Author: ammarh@

Navigation triage champion: mitri@

Objective

This document is a ~6 month retrospective on implementing a process for triaging and addressing field incident reports (aka bugs) as prescribed by the Navigation Field Triage Doctrine.

Key Outcomes and Learnings

- Setting up a fire department for navigation has enabled the collection of proper & detailed statistics on common field failures as outlined in the Navigation Field Report.

- Even though there is a 10X room for improvement, there has still been a marked reduction in the time it takes to triage a field bug as visible here. This reduction is due to two key factors:
    - More scripting & automation for root causing failures with logs.
    - Now able to make much more consistent, persistent and uniformly prioritized tools feature requests for the Platform team as detailed in Triage Tools Feature Requests.

- Observations and experience in the Navigation Field Report guided the creation of a scenario evaluation system for navigation, as well as informed the design of scenarios to test, as enumerated in the table of use cases under the Navigation Scenario Tests Design Doc.

- Enabled a better understanding of robot capability limitations and early spotting of systems regressions as is evident in b/150867120, b/150718120, etc.

- Detailing and constantly documenting the triage process in How to Debug Navigation Bugs increases the bus factor and bottleneck on information.

- Training a triage champion from CO to dive slightly deeper into the navigation stack enabled bridging the gap between SWE developers and robot experience on the field.

- Most field bugs are now being addressed in a timely manner and usually within 48 hours. Some examples: b/151157855, b/151152320, b/150629177, b/150118175, etc.

- Weekly high level summary of navigation related incidents facilitates better communication:

Nav bugs weekly summary report (Mon, 3/02 - Fri, 3/06)

Mitri Syriani <mitri@google.com>
to x-proxy-navigation, x-proxy-perception, Mike, Devashree

Hello all,

Attached below is a snapshot of the weekly report for navigation bugs.

Quick Summary:

- This week our robots (Meta's and HV's) navigated for 7.85 hours.
- We had a total of 14 navigation bugs (13 Meta and 1 HV), most of the bugs were related to the meta not being able to find/plan a route to navigate from HW-Eng station to Timeline station.
- Check out the new navigation weekly summary dashboard.

Here's a summary of the bugs, broken down into categories. You can see more info here. (5 bugs are pending more investigation).

Weekly summary of Nav bugs																	
		Navigation Run Time (hours)	Total count of Bugs	Capability Limitation							Upstream noise				Regression	True failure	Miscellaneous
	Week			Ego Motion / Jerkiness	Missing Semantics	CBr sensing limitation	Camera sensing limitation	Sensing blindspot	Ego occlusion	Total	CBr	Camera	Map	Total			
	3/2/2020	7.85	9	0	0	0	0	1	0	1	4	1	0	5	0	2	1

Top Issues reported from QA & Ops:

1. b/150867120 - META014 - BaseReplanner - ERROR: Goal is not reachable | Robot unable to navigate to the next station.
2. b/150629177 - META009 - Robot unable to plan a route, ghost obstacles blocking robot route.
3. b/150852360 - HV14 - [camera] detected floor as cluster points, causing robot to think it's in collision with the floor

- Training one (or two) triage champions to go relatively deep into root causing a particular capability is more effective than training all of CO in shallowly root causing all capabilities. Given the system complexity, specialization improves the quality of the outcome.

- Reduced context switching (going from strategic feature planning & development to tactical firefighting) time for SWEs, boosts productivity.

- Integrated view of overall navigation performance in the Proxy wide dashboard.