From Space to Street – Autonomy Lessons from Deep-Space Human Missions

2nd Space Robotics Workshop
IEEE SMC-IT/SCC 2025 | Los Angeles

Ignacio López-Francos – *ignacio@nasa.gov*

AI Research Lead

Intelligent Systems Division - NASA Ames Research Center

Mars Campaign Office / Earth Independent Operations Domain

Summary

- Crewed Mars missions will see unavoidable decreases in support from Earth for real-time onboard trouble-shooting and problem-solving.

- An integrated set of technologies is needed to ensure that crew can effectively respond to on-board situations in the absence of real-time ground support.

- This presentation describes describes early progress towards integrated hardware and software intended to reduce risk for a crewed Mars mission.
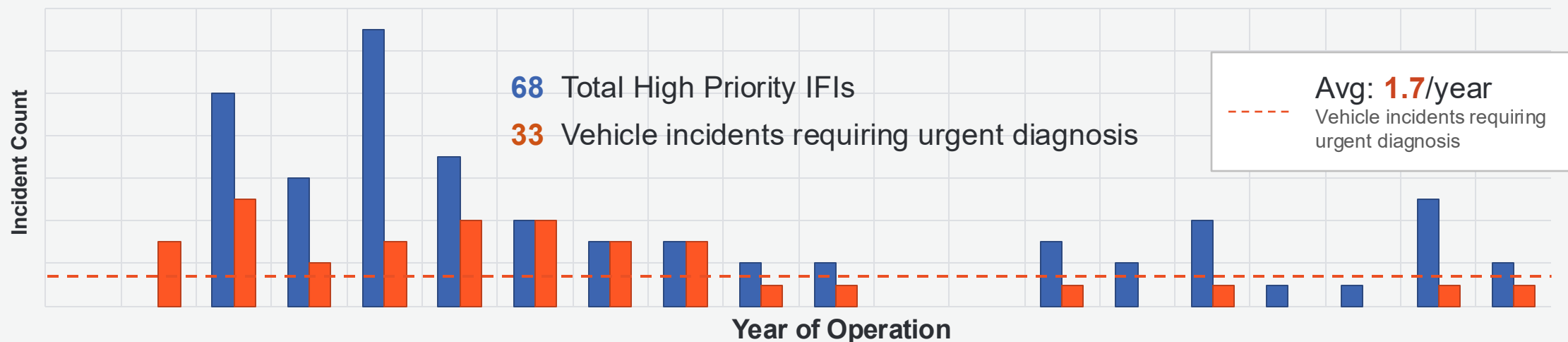
A Crewed Mission to Mars

- Long Mission : ~3 yrs
- No evacuation (a year to get back after 3-4 weeks)
- No resupply during transit
- Small crew, delayed ground support

~20 mins delay

~20 mins delay

~15 mins to work the problem

Support from ground will be 1 hour outdated

Anomalies and Alarms

ISS: Significant Incidents in Vehicle Systems Requiring Immediate Response

68 Total High Priority IFIs

33 Vehicle incidents requiring urgent diagnosis

Avg: **1.7**/year
Vehicle incidents requiring urgent diagnosis

Incident Count

Year of Operation

Apollo Anomalies

Avg: **3** significant incidents / mission
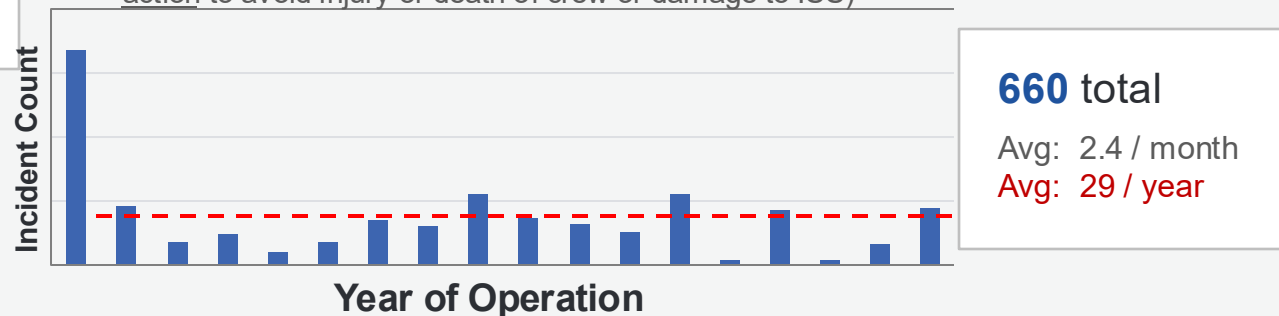

362 Total Anomalies

35 Significant Incidents

ISS: Class 2 Alarms

(indicate that crew or ground needs to take <u>immediate action</u> to avoid injury or death of crew or damage to ISS)

Incident Count

Year of Operation

660 total

Avg: 2.4 / month
Avg: 29 / year

This is not unique to human spaceflight missions

Unanticipated Anomalies

Aviation Failure Modes: 40% Unanticipated

3.2.3 Managing Malfunctions

Finding 3 - Managing Malfunctions.
Pilots successfully manage equipment malfunctions as threats that occur in normal operations. However, insufficient system knowledge, flightcrew procedure, or understanding of aircraft state may decrease pilots' ability to respond to failure situations. This is a particular concern for failure situations which do not have procedures or checklists, or where the procedures or checklists do not completely apply.

Over 40% of failure modes were unanticipated by designers, cases where pilots have to rely on their knowledge, skill and other aspects of airmanship to mitigate the risk because there was no procedure to follow
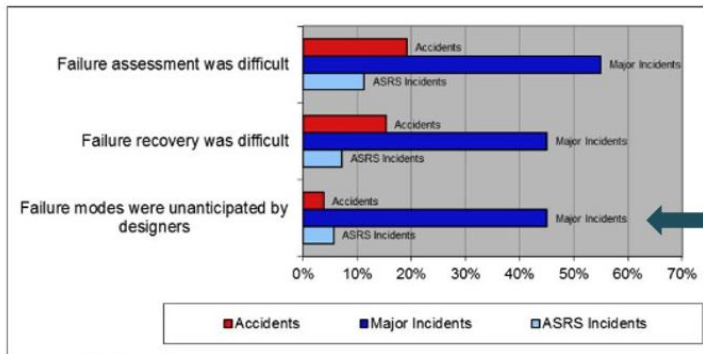
Figure 9. Failure Issues.

Source: National Transportation Safety Board. (2013). Final report of the performance-based operations aviation rulemaking committee / commercial aviation safety team flight deck automation working group (Docket No. SA-537, Exhibit No. 14-E).

3

Aerospace Software Failures: 40% Unanticipated

National Aeronautics and Space Administration
NASA Engineering and Safety Center Technical Bulletin No. 23-06

Considerations for Software Fault Prevention and Tolerance

Mission or safety-critical spaceflight systems should be developed to both reduce the likelihood of software faults pre-flight and to detect/mitigate the effects of software errors should they occur in-flight. New data is available that categorizes software errors from significant historic spaceflight software incidents with implications and considerations to better develop and design software to both minimize and tolerate these most likely software failures.

New Historical Data Compilation Summary

Previously unquantified in this manner, this data characterizes a set of 55 high-impact historic aerospace software failure* incidents. Key findings are that software is much more likely to fail by producing erroneous output rather than failing silent, and that rebooting is ineffective to clear these erroneous situations. Forty percent (40%) of software errors were due to absence of code, which includes missing requirements or capabilities, and inability to handle unanticipated situations. Only 18% of these incidents fall within the software discipline itself, with none being related to choice of platform or toolset. The origin of each error is [...] to focus specific development, test, and validation techniques [...] prevention in each category. This new data focuses on manifestation [...] unexpected flight software behavior independent of ultimate root cause. It is provided for considerations to improve software design, test, and operations for resilience to the most common software errors and to augment established processes for NASA software development.

Best Practices for Safety-Critical Software Design

Although best efforts can be made prior to flight, software behavior reflects a model of real-world events that cannot be fully proven or predicted, and traditional system design usually employs only one primary flight software load, even if replicated on multiple strings. Like designing avionic systems to protect for radiation and mistrusted communica-

❖ *From 2023 study by Lorraine Prokop, NASA Engineering and Safety Center Software Tech Fellow*

Forty percent (40%) of software errors were due to absence of code, which includes mission requirements or capabilities to handle unanticipated situations.

Similar number reported in recent interview with Los Angeles Class submarine crew member

4

Current Ground Response Capability

About 70 Flight Controllers across three daily shifts

Front Room - Mission Control Center (MCC-H FCR + MPSRs)

- 50+ operators on console

- 20+ specialists on call

- **~500 years combined *on-console* experience (operators only)**

- 600+ years combined *relevant* experience (operators only)

- 80% have 1+ engineering degrees

- 50% have a degree in Aerospace / Aeronautical / Astronautical Engineering

Estimated years experience by console:

Back Room - Mission Evaluation Room (MER)

- 30+ engineers on console

- **~161 years combined on-console experience**

- 556 years combined relevant experience (estimated based on average experience level of MART participants)

- In one MART meeting:
    - 747 years combined relevant experience
    - Average experience level: 17 years

~660 years combined console experience

22 unique console disciplines

MCC-H FCR: Mission Control Center-Houston Flight Control Room
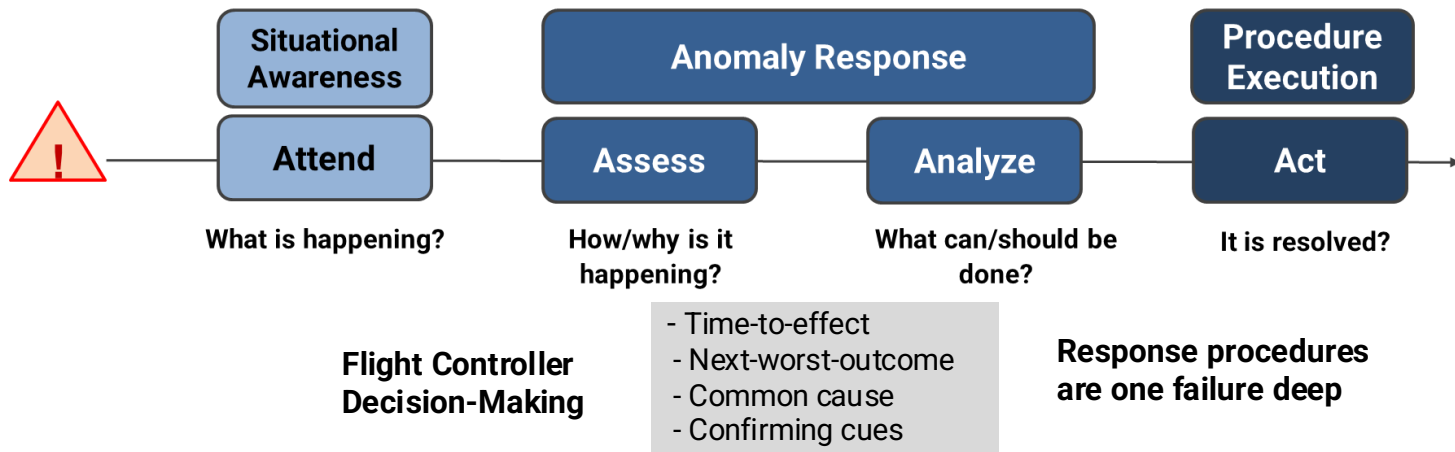MSPR: Multi-Purpose Support Room

Real-time Crew Interaction

During Mars missions, the crew-system team will need to take on critical capabilities currently performed by experts on the ground with access to vast datasets.

Currently, the ground:
- Guides and oversees procedure execution in real time, preventing crew error and advising when an unexpected result is reached
- Monitors system telemetry and automated actions to track and respond to major system state changes

There are three types of activities performed by Ground Controller that currently depend on real-time or near real-time communication:

⚠️	**Situational Awareness**		**Anomaly Response**		**Procedure Execution**
	Attend		**Assess**	**Analyze**	**Act**
	What is happening?		How/why is it happening?	What can/should be done?	It is resolved?

Flight Controller Decision-Making

- Time-to-effect
- Next-worst-outcome
- Common cause
- Confirming cues

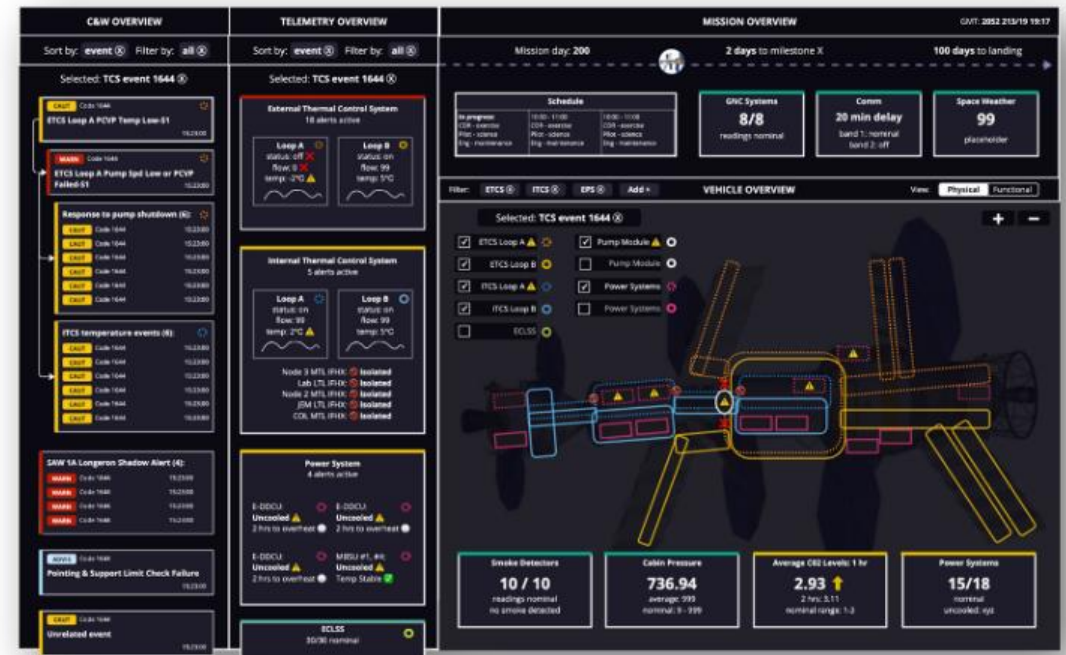**Response procedures are one failure deep**
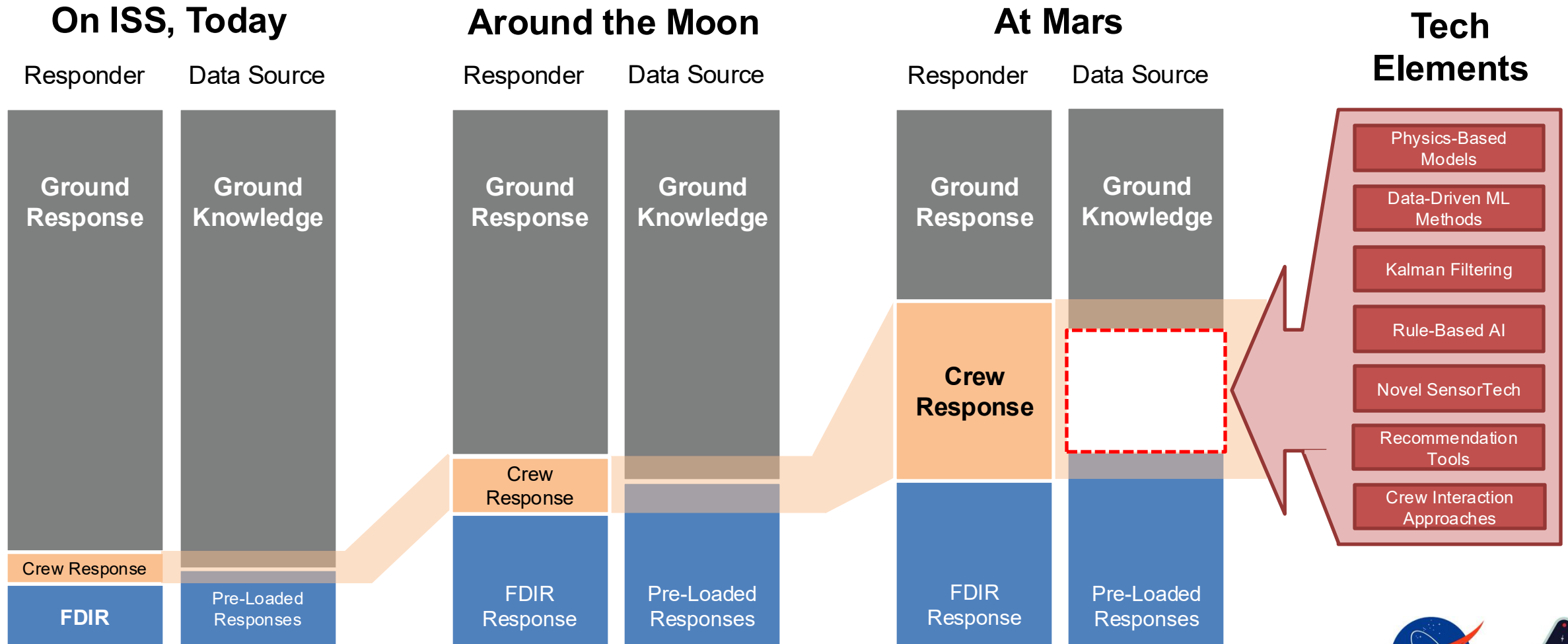
Decision Support Technology Gaps

Capabilities needed to support crew with time-critical *unknown knowns*:

1. Onboard real-time telemetry analytics for diagnostics

2. Integration of statistical and rule-based AI output

3. Real-time AI analysis with crew input

   – Crew as sensors and hypothesis generators

4. Crew interaction with complex engineered systems

   – Including autonomy inhibition

Humans and AI succeed and fail in different ways

A Layered Approach to Problem Solving

On ISS, Today

Responder | Data Source

Ground Response | Ground Knowledge

Crew Response |

FDIR | Pre-Loaded Responses

Around the Moon

Responder | Data Source

Ground Response | Ground Knowledge

Crew Response |

FDIR Response | Pre-Loaded Responses

At Mars

Responder | Data Source

Ground Response | Ground Knowledge

Crew Response |

FDIR Response | Pre-Loaded Responses

Tech Elements

- Physics-Based Models
- Data-Driven ML Methods
- Kalman Filtering
- Rule-Based AI
- Novel SensorTech
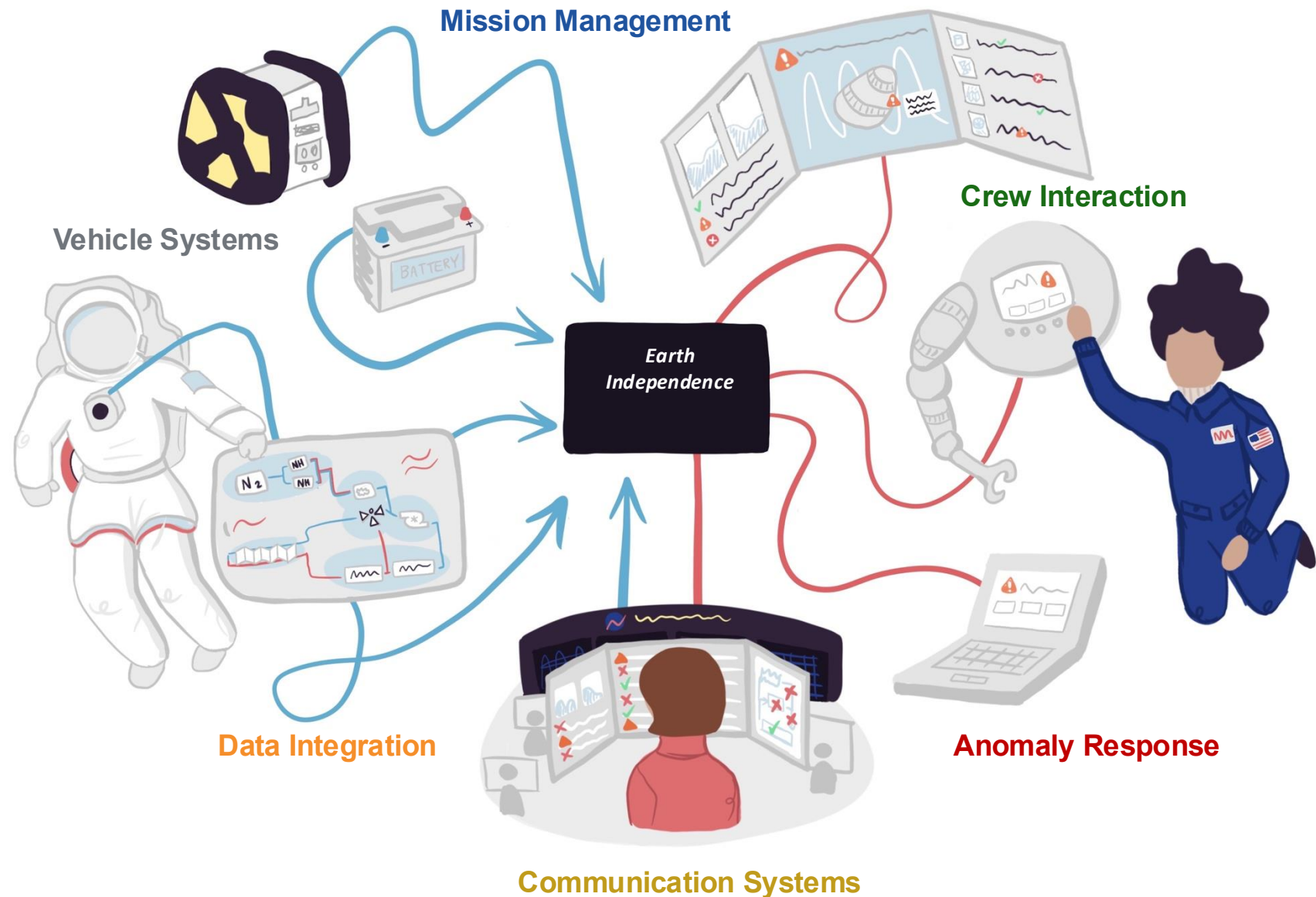- Recommendation Tools
- Crew Interaction Approaches

1 | 3

Path to Earth-Independent Operations

- Onboard systems automate as much as safely possible and work with crew and ground when it not possible
- Use crew as sensors and adaptive/flexible problem solvers
- Ground Control on Earth will manage everything that is not urgent

Leveraging major advances of the past ~decade:
- Sensor technologies
- Compute Hardware
- AI/ML

THANK YOU