

National Aeronautics and  
Space Administration



# VIPER Lunar Rover and the Exploration of the Moon

Izmir

Jay Trimble  
MARCH 14, 2023

The image shows a close-up view of the lunar surface, characterized by numerous dark, circular craters of varying sizes. The surface is illuminated from the side, creating deep shadows and highlighting the rugged terrain. A solid, bright blue horizontal band runs across the center of the image, serving as a background for the title text.

# The Mission

# The New Moon....

Not that long ago, we understood the Moon very differently...

We studied from the Earth, from the Moon's surface, and had returned samples to Earth.

General conclusion was:

- Surface was relative constant
- Essentially no atmosphere
- Bone dry

Recent Lunar Missions like LCROSS, LRO, LADEE, and others changed all that...



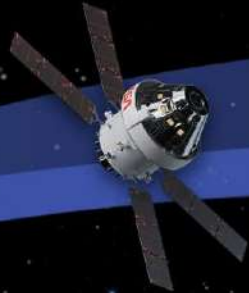
# Enter Artemis: Landing Robots & Humans On the Moon



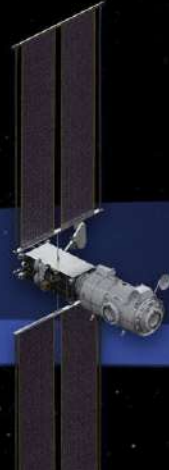
Lunar Reconnaissance Orbiter: Continued surface and landing site investigation



Artemis I: First human spacecraft to the Moon in the 21st century



Artemis II: First humans to orbit the Moon and rendezvous in deep space in the 21st century



Gateway begins science operations with launch of Power and Propulsion Element and Habitation and Logistics Outpost



Artemis III-V: Deep space crew missions; cislunar buildup and initial crew demonstration landing with Human Landing System



**Early South Pole Robotic Landings**  
Science and technology payloads delivered by Commercial Lunar Payload Services providers

**Volatiles Investigating Polar Exploration Rover**  
First mobility-enhanced lunar volatiles survey

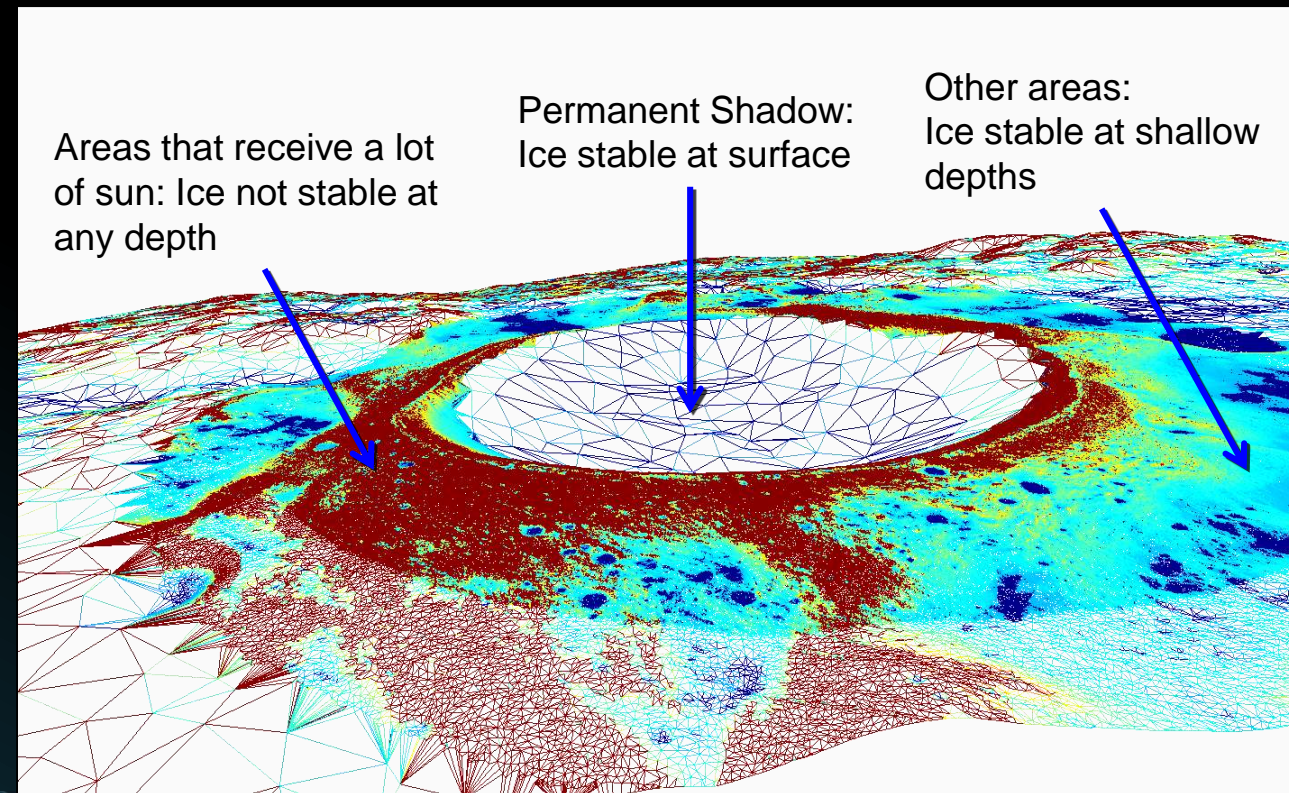
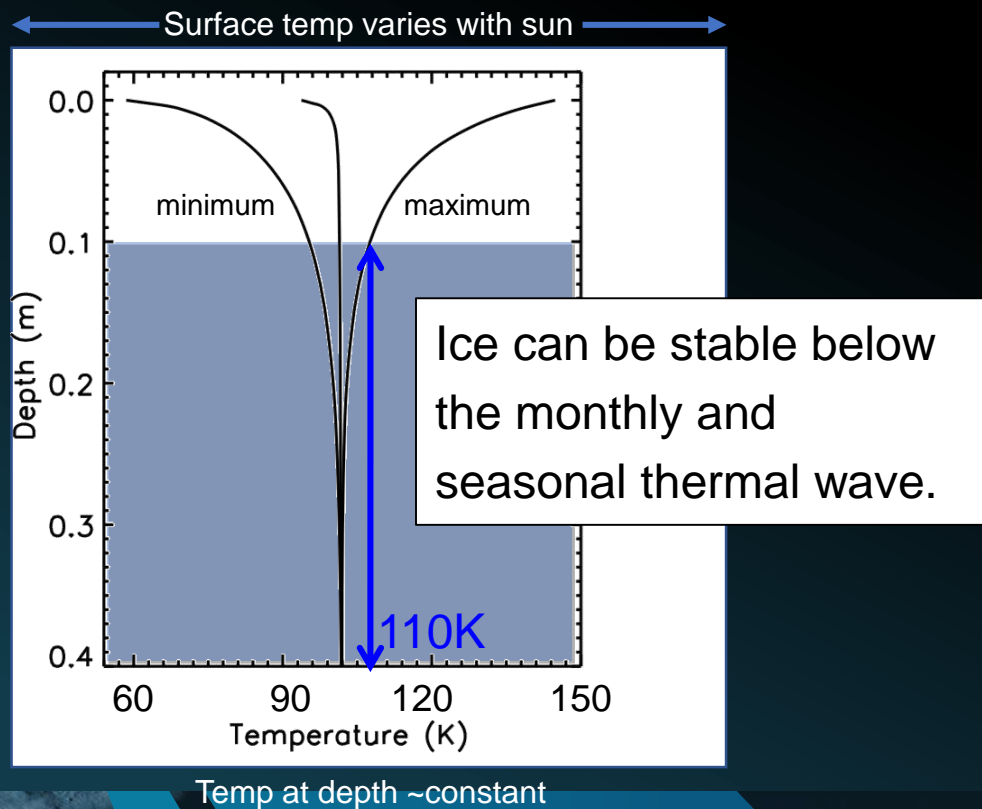
Uncrewed HLS Demonstration

**Humans on the Moon - 21st Century**  
First crew expedition to the lunar surface

**LUNAR SOUTH POLE TARGET SITE**

# If Ice were present, where could it last?

- Temperatures that remain below 110 K are necessary for long-term ice sequestration
- Where does it stay cold enough over long periods of time?
  - On the surface in permanent shadow
  - Below the surface ... how deep depends on transient surface heating by the sun.
- Summarized as 'Ice Stability Depth'



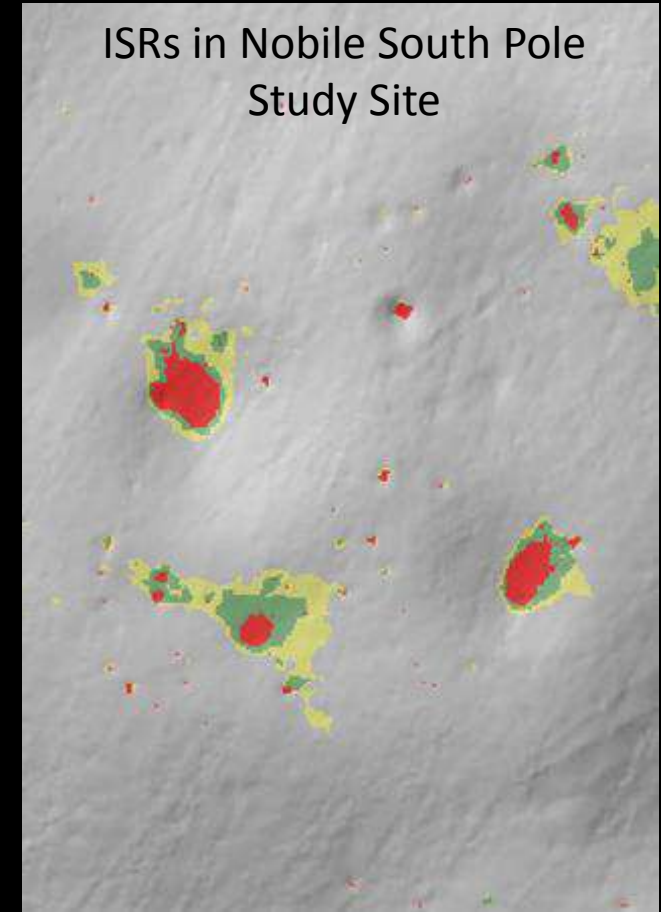
Maps generated by Matt Siegler (PSI) and Dave Paige (UCLA)

# Where will VIPER explore?

VIPER will explore four polar “**Ice Stability Regions**” (ISRs)\*:

- “**Surface**” - Ice expected stable on the surface (PSRs – Permanently Shadowed Regions)
- “**Shallow**” - Ice expected stable between 0-50cm of the surface
- “**Deep**” - Ice expected stable between 50-100 cm of the surface
- “**Dry**” - Ice *not* expected stable (0-100cm *too warm* to be stable)

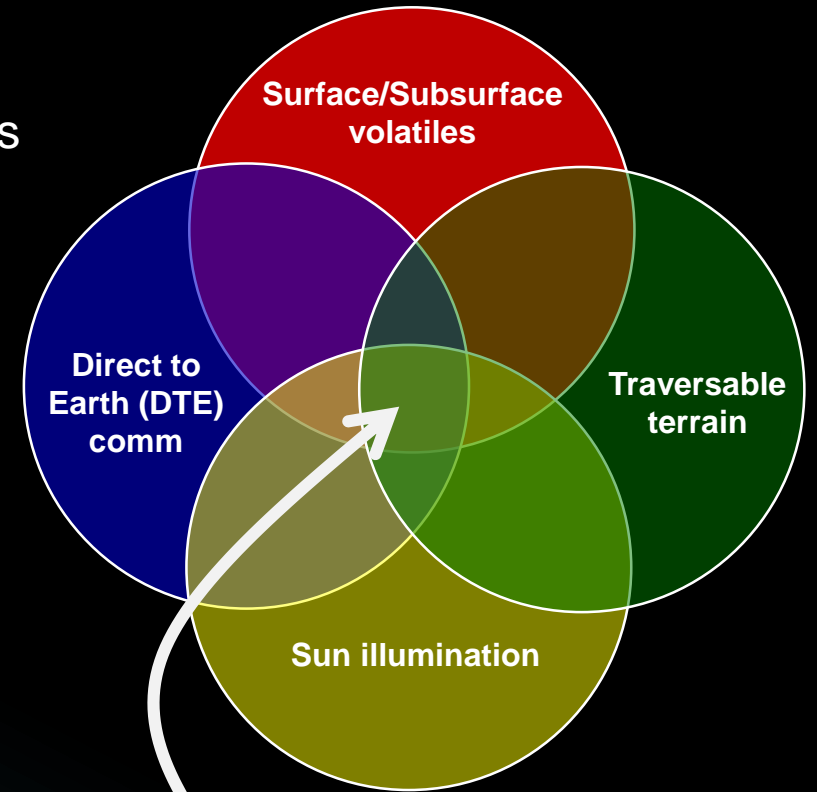
\* *ISR's are based on the predicted thermal stability of ice with depth*



# Key Requirements Paraphrased

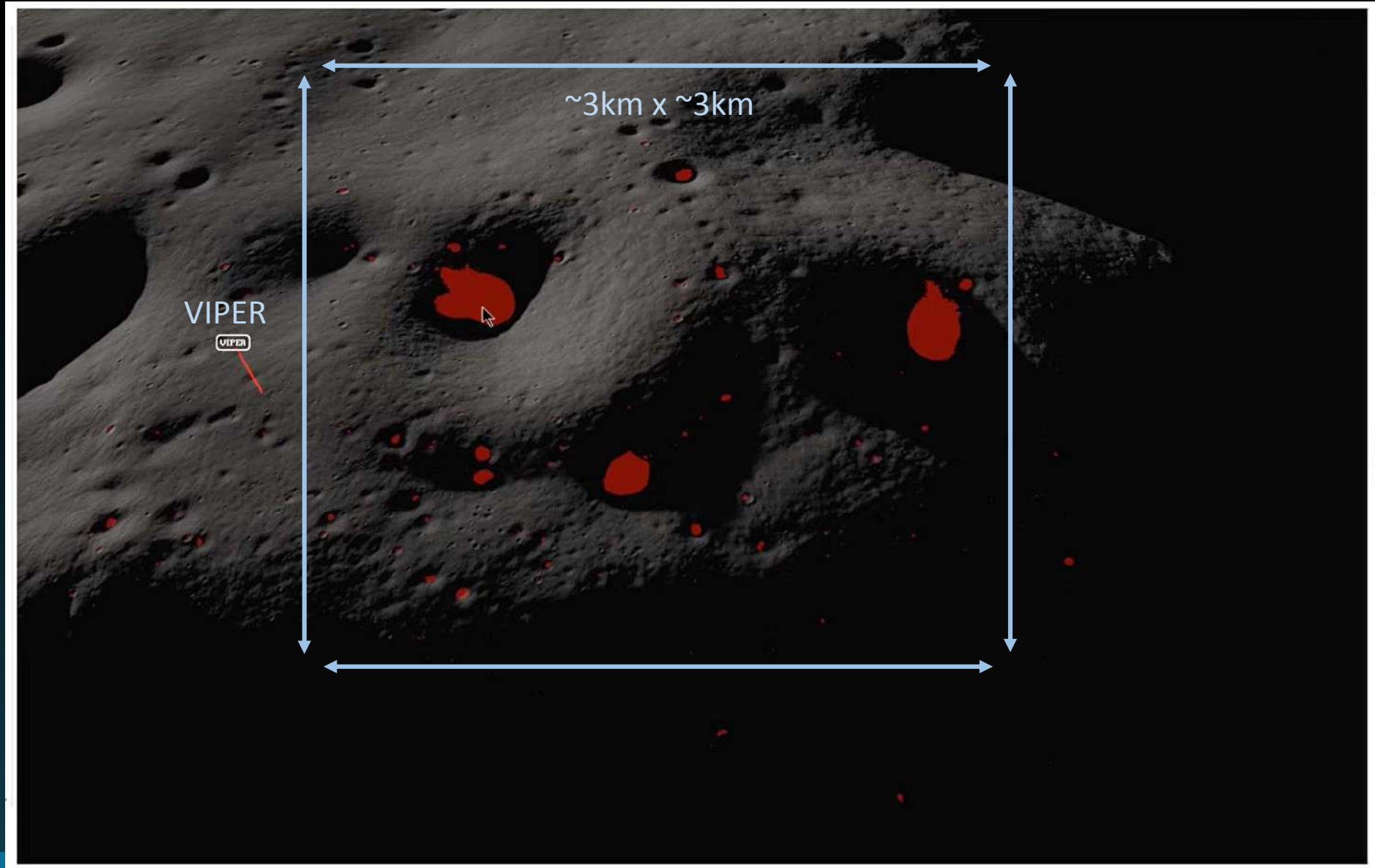
The Mission Site must meet these four criteria:

- Contain locations that plausibly contain surface & subsurface volatiles
- Provide sunlight for power (including when out-of-comms)
- Provide sufficient line-of-sight to Earth for radio communication
- Contain reasonable slopes for landing and traverse



Mission sites must meet all four criteria

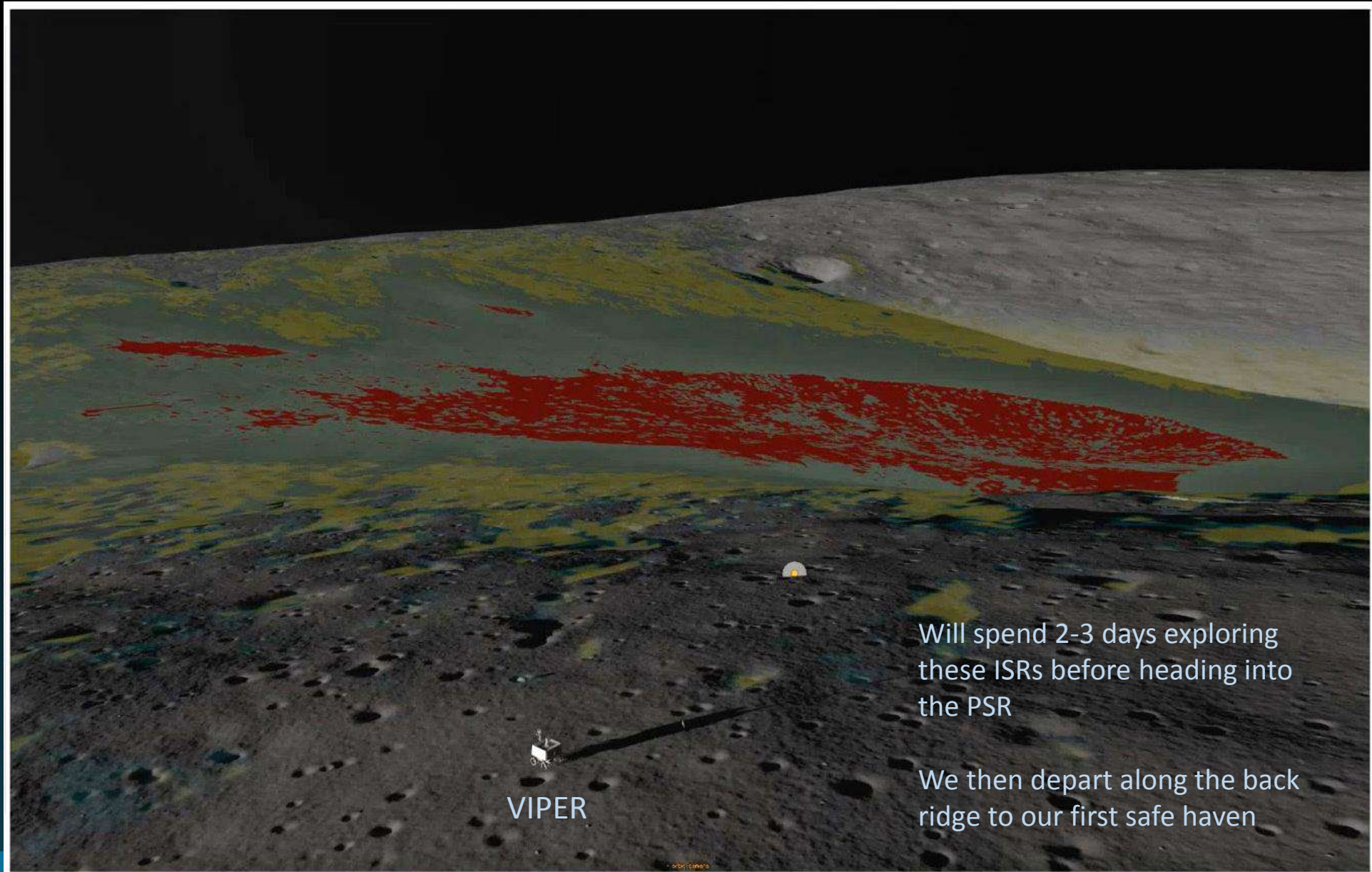
# VIPER Nobile mission region on the Lunar South Pole



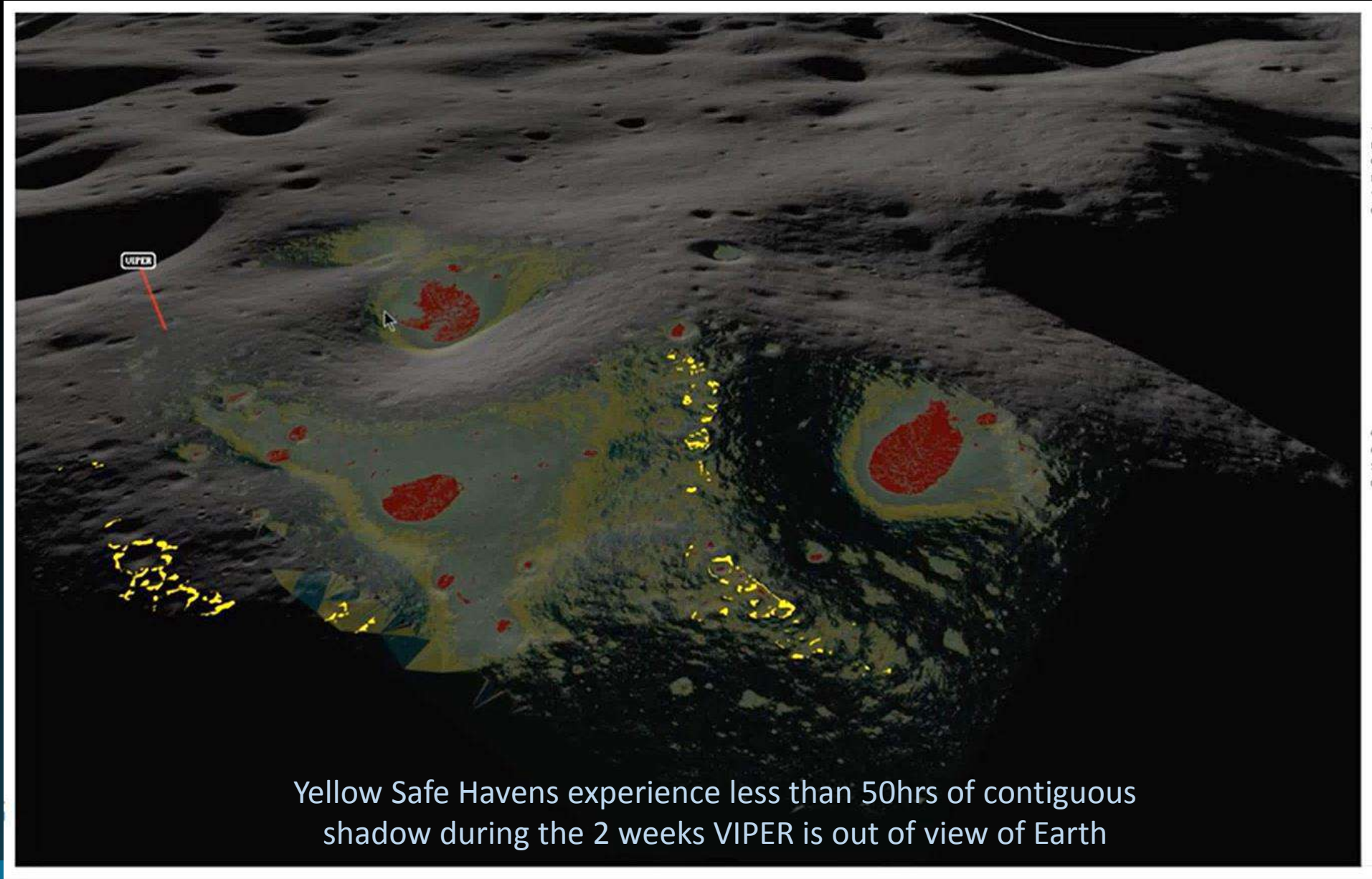
Screenshot  
from the  
VIPER  
Traverse  
Planning tool  
/M. Shirley



# VIPER and the four Ice Stability Regions (ISRs)



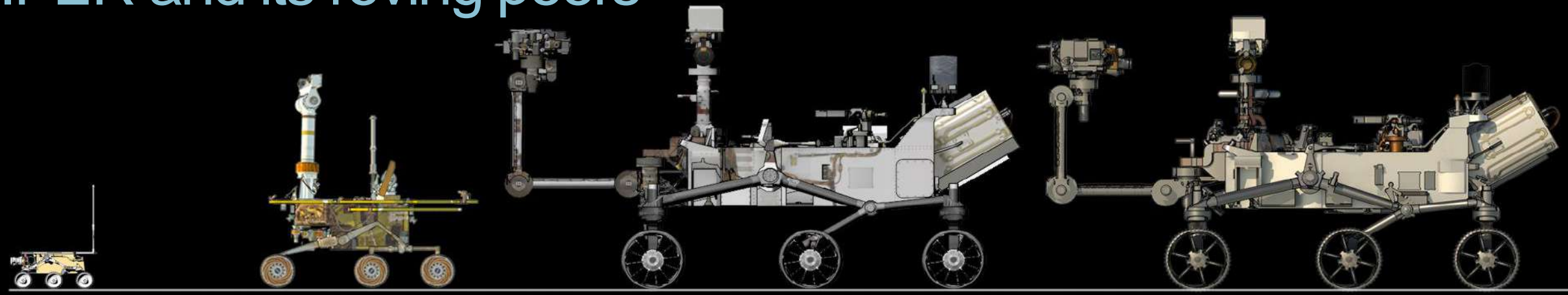
# Nobile region with Safe Havens



Yellow Safe Havens experience less than 50hrs of contiguous shadow during the 2 weeks VIPER is out of view of Earth

Screenshot  
from the  
VIPER  
Traverse  
Planning tool  
/M. Shirley

# VIPER and its roving peers

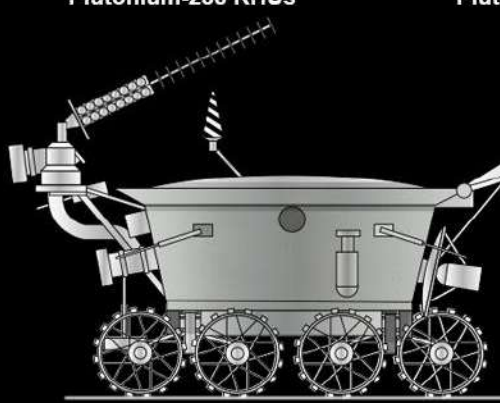


**Sojourner (1996)**  
 0.6m x 0.5m x 0.3m  
 11kg  
 Top Speed: 0.5cm/s  
 Plutonium-238 RHUs

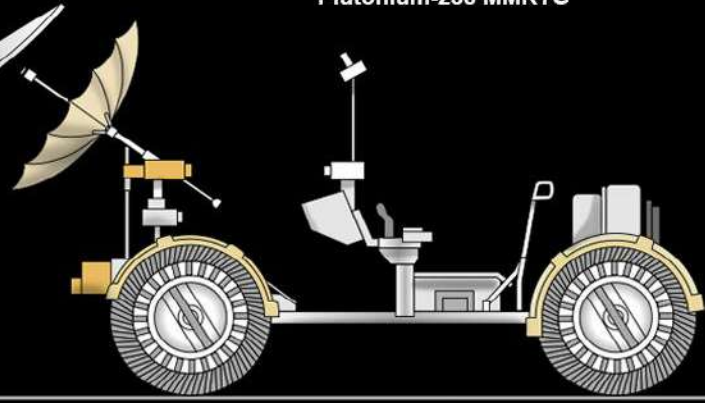
**Mars Exploration Rover (2004)**  
 1.6m x 2.3m x 1.5m  
 180kg  
 Top Speed: 5cm/s  
 Plutonium-238 RHUs

**Mars Science Laboratory (2011)**  
 3.0m x 2.8m x 2.1m  
 900kg  
 Top Speed: 4cm/s  
 Plutonium-238 MMRTG

**Mars 2020 Rover (2020)**  
 3.0m x 2.7m x 2.2m  
 1025kg  
 Top Speed: 4.2cm/s  
 Plutonium-238 MMRTG



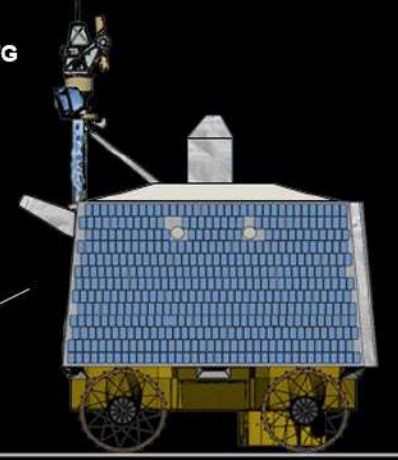
**Lunokhod 1 & 2 (1970/1973)**  
 2.3m x 1.6m x 1.5m  
 840kg  
 Top Speed: 55cm/s  
 Polonium-210 heat source



**Lunar Roving Vehicle (1971/1972)**  
 3.1m x 1.6m x 1.5m  
 210kg  
 Top Speed: 500cm/s  
 2 silver-zinc 36 volt batteries



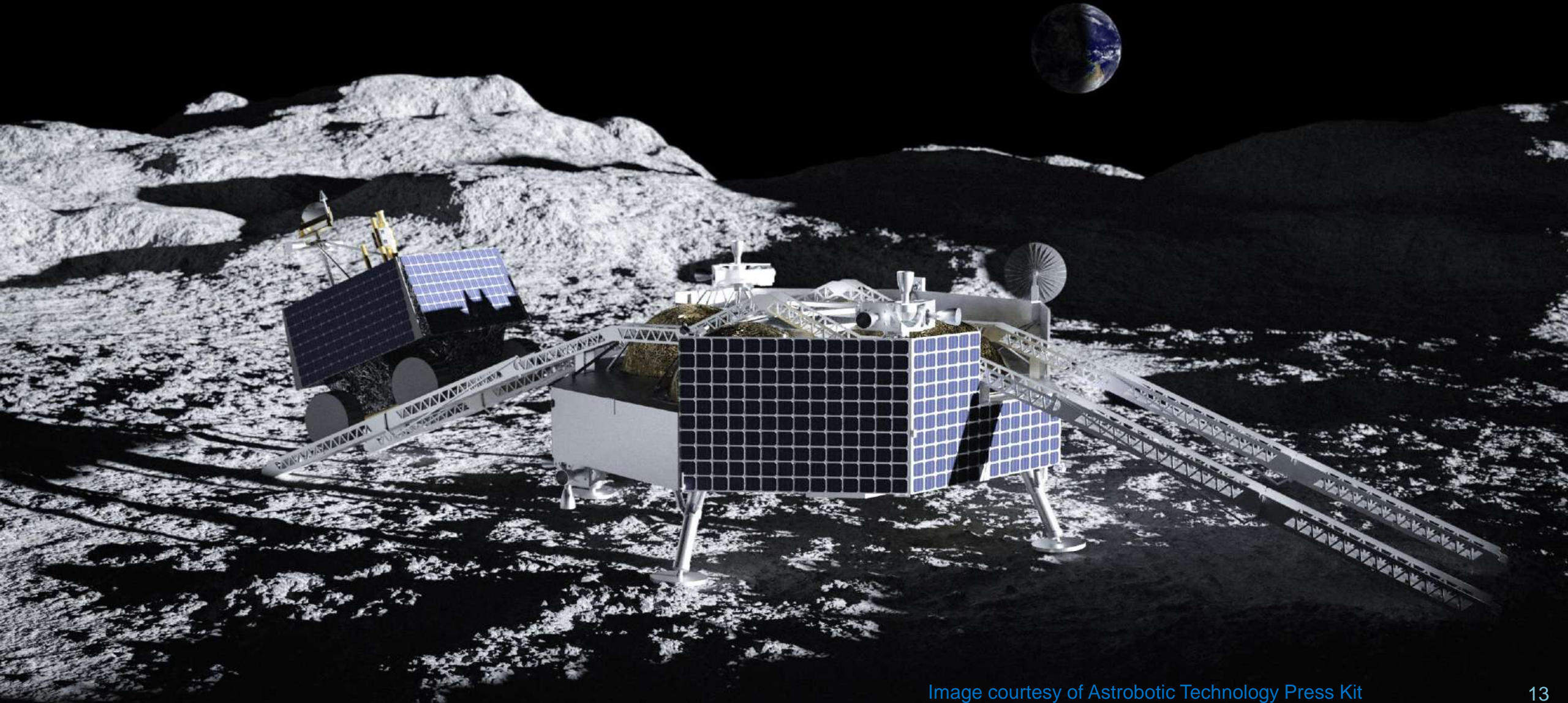
**Yutu (2013/2019)**  
 1.5m x 1.1m x 1.1m  
 140kg  
 Top Speed: 5cm/s  
 Plutonium-238 RHUs



**VIPER (2024)**  
 1.5m x 1.5m x 2.0m  
 430kg  
 Top Speed: 20cm/s  
 Electric heaters only



The NASA CLPS program has selected **Astrobotic Technology**, (Pittsburgh, PA, USA) for delivery of VIPER to the lunar pole in 2024 aboard their Griffin Lander



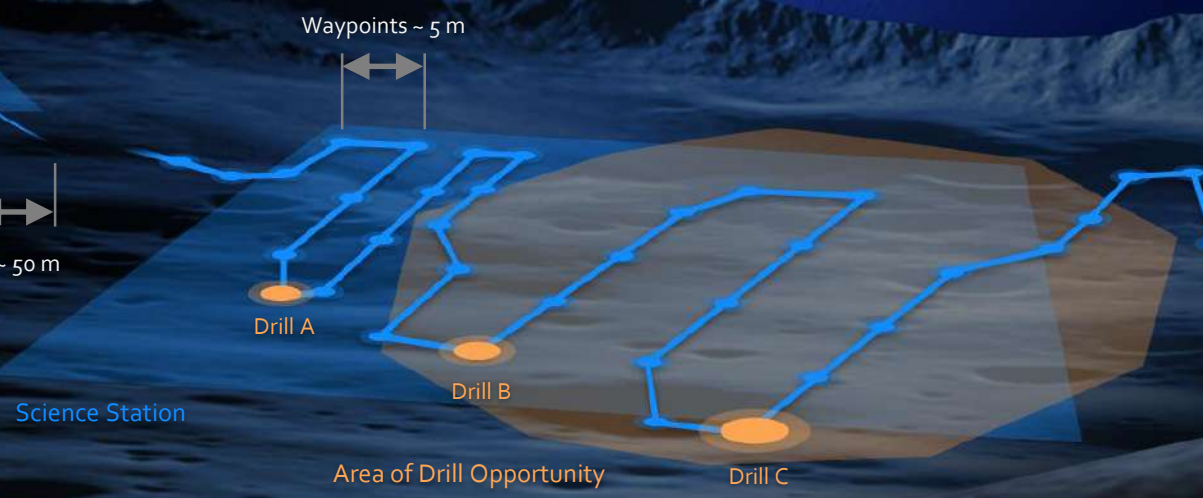
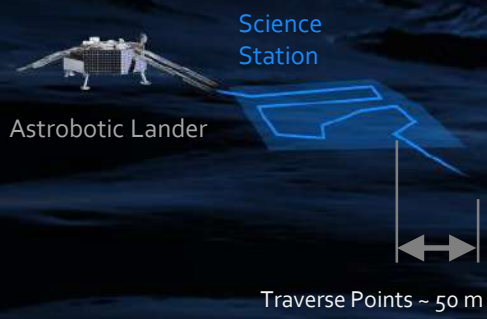
The image shows a close-up view of the lunar surface, characterized by numerous dark, circular craters of varying sizes. The surface is illuminated from the side, creating deep shadows and highlighting the rugged terrain. A solid, bright blue horizontal band runs across the center of the image, serving as a background for the title text.

# Operations

# VIPER Mission Systems

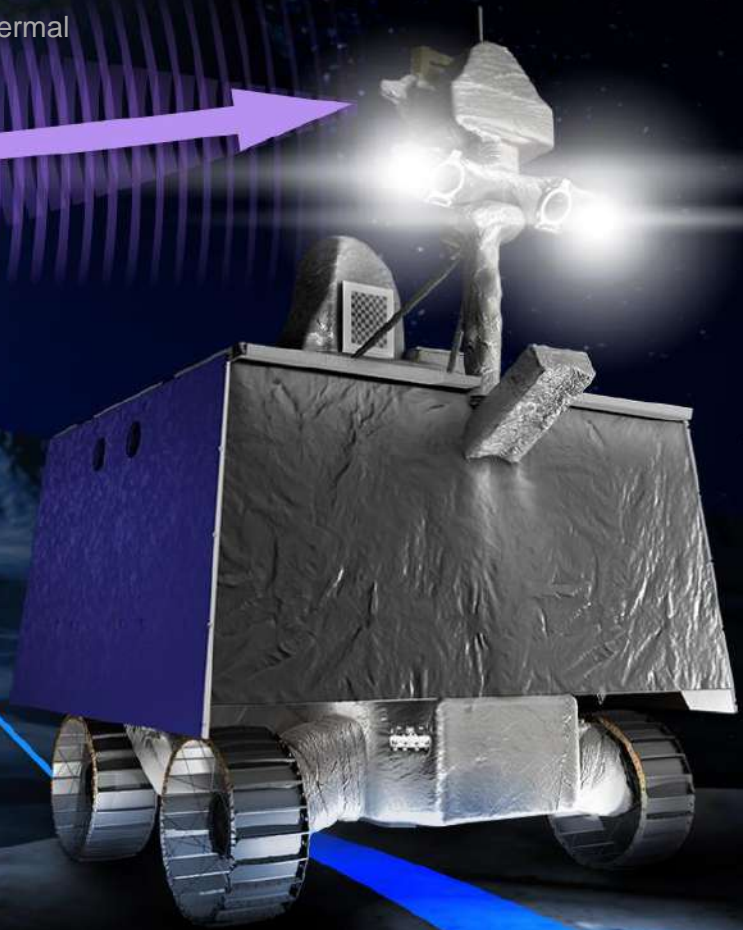
Rover Driving

Mission Monitoring

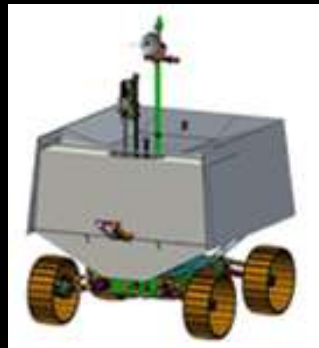


Nobile Crater Area, Moon South Pole

Low light angle at poles



# Ops Approach: VIPER vs. Mars Rovers vs. ISS



	VIPER	MER/MSL	ISS EVA Robotics
Comms	continuous	~once per Sol	continuous
Latency	6-25 sec	20-40 minutes	<=2 sec
Environment	unstructured	unstructured	engineered
Approach	waypoint commands	command sequences	direct teleoperation

# Surface Environment Key Operational Drivers

- Regolith/Rocks/Craters
  - Vehicle design to environmental spec
  - Drivers responsible to stay within vehicle mobility limits
  - Real-time science inputs
- Lighting
- Line of Site Comm
- Temperature



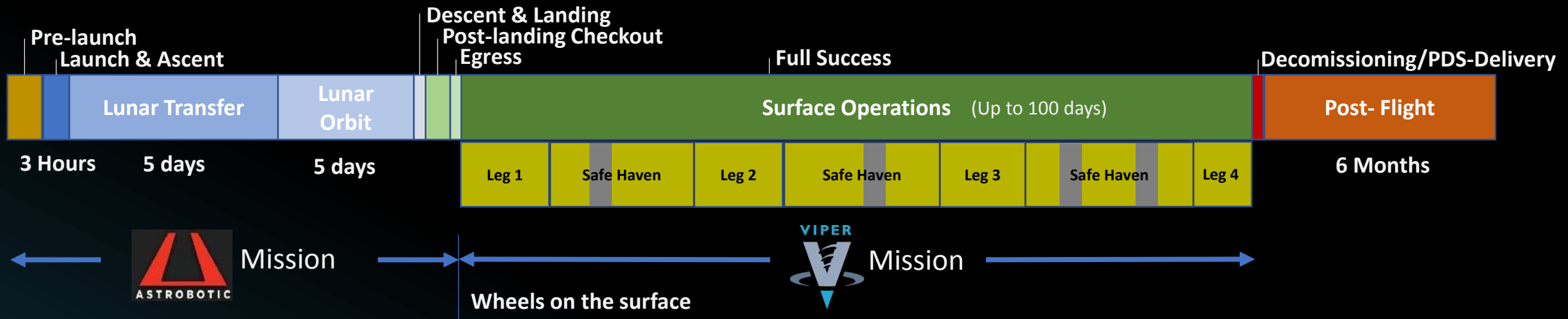
# Lighting & Comm at the S. Pole v Apollo



Courtesy ESA

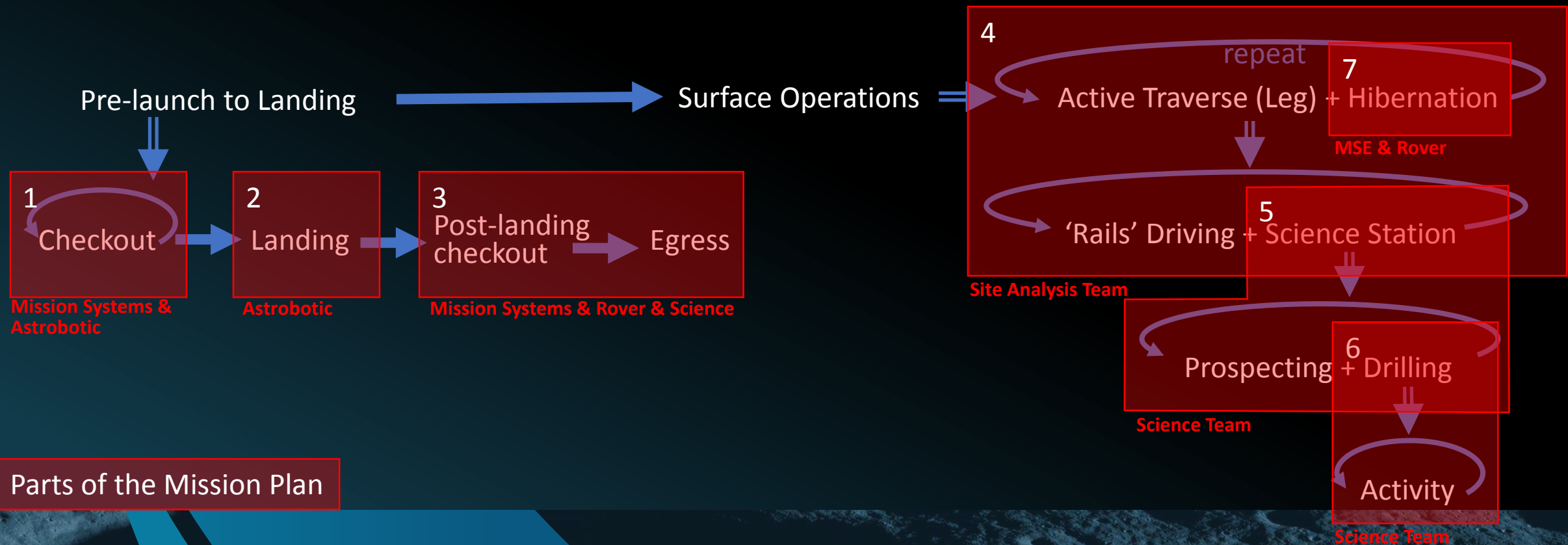
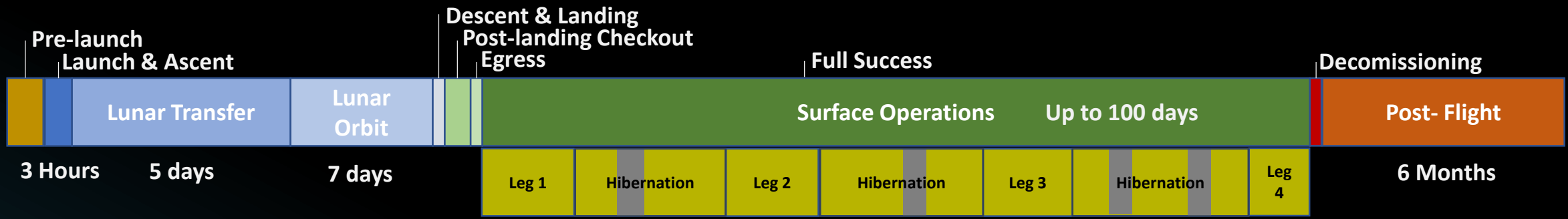


# VIPER Mission Phases



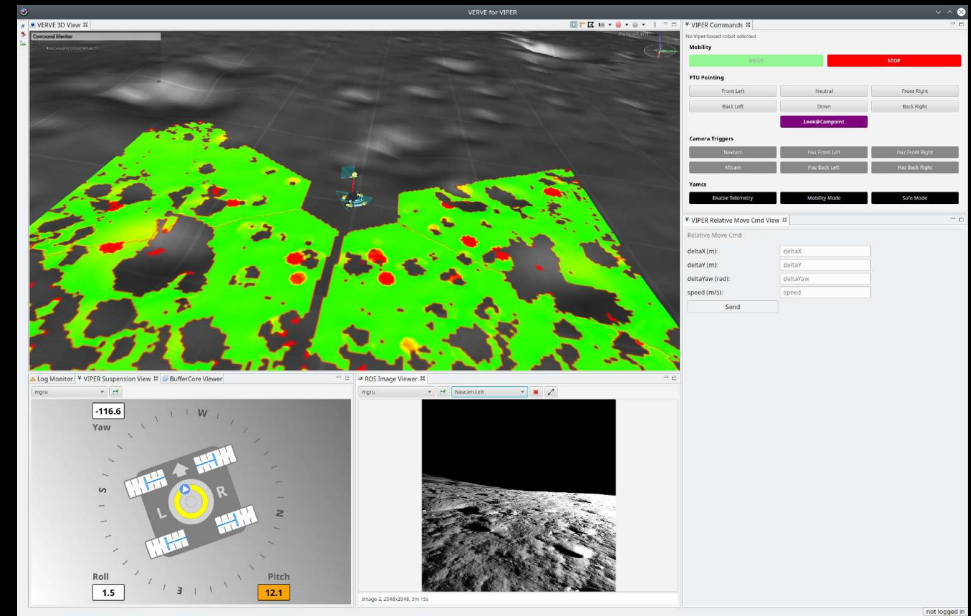
- Surface Ops consist of periods of activity (traverse “Legs”) and periods of inactivity (“Safe Haven”)
  - Traverse Legs are in view of Earth and sun (except for planned shadow (PSR) ops (<8 hours))
  - Safe Havens are NOT in view of Earth, but in view of sun, with periods of sun shadow (<70hrs hibernation)
  - Lunar Day = one Traverse Leg + one Safe Haven
- Mission Success:
  - Minimum Mission Success planned by end of Lunar Day 1
  - Full Mission Success planned by end of Lunar Day 2
  - Lunar Days 3 (and 4 if possible) offer either contingency time, or improved science data

# Mission Plan Structure



# Driving

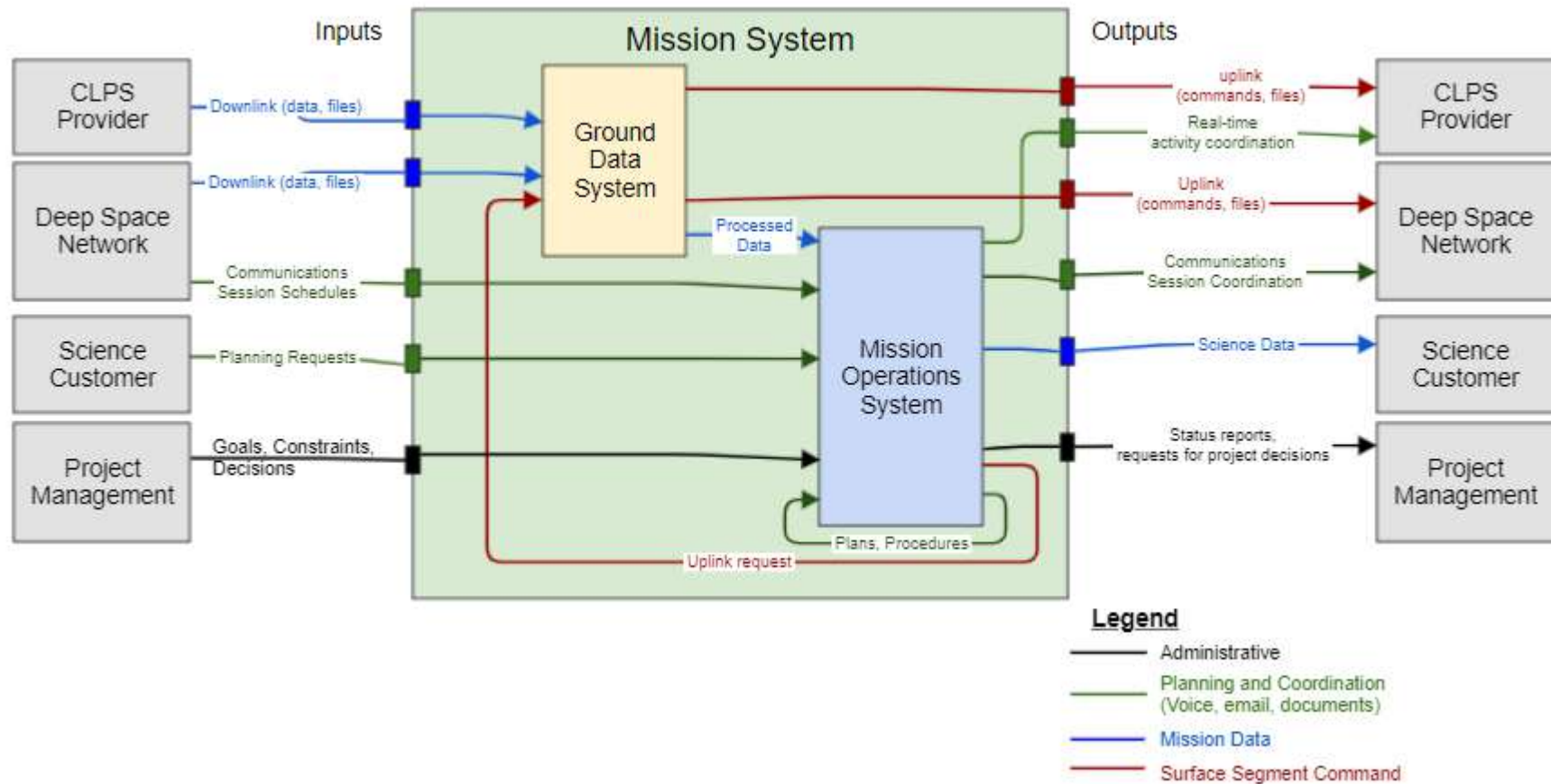
- Continuous communication with Earth
    - Low bandwidth: 256 kbps
    - Latency: 6-10 sec nominal round-trip
  - Limited computing, smarts on the rover
  - Waypoint Driving
    - Command rover to drive to a waypoint ~4.5m away, based on lookahead distance and predictability of mobility (e.g. wheel slip)
    - Rover onboard control achieves the waypoint, minus wheel slip or other errors
  - Need to drive using data from the rover
    - Driver/co-driver responsible for safe driving
    - Real-Time Science to interpret terrain when needed (Science Station)
- Not continuous “rate control” input, e.g. joystick or gas pedal
- Not sol command sequences, e.g. MER/MSL rovers



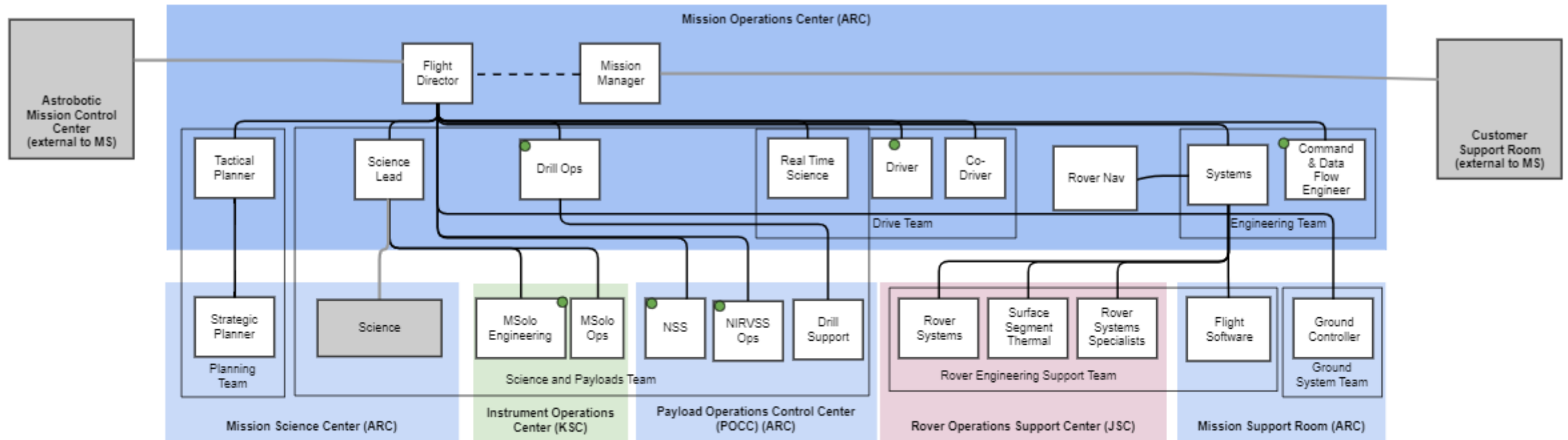
The image shows a close-up view of the lunar surface, characterized by numerous dark, circular craters of varying sizes. The surface is illuminated from the side, creating deep shadows and highlighting the rugged terrain. A solid blue horizontal band runs across the center of the image, serving as a background for the title text.

# The Mission System

# MS Architecture



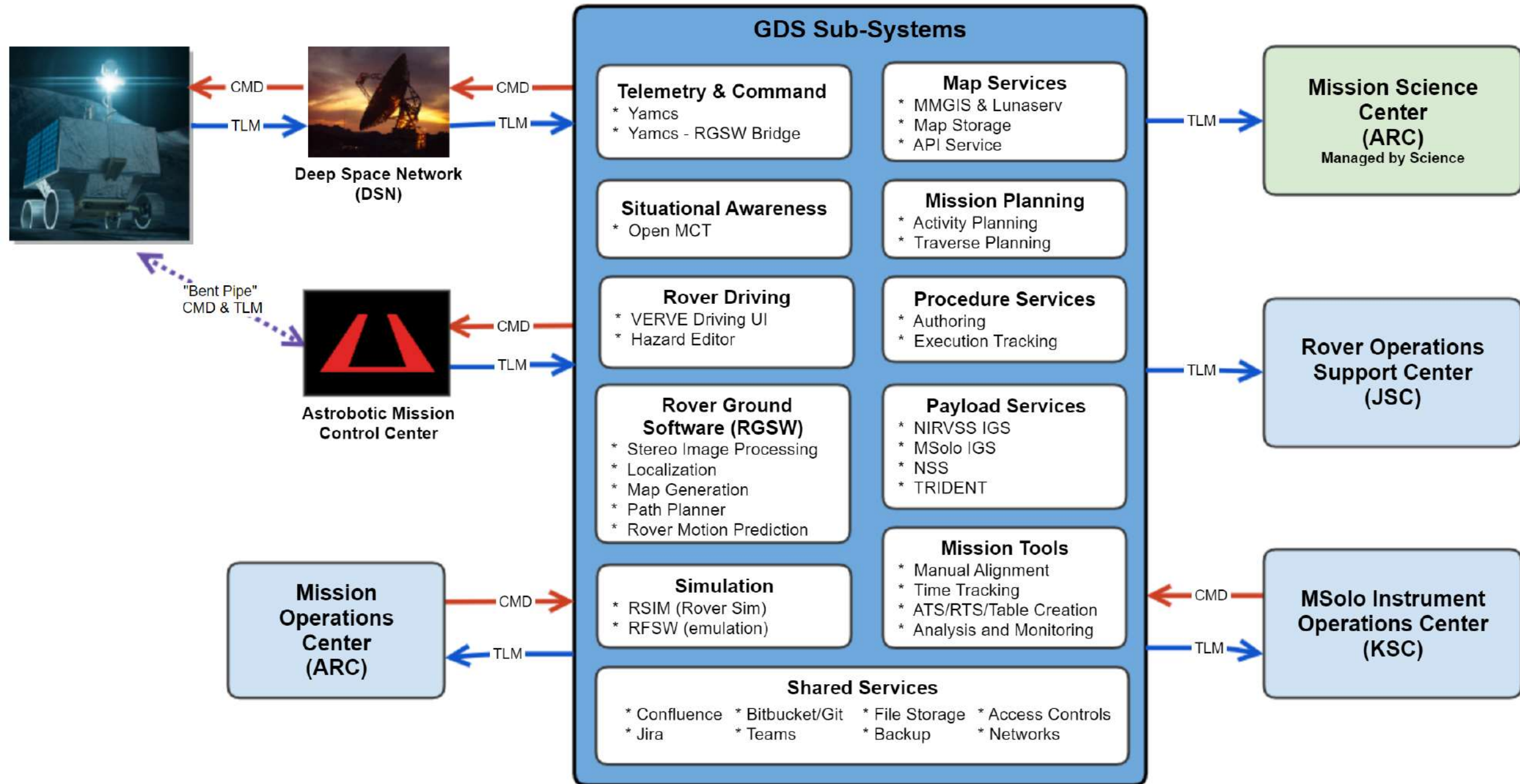
# Mission Operations Team



## Legend

- NASA ARC MS Facility
- NASA JSC MS Facility
- NASA KSC MS Facility
- Non-MS Facility
- Command Capability
- Primary Real-Time communication and decision flow
- External Interface
- near-real-time reporting and stakeholder coordination

# GDS Overview Diagram







# Test & Training

## Thread Tests

- Verify/validate selected parts of the MOS/GDS, initial process run throughs, does not require a full flight team

## Integrated Simulations

- Demonstrates ability to execute activities
- Mission simulations of portions of the nominal timeline with contingencies introduced in some circumstances

## End to End Testing

- Test end-to-end command and data flow, execute test as you fly activities

## Operational Readiness Testing

- Demonstrates ability to perform the mission
- Time realistic, final MOS training and cert, operational readiness

## Mission Rehearsals

- Maintain and enhance operational proficiency

## Observing Ops

- Console time as an observer at JSC and JPL

The image features a dark, cratered lunar surface, likely from the Apollo 16 mission, showing various sized craters and shadows. A solid blue horizontal band is overlaid across the center of the image, containing the word "Software" in white text.

# Software

# Open MCT Integrated Visualization

- Realtime and Historical Plots/Tables
- Rover Image Display
- 2D and 3D Map Layer Display
- Instrument Data Heatmaps
- Rover Traverse Plans
- Activity Timelines
- Events and Limit Violations
- Fault Manager
- Procedure Library
- Shared Procedure View
- User Annotations
- Console and User Notebooks
- Customized Layouts

The screenshot displays the Open MCT interface for a rover mission. The browser address bar shows the URL: `banner.ndc.nasa.gov/stable/#/browse/mine/cd333b91-e464-4bf8-9847-8b419e1425d2/12c0e46c-0295-4bfa-9e8e-1afb1dbb0313/64b11045-6e62-433d-8d2b-699fd01442fa/6bb2c571-20ac-49a...`. The interface is titled "OVERVIEW" and shows the following components:

- Top Panel:** Displays system status (UTC 21:34:35, MET +20D 03:55:43, SET +17D 03:55:04) and control buttons for PWR, BATT, COMM, MSOLO, TRID, NRVS, NSS, and DRIVING.
- Left Panel:** A "BROWSE" tree view showing a hierarchy of data sources, including "WheelOdometry", "commandCount", "deltaTime", "deltaXr", "deltaYaw", "deltaYr", "enabledFlag", "errorCount", "houseKeepingCount", "inhibitedFlag", "messageCount", "padding1", "suspElev", "suspPitch", "suspRoll", "timeConstraintErrorCount", "wheelInputErrorCount", "wVelocity", "xVelocity", and "yVelocity".
- Center Panel:** A 3D map of the lunar surface (CH MMGIS) with a rover's path and several red-outlined polygonal regions. A scale bar indicates 0 to 1000m.
- Right Panel:** A 3D model of the rover with a data table showing: 

driveSpeed	0.2
heading	-59.89
wVelocity	0.0000
xVelocity	-0.0000
yVelocity	-0.0000
suspElev	0.328
suspPitch	-0.000
suspRoll	0.000

 Below the model are two camera views: "Navcam\_left\_image" and "Navcam\_right\_image".
- Bottom Panel:** A "Plan View Standin 1" timeline from 04:00 to 11:00 UTC. It shows activities for ROVER (Rails Driving, Shadow\_Rails Driving) and VIPER (Shadow\_Panorama, Shadow\_Subsurface Assay - 40 to 60cm, Shadow\_Drill Location Targeting, Shadow\_Drilling Preparation, Shadow\_Subsurface Assay - Percussion, Shadow\_Drill Temperature Pause). A status bar at the bottom shows "Updated 2021-06-21 21:35:35.442Z" and a timer at "00:01:00".
- Far Right Panel:** A "Goals vs. Actual" section with four line graphs: "distanceToGoal", "headingToGoal", "goalX", and "goalY". Each graph compares a goal (teal) with actual values (green) over time.

# VERVE

- Orient Pan Tilt Unit
- Nav/Aft/Haz image commands
- 3D environment visualization
- Hazards (w/manual editor)
- Open MCT (embedded)
- View desired traverse
- Set waypoints
- View predicted path
- Command rover motion

*Visual Environment for  
Remote Virtual  
Exploration*

The screenshot displays the VERVE software interface, titled "VERVE for VIPER". The main window, "VERVE 3D View", shows a 3D visualization of a rover on a lunar surface. The terrain is rendered with a color-coded hazard map, where green and yellow indicate low to medium hazard levels, and red and black indicate high hazard areas. A small rover model is positioned on the surface, and a predicted path is visible. The interface includes several control panels:

- VIPER Commands:** A "Mobility" section with a green "GO" button and a red "STOP" button. Below it, a "PTU Pointing" section with buttons for "Front Left", "Neutral", "Front Right", "Back Left", "Down", and "Back Right", along with a purple "Look@Campoint" button.
- Camera Triggers:** A section with buttons for "Navcam", "Aftcam", "Haz Front Left", "Haz Back Left", "Haz Front Right", and "Haz Back Right".
- Yamcs:** A section with buttons for "Enable Telemetry", "Mobility Mode", and "Safe Mode".
- VIPER Relative Move Cmd View:** A section for sending relative movement commands, with input fields for "deltaX (m)", "deltaY (m)", "deltaYaw (rad)", and "speed (m/s)", and a "Send" button.
- Log Monitor:** A panel showing "VIPER Suspension View" and "BufferCore Viewer".
- ROS Image Viewer:** A panel showing a live video feed from the "Navcam Left" camera, displaying a grayscale image of the lunar surface. Below the image, it shows "Image 2, 2048x2048, 3m 15s".

At the bottom right of the interface, it says "not logged in".

# Open Source

## Open Mission Control Technologies - Open MCT

### Info

<https://nasa.github.io/openmct/>

From info site, click on Try It Now

### Code

<https://nasa.github.io/openmct/>

**openMCT** ABOUT GETTING STARTED CONTRIBUTING DOCUMENTATION TRY IT LIVE

Open MCT is a next-generation mission control framework being developed at NASA's Ames Research Center in Silicon Valley, in collaboration with the Jet Propulsion Laboratory.

Web-based, for desktop and mobile.

**TRY IT NOW**  
**GET SOURCE**

**HOW IS NASA USING OPEN MCT?**  
Software based on Open MCT is being used for mission planning and operations in the lead up to the *Resource Prospector* mission at NASA's Ames Research Center, and as a data visualization tool at the Jet Propulsion Laboratory.  
[FIND OUT MORE](#)

**HOW CAN YOU USE OPEN MCT?**  
Open MCT can be adapted for planning and operations of any system that produces telemetry. While Open MCT is developed to support space missions, its core concepts are not unique to that domain. It can display streaming and historical data, imagery, timelines, procedures, and other data visualizations, all in one place.  
[LEARN MORE](#)

**HOW TO CONTRIBUTE**  
We are looking for enthusiastic people who want to help contribute to NASA's exploration of the solar system. Are you a student, professional software developer, or just a space enthusiast? We'd love to hear your ideas for new features or ways of visualizing data. If you're a coder you can help us develop new features or capabilities, and fix bugs.  
[GET MORE INFO](#)

# Open Source Benefits

Collaboration that works

Use, adopt, make it your own, contribute

No ownership issues

Instant access

A screenshot of a GitHub repository page for 'openmct'. The repository is titled 'A web based mission control framework' and is located at 'https://nasa.github.io/openmct/'. The repository statistics show 5,339 commits, 247 branches, 23 releases, and 38 contributors. The current branch is 'master'. The repository is owned by 'charlesh88 and psarram'. The latest commit is 'e19ce4a' from 5 days ago. The repository contains several files and folders, including 'docs', 'example', 'platform', 'scripts', 'src', 'gitignore', 'jscsrc', 'jshint', 'npmignore', 'API.md', 'CONTRIBUTING.md', 'LICENSES.md', 'Profile', 'README.md', 'app.js', 'bower.json', 'build-docs.sh', 'circle.yml', 'gulpfile.js', 'index.html', 'jdoc.json', 'karma.conf.js', 'openmct.js', 'package.json', and 'test-main.js'. Each file entry includes a brief description of the file's content and the date of the last commit.

File/Folder	Description	Last Commit
docs	[Documentation] Add security guide (#1900)	4 months ago
example	Handle string states (#2019)	21 days ago
platform	[Frontend] Adds overflow scrolling to J-view-section (#2039)	5 days ago
scripts	[Copyright] Update copyright year across platform code references	9 days ago
src	[Copyright] Update copyright year across platform code references	9 days ago
gitignore	Add functionality to allow users to add hideParameters to the url, wh...	9 months ago
jscsrc	[Code Style] Add JSHint rules	2 years ago
jshint	Disabled late definition check for functions	4 months ago
npmignore	[Build] Add npmignore to allow packing	2 years ago
API.md	Summary widget telemetry provider (#1843)	a month ago
CONTRIBUTING.md	[Documentation] Edit for style	a year ago
LICENSES.md	Include 2017 in copyright. #1517	a year ago
Profile	Merging in latest github/master	3 years ago
README.md	prepare -> prepublish	5 months ago
app.js	Revert "[proxyUrl] pass URL parameters to proxied URL"	2 years ago
bower.json	Lock filesaver version (#1956)	2 months ago
build-docs.sh	[Licenses] Update copyright year to 2017	a year ago
circle.yml	Bump Node Version	23 days ago
gulpfile.js	adds v8-compile-cache	5 months ago
index.html	[Autoflow] Rewrite Autoflow Tabular using new APIs (#1816)	6 months ago
jdoc.json	Squashed commit of the following:	2 years ago
karma.conf.js	[Plugin] Add imported root plugin (#1784)	3 months ago
openmct.js	new-plot import (#1557)	3 months ago
package.json	Enterprise-galactica (#1993)	a month ago
test-main.js	d3 selection filepath changed (#1898)	4 months ago

<https://github.com/>

The image features a dark, cratered lunar surface, likely from the Apollo 16 mission, showing numerous craters of various sizes. A solid teal horizontal band is superimposed over the center of the image. The word "Agile" is written in white, sans-serif font within this band.

# Agile



# VIPER: A new approach to planetary missions

- VIPER follows **NPR 7120.8 with augmentation**
  - Apply ARC and NASA engineering processes and standards with **tailoring** to improve efficiency while reducing cost and schedule
  - VIPER Review Team (VRT) provides streamlined approach to continuously review the project
  - VIPER does not have a formal risk classification, but is “like” Class D
- Adapt and use **NPR 7120.5 “constructs” as needed**
  - Control plans should be baselined and used when needed
  - Reviews should focus on content, not on slide preparation and travel
  - Focus on what provides value & rigor, not just satisfying process
- Rapid design and development
  - Apply agile software development practices as much as possible
  - Use iterative design and test rather than lengthy “waterfall” process
  - Extensive use of collaboration tools



# In the Beginning

Software, years before VIPER

Delivery every 6 months

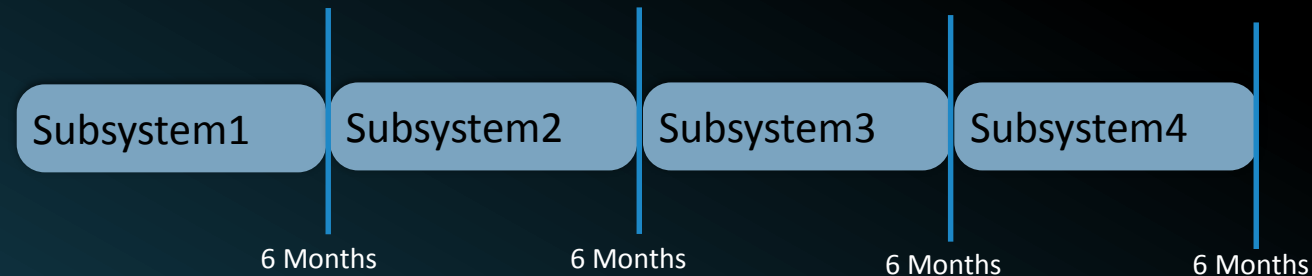
The 6 month delivery cycle created too much time for customer expectations to diverge from what we were building

Customers needed to see

Progress difficult to measure

Long and formal design specs

Too much time talking, having meetings, writing documents, not enough time doing



# Agile Software Sprint example, external customer

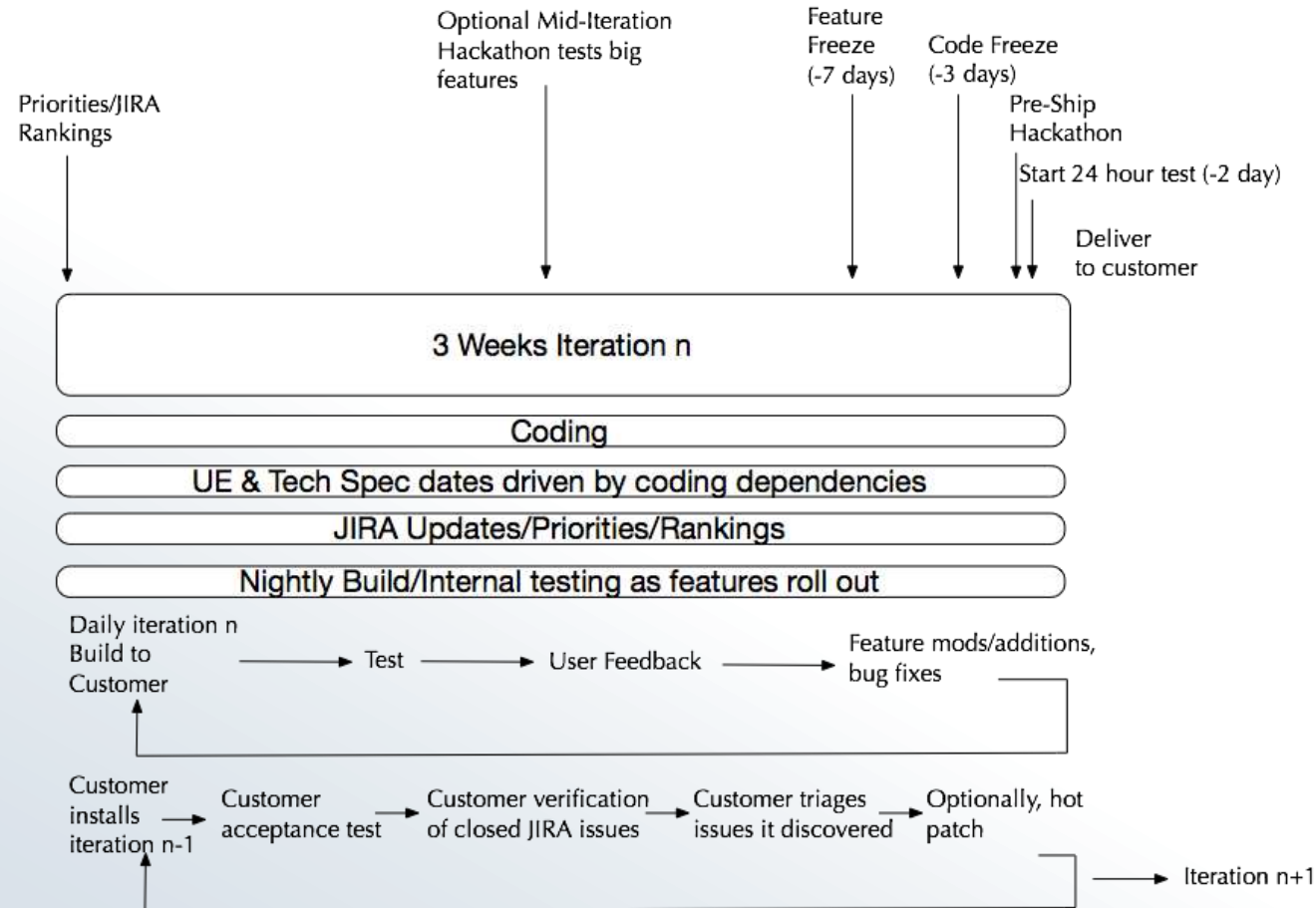
Agile Tailored for our team

Deliver to customer every 3 weeks

Nightly build

Release every 3 months

Emphasis on constant interaction and use



# Key attributes of our tailored agile process for software

The measure of progress is working code

Demonstrations, not presentations

Constant use of the software in a relevant environment, such as mission simulations

Visible progress - nightly or continuous builds

Ship on time, features that are not ready go into the next sprint or release

Validation using both QA and customer use in context

The image shows a dark, cratered lunar surface. A prominent horizontal band of bright blue color runs across the center of the image. The text 'Agile Mission Operations' is centered within this blue band in a white, sans-serif font.

# Agile Mission Operations

# Agile for the Mission System

## Mission System (MS)

Mission Operations System (MOS) – people and processes

Ground Data System (GDS) – Software, hardware, infrastructure, control center

Agile well established in GDS software

Are there potential benefits from agile in extending it beyond the GDS, to the MOS?

Emphasis on doing rather than documenting

Continuous integration and interaction

Visibility of the system, knowing where you are

Fail early so you can succeed sooner

# Tailored Agile

## MOS

Assessment of capability through demonstrated execution of mission capability

Maturation, evaluation and iterative development of the system through continuous use

Early and frequent builds and tests

## Software

The measure of progress is working code

Demonstrations, not presentations

Interactions and use over meetings

Continuous visibility of progress (continuous builds)

Sprints ship on time, features that are not ready go into the next sprint

# Words to Remember

"Test as you fly, fly as you test"

"Say it then sim it"

# Agile Methods from VIPER: Dev Sims

Dev Sims = Development Simulations

Location: Lab

Enables use of the latest running software build

Try the software in a controlled usage scenario, such as driving

Get data to help answer targeted questions

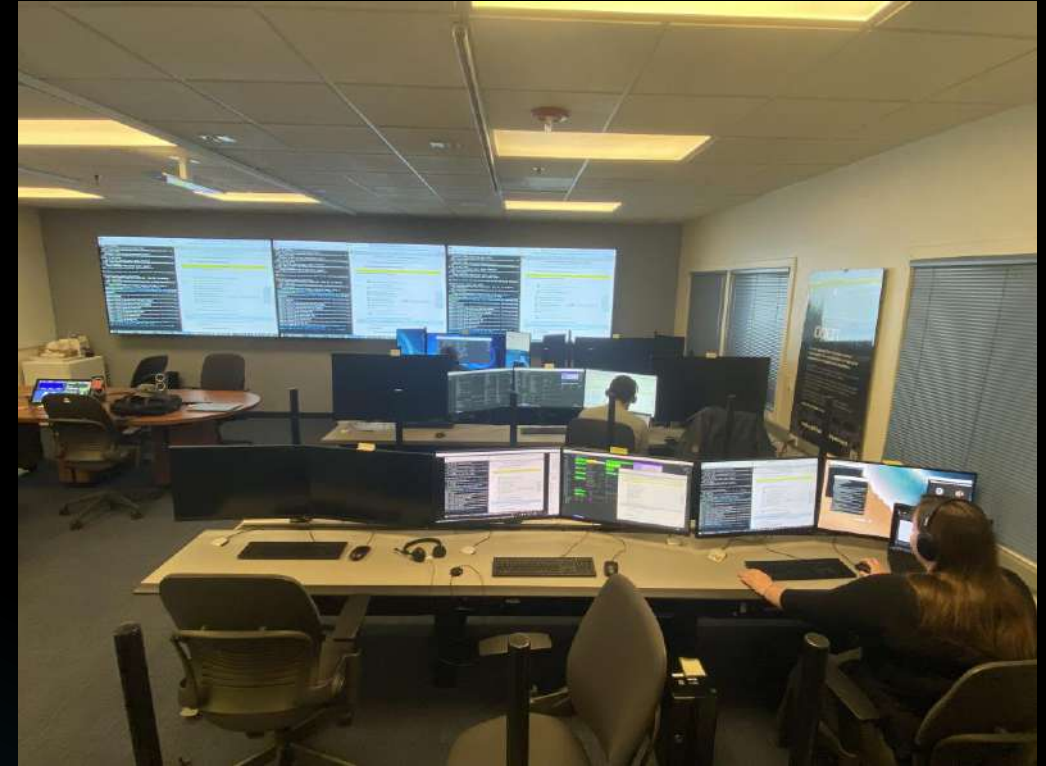
Example: Driver decision time between command cycles

Find things you may not be looking for

Example: Aft driving issues

Develop and mature the system by using it

Note: This requires a simulator and a budget





# Agile Methods from VIPER: Ops Products

A traditional waterfall ops product cycle might consist of a small number of major releases

Draft

Engineering Release

Test & Training Release

Mission Release

These releases would each come with a release review, which would involve document review, signoff and likely a set of presentations

Ops products would then be updated during test and training, culminating in a mission release

# Agile Methods from VIPER: Ops Products

We develop ops products based on dependencies from test & training activities

For each activity, such as a simulation, or an integration and test activity, develop the ops products required to support it

Rather than a large procedure drop, the team focuses on a limited number of products and matures those products in a structure similar to a software sprint

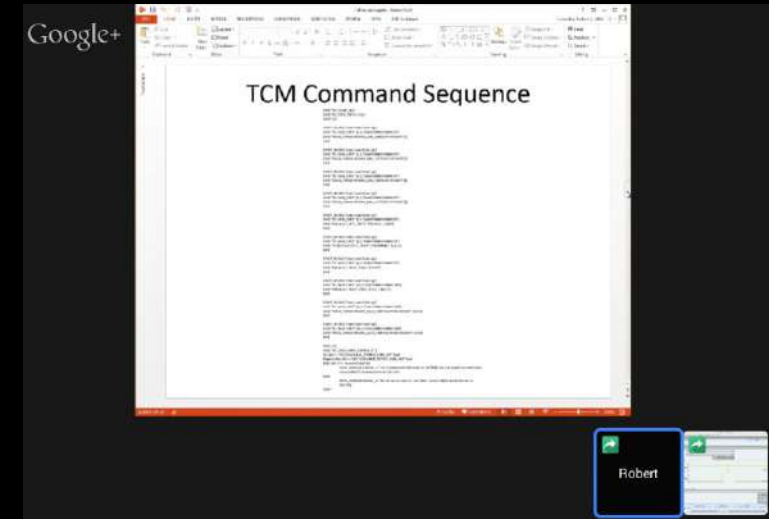
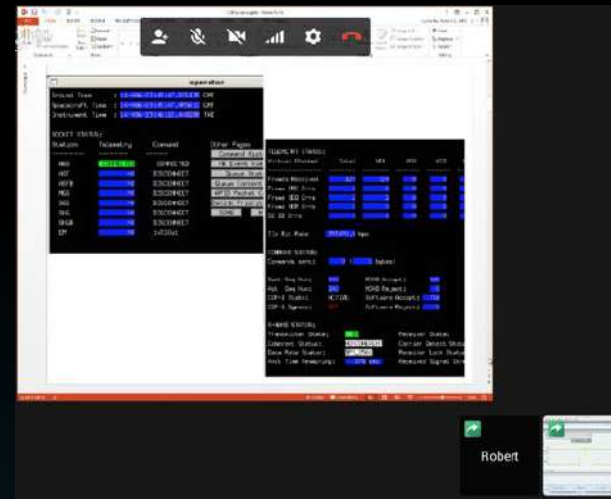
Draft → walkthrough in lab → walkthrough in control center → simulation → iterate and repeat

# Procedure walkthrough

You don't have to wait for mature procedures and software to design your ops products

This is an example from the Resource Prospector Project of an early walkthrough

We used Powerpoint, Google Hangouts and prototype timeline software



# Test & Training

Train the team, test the system

Location: Control Center

Get ready for the mission

- Train and certify the operations team

- Verify and validate the Ground Data System (GDS)

Agile elements of test & training

- Structure, iterations

- Rapid response to change in process, products and software (dependent on build and installation cycle)



# Parting Thoughts

Create a culture of doing, rather than a culture of documents and meetings

Agile development, for both software and mission operations processes, is compatible with mandated system engineering processes