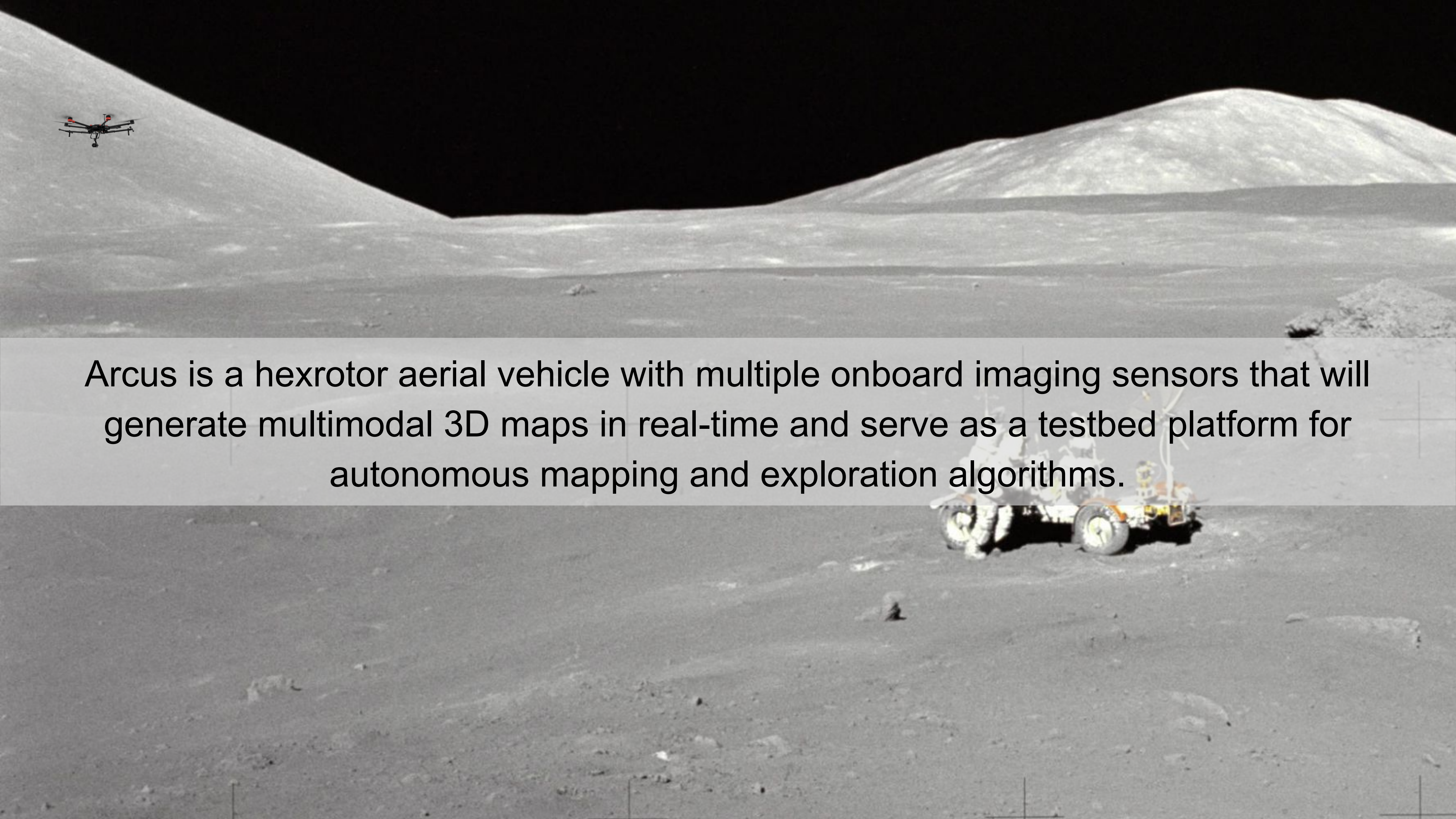
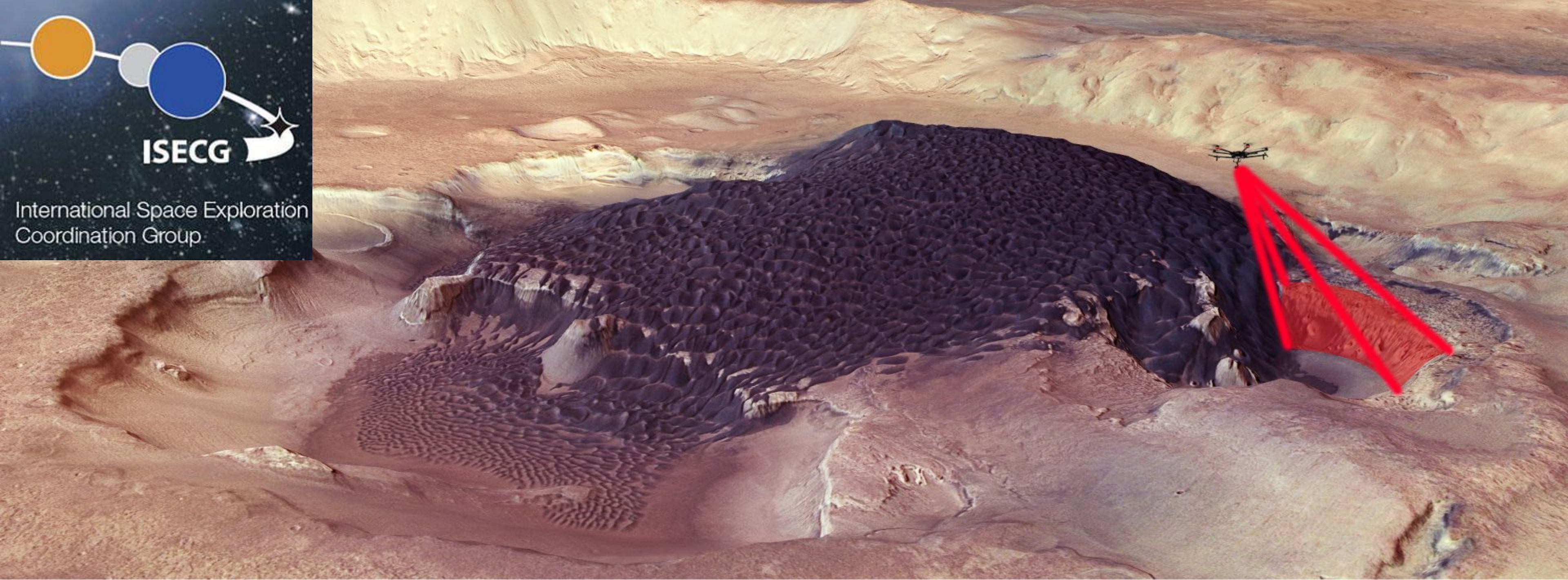
# System Development Review

**Team B: Arcus**

Clare Cui, Maitreya Naik, Angad Sidhu, Logan Wan



Arcus is a hexrotor aerial vehicle with multiple onboard imaging sensors that will generate multimodal 3D maps in real-time and serve as a testbed platform for autonomous mapping and exploration algorithms.



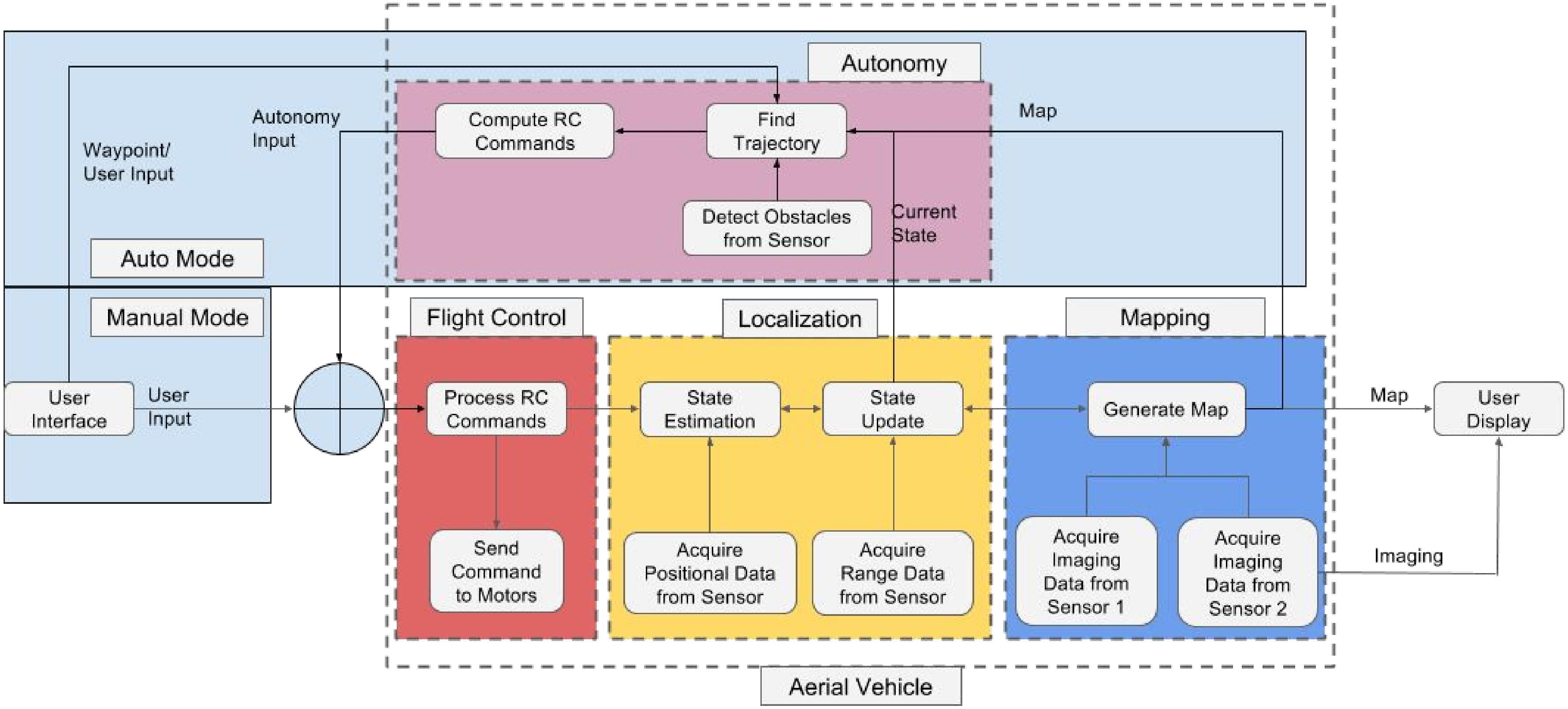
# Concept: Extraplanetary Exploration

- Space shuttle lands on planet
- UAV deploys and autonomously maps unknown environment
- Uses sensors to further investigate points of interest

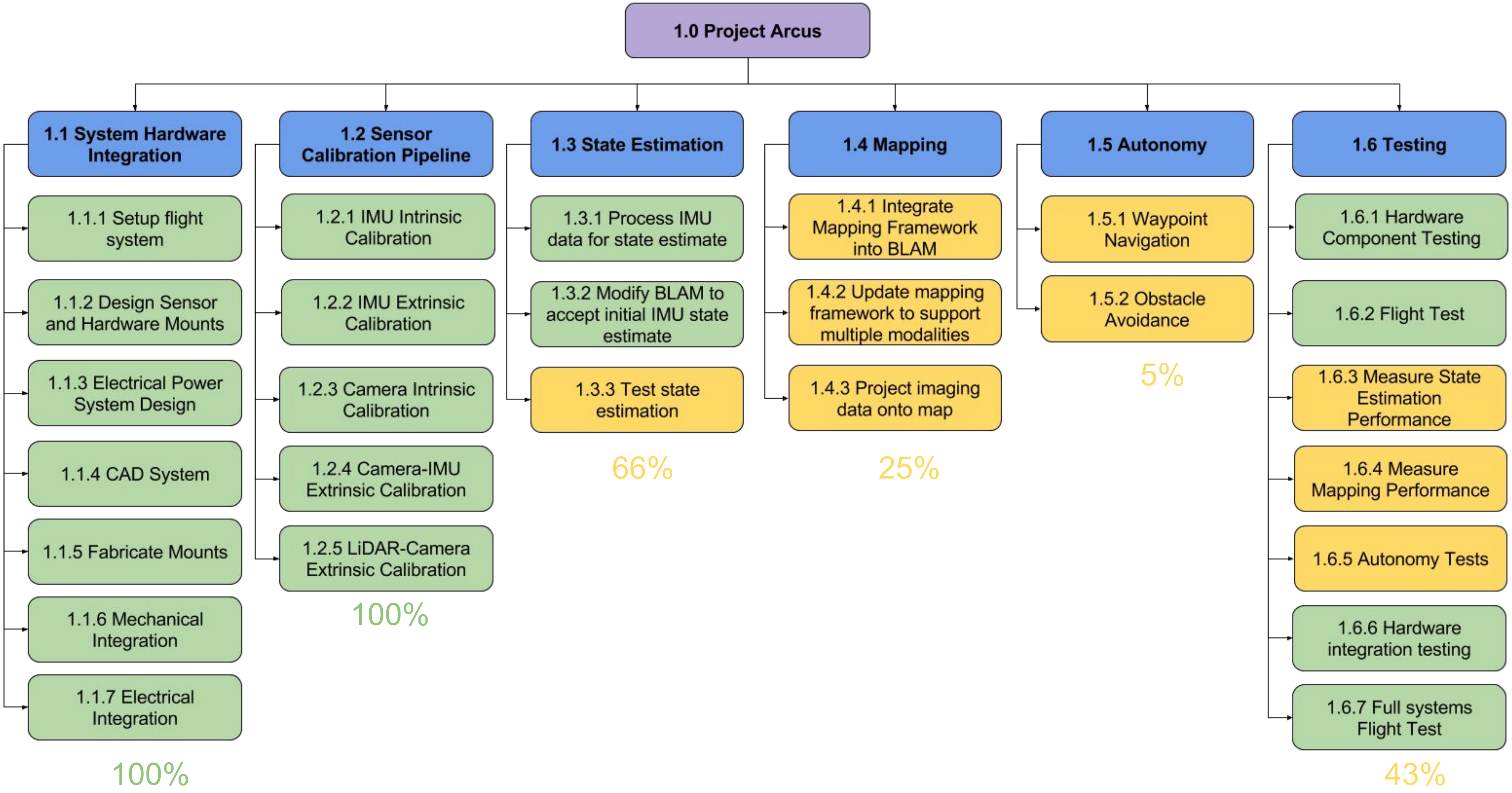
# Use Case: On Earth

- Researcher develops autonomous exploration algorithms
- Researcher selects imaging sensors
- Vehicle executes autonomous exploration code, researcher can optionally use RC control for development
- System generates map for post-experiment evaluation

# Functional Architecture

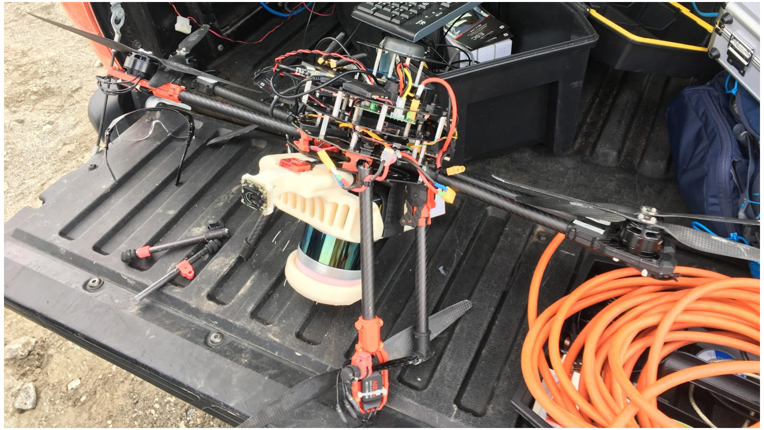


# Work Breakdown Structure



- **1.0 Project Arcus**
  - **1.1 System Hardware Integration**
    - 1.1.1 Setup flight system
    - 1.1.2 Design Sensor and Hardware Mounts
    - 1.1.3 Electrical Power System Design
    - 1.1.4 CAD System
    - 1.1.5 Fabricate Mounts
    - 1.1.6 Mechanical Integration
    - 1.1.7 Electrical Integration
    - 100%
  - **1.2 Sensor Calibration Pipeline**
    - 1.2.1 IMU Intrinsic Calibration
    - 1.2.2 IMU Extrinsic Calibration
    - 1.2.3 Camera Intrinsic Calibration
    - 1.2.4 Camera-IMU Extrinsic Calibration
    - 1.2.5 LiDAR-Camera Extrinsic Calibration
    - 100%
  - **1.3 State Estimation**
    - 1.3.1 Process IMU data for state estimate
    - 1.3.2 Modify BLAM to accept initial IMU state estimate
    - 1.3.3 Test state estimation
    - 66%
  - **1.4 Mapping**
    - 1.4.1 Integrate Mapping Framework into BLAM
    - 1.4.2 Update mapping framework to support multiple modalities
    - 1.4.3 Project imaging data onto map
    - 25%
  - **1.5 Autonomy**
    - 1.5.1 Waypoint Navigation
    - 1.5.2 Obstacle Avoidance
    - 5%
  - **1.6 Testing**
    - 1.6.1 Hardware Component Testing
    - 1.6.2 Flight Test
    - 1.6.3 Measure State Estimation Performance
    - 1.6.4 Measure Mapping Performance
    - 1.6.5 Autonomy Tests
    - 1.6.6 Hardware integration testing
    - 1.6.7 Full systems Flight Test
    - 43%

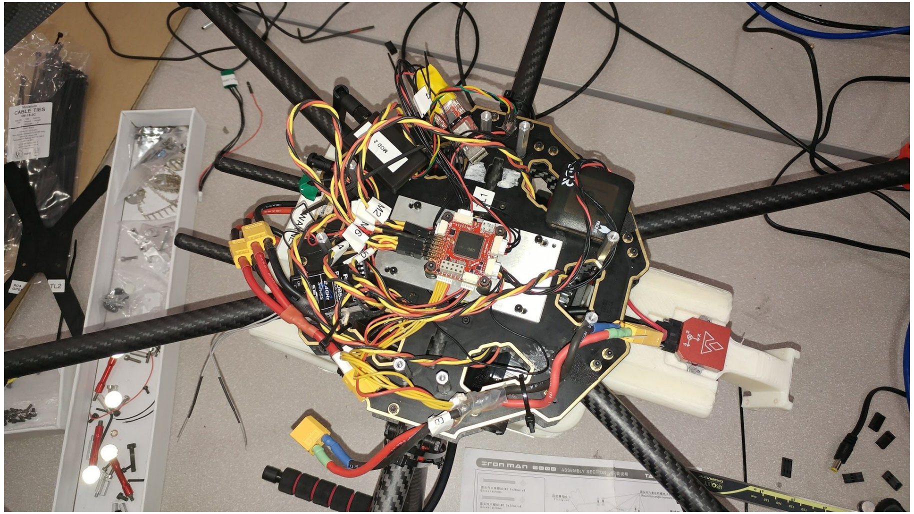
# System Status



# System Status: Hardware

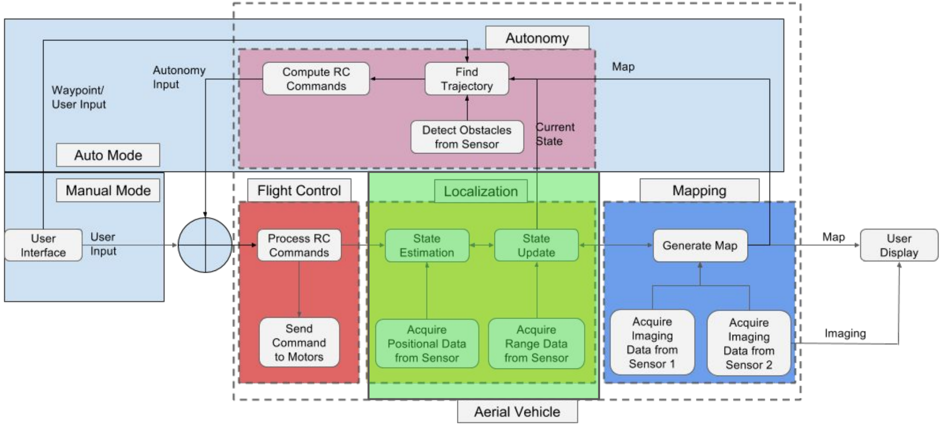
Hardware
- Drone rebuild
- State of hardware
  - New ESCs with closed-loop feedforward control
  - Pixhawk replaced by Pixracer
  - Newly tuned gains

Newly ordered
- PCB
- Spare drone chassis
- Spare propellers

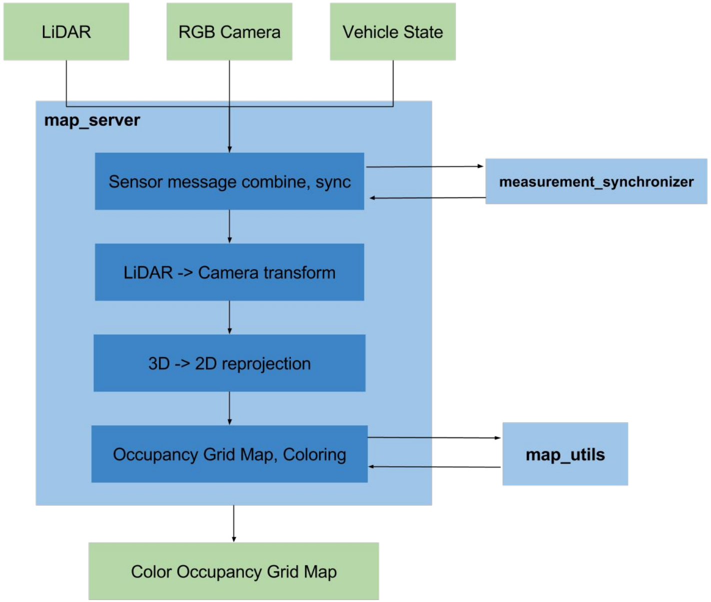


# System Status: Localization

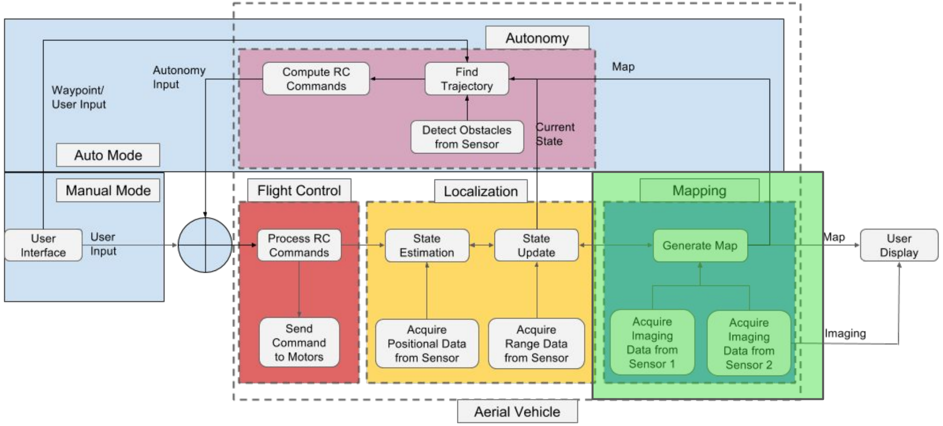
- Localization requirements met; transitioning to vehicle planning, autonomy
- Currently usable, advisor would like more robust and accurate state estimation using SCUKF from RASL



# System Status: Mapping

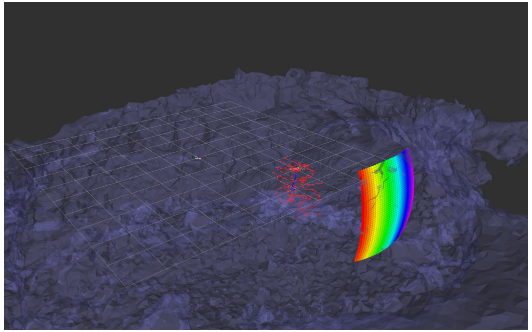


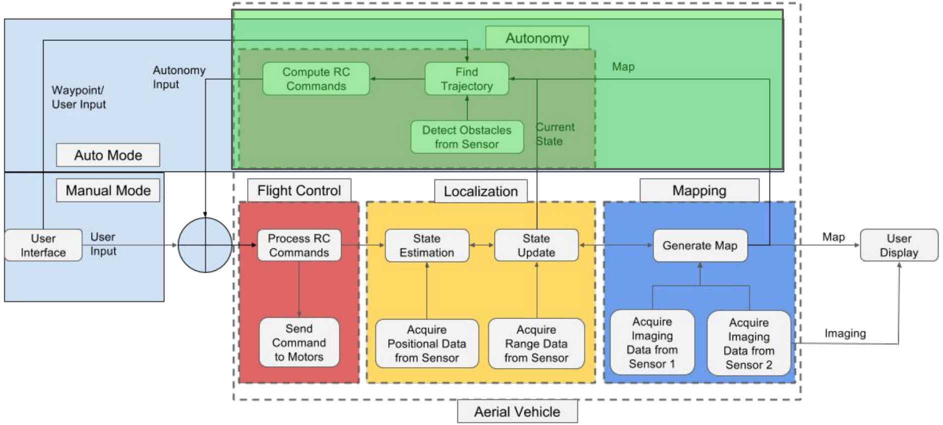
- Map_server extension:
  - Take ROS sensor messages, combine into colorized point cloud
- Map_utils
  - Take input colorized point cloud, populate a colorized occupancy grid map for autonomy planning



# System Status: Autonomy

- Obstacle Detection
  - Rudimentary collision detector in package “collision detection”
- Trajectory Planning
  - Separated into a number of packages but main one is “pit-exploration-planner”
  - The packages currently implement our advisor’s exploration algorithm in simulation, we would like to add in the ability to navigate waypoints in map.





# Budget

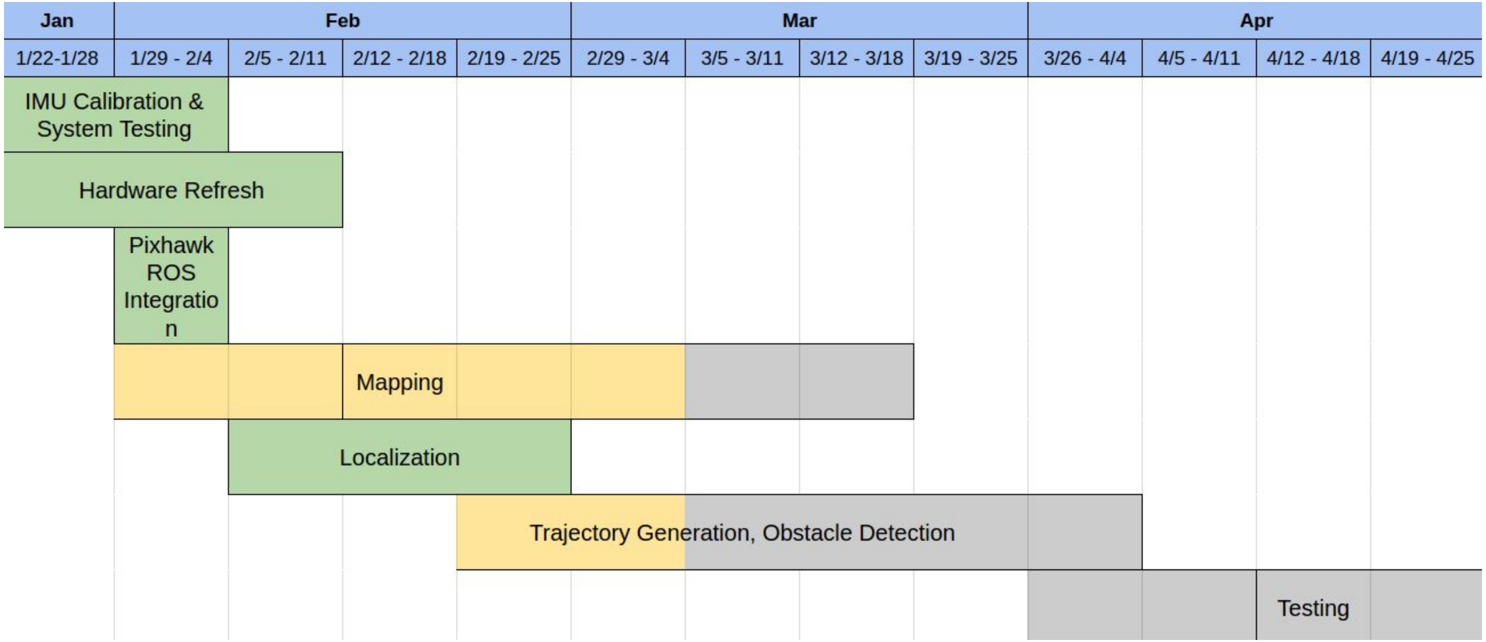
- Total budget $5000.00
- Spent: $1194.60 (24%)
- Remaining: $3805.40

Sponsor Contributions: Priceless

Sponsor purchases:

- 2 extra drone chassis
- 12 extra propellers
- Extra RGB cameras
- Extra computer

# Schedule



<table>
<tr><th>Jan</th><th colspan="4">Feb</th><th colspan="4">Mar</th><th colspan="4">Apr</th></tr>
<tr><th>1/22-1/28</th><th>1/29 - 2/4</th><th>2/5 - 2/11</th><th>2/12 - 2/18</th><th>2/19 - 2/25</th><th>2/29 - 3/4</th><th>3/5 - 3/11</th><th>3/12 - 3/18</th><th>3/19 - 3/25</th><th>3/26 - 4/4</th><th>4/5 - 4/11</th><th>4/12 - 4/18</th><th>4/19 - 4/25</th></tr>
<tr><td colspan="2">IMU Calibration & System Testing</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td colspan="3">Hardware Refresh</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td>Pixhawk ROS Integration</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td colspan="8">Mapping</td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td colspan="3">Localization</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td colspan="6">Trajectory Generation, Obstacle Detection</td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td colspan="4">Testing</td></tr>
</table>

# Anticipated Challenges

- Schedule delay
- Mapping pushed back but will deliver for PR10
- Still need to learn about autonomy stack
- Integration difficulties
    - Moving from vehicle simulation to physical system
    - Moving from simulation data to actual sensor data

# Milestones

| PR 10 (3/22) | PR 11 (4/5) | PR 12 (4/17) |
|---|---|---|
| Colorized dense voxel grid mapping in simulation | UAV Waypoint Navigation in Simulation | UAV Waypoint Navigation on Vehicle |
| Development for trajectory planning and obstacle detection | Finish trajectory planning and obstacle detection | Test flights at Lafarge |
| Drone rebuilt with new hardware integrated | Development for waypoint navigation | |

## Spring Validation Experiment 1

| | |
|---|---|
| Location | LaFarge Duquesne Quarry |
| Equipment | 120VAC, hexrotor, base station, networking equipment |
| Procedure | 1. Follow preflight checklist (appendix)<br>2. Set up UAV on landing pad<br>3. Set up the ground control station<br>4. Set up camera, waypoint pad, and wooden beacon at chosen waypoint location<br>5. Tele-operate vehicle and map for 5 minutes, visualize real-time map on base station monitor<br>6. Visualize real-time mapping error on base station<br>7. Land UAV on landing pad<br>8. Replace batteries if batteries have less than 15 V<br>~~9. Show video of previous map with comparison to ground truth FARO scan~~ |
| Validation Requirements | SPR1: Reduce odometry error to less than 0.1m / meter traveled<br>SPR2: Demonstrate textured mesh map<br>SPR3: Map sent to ground control refreshes at minimum rate of 0.5 Hz<br>SPR4: Visually inspect a generated colorized dense voxel grid map<br>SPR5: Visually inspect loop closure occurring<br>SPR6: Visually check that teleoperation distance is greater than 20 m |

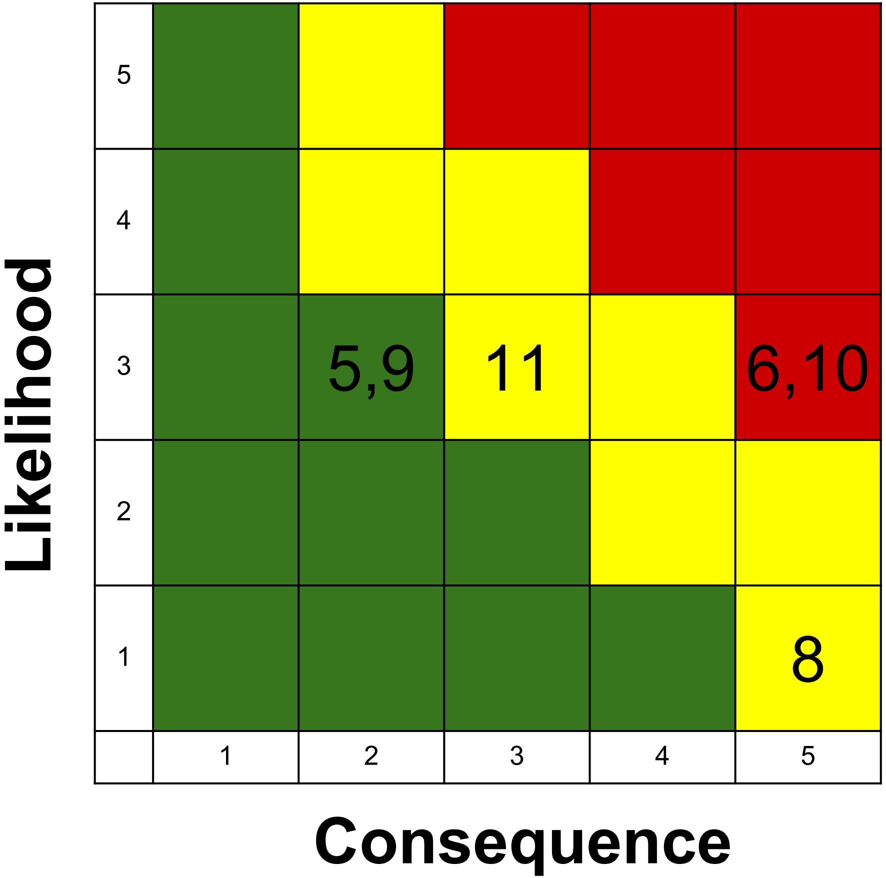
| **Spring Validation Experiment 2** | |
|---|---|
| Location | LaFarge Duquesne Quarry |
| Equipment | 120VAC, hexrotor, base station, networking equipment |
| Procedure | 1. Follow preflight checklist (appendix)<br>2. Set up UAV on landing pad<br>3. Set up the ground control station<br>4. Populate navigation map with 50cm x 50cm virtual pillars<br>5. Command UAV to specified distant waypoint location using the map formed in 13.1 SVE<br>6. Command vehicle to return back to base<br>7. Visualize voxel grid map on base station |
| Validation Requirements | SPR7: Vehicle is localized with less than 3 m of error from actual position<br>SPR8: Vehicle detects obstacles of minimum map resolution size (50cm x 50cm x 50cm)<br>SPR9: Vehicle flight path avoids obstacles by 3m |

# Risk Management: In Progress

| No. | Risk | Type | Description | L | C | Risk Reduction Plan |
|---|---|---|---|---|---|---|
| 5 | Asynchronous timing between Velodyne and RGB camera | Tech. | LiDAR scans at 10 Hz, RGB scans at 100 Hz (check these numbers), so pixels may be assigned incorrectly, resulting in a warped/inaccurately sensor-fused map | 3 | 2 | - delay RGB colorization until LiDAR point cloud map developed or attempt to predict the UAV's pose in the future based on current velocity/FOV & colorize immediately<br>- throw out extra RGB and synchronize with LiDAR<br>- try to trigger the cameras with the LiDAR |
| 6 | PIlot crashes UAV during flight | Tech., Sched, Budg. | Vehicle damage, bodily, property harm possible | 3 | 5 | - get extensive experience flying UAVs for all team members<br>- emergency landing procedure<br>- multiple vehicles |
| 8 | RF interference | Tech. | Operating the UAV at 2.4 GHz for RC control and at 5.0 Ghz for wifi / GPS may cause us to lose control of the UAV or online mapping goes down | 1 | 5 | - Static testing in RF noisy environments<br>- Construct a test to validate UAV communications performance<br>- Look at flight logs |
| 9 | RASL's mapping framework cannot be integrated into BLAM | Tech. | Mapping framework incompatible or too difficult to integrate with BLAM | 3 | 2 | - Find another framework to use<br>- Modify current framework code to make it more compatible with our purposes |
| 10 | Robot crashes while flying autonomously | Tech. | Vehicle damage, bodily, property harm possible | 3 | 5 | - testing in simulation<br>- extensive testing indoors in a safe environment |
| 11 | Lafarge Quarry inaccessible | Sched. | Cannot access the quarry or fly due to inclement weather/permissions issues | 3 | 3 | - backup flight test days<br>- show video instead for SVE |

# Risk Management: Likelihood-Consequence Table

| No. | Risk |
|---|---|
| 5 | Asynchronous timing between Velodyne and RGB camera |
| 6 | PIlot crashes UAV during flight |
| 8 | RF interference |
| 9 | PRL's mapping framework cannot be integrated into BLAM |
| 10 | Robot crashes while flying autonomously |
| 11 | Cannot access Lafarge |



**Likelihood**

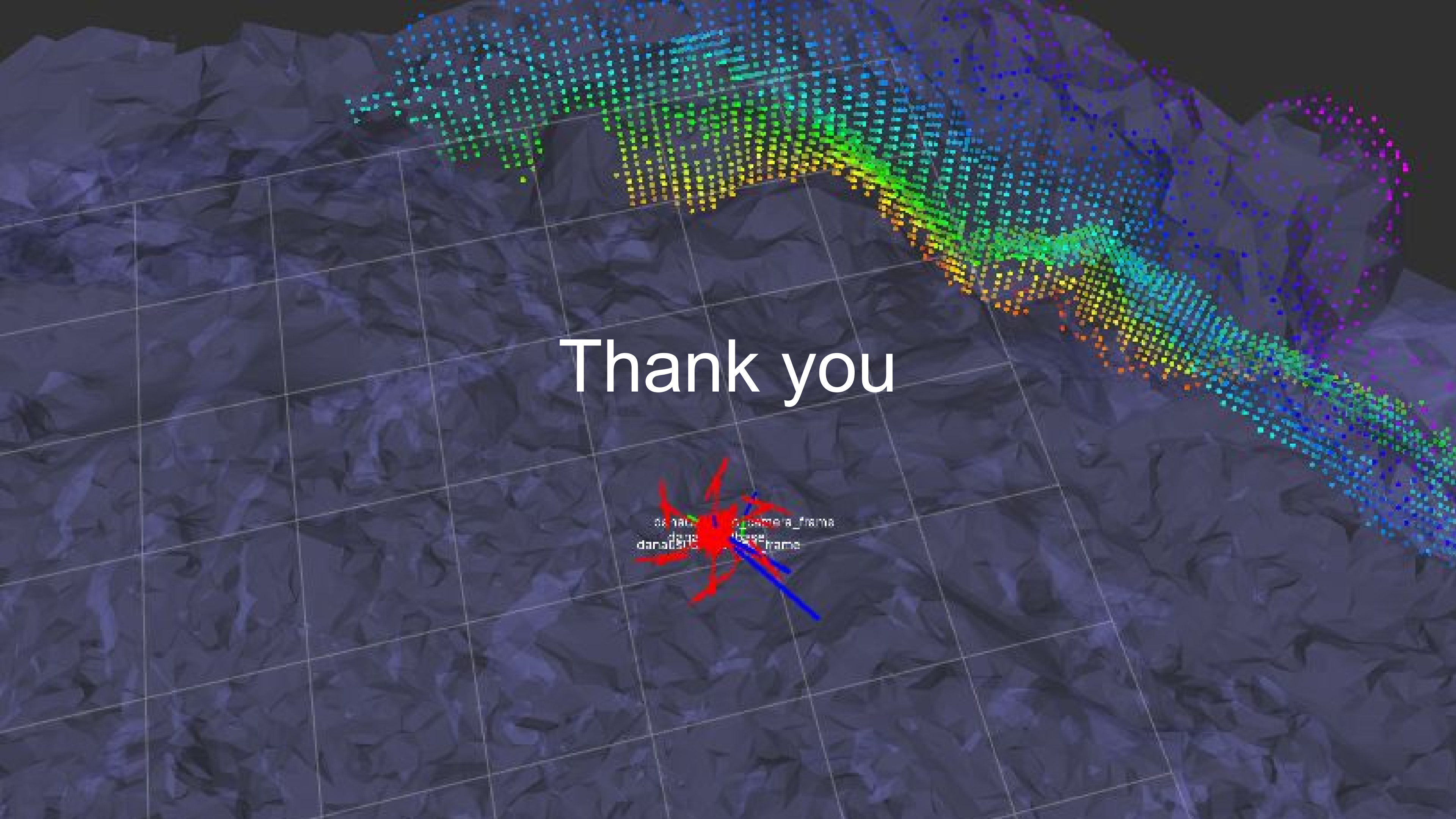
| | | | | | |
|---|---|---|---|---|---|
| 5 | 🟩 | 🟨 | 🟥 | 🟥 | 🟥 |
| 4 | 🟩 | 🟨 | 🟨 | 🟥 | 🟥 |
| 3 | 🟩 | 🟩 5,9 | 🟨 11 | 🟨 | 🟥 6,10 |
| 2 | 🟩 | 🟩 | 🟩 | 🟨 | 🟨 |
| 1 | 🟩 | 🟩 | 🟩 | 🟩 | 🟨 8 |
| | 1 | 2 | 3 | 4 | 5 |

**Consequence**

# Risk Management: Mitigated Risks

**4 of 8 original risks have been mitigated**

| **Risk No.** | **Risk** | **Risk Type** | **Mitigation Method** |
|---|---|---|---|
| 1 | Velodyne damage | Technical, Budgeting | Fabricated a padded crash cap |
| 2 | Unable to map online | Technical | Computer (Brix) has sufficient processing power and is able to keep up with online mapping algorithms |
| 4 | Unable to form loop closures with BLAM | Technical | Fixed loop closures while getting IMU pre-integrated |
| 7 | Unable to get Kalibr running | Technical, Scheduling | Found the correct setting to use when starting up Kalibr (video needed to be 8 bit grayscale) |



# Thank you

# Risk Management L-C Scales

**Consequence**

| Level | Technical Performance | Safety | Schedule | Cost ($) |
|---|---|---|---|---|
| 1 | Minimal or no consequence to performance | No bodily harm | Minimal to no impact | 0 - 20 |
| 2 | Minor reduction in performance, tolerable | No bodily harm | Able to meet key dates | 21- 100 |
| 3 | Moderate reduction in performance, limited impact on final objectives | No bodily harm | Able to meet milestones | 100 - 300 |
| 4 | Significant degradation in performance | No bodily harm | Milestone objectives altered | 300 - 1000 |
| 5 | Complete loss of performance | Any bodily harm | Unable to meet milestones altogether | >1000 |

**Likelihood**

| Level | Likelihood | Probability of Occurrence (%) |
|---|---|---|
| 1 | Not Likely | 0 - 20 |
| 2 | Low Likelihood | 20 - 40 |
| 3 | Likely | 40 - 60 |
| 4 | Highly Likely | 60 - 80 |
| 5 | Near Certainty | 80 - 100 |