A Robotic Dune Buggy Named ‘Tommy’

Paul J. Perrone

Founder/CEO
Paul J. Perrone
www.perronerobotics.com

TS-3966

Goal of Talk

Story and use of Java™ technology in a robotic dune buggy

Learn how a 100% Java technology-based robotic dune buggy was developed for an historic race of driverless robotic vehicles, and the unique and dramatic story of this vehicle named ‘Tommy’

Agenda

Autonomous Robotic Vehicles

The DARPA Grand Challenge

“Tommy”

Tommy at the Grand Challenge

Tommy Evolves

Concluding Remarks

Agenda

Autonomous Robotic Vehicles

The DARPA Grand Challenge

“Tommy”

Tommy at the Grand Challenge

Tommy Evolves

Concluding Remarks

Autonomous Robotic Vehicles?

Transfer of control from man to machine

- UGV = Unmanned Ground Vehicle
  - No human required onboard
  - Tele-operated, autonomous, or hybrid
- AGV = Autonomous Ground Vehicles
  - No human onboard or in remote operation
  - Complete a mission over terrain in time
  - Navigate and avoid obstacles
- Evolving autonomy, distances, and speeds
- Key to remember
  - Transfer of control not complete
  - Experimental technology
  - Human emergency safety override is critical

The Applications

Why is this happening?

- Surveillance and security
- Payload delivery and transport
- EOD and mine detection
- Force protection and replacement
- Farming
- Auto safety overlays
- Self-driving cars
- Related applications
  - Hospital carts
  - Personal robotics
  - Professional robotics
  - And much more

Agenda

Autonomous Robotic Vehicles

The DARPA Grand Challenge

“Tommy”

Tommy at the Grand Challenge

Tommy Evolves

Concluding Remarks

What Was the DARPA Grand Challenge?

AGV desert robot competition

- Race:
  - @150 mile race through desert terrain
  - First to complete course in < 10 hours
  - Coordinates given 2 hours prior to race
  - Winner gets cash prize
- Vehicle:
  - Completely autonomous
  - No external communication
  - DARPA has E-stop controls
  - No fed funded technology

Footnote: Image from DARPA sponsorship promotion package

Grand Challenge Highlights

Major advancement in one year's time

- 2004 Challenge
  - $1M prize, 142 miles, rough terrain
  - Farthest was 7.4 miles, others 5–6 miles, 1 mile, start area, few withdrawn
- 2005 Challenge
  - NQEs: @2 mile obstacle course
  - $2M prize, 132 miles, smoother terrain
  - Winner in @7 hrs, 3 others completed in time, 1 more completed in @13 hrs

Agenda

Autonomous Robotic Vehicles

The DARPA Grand Challenge

“Tommy”

Tommy at the Grand Challenge

Tommy Evolves

Concluding Remarks

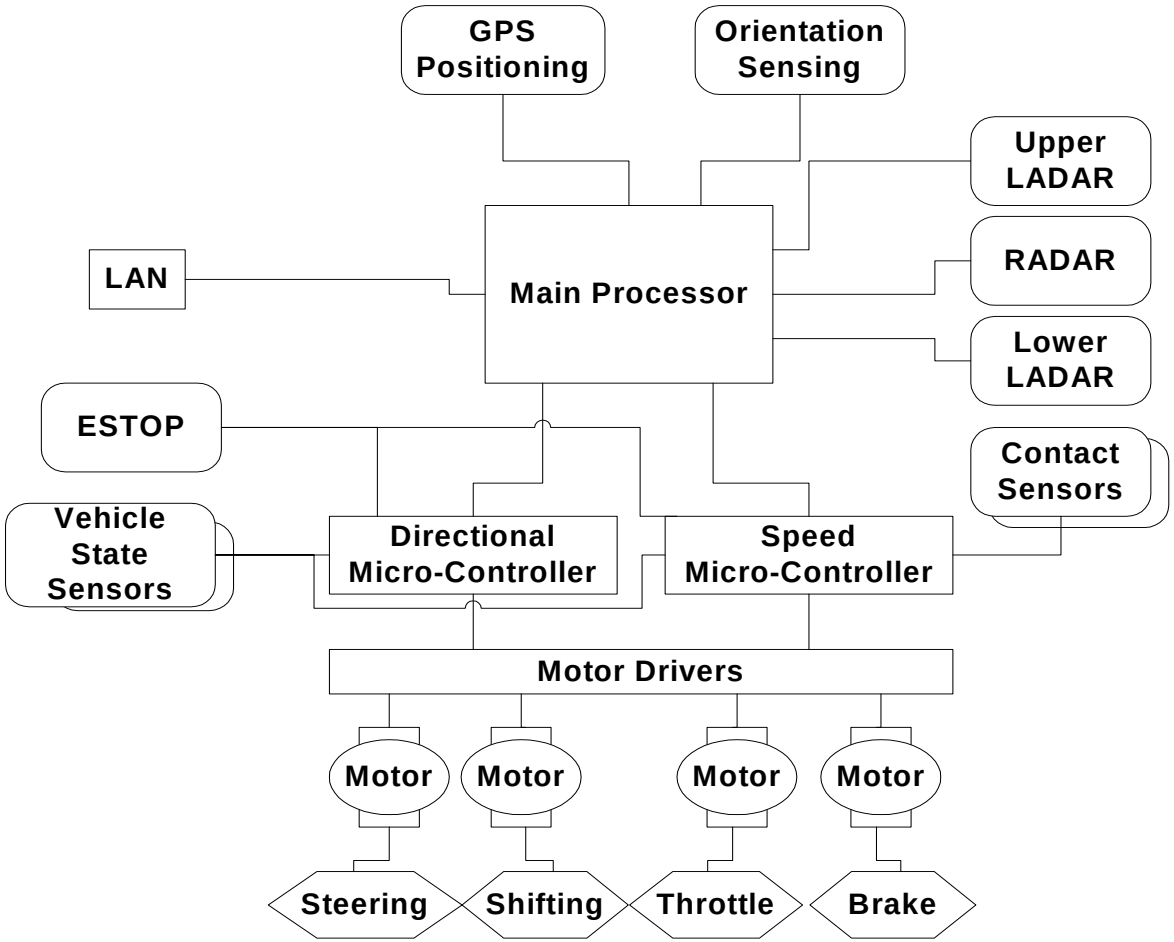
Overview of Tommy

The silver egg that could

- Custom built dune buggy
- Silver egg shell
- COTS Electronics
- COTS Sensors
- COTS Actuators
- MAX™ Software

System Architecture

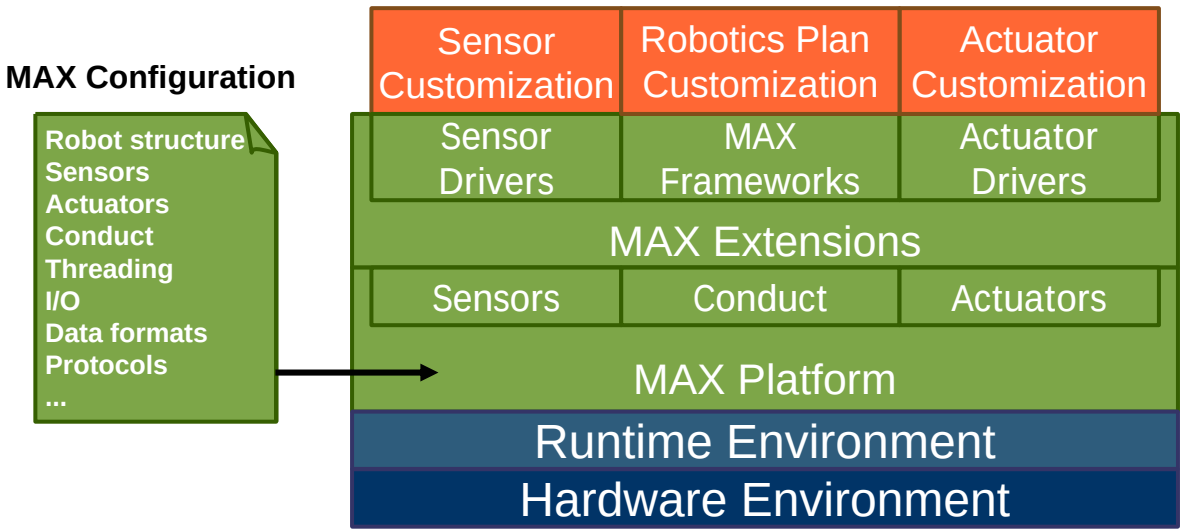
Rapid configuration of low-cost COTS hardware



Robotics Software Engine

Faster and cheaper path to realization

- Perrone Robotics' patent pending MAX platform
- General purpose robotics platform
- Configuration of built-in services
- Used available and added drivers and extensions

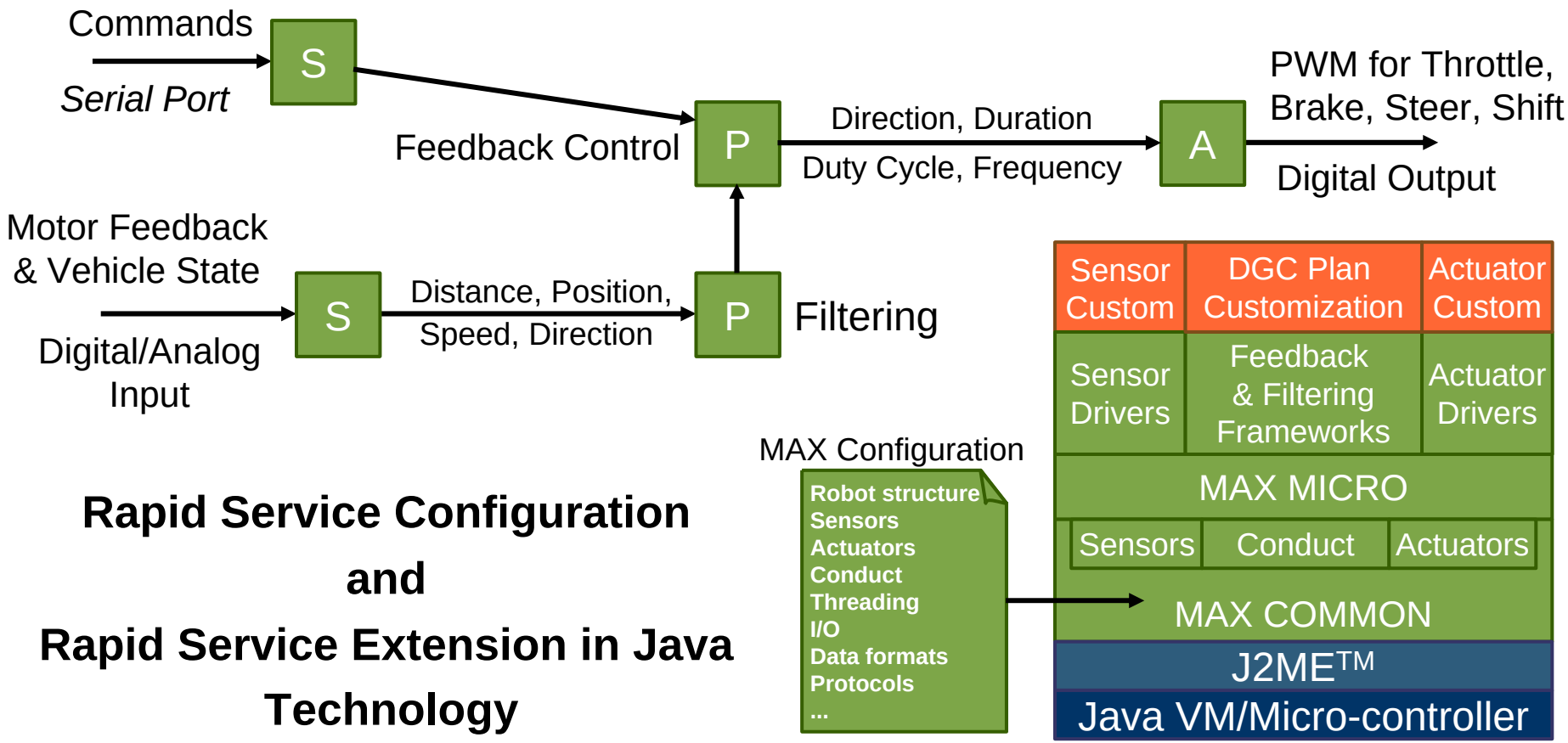


General MAX Operation Example

```
import com.perronerobotics.plan.*;
import com.perronerobotics.sensor.*;
import com.perronerobotics.actuator.*;
import com.perronetech.system.*;
import com.perronetech.id.*;

public class MyPlan extends PlanGeneric{
  public void init(ID callerID){
    // Get components (threading, I/O, management, distribution, etc configured)
    sensor = (Sensor) super.getSensor(SENSOR_ID);
    actuator = (Actuator) super.getActuator(ACTUATOR_ID);
    actuatorPlan = actuator.getActuatorPlan();
    component = (Component) super.getComponent(COMPONENT_ID);
  }
  public void trigger(ID callerID){
    // Get some sensor state value
    SensorState state = sensor.getSensorState();
    double stateValue = state.getDouble(STATE_VALUE_ID);

    // Get a desired state value from some example component
    double desiredValue = component.getDouble(DESIRED_VALUE_ID);

    // Schedule the actuator to move to a position based on difference
    actuatorPlan.setPosition(desiredValue – stateValue);
  }
}
```

Java Platform, Micro Edition Control

Java technology-based control of actuation and feedback



Rapid Service Configuration

and

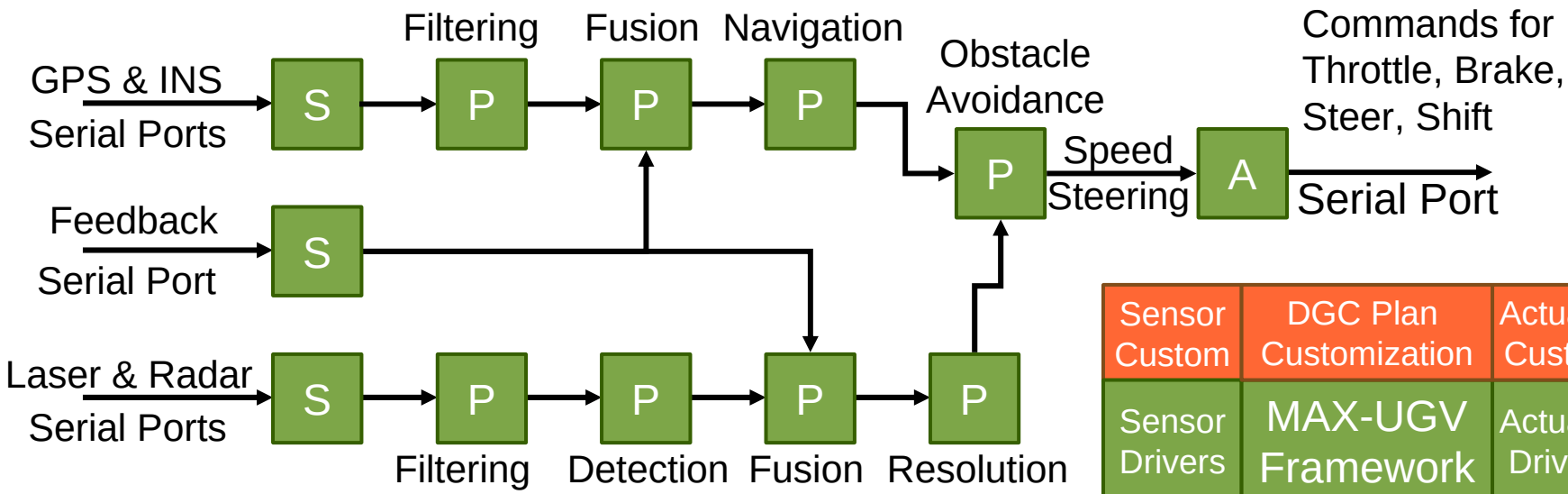
Rapid Service Extension in Java Technology

MAX Micro Profile Example

```
import com.perronerobotics.plan.*;
import com.perronerobotics.command.*;
import com.perronerobotics.actuator.*;
import com.perronerobotics.feedback.*;
import com.perronetech.id.*;

public class MyCommandFeedbackControlPlan extends PlanGeneric{
  public void init(ID callerID){
    // Get components (threading, I/O, management, distribution, etc configured)
    commandSensor = (CommandSensor) super.getSensor(COMMAND_SENSOR_ID);
    steerControl = (FeedbackControl) super.getComponent(STEER_FEEDBACK_COMPONENT_ID);
    steeringFeedbackSensor = (Sensor) super.getSensor(STEERING_FEEDBACK_SENSOR_ID);
    steeringMotor = (Actuator) super.getActuator(STEERING_MOTOR_ID);
    steerControl.registerSensor(steeringFeedbackSensor);
    steerControl.registerActuator(steeringMotor);
  }
  public void trigger(ID callerID){
    // Get commanded steering value
    SensorState state = commandSensor.getSensorState();
    double steerValue = state.getDouble(DESIRED_STEERING_VALUE_ID);

    // Set desired steering value
    steerControl.setDesiredPosition(steerValue);
  }
}
```

Java Platform, Standard Edition AI

Java technology-based brainwork



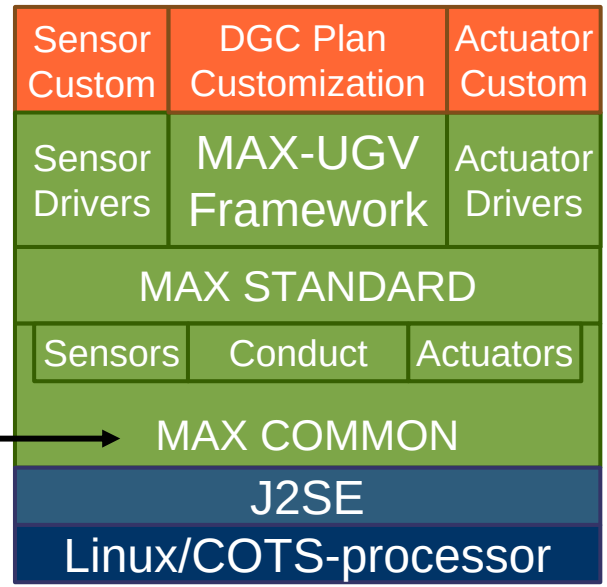
Rapid Service Configuration

and

Rapid Service Extension in Java Technology

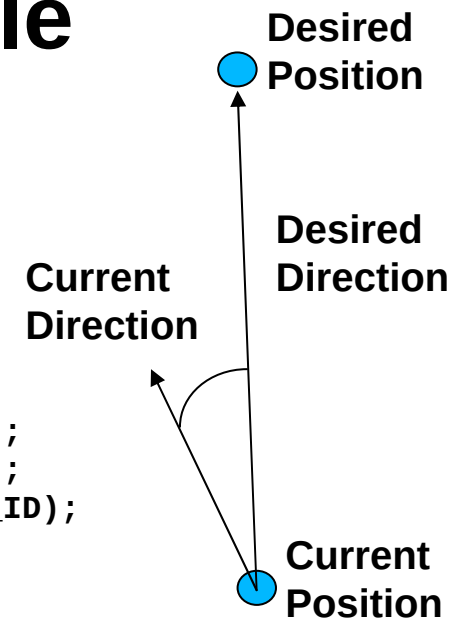
MAX Configuration

- Robot structure
- Sensors
- Actuators
- Conduct
- Threading
- I/O
- Data formats
- Protocols
- ...



MAX Standard Profile Example

```
import com.perronerobotics.plan.*;
import com.perronerobotics.reference.*;
import com.perronerobotics.actuator.*;
Import com.perronerobotics.navigation.route.*;
import com.perronetech.id.*;

public class MyNavigationPlan extends PlanGeneric{
  public void init(ID callerID){
    // Get components (automatically configured)
    gpsSensor = (ReferenceSensor) super.getSensor(GPS_SENSOR_ID);
    insSensor = (ReferenceSensor) super.getSensor(INS_SENSOR_ID);
    steeringMotor = (Actuator) super.getActuator(STEERING_MOTOR_ID);
    steeringPlan = steeringMotor.getActuatorPlan();
    route = (Route) super.getComponent(ROUTE_ID);
  }
  public void trigger(ID callerID){
    // Get current & desired position…and calculate desired direction
    Position currentPosition = gpsSensor.getPosition();
    Position desiredPosition = route.getDesiredPosition();
    Orientation direction = desiredPosition.calculateDirection(currentPosition);
    // Get current orientation…and set desired steering based on desired direction
    Orientation currentOrientation = orientationSensor.getOrientation();
    Orientation directionDiff = currentOrientation.calculateDifference(direction);
    steeringPlan.setPosition(directionDiff.getHeading().getValue());
  }
}
```



Source: Perrone Robotics, Inc. Copyrights reserved, MAX trademarked, & MAX technology patent pending.

Off-Line Route Planning

Java technology-based planning of route

- Inputs
  - RDDF file: waypoints, boundaries, and max speeds
  - Geo data: elevation and features
- Fair-Isaac Blaze Advisor Rules
  - Import RDDF data
  - Parse geo data
  - Refine route
- Output:
  - Modified route
  - Intermediate waypoints
  - Refined speeds

Agenda

Autonomous Robotic Vehicles

The DARPA Grand Challenge

“Tommy”

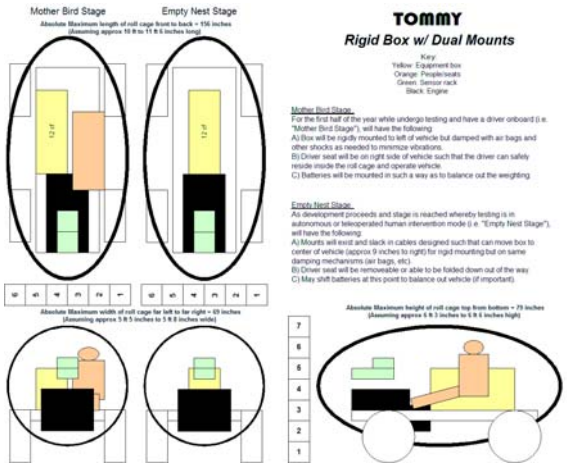
Tommy at the Grand Challenge

Tommy Evolves

Concluding Remarks

Tommy's Journey to the Challenge

From concept to trip down Route 66 with bot 06



Tommy at the Qualifying Events

Vetting the “Fast and Cheap” mantra

- Tommy as media darling
- E-stop snafus
- Increasing rank
- Perfect 3rd run

Tommy's Brush with a Wall

The "Out of Control" part

- Day after perfect run…
- Unexplained acceleration
- Requests for an E-stop
- No E-stop issued
- 13 seconds to impact
- Tommy is totalled

Tommy's Repair

Totalled AGV = game over?

- 36 hour rebuild
- Source new engine
- Replace front end
- Calibrate in dark
- Tommy back to track
- Bar raised on last run

Alive and Well in the Desert

Fear and loathing behind Wild Bill's

- Few degree steering offset
- Scores still indicated were in
- Not allowed in race
- Quick steering offset fix
- All systems go again
- Went for desert run

Tommy's Comparison

How Tommy stacked up

- Tommy project:
  - $60k in parts/tools and $30k travel
  - 10 man-months software development time
  - Used single processor card (< $200 retail)
- Other project examples:
  - Well financed institutions with prior/current federal funding
  - $3M+ budgets
  - 100+ man-years
  - Banks of powerful processors
  - Some teams had multiple spare vehicles
- Tommy's scores compared to some others allowed in:

	Time	Gates	Obstacles	Time	Gates	Obstacles	Time	Gates	Obstacles	Time	Gates	Obstacles
Team Jefferson	x	10	0	x	16	0	x	36	1	18	49	5
Mitre	x	0	0	x	0	0	x	10	0	21	44	5
Gray Team	x	2	0	x	14	0	x	15	0	16	48	5
Va Tech GC Team	x	0	0	x	5	0	x	16	0	17	44	3
Cajunbot	x	0	0	x	18	0	x	29	1	16	49	4

VIDEO

Tommy and the DARPA Grand Challenge

Agenda

Autonomous Robotic Vehicles

The DARPA Grand Challenge

“Tommy”

Tommy at the Grand Challenge

Tommy Evolves

Concluding Remarks

Tommy Now

Robots are stubborn

- Tommy had some cosmetic repairs
- Used as R&D platform
- Demos/exhibitions: here at JavaOne^{SM} conference!
- Basis for turn-key AGV dune buggy platforms
- Ready for next Grand Challenge
- Star in upcoming documentary “Autopilots”

Tom Jr.

Carrying on the Tommy legacy

- Smaller integrated hardware/ software/mobility platform
- Mobility based on off-road go-kart
- Starter platform for AGV R&D and applications
- Modular software, hardware, and mechanical parts
- R&D, hobbyist, commercial, and military profiles

Agenda

Autonomous Robotic Vehicles

The DARPA Grand Challenge

“Tommy”

Tommy at the Grand Challenge

Tommy Evolves

Concluding Remarks

Reflections

Why did I pursue the DARPA Grand Challenge?

- How general purpose and scalable was MAX?
  - Started with rat and cat sized bots
  - Would MAX scale to complex elephant sized AGV aps
- Results
  - $90K and 10 months later
  - Faster and cheaper approach to complex AGV development
- Cautions
  - Any part of a system can fail (electrical in our case)
  - Human vigilance and E-stop caution is key

What Now?

Tommy as ‘robot pinup’ for our Java-based dreams

- Build MAX community (more fun in numbers)
- Provide MAX, MAX extensions, and services
- Provide Tom Jr. for others to build AGV aps
- Evolve Tommy for next Grand Challenge
- Watch Tommy hit the big screen with ‘Autopilots’

For More Information

- Sponsors (Tier I)
  - Sun Microsystems
  - Fair-Isaac
  - Perrone Robotics
  - Assured Technologies
- Technology
  - Java technology (java.sun.com)
  - MAX (www.perronerobotics.com)
  - Tommy (www.teamjefferson.com)
- Documentary
  - Autopilots (www.robotworldmedia.com)

Q&A

Paul J. Perrone

CEO, Perrone Robotics

Team Lead, Team Jefferson

A Robotic Dune Buggy Named ‘Tommy’

Paul J. Perrone

Founder/CEO
Paul J. Perrone
www.perronerobotics.com

TS-3966