Emilie Phillips

53 Babb Meadow Ln
Mason, NH 03048

emilie@j3.org
(603) 809-0556

Principal software engineer with nine years of experience in Linux software development and data analysis within the robotics industry. Uses creativity and detailed reasoning to solve challenges. Strong skills in software design and testing for complex systems.

QUALIFICATIONS

- Soft real-time, concurrent, embedded Linux applications
- Framework design: components, model-view-controller, data driven configuration, and test driven development
- Physics based robotics: inverse kinematics, planning, sensor processing, obstacle detection and avoidance, robot navigation, PID control
- Peripheral control via serial, USB, and Posix and raw Ethernet sockets
- Team leadership, agile development, scrum, peer reviews
- C++, Boost, STL, Python, bash, XML, Make, subversion, Code Collaborator
- Proficient in French

PATENTS

8,577,517 8,447,440 8,396,611 8,326,469 8,255,092 8,108,092

WORK ACCOMPLISHMENTS

Principal Software Engineer, Sr. Software Engineer, iRobot
Bedford, MA 2012-present

- Led four software engineers to release software for customer shipment. Software included significant video improvements, two new peripherals, OS upgrade, and support for new manufacturer. Continually monitored and adjusted scope to achieve tight deadline. Focused team on necessary tasks. Ramped up new contractors.
- Designed and implemented plug-in framework for rapid integration of FirstLook USB peripherals. Defined protocol extension and XML description for automatic GUI control. Established hardware requirements for discovery via udev. Ensured framework robust to USB glitches. Incrementally added error reporting and hardware emulation.
- Refactored prototype FirstLook robot and controller into product quality software. Jumped into unfamiliar code base using Qt and fixed numerous race condition and threading bugs. Reduced start time. Incrementally improved protocol support. Implemented accelerated life time test for flippers. Systematically added unit tests and run time debugging.
- Implemented features as needed for customer acceptance: low latency two way audio, GUI control of iRobot SUGV lights, udev rules to detect peripherals, and plug-ins to control them. Wrote fault tolerant serial control for actuators. Enabled superior video by significantly reducing CPU usage and refactoring camera control to handle dynamic resolution.
- Researched academic literature and wrote generalized inverse kinematics library. Significantly improved linearity of end effector motion on iRobot PackBot and Warrior manipulators.
- Evaluated open source libraries ROS and OMPL to replace legacy code. Prototyped critical FirstLook features in ROS. Presented report of tradeoffs and concluded that ROS is too immature. Work in progress to replicate PackBot poses with OMPL.

Emilie Phillips

emilie@j3.org

Sr. Research Scientist, iRobot
Bedford, MA 2009-2011

- Developed software control of various sensors and actuated peripherals to demonstrate special capabilities on iRobot systems. Device maturity ranged from off the shelf products, to custom in house hardware, and customer supplied hardware. Supported USB, serial, and TCP devices. Implemented PID control for simple actuators. Filtered sensors as needed. Sped up development by implementing hardware emulator.
- Analyzed existing motion control and demonstrated user visible improvements. Improved torque limiting for better stair climbing on PackBot. Demonstrated precise and delicate manipulation on PackBot and Warrior using torque control, improved inverse kinematics, and haptic feedback from force sensors.
- Developed proprietary Ethernet protocol from initial design to functional proof of concept demonstrating low latency CAN tunneling over Ethernet. Assisted in initial evaluation of protocol and open source raw socket libraries. Delivered Linux and Windows libraries to customer with detailed API docs.
- Implemented custom features on robot and GUI, and augmented external UDP protocol for customer contracts. Promoted generic features to internal common software. Provided detailed documentation for all software interfaces.
- Successfully managed schedule, budget, technical decisions, and second engineer on small projects. Ensured projects were well documented for follow on contracts and demonstrations.
- Increased productivity by actively tutoring other engineers in C++ and in house software.

Software Engineer, iRobot
Bedford, MA 2005-2009

- Implemented generalized hardware control arbitration which has allowed code reuse across robots and stable hardware control implementations despite multiple generations of high level applications.
- Co-designed iRobot’s first production operator controller framework with support for multiple robots.
- Optimized run time performance of new robot code base so it could run on all current and legacy systems.
- Rewrote autonomous vehicle software to produce a reliable system. Contributions include CAN and USB hardware control and platform independent autonomous navigation behaviors using the new robot framework.
- Created prototype autonomous retrotraverse, self righting, and cruise control behaviors documented sufficiently for marketing to demonstrate the value of software upgrades on existing PackBot robots.

Grad. Research Asst.: Software Librarian, Cornell University
Ithaca, NY 2004-2005

- Oversaw the daily acceptance testing of over 800 data acquisition and physics analysis software packages. Maintained build system and releases. Created a parallelizable test infrastructure and simplified the procedure for creating and patching production releases.

EDUCATION

Carnegie Mellon University
BS Computer Science & Physics – 1999-2003