Luke Palmer

Software Engineer

720-837-6741
lrpalmer@gmail.com

I have nine years of programming experience with focus on software architecture and abstraction, scripting languages[1], automated testing and quality assurance, and artificial intelligence. I am very interested in automated testing techniques for games, ensuring code quality, maintainability, and documentation. I am also interested in core gameplay dynamics (a game that is all polish and no core is nothing) and founding designs in mathematical theory for emergent dynamics.

Accomplishments

- Worked with Larry Wall as a member of the Perl Cabal (Perl 6 Design Team) since 2004.
- Implemented the Perl Grammar Engine in Perl 6, an essential piece of the bootstrapping process, using a very modular design ("virtual classes").
  `http://svn.pugscode.org/pugs/ext/Parse-Rule/`
- Authored 15 Perl 5 modules including a large predicate logic engine, several object-oriented design tools, an attribute grammar engine (complex data structure processing tool), among other things, all with extensive documentation and test suites. `http://search.cpan.org/~lpalmer/`
- Authored and co-authored many small, cross-platform games, some of which are at `http://soylentsoft.net/`. My largest contributions

[1] Embedding, using, and designing.

to the projects there were: the fluid dynamics engine for Ichor, the fast automated solver for Minesweeper Infinity (to ensure that the user never gets an unsolvable board), and the transactional logic engine for Telegnosis (well, all of Telegnosis, actually).

- Presented as an "emerging topics" speaker at the O'Reilly Open Source Convention 2004 a talk entitled "A Language for Games". `http://conferences.oreillynet.com/cs/os2004/view/e_sess/4969`

Career History/Education

- BA/MA student of mathematics at the University of Colorado since 2002, presently on a year-long hiatus (until spring 2008). GPA: 3.32; Major (mathematics) GPA: 3.80.
- Worked on the large, open source Perl 6 project continuously since 2001.
- Teaching Assistant in Physics 2 at CU in 2006 under Dr. Steven Pollock.
- Teaching Assistant in Introductory Computer Science at CU in 2006 under Dr. Susanne Sherba.
- Worked on MVT (mathematics visualization toolkit) for the summer of 2005. `http://amath.colorado.edu/java/index.php`
- Worked on a medical natural language processing project in 2004 with Dr. Robert Bruegel and Dr. Ngo Thanh Nhan (not associated with the University of Colorado).
- Teaching Assistant in Calculus 2 at CU in 2004 under Dr. Mark Ablowitz.
- Worked on a medical expert system in 2003 with Dr. Robert Bruegel.
- Teaching Assistant in Physics 2 at CU in 2003 under Dr. Mike Dubson.

Qualifications

- 7 years experience with C and C++; I know all weird the nooks and crannys of C++, and can use them in order to achieve the best design.
- Plenty of experience with physics programming (wrote a physics engine for a game early on, and have made several games using the Open Dynamics Engine physics library).
- Experience with *many* other languages, including Perl, Ruby, Lua, Python, Java, Haskell, Lisp, Prolog, Curry, Scala, ... I can learn new languages very quickly. Experience with embedding Perl, Ruby, and Lua into C++ projects.
- Lots of experience with Subversion and some with CVS.
- Experience with JUnit, Test::More, and building my own unit testing frameworks with Test::Builder.
- Fundamental desire for clean, abstract software design. Lots of experience refactoring code into such designs.
- Desire and experience in building extremely simple but powerful systems, usually strongly founded in mathematical theory.

References

- Larry Wall. Mountain View, CA. 650-691-9038 (home). `larry@wall.org`
- Robert Bruegel. Lafeyette, CO. 303-499-1685. `rbruegel@aol.com`