Toward a Simulation Model of Open Source Software

Patrick Wagstrom

Department of Engineering and Public Policy

Center for the Computational Analysis of Social and Organizational Systems

Carnegie Mellon University

June 28, 2004

pwagstro@andrew.cmu.edu

http://patrick.wagstrom.net/

Overview

Carnegie Mellon ENGINEERING

- The Chaos that is Open Source
- The NK Model and the edge of chaos
- OSSim
- Some interesting results
- Moving on

Open Source...Chaotic? Never!

Carnegie Mellon ENGINEERING

- Anyone can create a project
- Anyone can use a project
- Anyone can contribute to a project
- Anyone can fork a project
- Anyone can distribute a project
- Diverging user requirements
- No formal training for new community members
- Usually lack specification and design documents

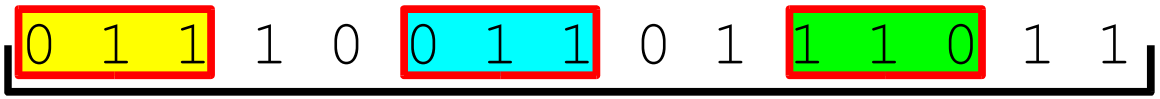
Now With 100% Less Sarcasm

Carnegie Mellon ENGINEERING

- Not everyone has the skill/knowledge to create projects
- Many projects are so poorly documented that they're nearly impossible to use
- Often times patches are ignored or rejected
- Forking a project takes time and is frequently frowned upon
- Bandwidth costs money
- Birds of a feather flock together
- Communities have indoctrination processes
- The process still makes software engineers cry

NK Model

Agents have a genotype bit string of length N



N=15

Each bit is associated with K neighbors to create alleles

011	011	110
111	110	101
110	101	011
100	011	110
001	111	101

K=2

Each allele has is evaluated against the randomly generated fitness landscape for that position. Overall fitness is the average of these values.

$$\frac{0.15 + 0.19 + 0.20 + 0.46 + 0.34 + 0.85 + 0.67 + 0.12 + 0.77 + 0.91 + 0.85 + 0.55 + 0.72 + 0.06 + 0.59}{15}$$

Mutation in the NK Model

Selects a bit at random, flip it, and evaluate new fitness

0 1 1 1 **1** 0 1 1 0 1 1 1 0 1 1

011	011	110
111	110	101
111	101	011
110	011	110
101	111	101

K=2

$$\frac{0.15 + 0.19 + \mathbf{0.82 + 0.31 + 0.74} + 0.85 + 0.67 + 0.12 + 0.77 + 0.91 + 0.85 + 0.55 + 0.72 + 0.06 + 0.59}{15}$$

0.5493

OSSim

- **Pronounced AWESOME!!**
  - **Make sure to say the exclamation points**
- **Multi-agent simulation**
- **Two classes of agents**
  - **People**
  - **Projects**
- **People agents have a genotype string**
  - **Represents desired feature set**
- **People may have ability to modify projects**
  - **Like real people, skill varies**
- **Project agents have a fitness landscape**

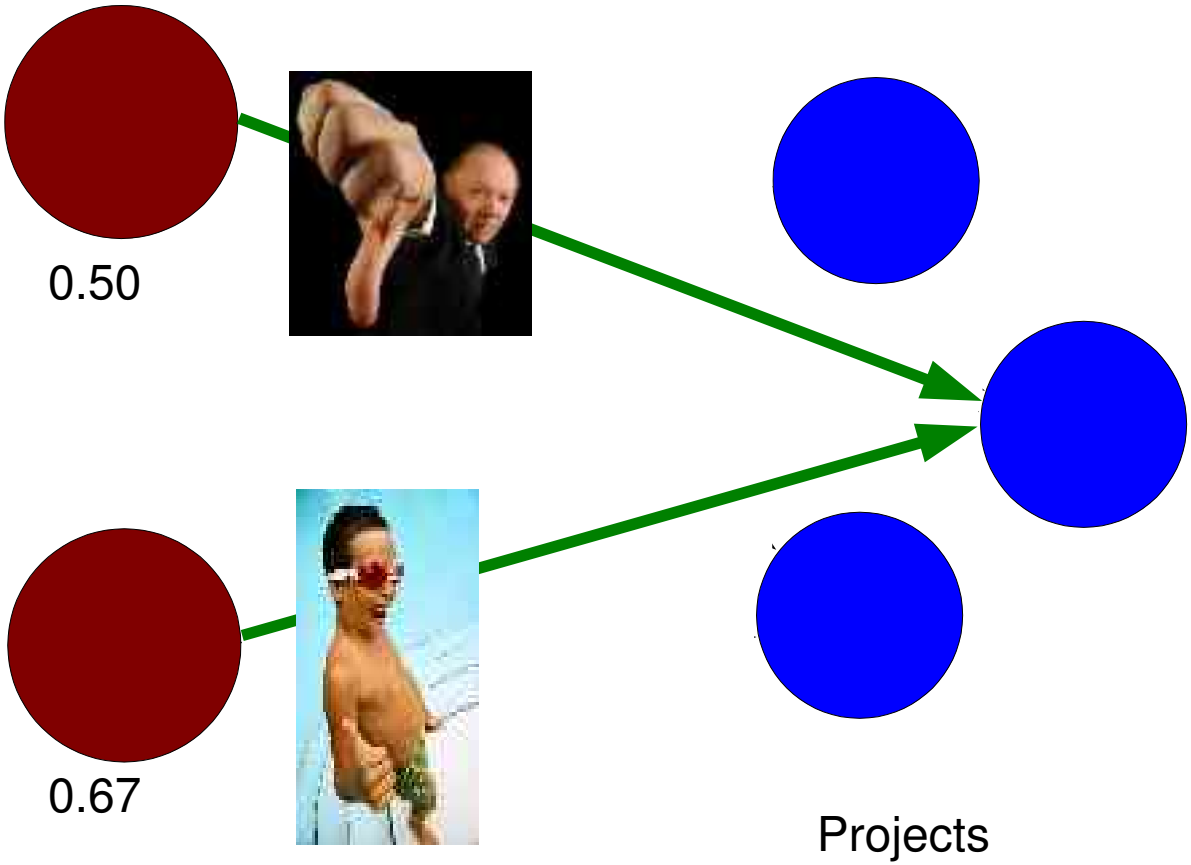
OSSim

- Initially, people associate with project that gives highest fitness
- Each tick, agents that can modify the project, make some random modifications
  - Increase fitness of one phenotype and decrease fitness of another
  - Example: Emacs is great editor, but a lousy web browser
- People vote on which modifications to accept
- Cycle repeats per simulation parameters
- If too long goes without improvement
  - Tolerance decreases
  - More likely to switch to another project

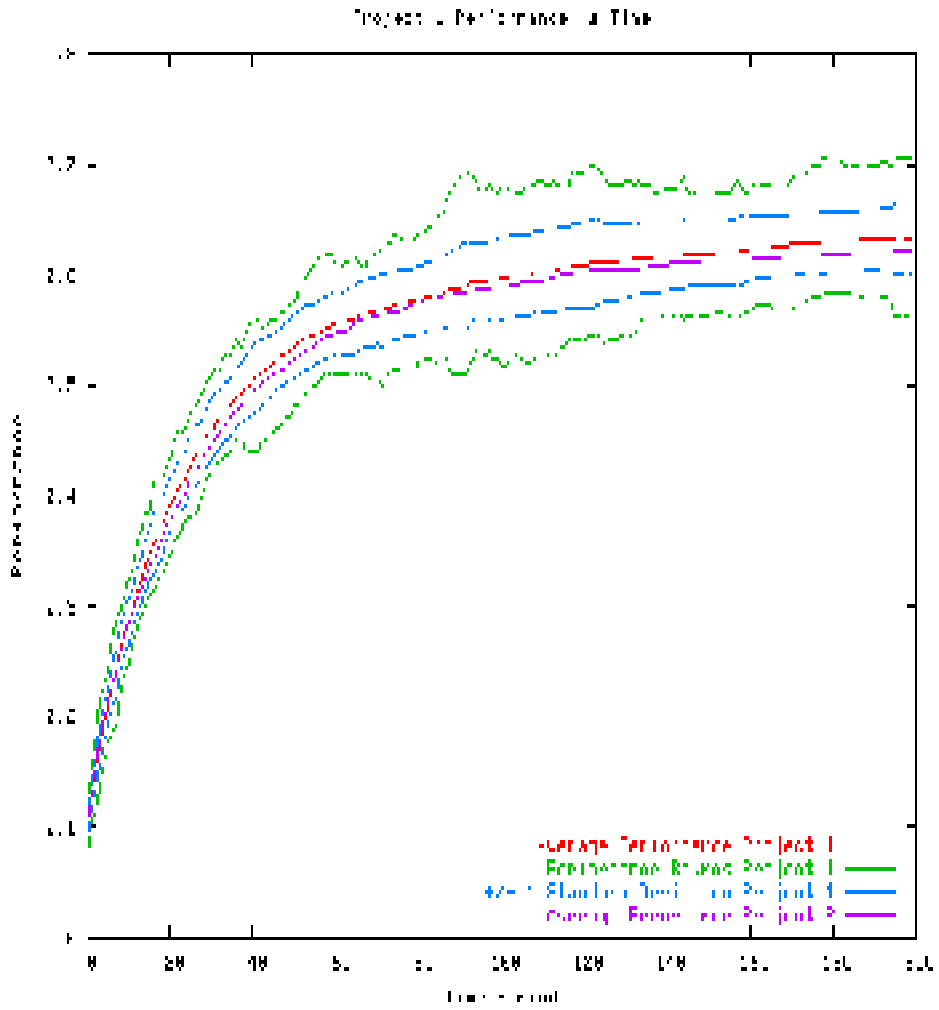
The OSSim Process

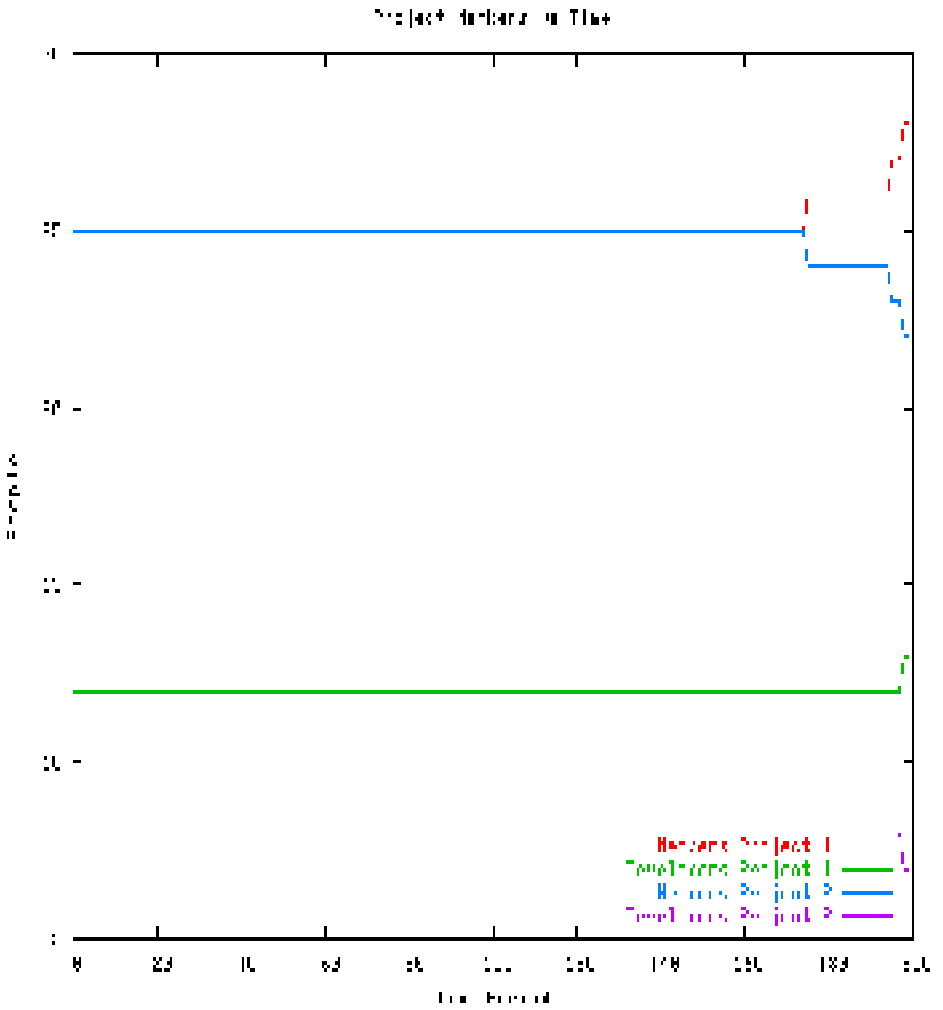
Cycle repeats until terminated

Agent 1
Developer
Skill 30
021211112220112

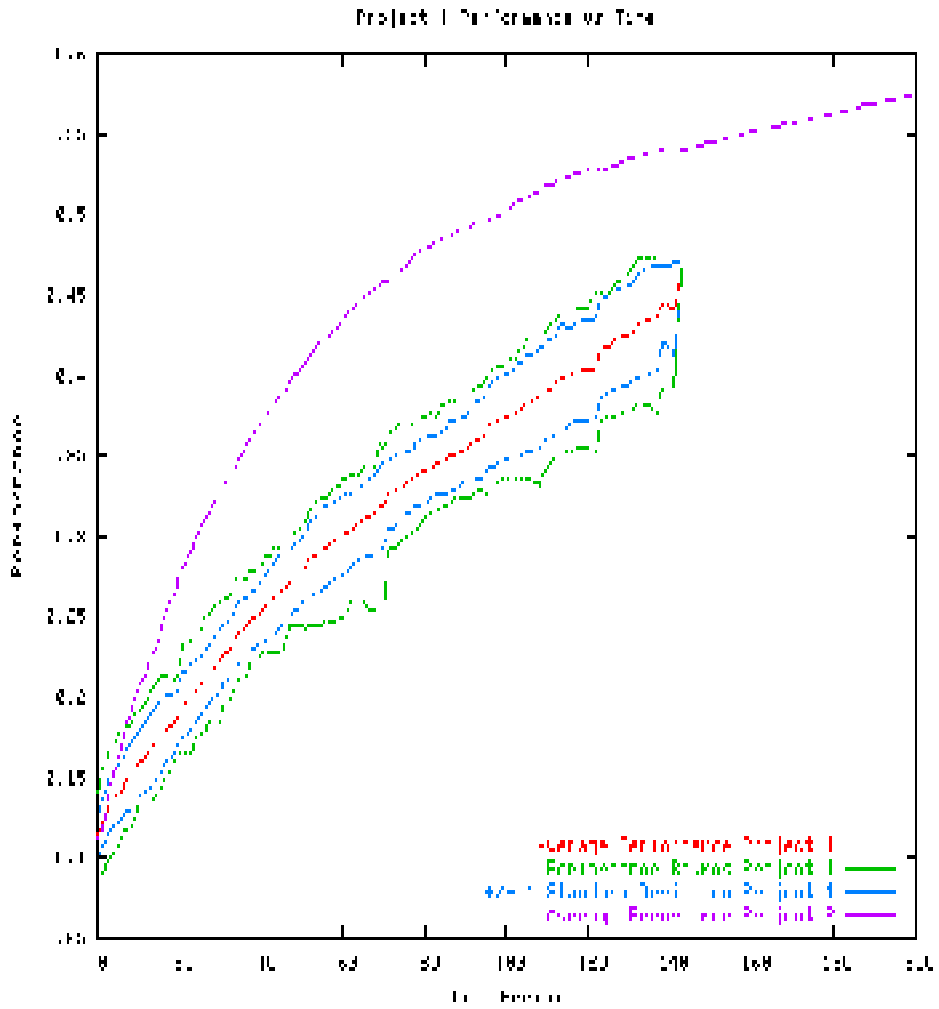


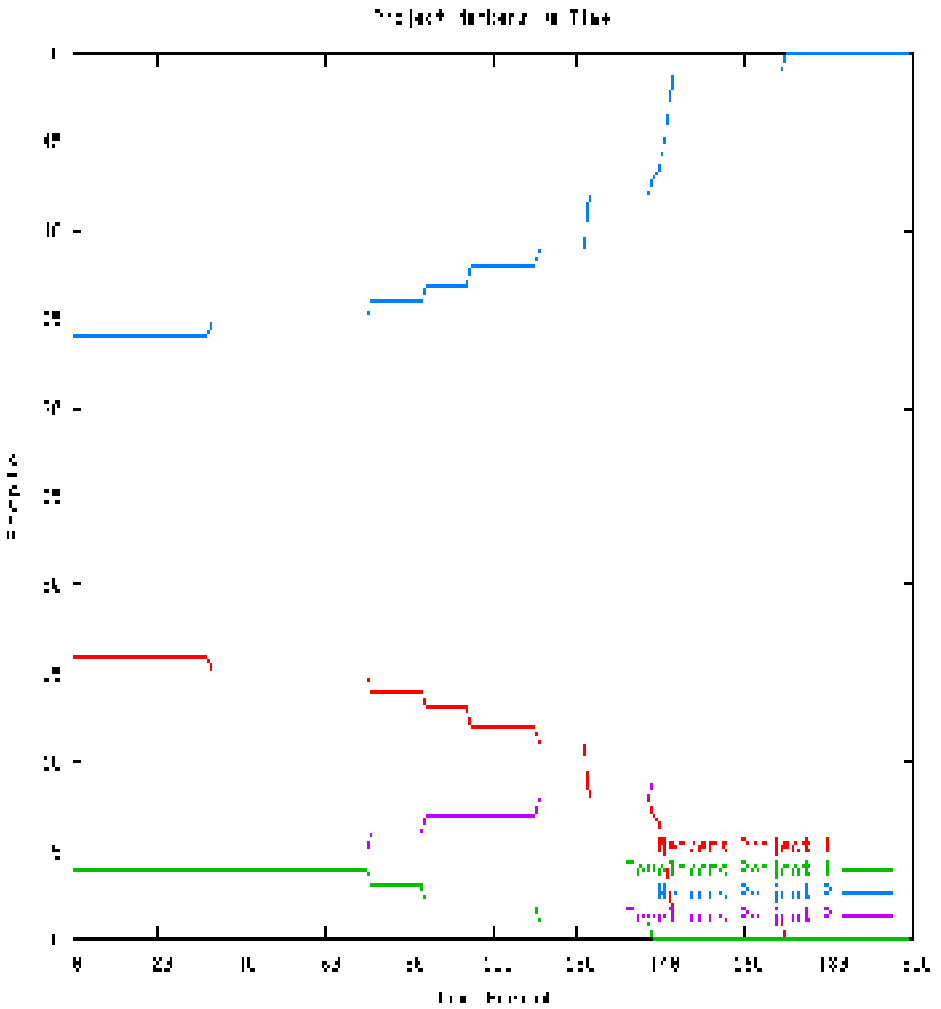
Basic Results





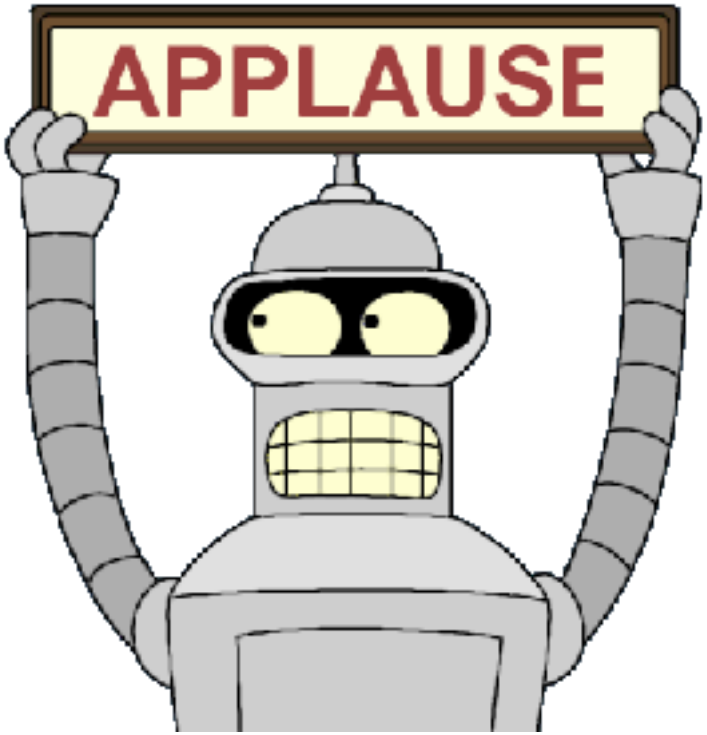
Abandon Ship!





In The Future...

- Validation, Validation, Validation
  - Face validity currently
  - Have data from OSS projects, need to code it
- Improved social networks
  - Preliminary concept of friendship has been integrated
  - Knowledge obtained about new projects from friends
- Integration with other CASOS tools
  - Export data in DyNetML
- More realistic developers


APPLAUSE