



WPI

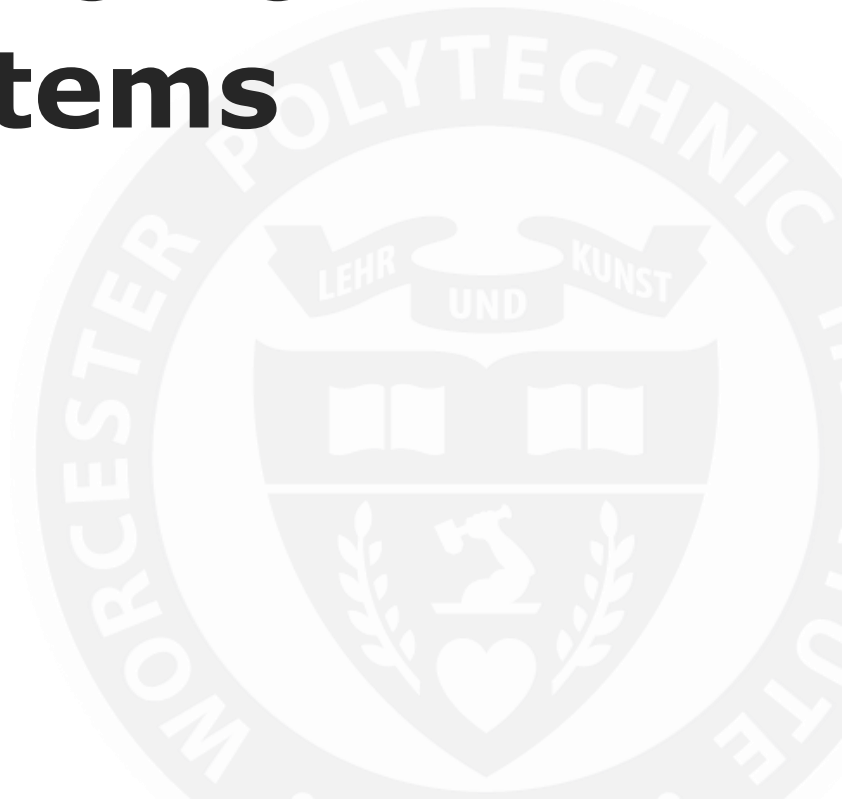
Human Supervision of Multi-Robot Systems

A Major Qualifying Project by:

Donald Bourque

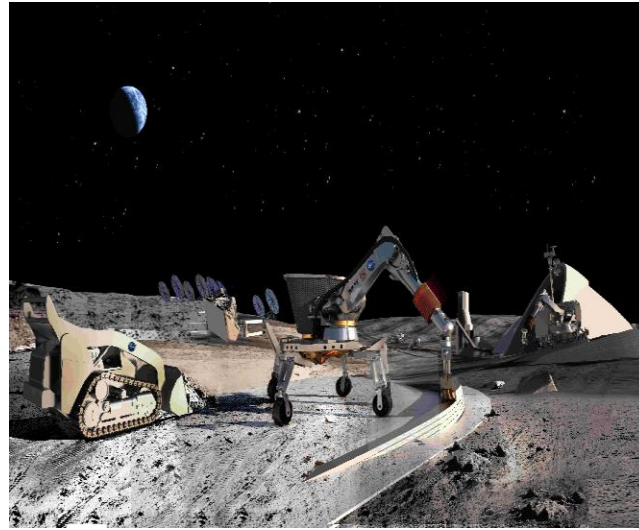
Thomas DeSilva

Nicholas Otero



Motivation

- Tasks often require multiple agents
- Robots are quick, safe, and/or reliable
- Humans offer judgment and flexibility



[1], [2], [3]

Project Goals

- Develop a framework for human supervision of multi-robot systems
- Devise a test to evaluate the framework
- Assemble a team of robots to perform the test

Research and Inspiration

- Distributed coordination
- Task and role assignments
- Utility function calculations
- Human-robot interface design

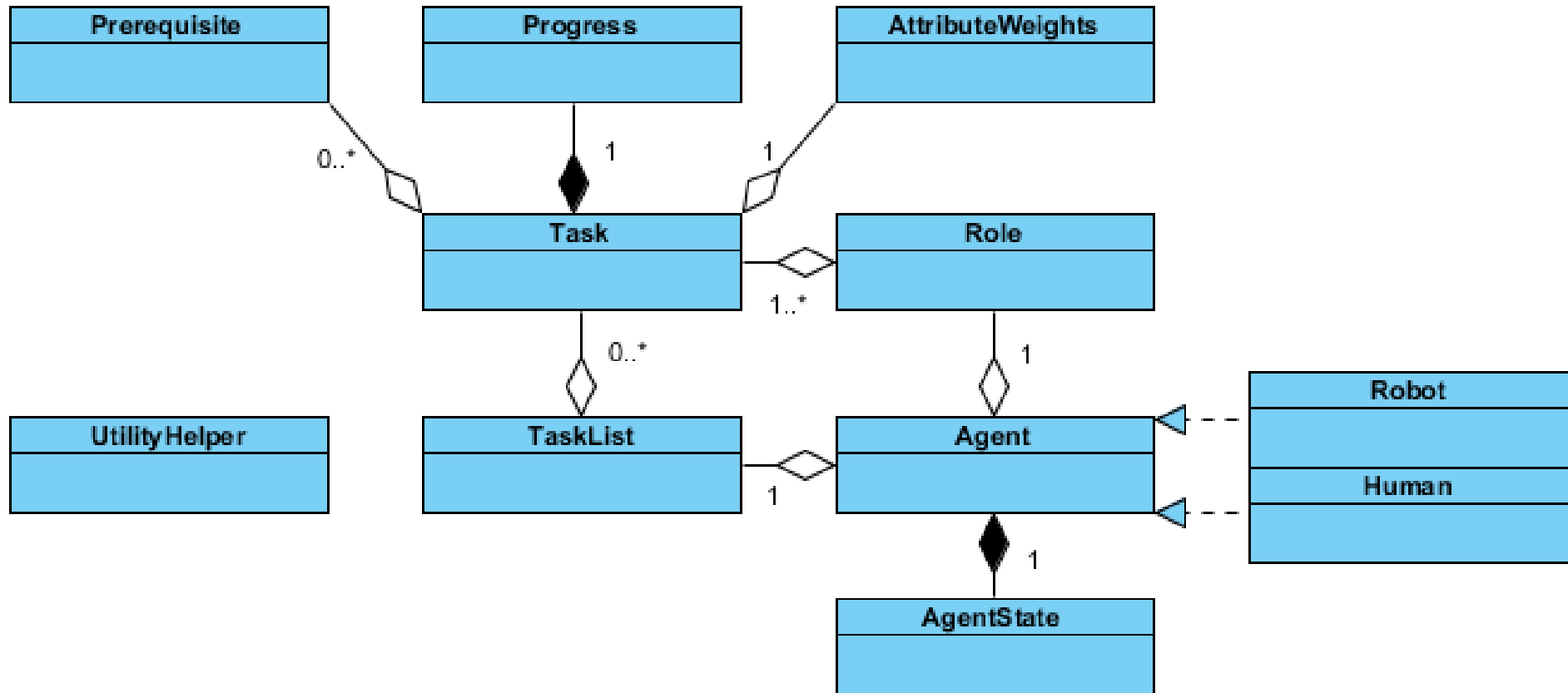


[4]

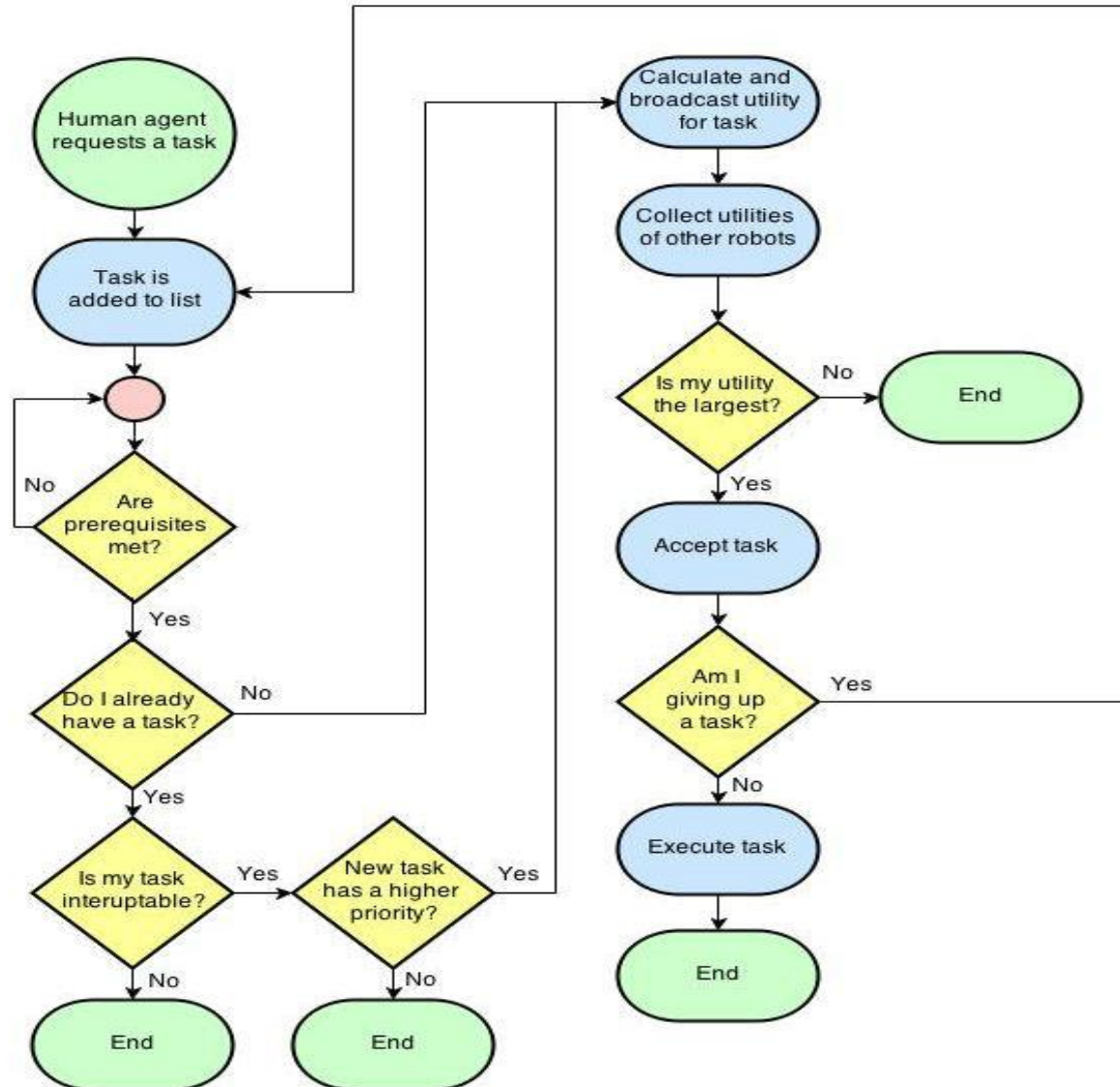


[5]

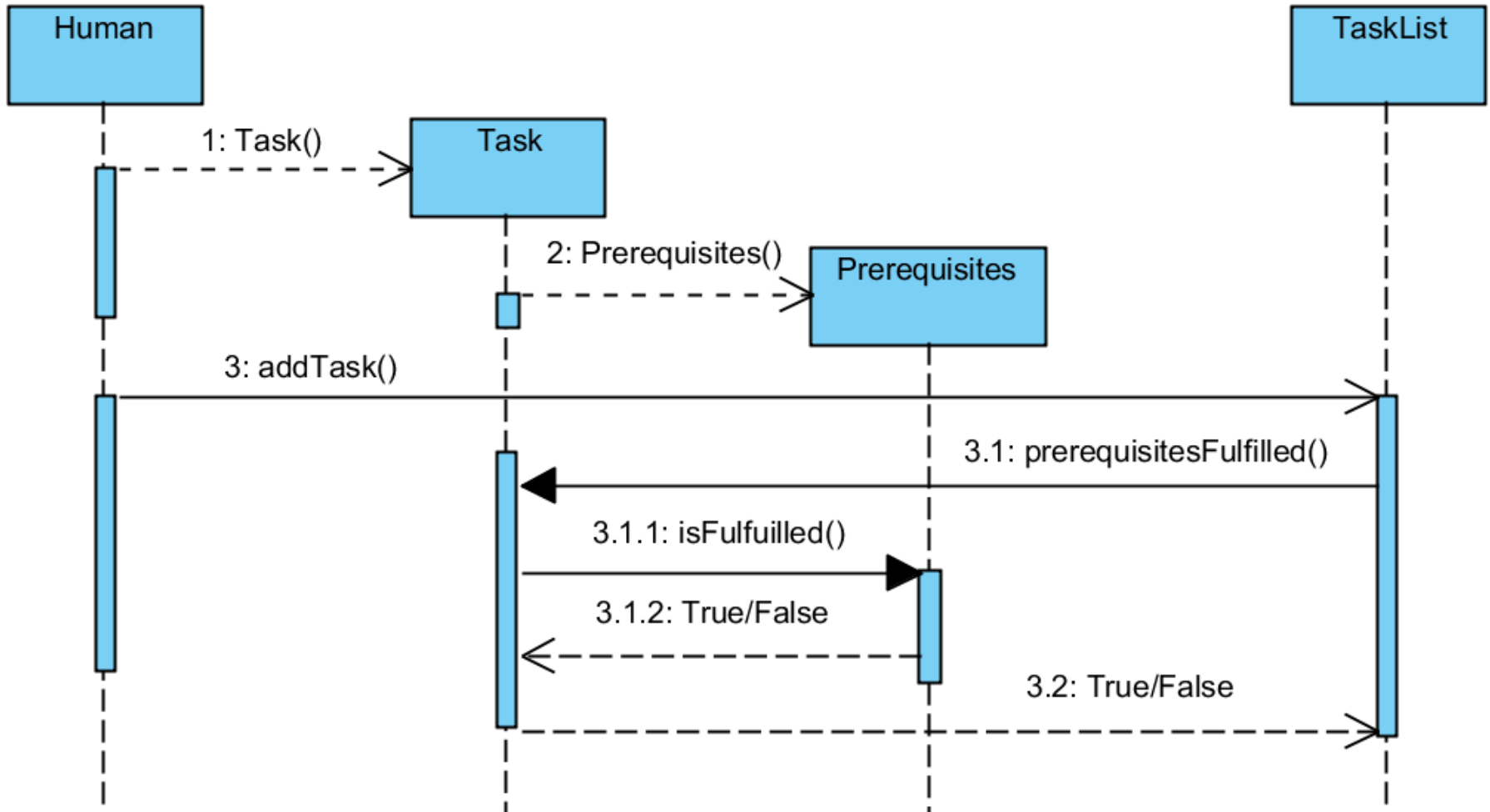
Framework Design



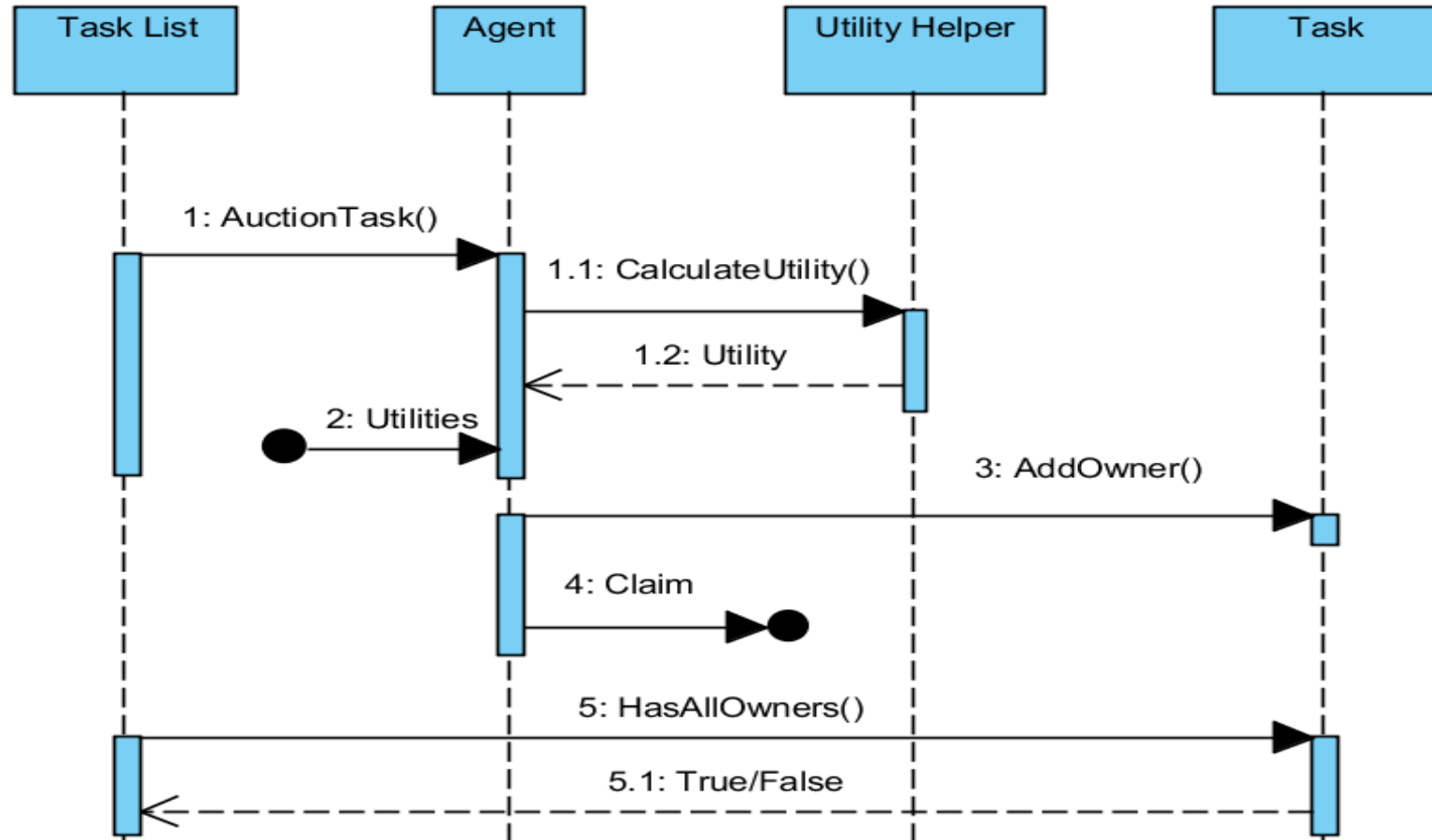
Framework Workflow (cont'd)



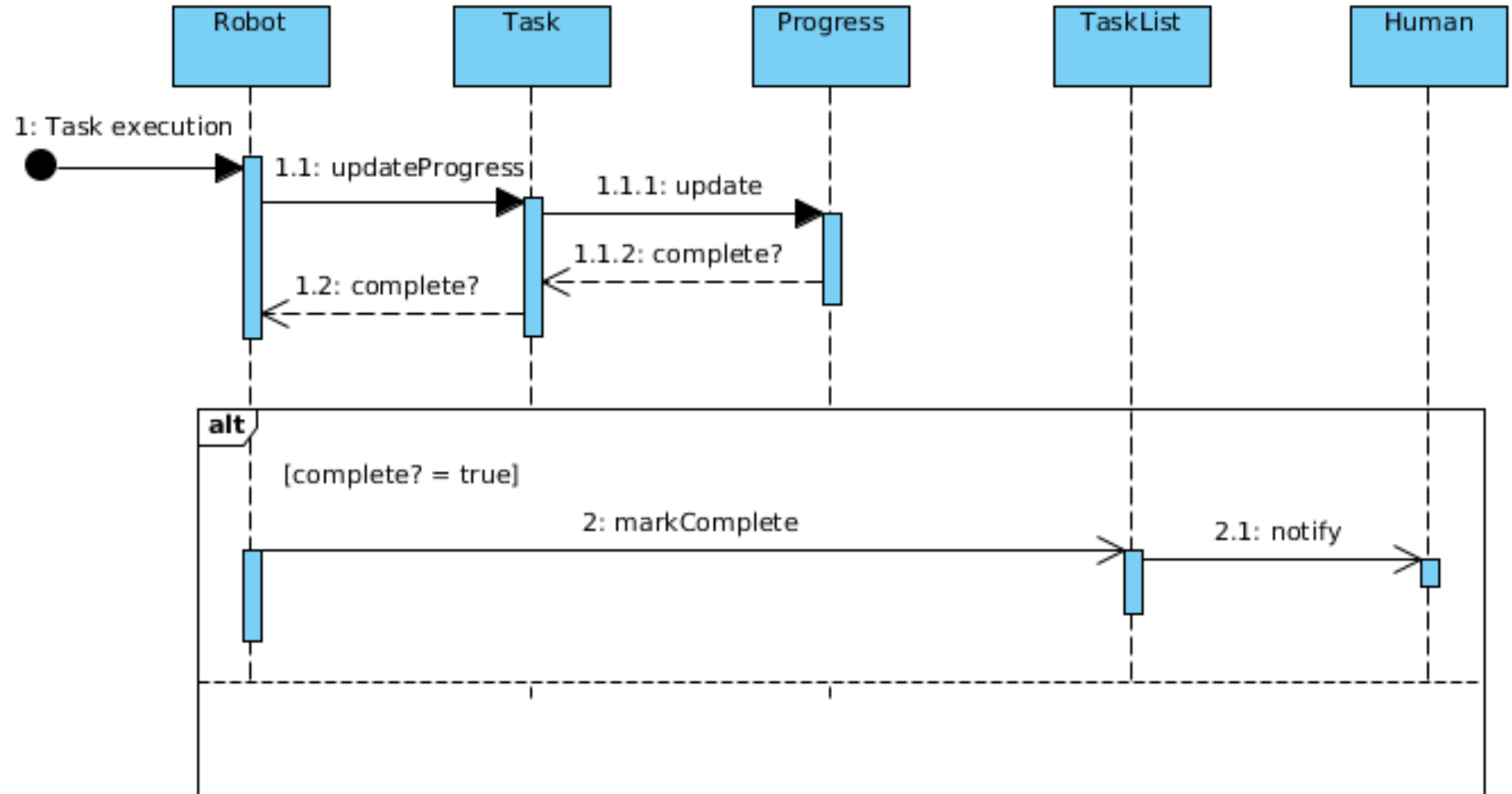
Framework Workflow



Framework Workflow (cont'd)



Framework Workflow (cont'd)



Testing The Framework

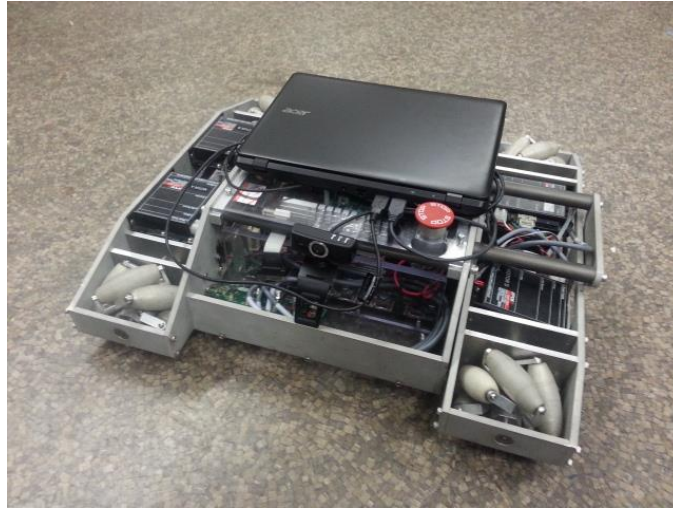
Evaluate the effectiveness of the framework:

- Unit tests of specific framework functions
- Search and discover mission

Robots



Turtlebot

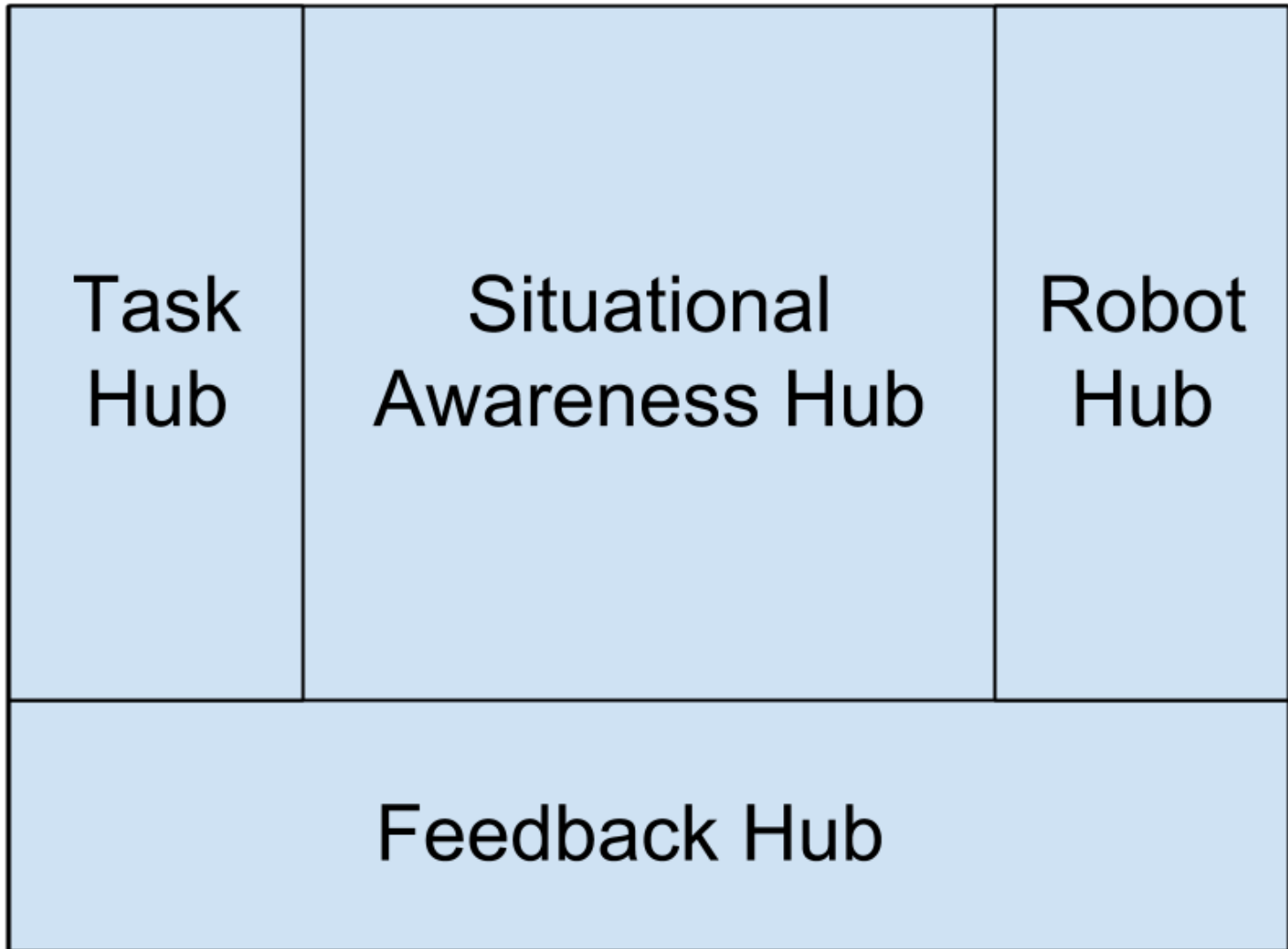


Hermes



Husky

Graphical User Interface



Graphical User Interface (cont'd)

Task List

New task	Remove task
GoTo	
Status: Not started	
Owners:	
FollowTag	
Status: Not started	
Owners:	
Search	
Status: Not started	
Owners:	

Robot List

■ Hermes
Status: Connected
Task: Idle
■ Husky
Status: Connected
Task: Idle
■ Turtlebot
Status: Connected
Task: Idle

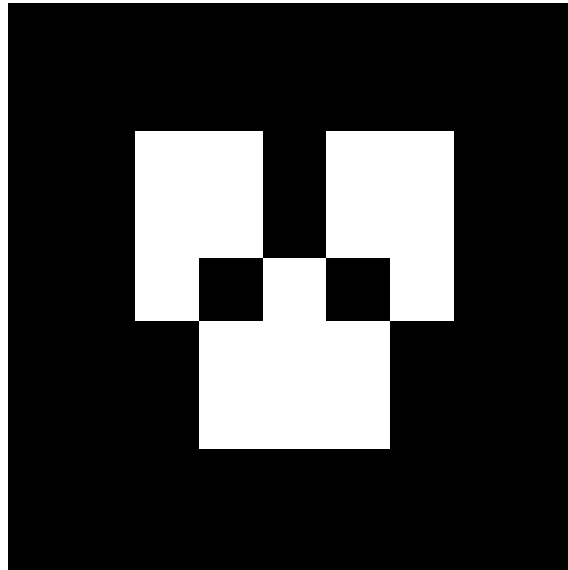
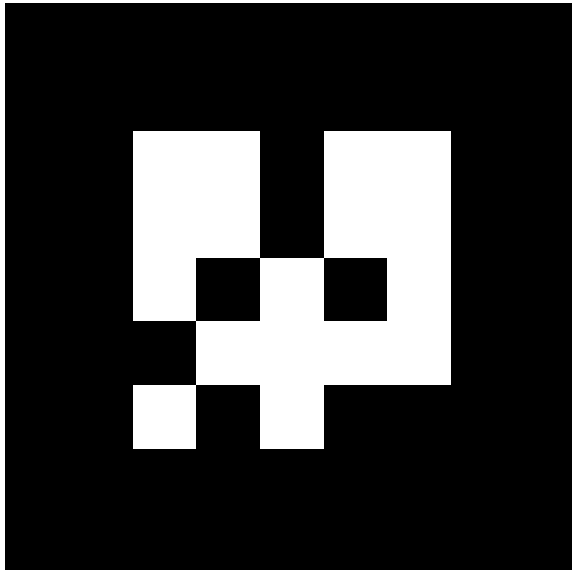
Log

All System Hermes Husky Turtlebot

[1428698102] System: New agent has been successfully registered: Hermes
[1428698112] System: New agent has been successfully registered: Husky
[1428698122] System: New agent has been successfully registered: Turtlebot

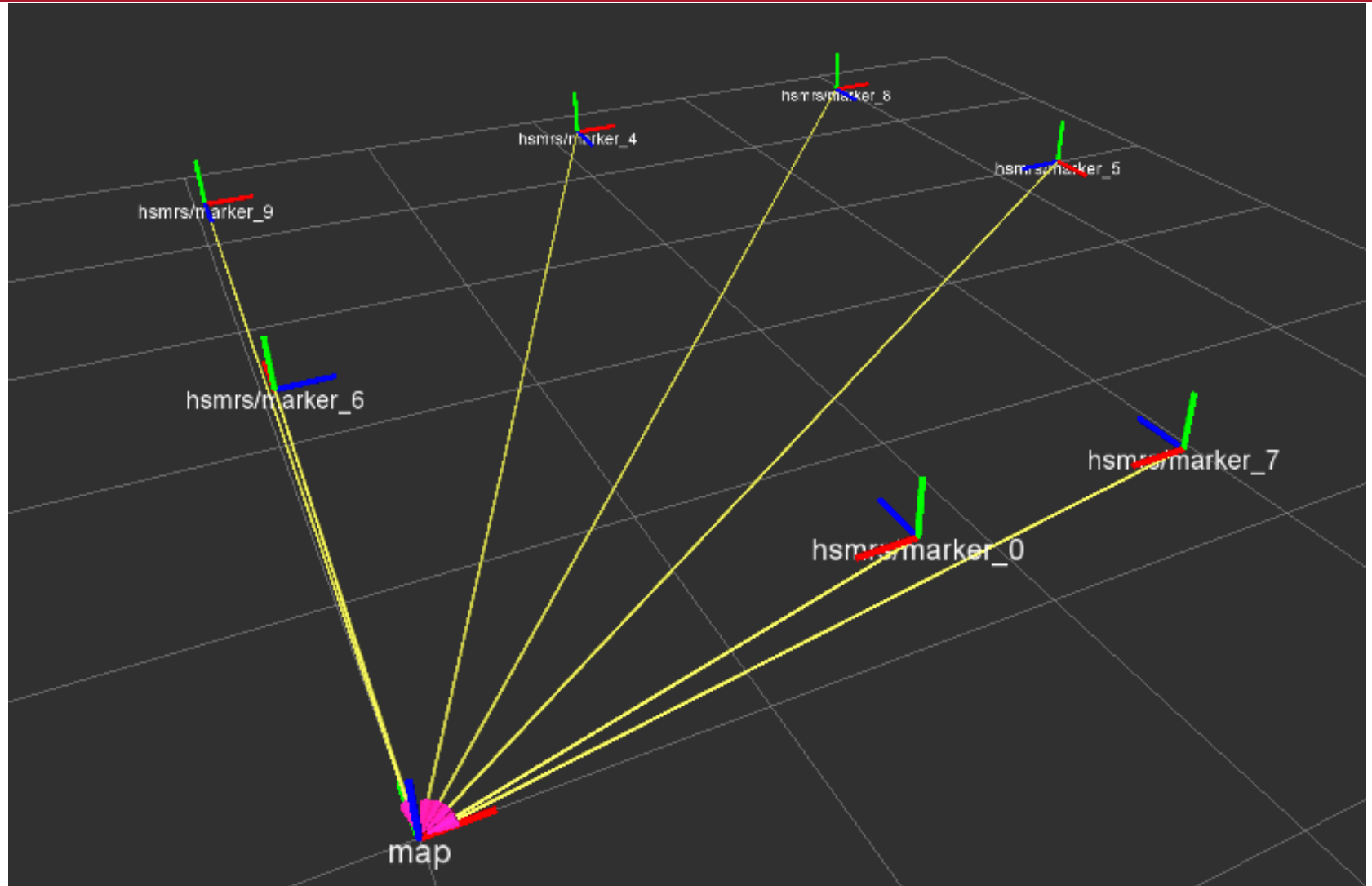
Localization

- Needed to determine position of robots within the operating area
- Used Augmented Reality(AR) tags and wheel odometry
- Position belief was maintained by a rolling average filter



[6]

Localization



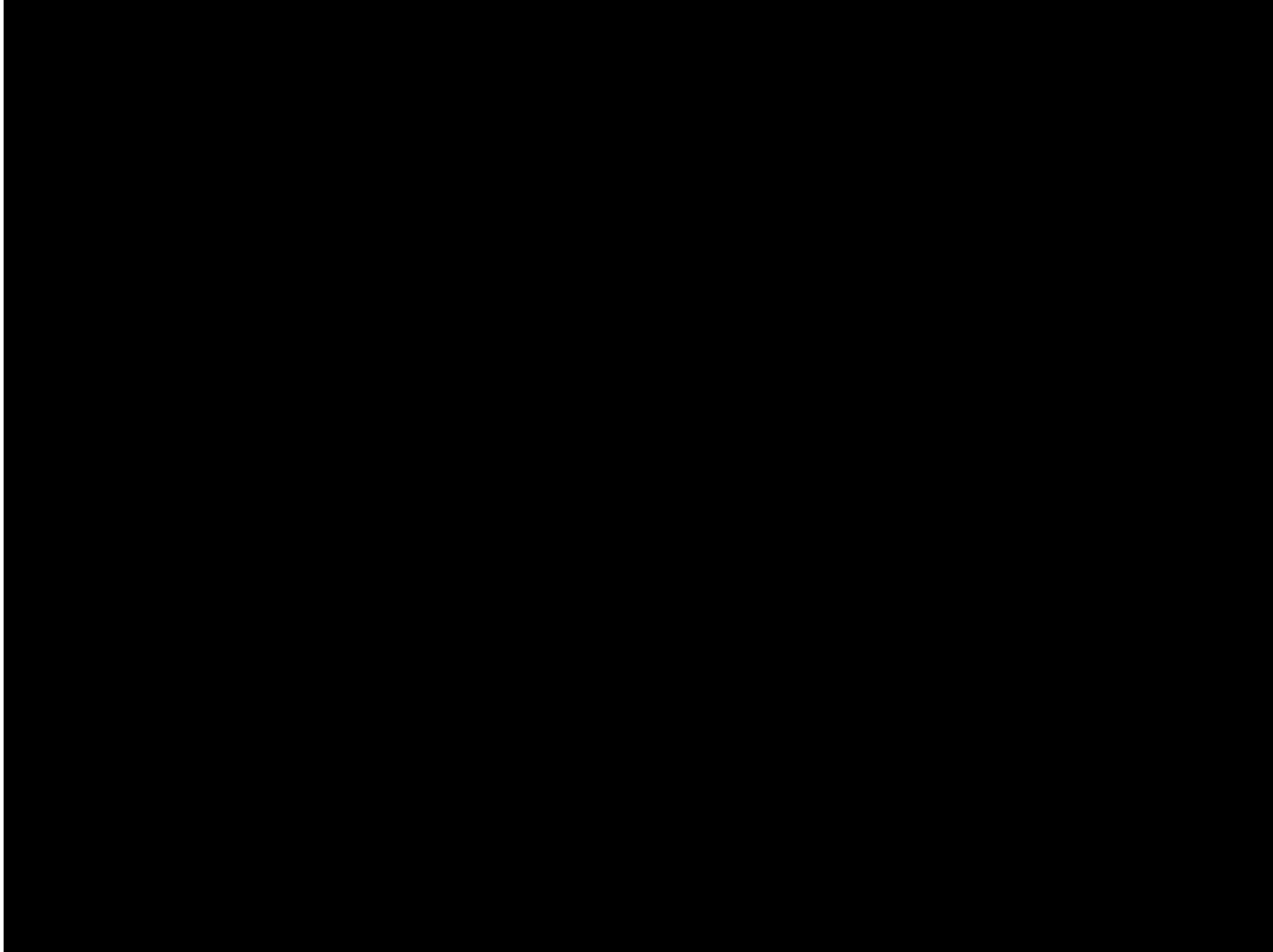
Results

- Hermes, Husky, and the Turtlebots were able to be represented in the system with different attributes.
- The bidding algorithm was able to assign tasks to robots who were the most capable of doing them.
- Roles allowed the user to exert coarse control over the task allocation process.
- The human supervisor was able to gain situational awareness using the camera view, the map view, and the help alert.
- The human supervisor could directly assign tasks, interrupt tasks and directly control one or more robots at a time.

Results (cont'd)

- Large amounts of latency in communications slowed the execution of the system.
- Prerequisites needed too much information to be practical.
- Localization methods had different levels of effectiveness on different platforms.

Video



References

- [1] <http://www.unocha.org/roap/about-us/emergency-response>
- [2] <http://www.contourcrafting.org/space-colonies/>
- [3] <http://www.dailymail.co.uk/news/article-2585981/Workers-casually-dismantle-cranes-used-build-Shanghai-Tower-knee-wobbling-2-000ft-up.html>
- [4] people.csail.mit.edu/rak/www/sites/default/files/pubs-/KneEtal13.pdf
- [5] <http://www.robocup2014.org/?p=893>
- [6] http://wiki.ros.org/ar_track_alvar
- [7] <http://www.bls.gov/news.release/pdf/cfoi.pdf>

Questions?



Motivation Follow Up

- As of 2013 [7]:
 - 100 fatalities(33%) were reported in the manufacturing industry which may have been preventable through robots.
 - 294 fatalities(37%) involving falling and 80 fatalities(10%) involving proximity to dangerous machines were reported in the construction industry. These could be prevented with human supervised robot teams.
- In space environments, teams of humans are difficult to maintain and could be replaced by robots. However, human supervision and judgment is still needed.
- Hazardous environments such as Fukushima Daiichi could require multi-robot systems with human supervision.

GUI Follow Up – Request Help

The screenshot displays a robot control interface. On the left is a 'Task List' panel with 'New task' and 'Remove task' buttons. The central area shows a camera feed of a white turtlebot in a room with bookshelves. On the right is a 'Robot List' panel showing two robots: 'turtlebot' (red square, Status: Connected, Task: Idle) and 'hermes' (green square, Status: Tele-Op, Task: Idle). An 'Attention Required' dialog box is overlaid on the right, containing a warning icon and the text 'turtlebot has requested assistance!' with an 'OK' button. At the bottom, a log window shows system messages: '[1429640825] System: New agent has been successfully registered: turtlebot', '[1429640843] System: New agent has been successfully registered: hermes', '[1429640880] turtlebot: My center bumper was released!', and '[1429641066] turtlebot: My center bumper was released!'. The log window has tabs for 'All', 'System', 'turtlebot', and 'hermes'.

Task List	
New task	Remove task

Robot List	
■ turtlebot	Status: Connected Task: Idle
■ hermes	Status: Tele-Op Task: Idle

Attention Required

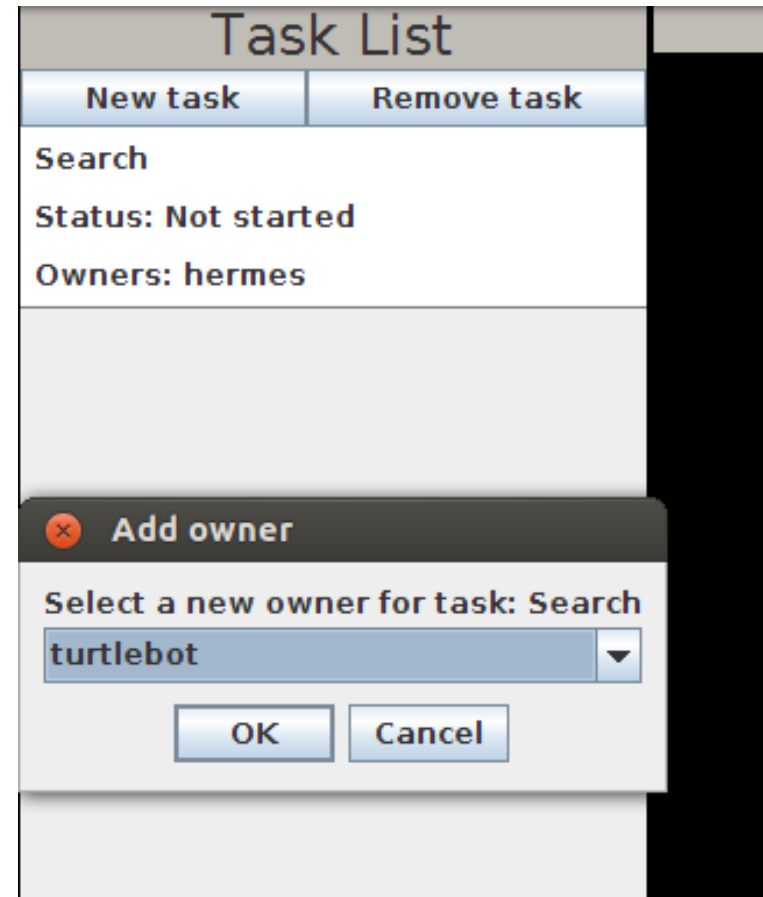
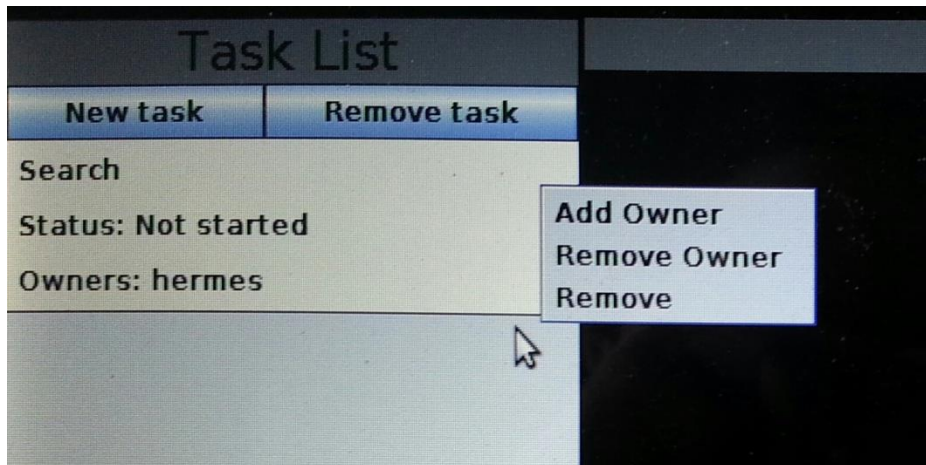
! turtlebot has requested assistance!

OK

All System turtlebot hermes

[1429640825] System: New agent has been successfully registered: turtlebot
[1429640843] System: New agent has been successfully registered: hermes
[1429640880] turtlebot: My center bumper was released!
[1429641066] turtlebot: My center bumper was released!

GUI Follow Up – Edit tasks




GUI Follow Up – Roles

New Role

Name


GoTo
FollowTag
Search



Role List


Role

My New Role
My Second New Role



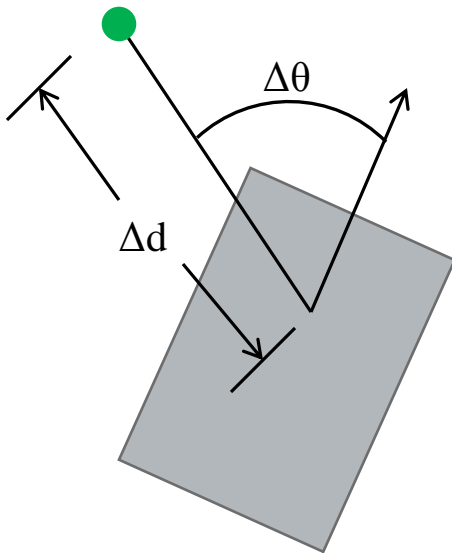
Robot List

■ turtlebot - My New Role Status: Connected Task: Idle
■ hermes - My Second New Role Status: Connected Task: Idle



Path Planning and Execution

- Path planning was implemented using A*
- Path execution was accomplished using a piecewise proportional controller:



$$\omega = K_{\omega}\Delta\theta$$

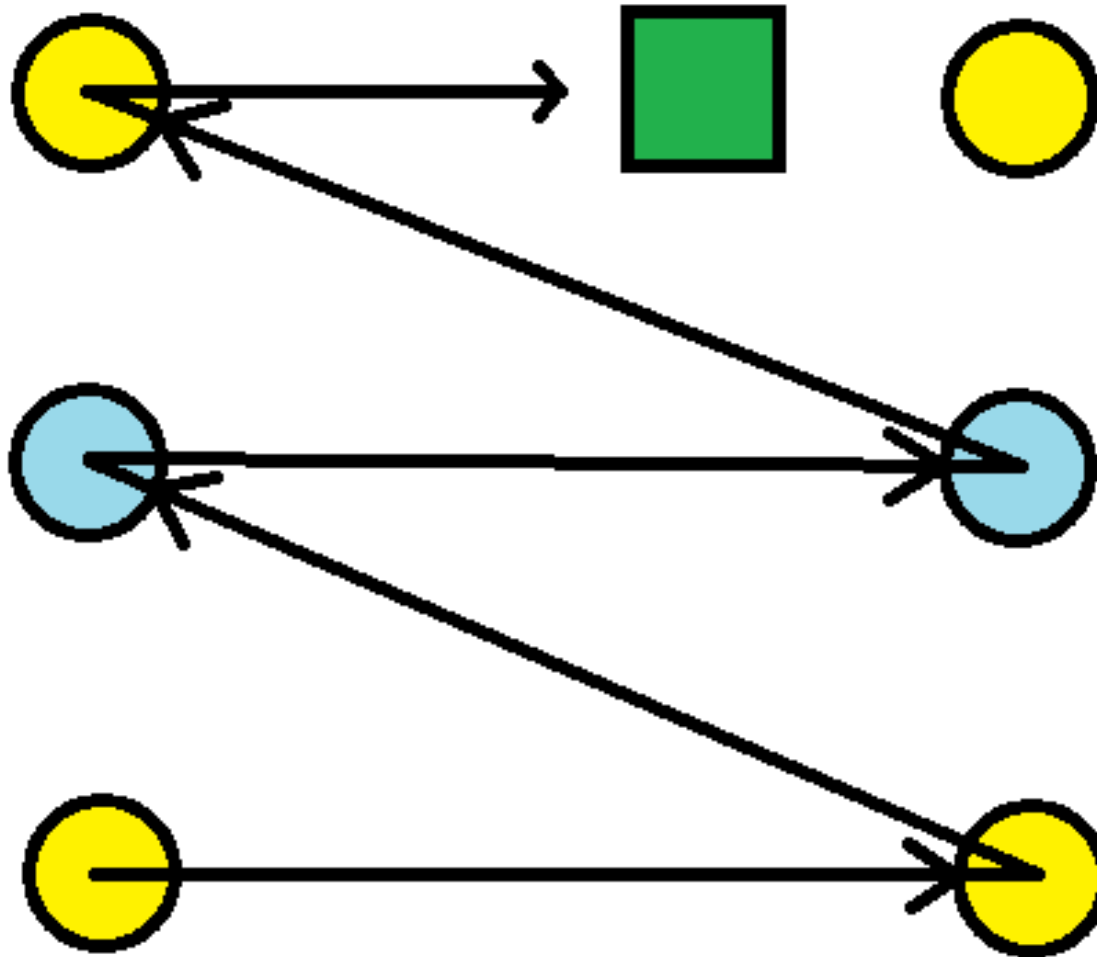
$$v = \begin{cases} 0, & \Delta\theta > \varepsilon \\ K_v\Delta d, & \Delta\theta \leq \varepsilon \end{cases}$$

Where:

ω and v are the robot's angular and linear velocities
 K_{ω} and K_v are the proportional gains for angular and linear velocity

ε is the allowable angular error before linear motion is initiated.

Navigation Follow Up



Communications Follow Up

- Pings to router > 10s
- Network traffic generated by system < 60kb/s
- Processor load on supervisor's computer <20%
- Improved performance when operating in Gateway Garage

- Possible cause was using overcrowded wireless channels

Utility Function

$$\sum aw$$

a = attribute value
w = weight