CS 393R - Fall 2015

Lab Introduction

Jake Menashe
jmenashe@cs.utexas.edu

Original slides created by Todd Hester and Katie Genter

Outline

- Administrative
- Lab Information
- Robots
- Codebase
- Assignments
- Rules
- Tips

My information

- Office hours
  - Tuesdays and Thursdays 3:30pm to 5:00pm
  - By appointment
  - GDC 3.710A
- jmenashe@cs.utexas.edu
- Put 393R in subject lines

Teams

- Form a team of 2-3 students (3 preferred)
- Send me:
  - Team Info
  - CS Usernames
- Use the subject: [393R Team Info]

Lab information - Access

- Machine login
  - Username: your cs user name (____@cs.utexas.edu)
  - Send me your team info for access
- Permissions
  - Your directories and files will be readable by classmates by default
  - Your responsibility to change permissions and protect your work

Lab information - Hardware

- The lab has 10 workstations + 1 server
- Wireless access to robots through the server
- The server: r1-g4.csres.utexas.edu
  - DO NOT REBOOT
  - Avoid using
- Workstations
  - DO NOT REBOOT
  - /home and /usr/local are NFS mounted
- Let me know of any hardware problems
- Let me know if you need any new software

Lab Information - Robots

- Each team locker contains:
  - One Nao
  - One Charger
  - One Ethernet Cable
- You are responsible for all three.
- Breaking a Nao will slow you down and make your work significantly harder. Be careful with them!

Aldebaran Nao H25

- Multiple Sensors
  - Vision
  - Touch/Pressure Sensors
  - Accelerometers/Gyros
  - Sonar, Microphone
  - Buttons
- Multiple Effectors
  - Arms/Legs with 5 DOF
  - Head with 2 DOF
  - Pelvis, Hand, LEDs
- 1.6 GHz Single Core Atom Processor
- 1GB RAM
- Communication over LAN/WLAN

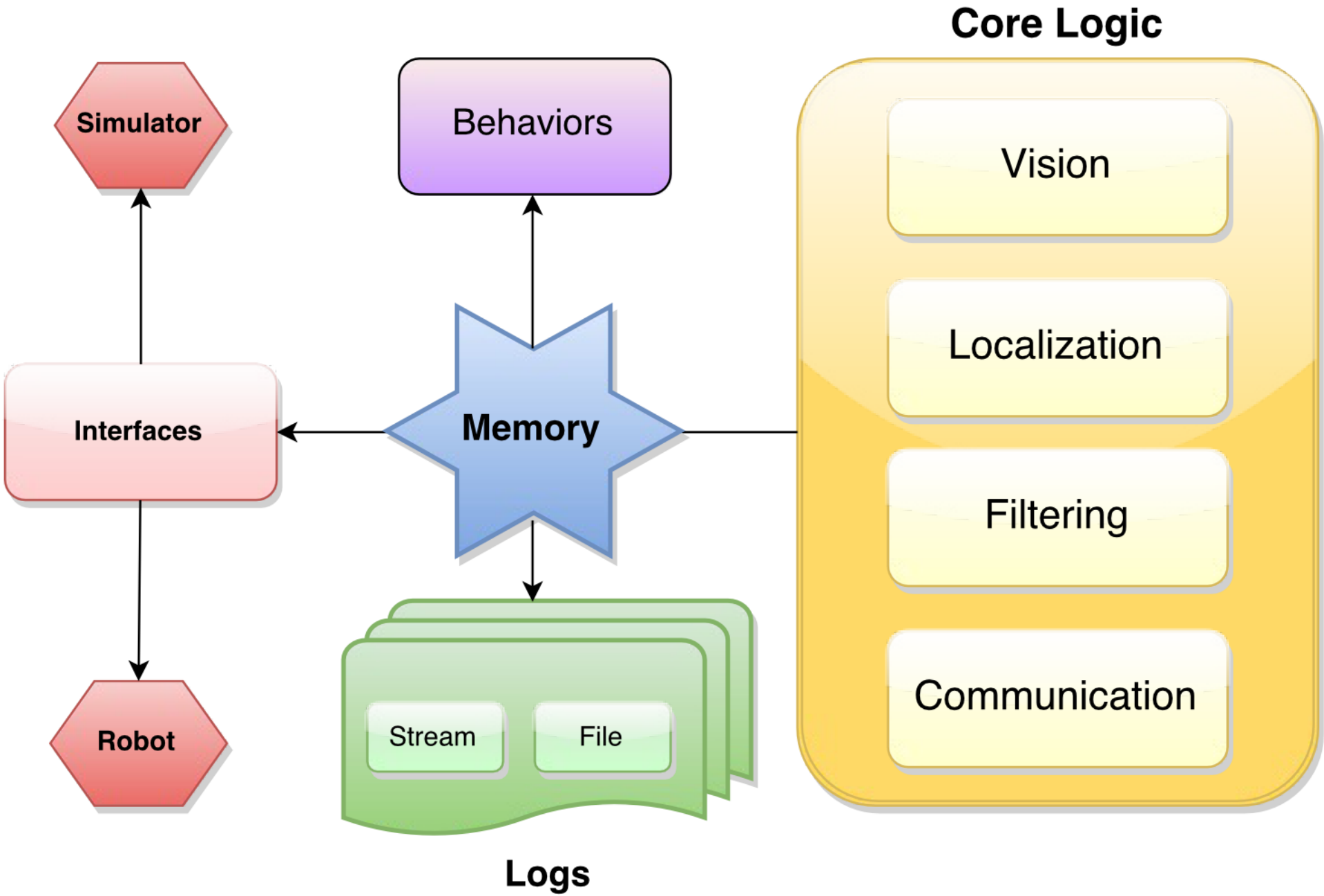
UT Austin Villa Codebase

- Assignments will use a stripped version of the UT Austin Villa codebase
  - C++11 modules
  - Python behaviors
- Contains many built in features you will need:
  - Pre-programmed walking/kicking
  - State machines
  - Sensor/Actuator interfaces
  - Debugging tools
- Documentation and access details on the course website

UT Austin Villa Architecture

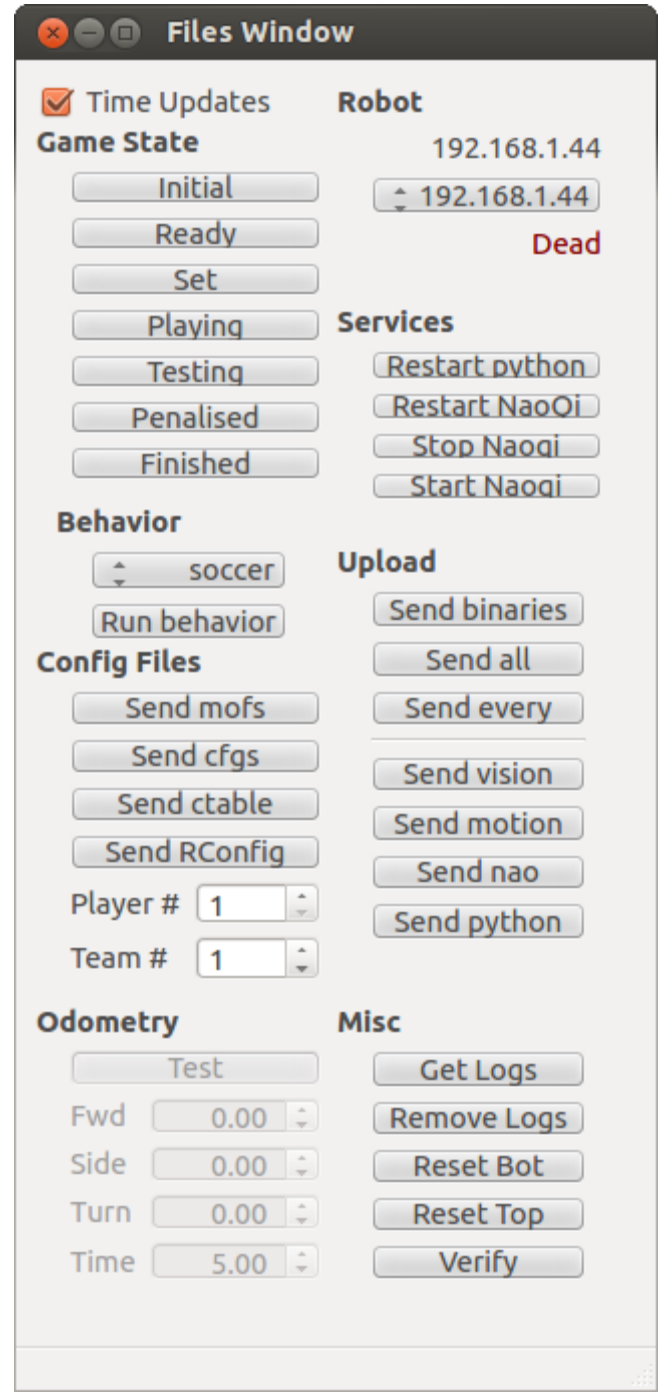
- You will be writing Nao behaviors in Python
- You will be writing vision and localization in C++11
- Behaviors
  - High level descriptions of robot actions and decision making
- Vision Module
  - Identify object locations from streaming images
- Localization Module
  - Determine the robot's location from nearby landmarks

UT Austin Villa Architecture



UT Austin Villa Tool

- View of camera and segmented image
- Can alter robot game states
- Can view and transfer files and logs
- Can simulate observations and behavior



Robot Care

- Robots - Fragile and Expensive
  - Don't set them on tables or chairs
  - Be aware of where they are at all times
  - Don't step on them or roll your chair into them
  - Don't let them walk into anything repeatedly
  - Don't force the joints to move once stiffness is enabled

- Let me know if you think your robot is broken or breaking

Power Management

- A battery will last up to 45 minutes depending usage
- Recharging can take some time - Keep the Nao charging whenever possible
- The bottom left eye LEDs indicate power. White is >90%, orange is >75%, red is <75%.

Heat Management

- Using the robot will cause its joint motors to heat up
  - Walking
  - Standing
  - Sitting down with stiff joints
- As the robot heats up its motors will become less responsive
  - Lots of falling over
- The top left eye LEDs indicate heat. White is <= 54°, Yellow is <= 64°, Orange is <= 74°, Red is > 74°

Lab Rules

- No food or drink
- Clean up after yourself
- Do not leave your robots unattended!
- Robots can only be used in the lab. They are never allowed to be removed for any reason.
- If no team members are present, the team's robot must be locked in its locker.
- Never give your locker key to anyone outside your group.
- Never tell the door combination to anybody outside of class.
- Double check that your locker is secure.
- Make sure the lab door locks if you are last to leave.
- Never give your robotics machine password to anyone.
- People who are not enrolled in CS393R:
  - May not be in the lab unless a class member accompanies them.
  - May not use any of the lab computers.
  - May not use any of the robots.

Breaking rules or robots can affect your grade.

Assignments

- Demonstrate that you meet all the assignment criteria
- Evaluations are done in person
  - Signup sheet posted to Piazza
- Evaluations are 10 minutes long.
  - No credit after 10 minutes.
  - One chance to get it right.
  - I will actively test for flaws in your solutions.
- You will turn in your code and memo
  - By email before class time
  - One email per team

Assignment One

Memo/Demo due 11:59:59 PM, Wednesday, Sept. 2

- Establish contact between your machine and the Nao
- Demonstrate you can read the sensors and display them
- Make the Nao move its head and walk
- Get started using color identified through the camera image
- Write a couple of simple control programs
  - Control the Nao's gaze to track the ball
  - Walk towards a blue goal
- Details on the course website

Tips

- The provided color table should work for most situations, but you may want to specialize it for this room and these assignments
- Read through the setup and tutorial documents.
- Do timed practice evaluations.
- Ask questions
  - Use Piazza if your question is relevant to other teams.
- **Work on assignments early.**
  - We have a large class this year.
  - Lab space is limited.
  - Wireless performs **very poorly** when lots of people use it.
  - Assignments can take 20-40 hours to complete

Tips - Start Early!

- Lots of things can go wrong in robotics:
  - Segmentation faults
  - Broken sensors and motors
  - Corrupted files (transfer or hardware failures)
  - Crowded lab space
  - Bad hardware initialization
  - Small environment variations
  - Network latency/outages
- Murphy’s Law - Anything that can go wrong, will
- Test

Tips - Git

- Use git
  - Distributed version control system
  - Simplifies:
    - Collaborating with your partner
    - Integrating codebase updates from the TA
    - Keeping track of code changes
    - Cloud storage for your work
- Use online git hosting
  - GitLab - Unlimited users, unlimited storage, unlimited private repos
  - Bitbucket - 5 users, 2GB, unlimited private repos

Recap:

1. Take care of your robots and the lab
2. Set up online git repositories
3. Ask questions on Piazza
4. Start assignments early
5. Prepare and practice for assignment demos
6. Email me your team info and cs usernames