The MITRE Meteorites
2005 DARPA Grand Challenge Entry

Robert Bolling (rbolling@mitre.org)
Robert Grabowski, Ph.D. (rgrabowski@mitre.org)
Ann Jones (adjones@mitre.org)
Mike Shadid (mshadid@mitre.org)
Richard Weatherly, Ph.D. (weather@mitre.org)

Introduction

The MITRE Meteorites team is sponsored by the MITRE Corporation. MITRE is a collection of Federally Funded Research and Development Centers that support the DoD, FAA, IRS and other federal agencies. MITRE sponsors the MITRE Meteorites entirely on discretionary funds and chooses this investment in a belief that many of The MITRE Corporation's work programs would benefit from an investigation in the technologies that contribute to the DARPA Grand Challenge. With this in mind, the MITRE Meteor was specifically designed to respond to a variety of potential sponsor needs. The software was designed to be robust and portable to a wide variety of platforms. The sensors were selected to be low cost for affordability in high quantities. Missions such as convoy leader/following, surveillance, unmanned transport and cooperative robot missions have been considered during design decisions and tradeoffs. This paper is provided to DARPA as a top level technical description of the Meteor. Please contact the authors if additional detail is desired.

While the MITRE Corporation is the primary sponsor, additional companies that provided equipment and services include: ACTTechnico, Concurrent Technologies, Hybricon, Electronic Mobility Controls, Corp.., SuperLift Suspensions, Interco Tire, MSC Software, Top-Soil Precision Ag., OmniStar, PCB Piezotronics, Inc., and Tidewater Communications. We are grateful for the support they have provided.

1. Vehicle Description

1.1. *Describe the vehicle. If it is based on a commercially available platform, provide the year, make and model. If it uses a custom-built chassis or body, describe the major characteristics. If appropriate, please provide a rationale for the choice of this vehicle for the DGC.*

Unmodified Ford Explorer Sport Trac

The decision was made early to purchase a commercial vehicle rather than develop a custom platform. This has allowed the focus to be on issues more relevant to potential MITRE sponsors including vehicle control, localization, navigation, and sensing/responding to the environment. Several vehicle types were considered: racing buggy, ATV, 4WD SUV and 4WD pickup. We desired a vehicle that would be street legal with sufficient off-road capabilities as well as a protected interior that would keep the components cooled and not exposed to the elements. A 2004 Ford Explorer Sport Trac was selected. The Sport Trac has reasonably good off-road capability and has sufficiently cooled interior space for our computing equipment. Another consideration that influenced our decision is that Ford vehicles are well understood by Electronic Mobility Controls Corp (EMC), the vendor that provided our drive-by-wire capability.

1.2. Describe any unique vehicle drive-train or suspension modifications made for the DGC including fuel-cells or other unique power sources.

EMC Drive-By-Wire System

There were three major modifications to the vehicle in order to prepare it for the Grand Challenge – installation of the EMC system, a SuperLift chasis lift, and a heavy duty alternator. Immediately after purchase, the Meteor was modified by EMC (Electronic Mobility Controls Corp) to provide a drive-by-wire capability. This included modifications to both the transmission and steering column. EMC's main business is providing after-market modifications for the handicapped community. By using EMC, the Meteor has a robust drive-by-wire capability that leverages years of investment and experience. The EMC Grand Challenge installation includes a manual override capability that immediately returns steering and throttle control to an operator in the vehicle with the touch of a button.

Other modifications include a heavy duty 220 amp alternator that removed the need for additional batteries or bulky generator to power the computers and sensors. A 4" suspension lift (provided by SuperLift Suspensions) was added to increase ground clearance. Additionally, a set of four off-road Super Swamper tires was also added to increase reliability in rugged terrain.

2. Autonomous Operations

2.1. Processing

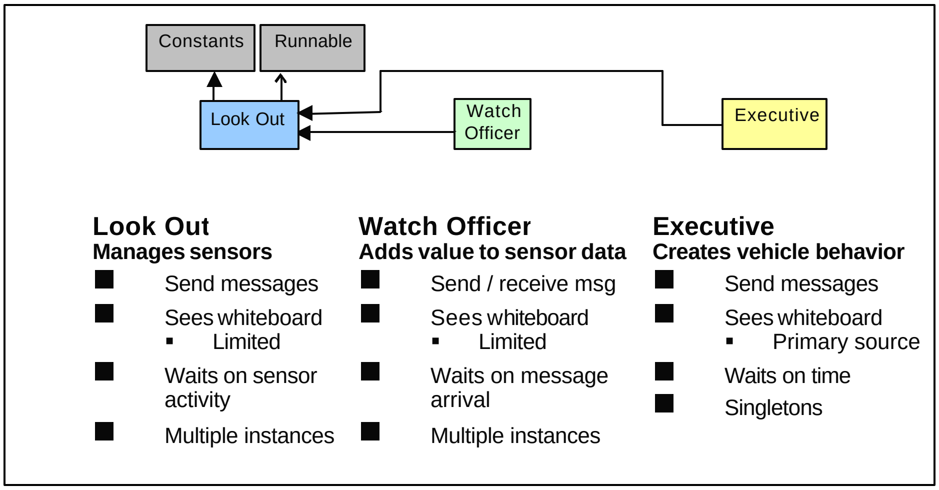
- *2.1.1. Describe the computing systems (hardware and software) including processor selection, complexity considerations, software implementation and anticipated reliability.*
- *2.1.2. Provide a functional block diagram of the processing architecture that describes how the sensing, navigation and actuation are coupled to the processing elements(s) to enable autonomous operation. Show the network architecture and discuss the challenges faced in realization of the system.*
- *2.1.3. Describe unique methods employed in the development process, including modeling-driven design or other methods used.*

The computing infrastructure is provided by a 9 board, military-hardened, VME computer array of 1.8 GHz Pentium processor boards connected by a gigabit Ethernet network shock mounted in a 19" rack. The software developed for the Meteor was written in Java and runs on these systems under the Linux operating system. The development environment consists of JDK 1.5, Subversion, Ant, and Eclipse. In addition to the computing hardware, the rear passenger bay contains a rack mounted monitor, keyboard, and mouse providing the ability to oversee and interact with the system during testing.

A major challenge of the system has been the self imposed requirement that the system be reusable and adaptable to the needs of a variety of our sponsors. By isolating specific hardware configurations and implementations, we believe that it would be straight forward to move the Meteor software architecture to a retrofitted Humvee or ATV. To support this challenge we have chosen an agent based architecture.

Functionality is achieved through agents that communicate with each other through messages. Agents are of three basic types: Look Outs, Watch Officers, and Executives. Look Out agents manage sensors and convert raw sensor information into system messages. Watch Officers process and fuse sensor data from one or more Look Outs to provide higher order information such as vehicle pose, obstacle definitions and ground plane estimates. Executives manage Watch Officers to make the decisions that ultimately control the actions of the Meteor.

The information flow is from the Look Outs managing the sensors to the Watch Officers to the Executives and finally to the actuators. Information that needs to be shared is posted to a whiteboard. This design allows for sensors to be dynamically added and subtracted to the system at runtime. Information dependencies are managed explicitly at the executive level. Meteor agents can run on any of the machines allowing for processor load balancing.



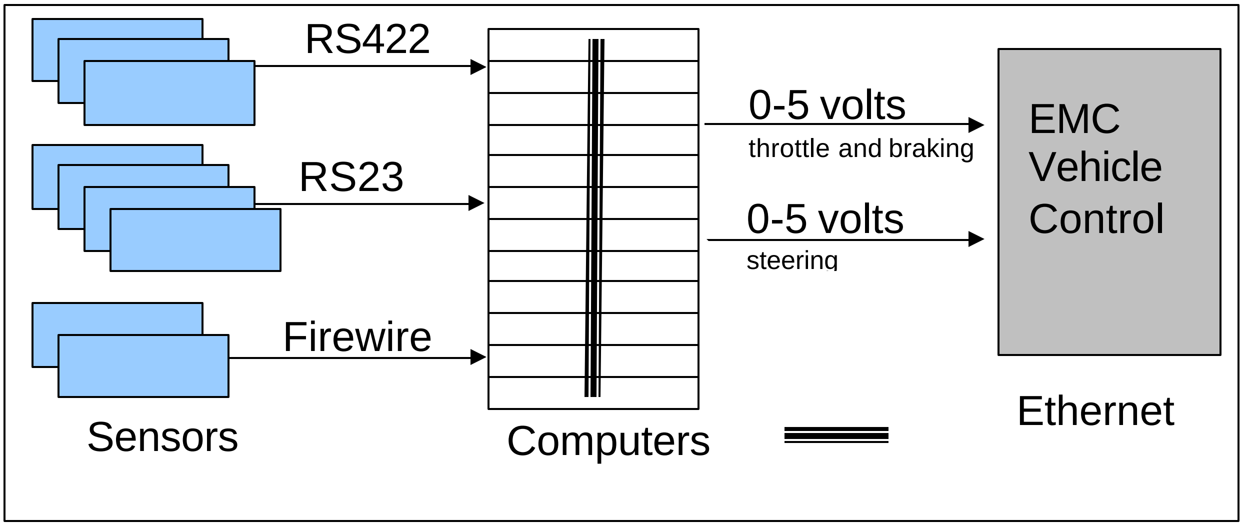
Look Out
Manages sensors
- Send messages
- Sees whiteboard
  - Limited
- Waits on sensor activity
- Multiple instances

Watch Officer
Adds value to sensor data
- Send / receive msg
- Sees whiteboard
  - Limited
- Waits on message arrival
- Multiple instances

Executive
Creates vehicle behavior
- Send messages
- Sees whiteboard
  - Primary source
- Waits on time
- Singletons

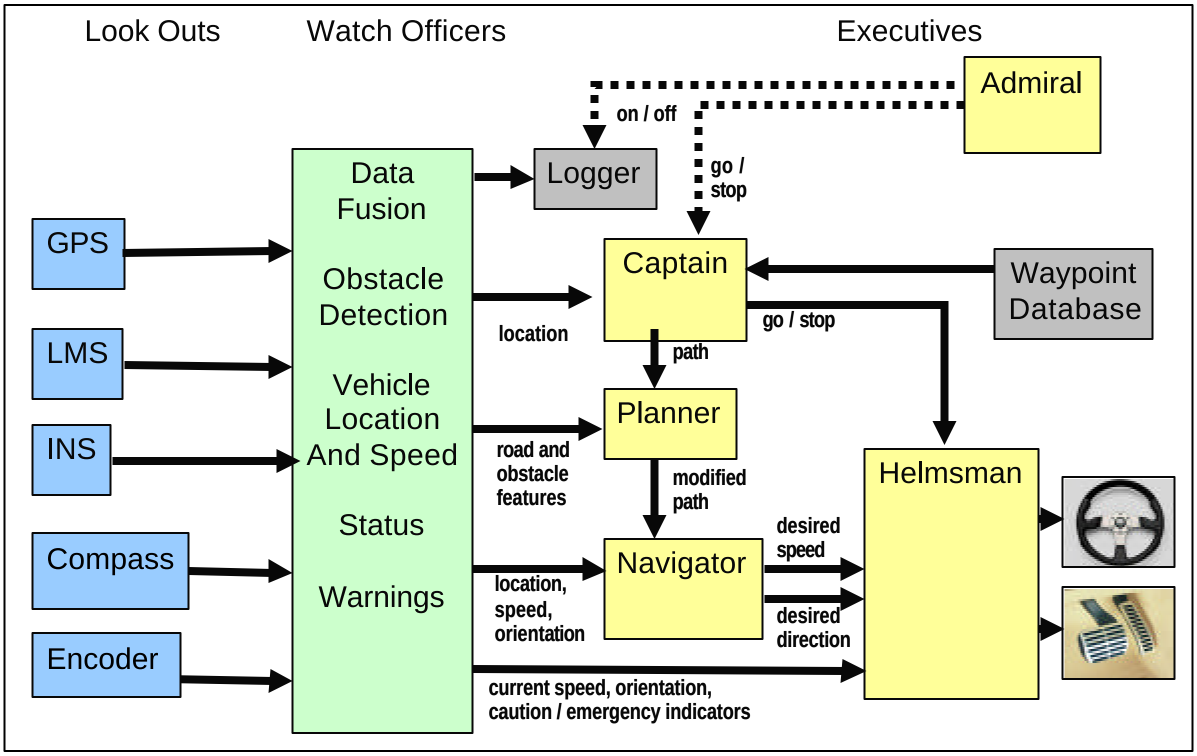
Basic Agent Classes

The network topology for this architecture is straight forward. Each of the sensors is connected to a port on one of the computers. The associated Look Out agent communicates with the port and makes the data available to the rest of the system. A high speed Ethernet logically connects the computers together. Ultimately every action that is preformed by the system results in the control of two 5v signals. One signal drives the brake/throttle servo of the EMC, the other controls the steering servo.



Network Topology

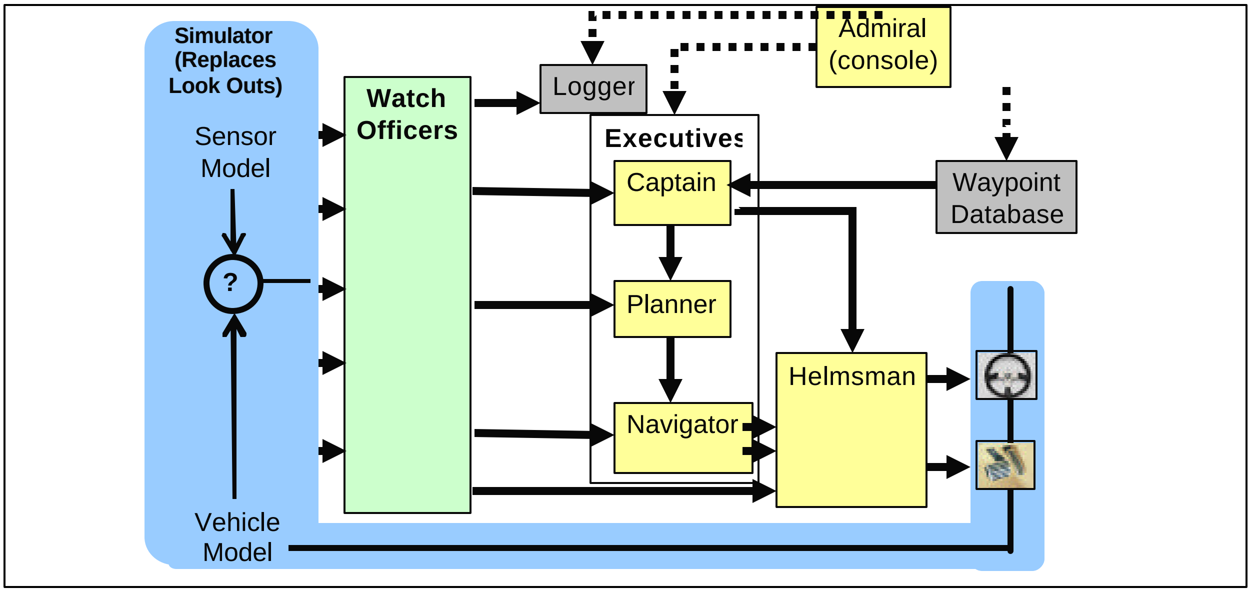
To implement this architecture we have developed a number of specialized executives. The Captain takes the RDDF file and generates the legal corridor in which the vehicle can operate. The Planner takes the legal corridor and obstacles to generate a path. The Navigator takes this path and determines a desired speed and direction based on the vehicle's current state. The desired speed and direction are then passed to the Helmsman which converts these parameters into the voltages that drive the EMC system.



Meteor High Level Architecture

The Meteor development has been driven by two overarching themes. The first is to do small increments of a develop, simulate, test, and regression cycle. The second is to continuously develop an end-to-end system built with agents of comparable complexity and quality. This approach means at any time the vehicle has all the necessary components to operate and shifts the emphasis from novel ideas to the interaction and integration of agents.

One of the unique features of our architecture is that it is insensitive to the input that drives it. The system behaves nearly identically whether driven by simulation, replay or live data. As new agents were developed during the design cycle they were first run in the simulator. Once the agent behavior was acceptable, it was tested in the field. All sensor data and inter-agent messages were recorded. Post analysis was performed by analyzing performance in replay. Once an agent passed its specific test, it was included in future regression cycles to evaluate its impact on the remainder of the system. Specific testing and regression testing was performed nearly daily for short focused evaluations. The behavior of the Meteor during the test would then (1) drive refinements to the simulator to more accurately reflect the demonstrations and (2) lead to new improvements in the software. Although our schedule was very aggressive, discipline was maintained to not allow several changes to be integrated simultaneously and to insist on regression testing to verify that previous performance was maintained with new versions. Regression is our term for periodically rerunning previous successful experiments verifying that any new changes have not 'undone' progress. Again, this testing was done in three modes, driven by the simulator, replaying of recorded data and vehicle integration test.



Bootstrapping with the Simulator

This incremental build philosophy has been in place from the start. For example, our vehicle was returned from SuperLift in early December and within the month the initial versions of the simulator, Look Outs, Watch Officers, and Executives were functioning and the Meteor was capable of driving autonomously. By starting with a simple configuration (one GPS, one LMS and a laptop) we had a system that was capable of driving from waypoint to waypoint while avoiding simple obstacles. We kept the system as simple as we could, only adding complexity when driven by performance requirements. A component, whether hardware or software, was moved from the test configuration to the operational configuration after it had demonstrated its utility and its performance characteristics and failure modes were understood.

2.2. Localization

2.2.1. Explain the GPS system used and any inertial navigation systems employed during GPS outages (as in tunnels). Include a discussion of component errors and their effect on system performance.

2.2.2. If map data was an integral part of the vehicles navigation system, describe the requirements for this data and the way in which it was used.

Localization is accomplished by fusing input from multiple GPS units, a magnetic compass, inertial navigation system, and several shaft encoders. Two Trimble GPS systems provide sub-meter accuracy through an Omnistar subscription. The Trimble GPS units are used primarily by the agriculture community for autonomous field preparation and harvesting. A third GPS is provided by a MIDG-2 inertial navigation system that comes from the remote controlled plane community. This GPS unit is augmented by an internal Inertial Measurement Unit that maintains location during GPS outages. In addition, a Honeywell magnetic compass is used as an alternative source of heading. Additionally, shaft encoders provide odometry at very slow speeds providing information that is needed for dead reckoning.

The team elected not to use terrain and elevation data. Instead we have outfitted the vehicle with multiple terrain sensors that provide local terrain information. While terrain analysis may be considered for specific applications in the future, the ability to perform without this information is of interest to many of our sponsors.

2.3. Sensing

2.3.1. Describe the location and mounting of the sensors mounted on the vehicle. Include a discussion of sensor range and field of view. Discuss any unique methods used to compensate for conditions such as vibration, light level, rain and dust.

2.3.2. Discuss the overall sensing architecture, including any fusion algorithms or other means employed to build models of the external environment.

2.3.3. Describe the internal sensing system and architecture used to sense the vehicle state.

2.3.4. *Describe the sensing-to-actuation system used for waypoint following, path finding, obstacle detection, and collision avoidance. Include a discussion of vehicle models in terms of braking, turning, and control of the vehicle*

The Meteor has three classes of sensors for detecting obstacles and ground terrain. Eight SICK laser range finders provide a two-dimensional range/distance map up to 40 meters with a 100 degree field of view. Two vertically mounted lasers provide ground plane information. Three lasers are mounted horizontally at different angles allowing for both short and long range detection of road obstacles. A fourth laser is mounted on a gimbal that allows for dynamic pointing in order to compensate for vehicle pitch and terrain variations. Finally, two downward looking lasers are mounted on the roof to derive road characteristics.

Additional sensors include an Eaton Vorad doppler range rate radar developed for the trucking industry. The radar provides detection of motorcycle-sized objects at a distance of 200 meters. The radar provides vehicle detection in smoke or fog. In addition to the obstacle detection sensors, the Meteor has a vision system to detect the edges of and center of the road. We are currently experimenting with this system to determine its best role in the architecture.

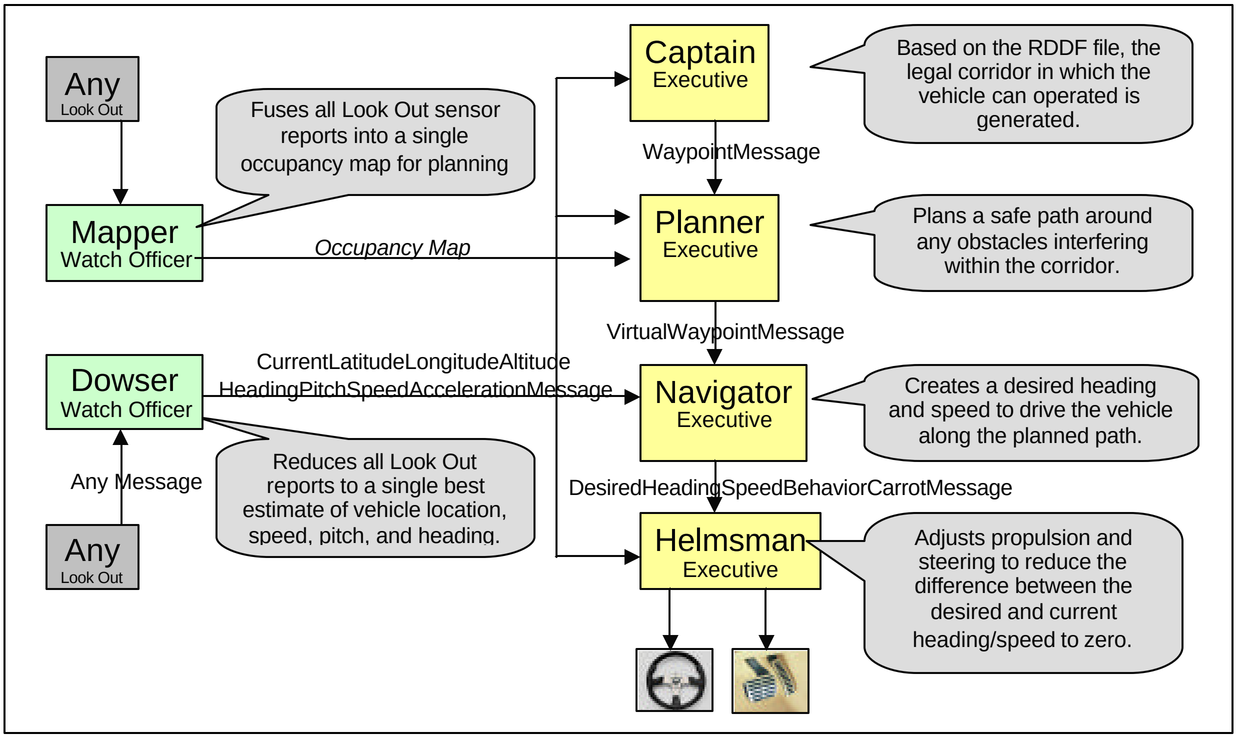
Sensor Configuration

Information from the obstacle detection sensors are fused in a sliding window occupancy map. As obstacles are repeatedly detected in a specific location, the occupancy map builds confidence as to the traversability of that space. During normal operations, the vehicle plots a path through occupancy space that best avoids obstacle while staying within the route boundaries.

2.4. *Vehicle Control*

- *2.4.1. Describe the methods employed for common autonomous operation contingencies such as missed-waypoint, vehicle-stuck, vehicle-outside-lateral-boundary-offset, or obstacle-detected-in-path.*
- *2.4.2. Describe the methods used for maneuvers such as braking, starting on a hill, or making a sharp turn without leaving the route boundaries.*
- *2.4.3. Describe the method for integration of navigation information and sensing information.*
- *2.4.4. Discuss the control of the vehicle when it is not in autonomous mode.*
  (covered in section 1.2)

The Meteor defines a legal corridor based on the RDDF file. Within the legal corridor and based on the occupancy map, the planner considers many alternative paths several times per second. In order to maintain stability, a preference is given to the present path. A set of rules is checked in order to determine whether a waypoint has been missed, if the vehicle is out of bounds, or if obstacles are in the vehicle's planned path. An energy based model is used to determine the desired adjustments to heading and speed to drive the vehicle along the planned path. These adjustments as well as vehicle pitch are used to compute the parameters that are sent to the EMC controller. This allows the throttle, brake, and steering wheel to be positioned in order to maneuver the vehicle along the planned path while taking into account vehicle state and orientation.



Watch Officers and Executives

2.5. System Tests

2.5.1. Describe the testing strategy to ensure vehicle for DGC, including a discussion of component reliability, and any efforts made to simulate the DGC environment.

2.5.2. Discuss test results and key challenges discovered.

The end to end testing of the Meteor was performed in stages. Early on we had many short specific tests on a nearly daily basis. Specific tests included a series of vibration and sensor fouling experiments. As the site visit approached we became focused on meeting the specific challenges of the site visit and focused specifically on the waypoint following and trash cans as obstacles. In July we went to the Mojave Desert to test the fully integrated vehicle. We tested for distance, responsiveness to the environment, effects of terrain and overall reliability. This was sufficient to convince us we could compete in the DGC. The final testing phase is emulating the NQE environment and identified NQE evaluation components.

Overall this activity has been an exciting challenge. Given our incredibly short time to prepare, a key challenge for us was to sustain a rapid pace of incremental development while maintaining system coherence. In order to ensure what we learn is of high utility to our sponsors we also had a self imposed challenge of reusability and extensibility of design and code.

Team Members

Meteor team members from left to right Mark Heslep, Dr. Richard Weatherly, Frank Carr, Ann Jones, Dr. Garry Jacyna, Dr. Robert Grabowski, Robert Bolling, Michael Shadid, Sarah O'Donnell, David Smith, Dr. Alan Christiansen, Dr. Thomas Bronez, Laurel Riek, Kevin Forbes, not shown: Keven Ring