.g. if they need to postpone a scheduled experiment or repeat a failed experiment).	S	2	4	Mitigate	Incorporate significant margin in schedule. Document development and testing plan with robot facility staff, including prioritized needs and descope options. Iterate with facility staff to update schedule and assess feasibility.
ISS availability	On-orbit testing requires reserving ISS resources, which may include robot hardware, crew time for robot setup, tear-down, and oversight, and use of ISS cabin volume (which may conflict with other crew activities). Schedule slips may be caused by unpredictable ISS resource conflicts.	S	2	4	Mitigate	Incorporate significant margin in schedule. Document development and testing plan with ISS program. Get on the Integrated Payload List as soon as possible. Iterate with ISS PIM and SpOC to update schedule and assess feasibility.
Diverse scope	ISAAC technology integrates multiple components (vehicle subsystems and robots). Beyond the core technology development, actually testing the integrated system requires interacting with a large number of components. It will be an ongoing challenge to control the level of effort invested per component (interfacing to the component, adapting the conops to make use of the component, ensuring component readiness for testing, etc.) so the testing can be completed within the available cost and schedule.	C	2	3	Watch	Estimate team velocity and level of effort feasibility. Descope or scale back investment in some components, if needed.
Coordinated execution	An ISAAC tall pole is developing a system for coordinated execution of high-level tasks with multiple robots and vehicle subsystems, including autonomous replanning for execution contingencies. The risk is that, when delivered, the coordinated execution system will not be considered mature enough for technology transfer due to shortcomings like (1) too much overhead required to implement new high-level tasks within the system, (2) inability to verify and validate the system reliability sufficient for mission assurance requirements, or (3) insufficient system expressiveness to implement the most critical high-level tasks.	T	2	3	Mitigate	Work with application subject matter experts and potential users to understand usability, quality, and task coverage requirements. Focus effort carefully on meeting documented user needs.
Staff recruiting	ISAAC staffing needs can't be met with current employees, so recruiting is needed. Competition for talent is tough right now, including competing with other local projects (VIPER).	S	3	2	Mitigate	Do aggressive recruiting early in FY20, including both external calls and NASA-internal recruiting where possible.

Astrobee testbed: ISAAC phase life cycle sketch





Software sim
- Ideal for initial testing, final ops planning with real ISS geometry
- HIL possible for some components



Granite Lab
- 3-DOF air bearing for Astrobee ground units
- Ideal for simulating closed loop motion control

Micro-Gravity Test Facility (MGTF)
- Active 6-DOF motion platform for Astrobee ground units
- Ideal for simulating localization and mapping
- *Motion platform currently non-operational.*

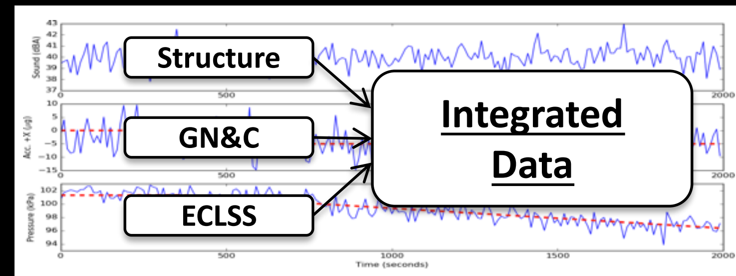
Multi-Mission Operations Center (MMOC)
- Includes Astrobee control station workstations
- Rehearse ground procedures with realistic comms

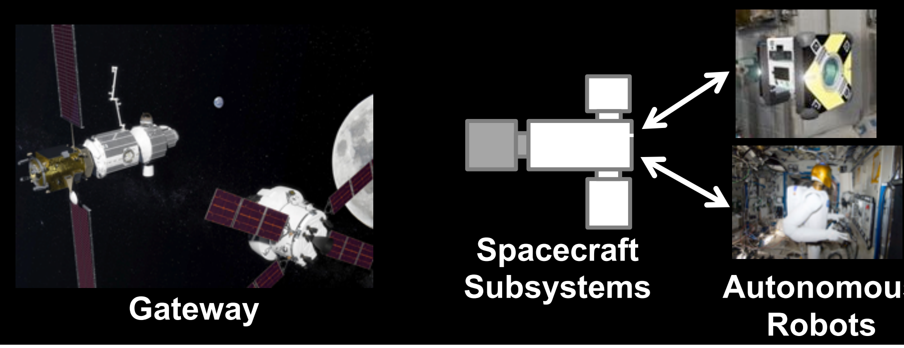
Integrated System for Autonomous and Adaptive Caretaking (ISAAC)

PT: Terry Fong (Autonomous Systems)
Thrust Area: ST5
PM: Trey Smith (ARC)
Deputy PM: Julia Badger (JSC)
Centers: ARC + JSC

Description and Objectives

- Develop a **critical capability to support autonomous caretaking of exploration spacecraft while uncrewed**
- Integrate autonomous robots, spacecraft infrastructure (avionics, sensors, network), and ground control
- Enhance **autonomous state assessment**, **autonomous logistics management**, and **integrated fault management**
- Focus on **capabilities required for the Gateway** (Human Exploration Requirements HEOMD-004: GTW-L2-0044, 0047, 0050, 0142, 0143, 0145) and applicable beyond the Earth-Moon system.
- Enable **important assessments of feasibility and relevance** for the design of future deep space spacecraft.
- Extend **autonomous system manager architecture** to enhance integrated analysis of data, operator productivity, and reliable coordinated execution of system-level tasks.





Customers

- **Gateway.** ISAAC-developed capabilities directly relevant to HEOMD-004 and other requirements for Gateway.

Partners

- **AES Autonomous Systems and Operations**. Support fault diagnosis and planning+execution technologies used by ISAAC architecture.
- **AES Astrobee Facility.** Support Astrobee testbeds and ops.
- **AES Logistics Reduction**. Support Robonaut2 testbeds and operations. Collaborate on logistics demonstration.
- **Gateway Intra-Vehicular Robotics (IVR) and Vehicle System Manager (VSM) Working Groups.** Provide Gateway guidance.

Leverage

- **GCD/HET2.** Developed analog robot platforms for Gateway IVR.
- **MAST.** Developed system architecture that ISAAC will extend.

Technical Approach

- Focus on three technical thrusts:
  - Integrated Data: Link models and telemetry across multiple spacecraft subsystems and robots
  - Coordinated Execution: Enable higher-level commanding and effective collaboration
  - Integrated Control Interface: Enable mission control to understand and control integrated autonomous systems
- Perform tests with the iPAS facility (JSC) and on ISS
  - Leverage existing testbeds and robots developed with STMD support
  - Capstone demo on ISS: Link embedded sensors and multiple robots to detect, isolate, and patch a simulated leak
- Proposing ISAAC development in **FY20-22**
  - Deliverables staged to respond to relevant Gateway milestones
- **Investment is needed now in order to meet Gateway needs**