FIRST Robotics and Java SE Embedded

CON5678

Derek White
Principal Member Technical Staff
Java SE team, Oracle

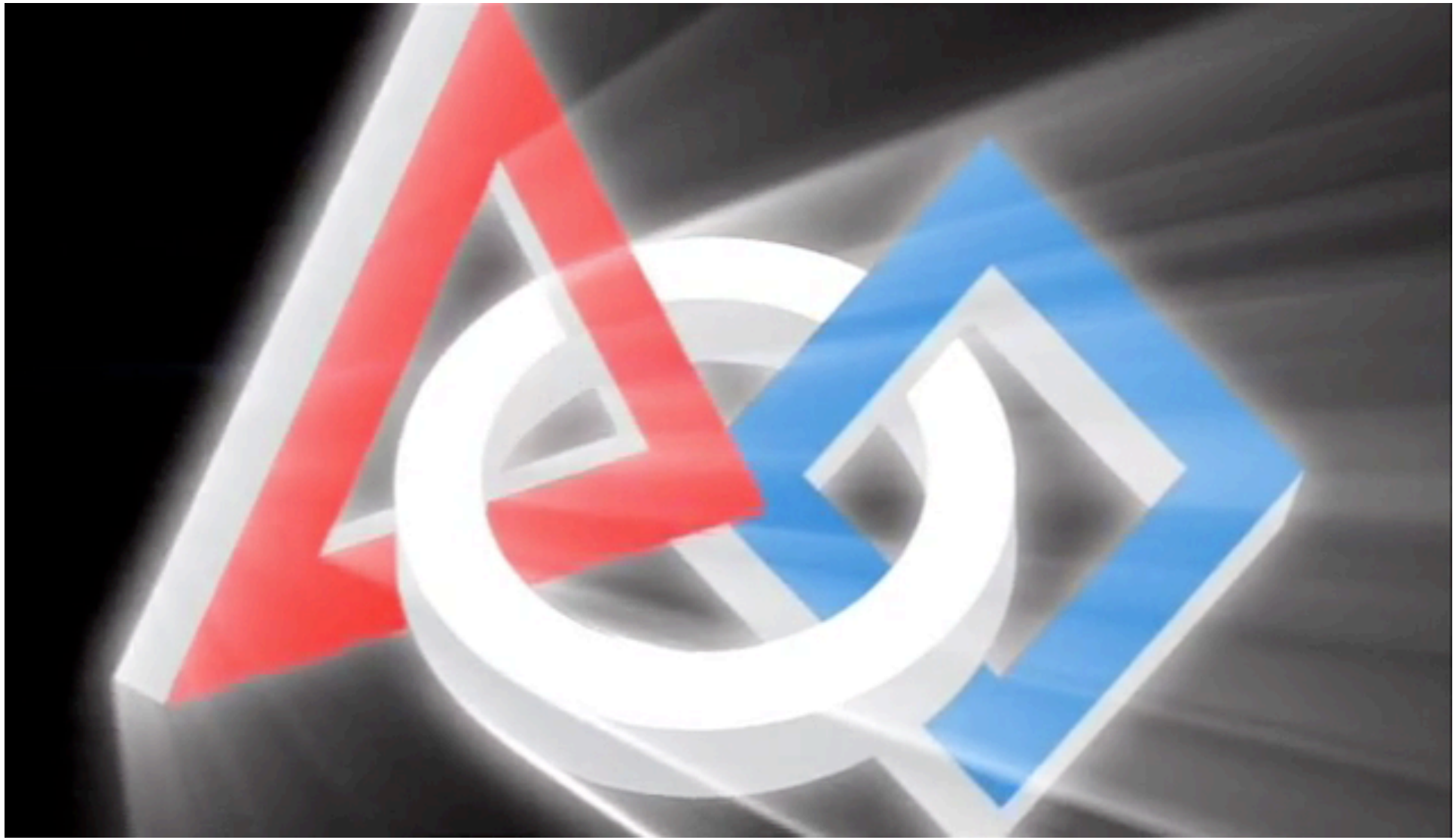
~~Brad Miller~~
~~Associate Director, Robotics Resource Center~~
~~Worcester Polytechnic Institute~~

October 1, 2014

~~Presented with~~

Outline

1. What is FIRST
2. FIRST Robotics Competition (FRC)
3. New for FRC in 2015
4. Java SE Embedded in Action
5. Next Steps…



What Is FIRST?

Set of 4 Robotics Competitions

- World-wide
- Ages 6-18
- New game every year
- Research
- Engineering
- Presentation
- Teamwork

FIRST Mission

"To transform our culture by creating a world where science and technology are celebrated and where young people dream of becoming science and technology heroes."

— Dean Kamen, Founder, FIRST

1) Junior FIRST LEGO League

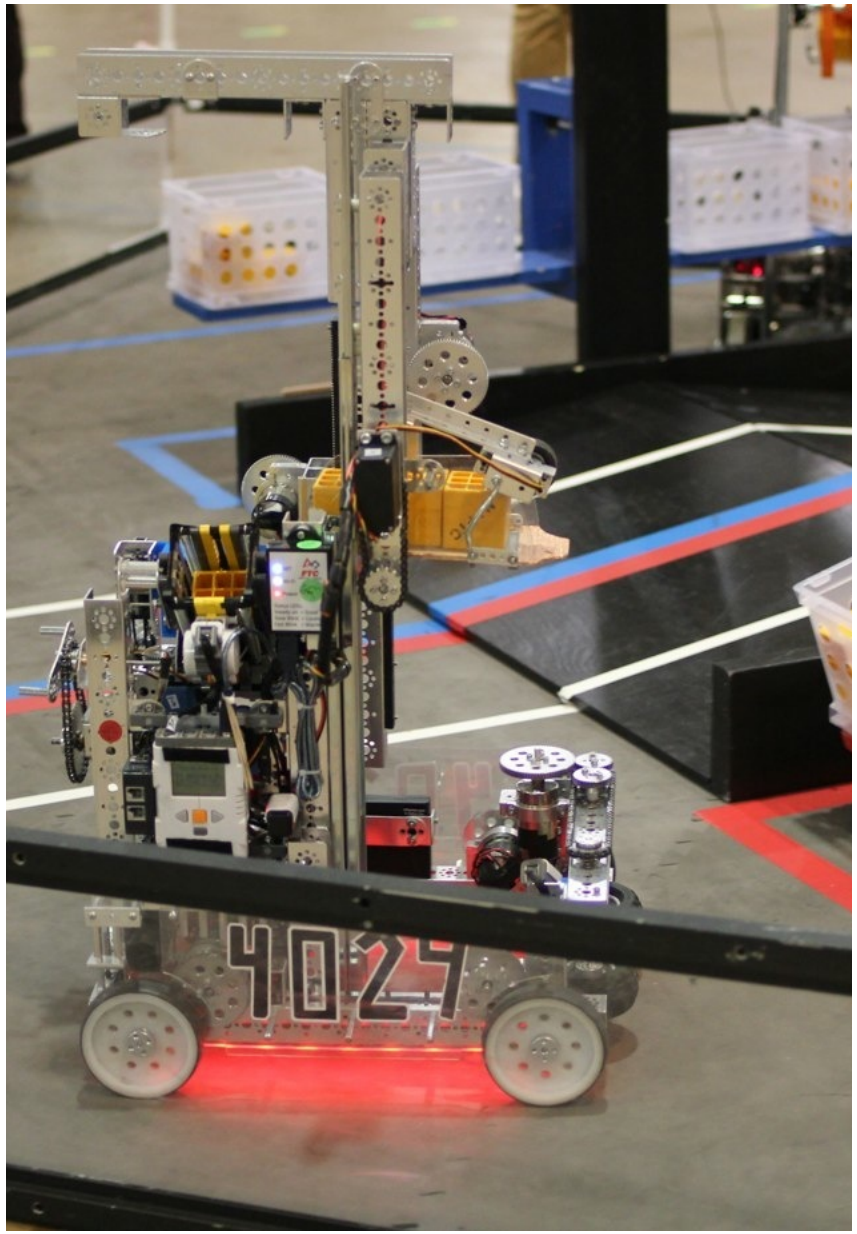
- Grades K-3
- 4,500+ teams
- 27,000+ students
- Models
- “Show-Me” posters

2) FIRST LEGO League

- Grades 4-8
- 27,000+ teams in 80 countries
- 267,000+ students
- LEGO Mindstorms and parts
- Fully autonomous

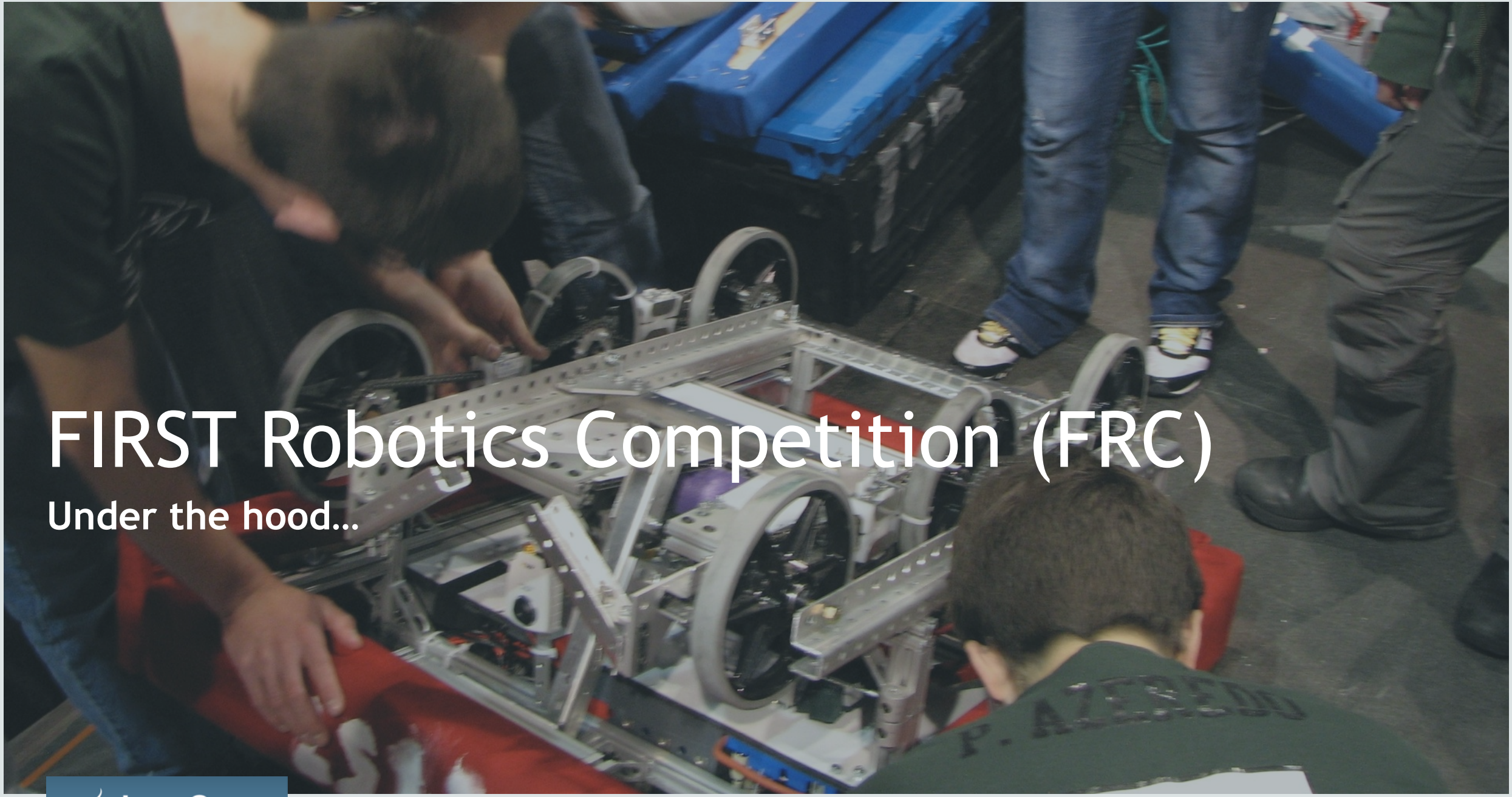
3) FIRST Tech Challenge

- Grades 7-12
- 4,450 teams in 15 countries
- 44,500 students
- LEGO Mindstorms, sensors, motors, custom parts
- Autonomous and tele-op
- 2 v. 2 game



4) FIRST Robotics Competition (FRC)

- Grades 9-12
- 3,000 teams in 17 countries
- 68,000+ students
- 16,000+ mentors and volunteers
- Autonomous and tele-op
- Industrial controller, pneumatics machined parts, sensors, vision, …
- 3 v. 3 game



FIRST Robotics Competition (FRC)

Under the hood…

How Does FRC Work?

- Kickoff in January
  - Game disclosed
  - Teams get kit of parts
    - Includes motors, control system, minimal base chassis, and some miscellaneous parts
- 6 weeks and 2 days later...
  - Decided how to play the game
  - Brainstormed robot ideas
  - Built prototypes
  - Built, programmed, tested and drove their robot

The Challenge

- New game every year
- Some similar characteristics
  - Autonomous period of game (10-15 sec)
  - Tele-op period (2 minutes)
- Teams play in 3 on 3 alliances
- Regional competitions lead to World Championship
  - Advance by winning matches or getting awards
    - Design, presentations, outreach, sportsmanship, etc.

2014 Game

The Robots

- Built from a kit of parts provided by FIRST
  - Motors, control system, Wi-fi, minimal chassis, assorted other components
- Teams supply other parts
- Robots specs:
  - 150 lbs MAX
  - About 3’x3’ and 5’ tall
  - Travel 8-15 ft/sec

2014 Robot Controller - National Instruments Compact RIO

- Commercial embedded controller
- 400 MHz PowerPC
- vxWorks
- 64Mb RAM and 128Mb flash
- FPGA chassis for custom I/O, control, and processing
- Ethernet and serial port
- Expandable through plug-in modules

Built-in FPGA - Real-time and Safety-Critical

- Analog Input
- Oversample / Average
- Accumulator
- Analog Trigger
- Digital Input / Output
- Slow Digital Output
- PWM Output
- I2C Bus
- Digital Input Filtering
- Solenoid Output
- Watchdog timers
- Counter / Timer
- SPI Engine
- Time / Alarm
- Routable Interrupts
- Direct Memory Access

FRC Driver Station

Programming (2010-2014)

- WPILib - FRC robot interface and framework library
- LabVIEW
  - Graphical programming system from National Instruments
  - Related to LEGO Mindstorms programming
- C++
  - Uses Wind River Workbench - eclipse-based IDE
- Java
  - Java ME CLDC 1.1
  - Squawk JVM (research project)

Anatomy of a Robot Program w/ WPILib

- Subclass one of the robot classes
  - **SimpleRobot**
    - Assumes code polls iteratively
  - **IterativeRobot**
    - More advanced model that handles devices with varying timing requirements
- Base classes handle field communication and match states

Robot Program Definition

```
import edu.wpi.first.wpilibj.SimpleRobot;

public class RobotExample extends SimpleRobot
{
    private RobotDrive drive;
    private Joystick stick;

// autonomous and operatorControl code
// goes here - override methods…

}
```

code to sequence through competition and communicate with driver station

Initialization (constructor)

```
public RobotExample()
{
    drive = new RobotDrive(1, 2);
    stick = new Joystick(1);
}
```

Create the robot drive

Joystick on Driver Station USB port 1

Autonomous part of program

```
public void autonomous() {
        drive.drive(0.6, 0.0);
        Timer.delay(2.0);
        drive.drive(-0.6, 0.0);
        Timer.delay(2.0);
        drive.drive(0.0, 0.0);
  }
```

Drive forwards for 2 seconds, then backwards for 2 seconds, then stop

`myRobot.drive(`*`speed, curve`*`)`

speed: a value from -1.0 to 1.0 where 0.0 is stopped

curve: a value from -1.0 to 1.0 where 0.0 is no turn

Tele-Op part of program

```
public void operatorControl()
{
    while (!isEnabled())
    {
        drive.arcadeDrive(stick);
        Timer.delay(0.01);
    }
}
```

Drive robot using one joystick

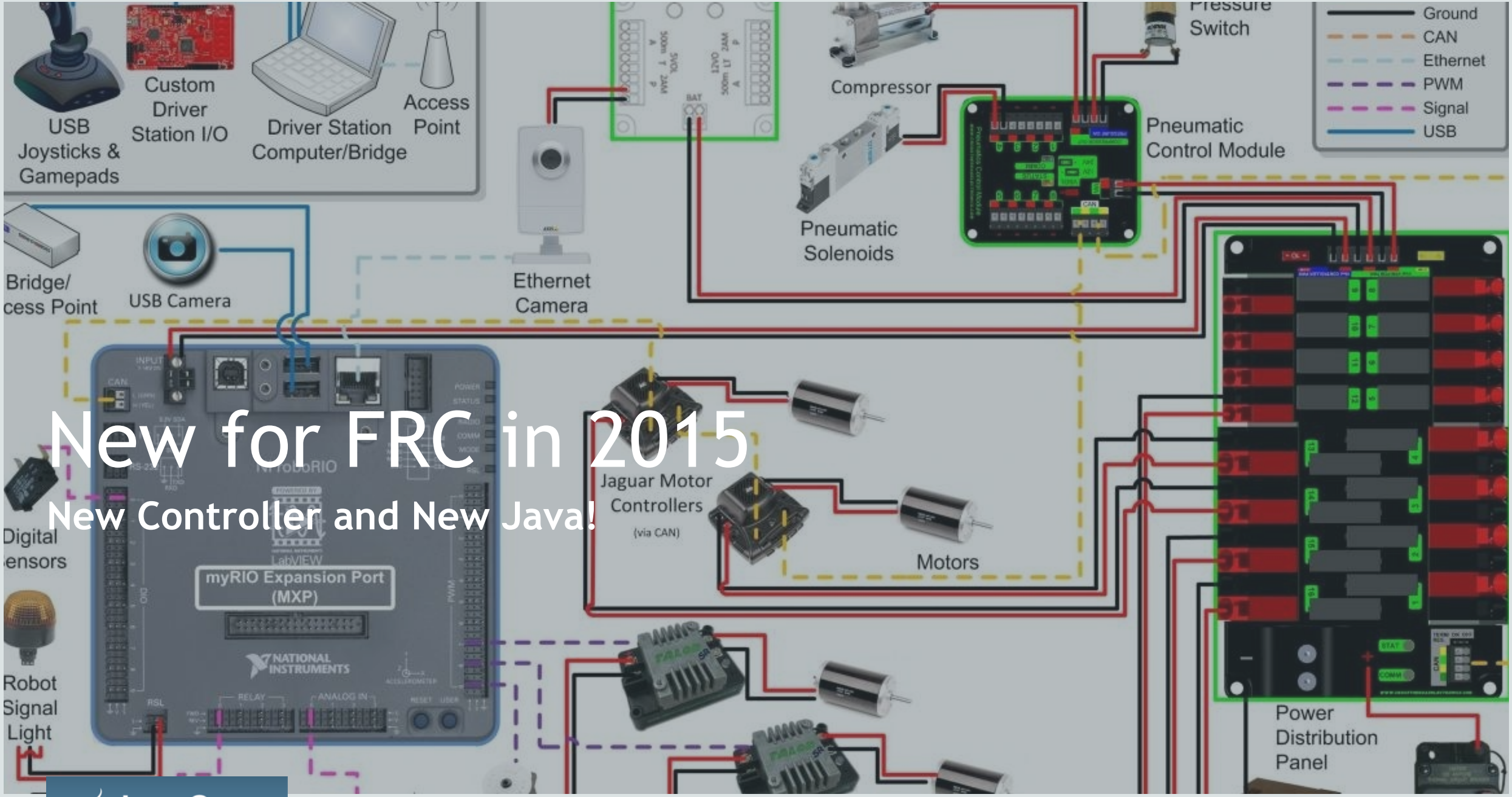
Wait

WPILib Goals

- Trivial to get started
- Access to all sensors and controllers
  - Pneumatics, servos, cameras, …
- Powerful enough to develop
  - Traction control
  - Gyroscopic heading control
  - Vision recognition
  - Auto-targeting
- Safe - FPGA and libs

New for FRC in 2015

New Controller and New Java!



roboRIO Controller

- Dual-core ARM A9 @667MHz
- Linux
- 256MB RAM
- 512MB Flash
- Built-in I/O (tons)

Java SE Embedded 8

- Perfect fit for roboRIO
- Runs out-of-the-box
  - Compact profiles 1-3
  - Client VM

ORACLE JAVA SE EMBEDDED VERSION 8 UPDATE 6

Note: Java SE Embedded 8 enables developers to create customized JREs using the JRECreate tool. Starting with Java SE Embedded 8, individual JRE downloads for embedded platforms are no longer provided. To get started, download an eJDK bundle suitable for your target platform and follow instructions to create a JRE that suits your application's needs. This change does not affect JRE downloads for Java SE Embedded 7 Update releases.

Product / File Description	File Size	Download
ARM, Linux		
ARMv7 Linux - VFP, HardFP ABI, Little Endian[1]	117 MB	ejdk-8u6-fcs-b23-linux-arm-vfp-hflt-12_jun_2014.tar.gz
ARMv7 Linux - VFP, SoftFP ABI, Little Endian[2]	116 MB	ejdk-8u6-fcs-b23-linux-arm-vfp-sflt-12_jun_2014.tar.gz

Java ME CLDC 1.1 -> Java SE 8

- **API Changes:**
  - Collection classes
  - java.io.*
  - NIO
  - Logging
  - Regular expressions
  - java.util.concurrent
  - unsigned support
- **Language Changes:**
  - *assert*
  - Generics
  - Annotations
  - Enumerations
  - New *for* loops
  - Lambda

Java SE 8 Benefits

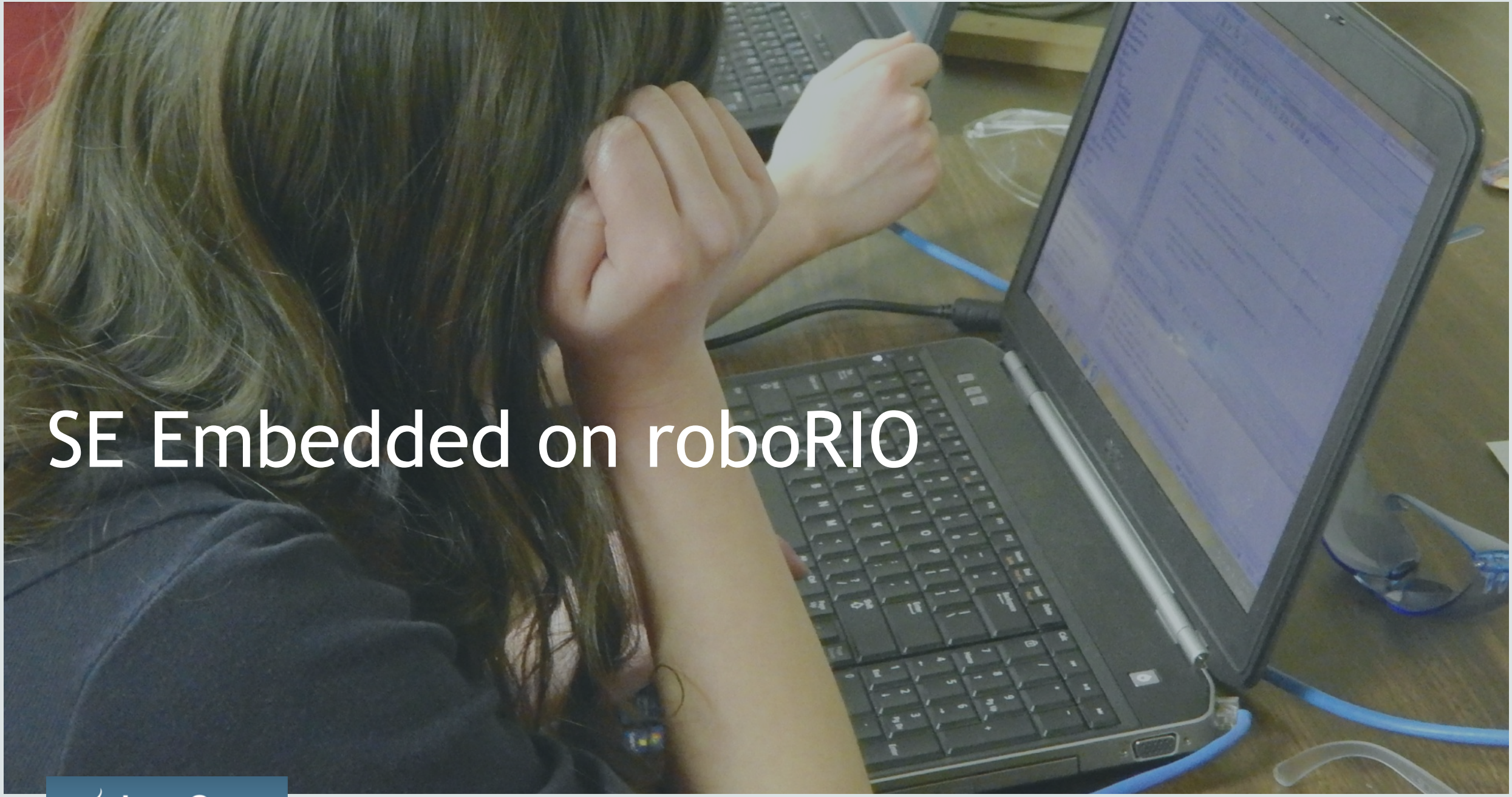
- What mentors know
- What kids learn
- Common code
- Opens up huge world of libraries
  - Do you really want…?
- Invoke Dynamic allows?

Squawk VM -> HotSpot

Small vs. Fast

- JIT compiler!
- Faster interpreter
- Multiple garbage collectors
- Full tool support
  - Better debugging
  - Profiling
  - visualvm, jstat, jconsole, …

SE Embedded on roboRIO



Faster Execution

- Simple benchmark code
  - From Team 2846 “FireBears”
- Compare Squawk VM to SE Embedded
  - Running on PPC vs. ARM
  - 400MHz vs. 667MHz

Faster Execution

System	BearMarks*	Ratio
Squawk - cRIO	*650,664*	*1x*
HotSpot (interpreter) - roboRIO	*340,582*	*1.9x*
HotSpot (JIT) - roboRIO	*53,261*	*12x*
roboRIO vs. cRIO (MHz)	*-*	*1.68x*

HotSpot Interpreter

- Assembly-based interpreter loop
- Machine code generated at startup
- Tuned to actual CPU(s)
- vs. generic C-based code in Squawk VM

JIT Overhead Issues?

- *Preliminary:*
- Avg 2.8ms / method
- JIT runs in parallel with interpreter
- Dual-core processor
- FYI - If you want to impress a programming team…

GC in SE Embedded

- Serial
- Parallel
- Concurrent Mark-Sweep
- *Garbage First (G1) not available*

Initial GC Benchmarking

- Synthetic app creates a FIFO queue of byte arrays (treadmill)
- Runs a short treadmill a fast speed (1ms)
- 2 larger treadmill of treadmills at slower speeds
- 2 java.util.TimerTasks that measure jitter (running every 5ms)

GC Info

	GC-pause Max	GC-pause Median	Jitter-delay Max	Jitter-delay Mean	Jitter-rate Max	Jitter-rate Mean
Serial	*37*	*16*	*1*	*0.007*	*36*	*1.3*
CMS	*31/25*	*7*	*8*	*0.009*	*23*	*0.9*
Parallel	*54*	*9*	*1*	*0.005*	*64*	*4.4*

GC Results

- Is that good or bad?
- Robots often have several periodic tasks
  - Control data arrives every 20ms
  - PID loops often run at 50ms
- Synthetic app is pushing too hard (too many full GCs)
- Need real data from real robotic programs
  - FIRST is beta testing now

Java Tools

- Full range of Java tools available
- Debugger
- Profiler
- jconsole, jstat
- VisualVM…

Start Page | 10.1.90.2:3000 (pid 5325)

Overview | Monitor | Threads | Sampler | MBeans | Buffer Pools | JConsole Plugins | Visual GC | Tracer

10.1.90.2:3000 (pid 5325)

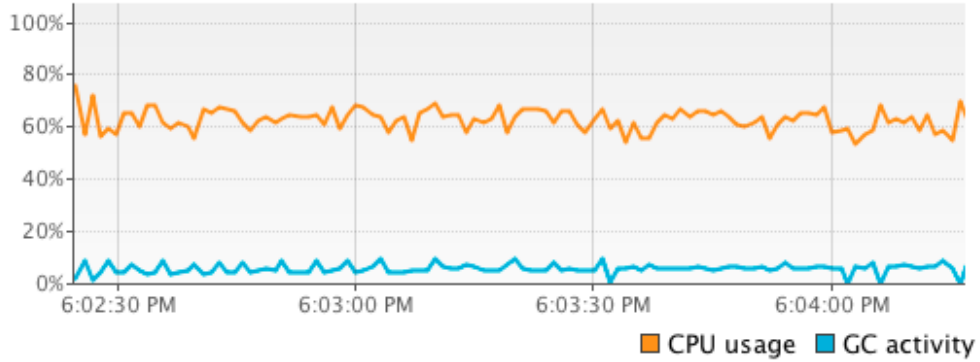
Monitor — CPU ☑ Memory ☑ Classes ☐ Threads ☑

Uptime: 2 min 01 sec

Perform GC | Heap Dump

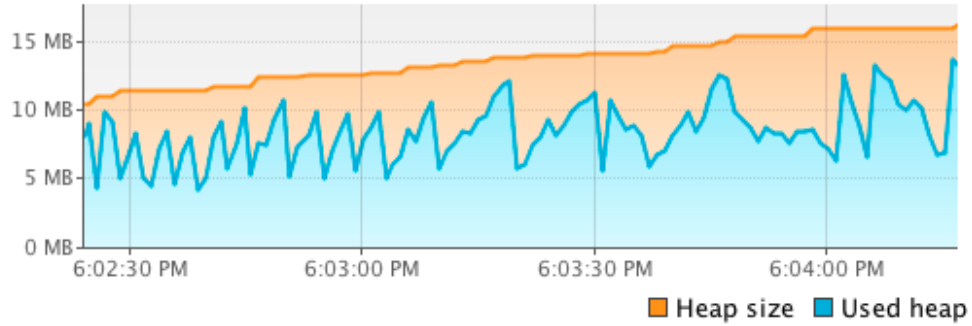
CPU

CPU usage: 64.5% **GC activity:** 7.2%



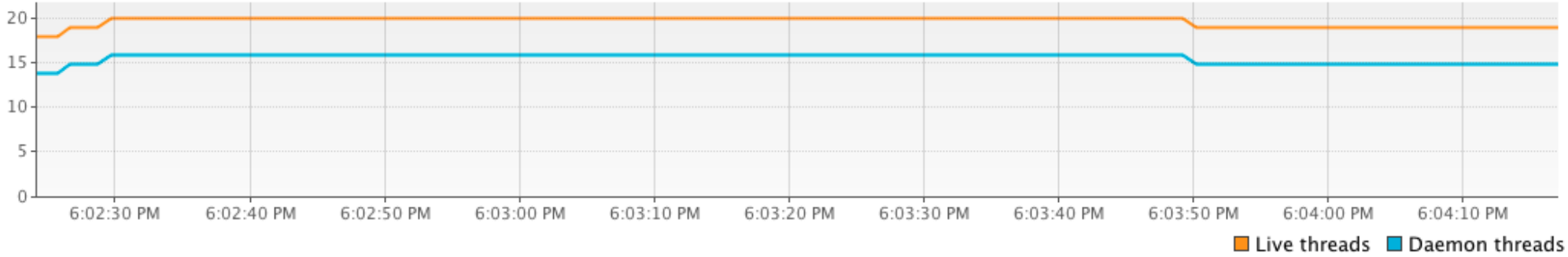
Heap | Metaspace

Size: 17,113,088 B **Used:** 14,104,200 B
Max: 97,386,496 B



Threads

Live: 19
Live peak: 20
Daemon: 15
Total started: 20



Start Page | 10.1.90.2:3000 (pid 5325)

Overview | Monitor | Threads | Sampler | MBeans | Buffer Pools | JConsole Plugins | Visual GC | Tracer

10.1.90.2:3000 (pid 5325)

Sampler — Settings

Sample: CPU | Memory | Stop

Status: memory sampling in progress

Heap histogram | Per thread allocations

Deltas | Snapshot | Peform GC | Heap Dump

Classes: 737 **Instances:** 120,312 **Bytes:** 14,664,792

Class Name	Bytes [%]	Bytes	Instances
byte[]		10,623,552 (72.4%)	3,739 (3.1%)
java.util.**TreeMap$Entry**		831,936 (5.6%)	25,998 (21.6%)
int[]		670,280 (4.5%)	917 (0.7%)
java.util.**TreeMap**		386,016 (2.6%)	8,042 (6.6%)
char[]		375,728 (2.5%)	6,263 (5.2%)
java.lang.**Long**		262,320 (1.7%)	16,395 (13.6%)
java.lang.**Object[]**		150,760 (1.0%)	5,546 (4.6%)
java.lang.**Class**		144,376 (0.9%)	1,625 (1.3%)
java.util.**TreeMap$KeySet**		128,352 (0.8%)	8,022 (6.6%)
javax.management.openmbean.**CompositeDataSupport**		128,160 (0.8%)	8,010 (6.6%)
java.util.**LinkedHashMap$Entry**		123,744 (0.8%)	3,867 (3.2%)
java.lang.**String**		96,160 (0.6%)	6,010 (4.9%)
java.util.**HashMap$Node[]**		76,656 (0.5%)	961 (0.7%)
java.util.**Arrays$ArrayList**		58,608 (0.3%)	3,663 (3.0%)
java.util.**Collections$UnmodifiableRandomAccessList**		58,400 (0.3%)	3,650 (3.0%)
java.lang.reflect.**Method**		44,704 (0.3%)	508 (0.4%)
java.util.**LinkedHashMap**		41,608 (0.2%)	743 (0.6%)

Class Name Filter (Contains)

Timeline

View: All threads

Name	Running		Total
JitterBug Timer	970 ms	(1%)	99,095 ms
JMX server connection timeout 20	0 ms	(0%)	99,095 ms
main	0 ms	(0%)	99,095 ms
Reference Handler	0 ms	(0%)	99,095 ms
RMI Scheduler(0)	0 ms	(0%)	99,095 ms
RMI TCP Accept-0	99,095 ms	(100%)	99,095 ms
RMI TCP Accept-0	99,095 ms	(100%)	99,095 ms
RMI TCP Accept-3000	99,095 ms	(100%)	99,095 ms
RMI TCP Connection(1)-169.254.12	24,048 ms	(28.6%)	84,076 ms
RMI TCP Connection(2)-169.254.12	99,095 ms	(100%)	99,095 ms
RMI TCP Connection(3)-169.254.12	0 ms	(0%)	99,095 ms
RMI TCP Connection(4)-169.254.12	60,417 ms	(61.4%)	98,382 ms
RMI TCP Connection(5)-169.254.12	96,042 ms	(100%)	96,042 ms
sampler	0 ms	(0%)	99,095 ms

6:10:05 PM 6:10:10 PM 6:10:15 PM 6:10:20 PM

Running Sleeping Wait Park Monitor

Threads inspector

- [ ] Finalizer
- [ ] JMX server connection timeout 20
- [x] JitterBug Timer
- [ ] JitterBug Timer
- [ ] RMI Scheduler(0)
- [ ] RMI TCP Accept-0
- [ ] RMI TCP Accept-0
- [ ] RMI TCP Accept-3000
- [ ] RMI TCP Connection(2)-169.254.1
- [ ] RMI TCP Connection(3)-169.254.1
- [ ] RMI TCP Connection(5)-169.254.1
- [ ] RMI TCP Connection(idle)
- [ ] RMI TCP Connection(idle)

```
2014-09-26 18:09:31

"JitterBug Timer" - Thread t@12
   java.lang.Thread.State: TIMED_WAITING
        at java.lang.Object.wait(Native Method)
        - waiting on <1c28fcb> (a java.util.TaskQueue)
        at java.util.TimerThread.mainLoop(Timer.java:552)
        at java.util.TimerThread.run(Timer.java:505)

   Locked ownable synchronizers:
        - None

"Thread-4" - Thread t@14
   java.lang.Thread.State: TIMED_WAITING
        at java.lang.Thread.sleep(Native Method)
        at memtreadmill.Allocator.run(Allocator.java:46)
        at java.lang.Thread.run(Thread.java:745)

   Locked ownable synchronizers:
```

Java Tools - More Info

- See James Gosling's talk:
- *Debugging and Profiling Robots* - CON6699
- TODAY, Oct 1, 1:00 PM - 2:00 PM
- Hilton - Continental Ballroom 4

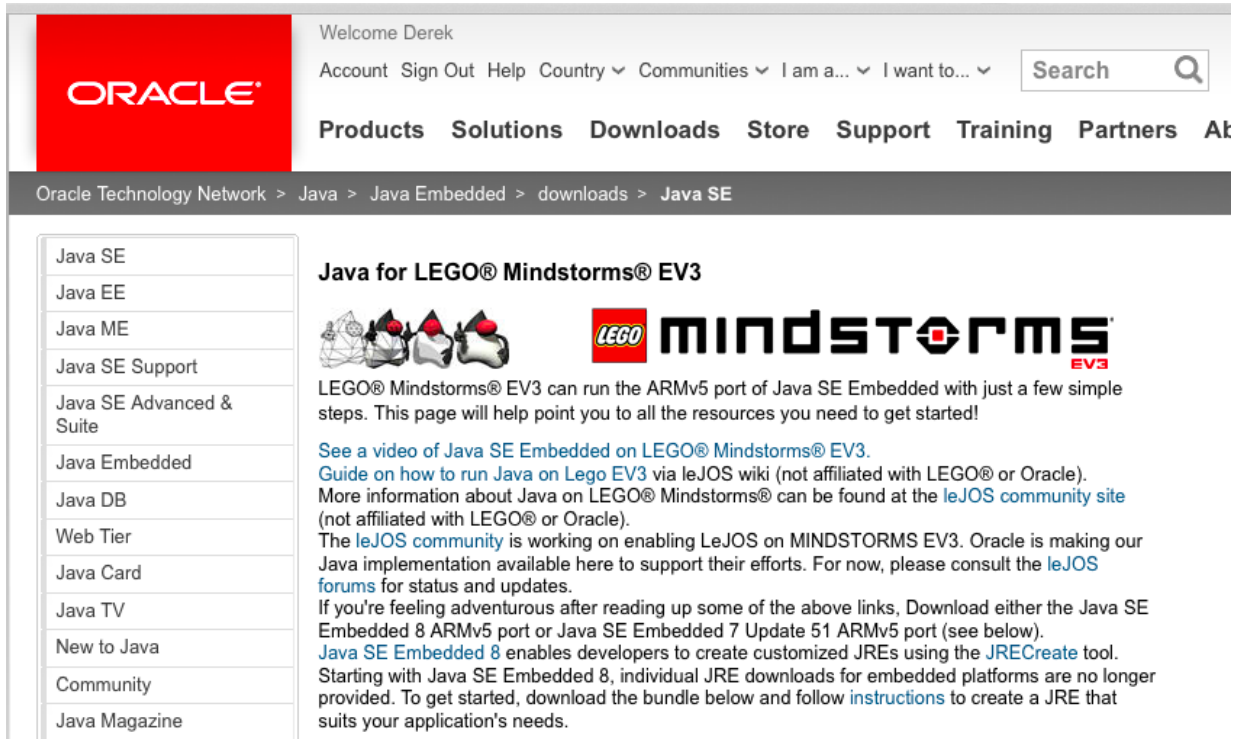
Next Steps…

Sound like fun? Get Involved!

- FRC Teams need mentors who know Java
- Volunteer for events
- Get kids to join a team
- www.usfirst.org/whats-going-on

Java for other robots

- LEGO Mindstorms EV3
- Runs linux on ARM9 @300MHz
- 64MB RAM, 16MB flash
- Flash card slot
- Search web for “java lego ev3”
- FLL and FTC someday?



Safe Harbor Statement

The preceding is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

Credits

- Photos courtesy of
  - FIRST Team 662, Rocky Mountain Robotics
  - USFIRST (credit Adriana M. Groisman and Argenis Apolinario)
- “BearMark” from FRC Team 2846, FireBears

Q+A

Hardware and Software
Engineered to Work Together