How Many Developers?

Staffing Strategies for Maintenance of Critical Software Systems at the Jet Propulsion Laboratory

Bill Taber (JPL)
Dan Port (University of Hawaii)

Jet Propulsion Laboratory
California Institute of Technology

10th International Symposium on Empirical Software Engineering and Measurement (ESEM 2016)
©2016 California Institute of Technology. Government sponsorship acknowledged.

Mission Critical Software

- How do you get to another planet?
  - You're going need a couple million lines of code.
    - That code is going to have bugs
    - Bugs can be mission ending
  - You're going to have to fix bugs
    - You are going to need software developers
    - The laws of physics don't wait for you to respond
  - How do I staff for this?
    - Not enough, I can't fix things
    - Too many, I'm losing resources elsewhere

©2016 California Institute of Technology. Government sponsorship acknowledged.

JPL Navigation Software

- About 800,000 lines of code
- We know our process
  - Productivity
  - Number of bug discoveries per day
  - Defect density
  - Distribution of time required to fix a bug.
- Staffing based on intuition
  - Staffing justification "*If we don't have this many people, things will be bad*"

©2016 California Institute of Technology. Government sponsorship acknowledged.

Staffing Based On Size—A Simple, Wrong Idea

- It would be nice to have a simple rule of thumb
  - I've got N lines of code
  - A developer can maintain M lines of code
  - To maintain my code I need N/M developers
- It has an element of plausibility
  - Larger systems have more bugs
  - A developer can only know a portion of a large system
- The problem is I cannot get a value for M
  - How do I determine M ?
  - How do I defend my value?
- This model ignores
  - Maintenance response to the number of issues received
  - Maintenance depends upon the time needed to respond to an issue.

©2016 California Institute of Technology. Government sponsorship acknowledged.

Staffing for Maintenance as Risk Mitigation

- Mission failures (like most failures) are generally the result of many small failures.
  - We address issues to enhance the odds of success—removing the small failures.
  - Schedules may be driven by physics.
  - Not being able to address a new issue allows the many small failures to accumulate
    - Ability to address issue depends upon staff available
- Staffing needs to be adequate to ensure timely resolution of issues
    - Paying for staff is risk mitigation
      - Staffing is a running taxi meter
    - What is the smallest staff that mitigates the risk?

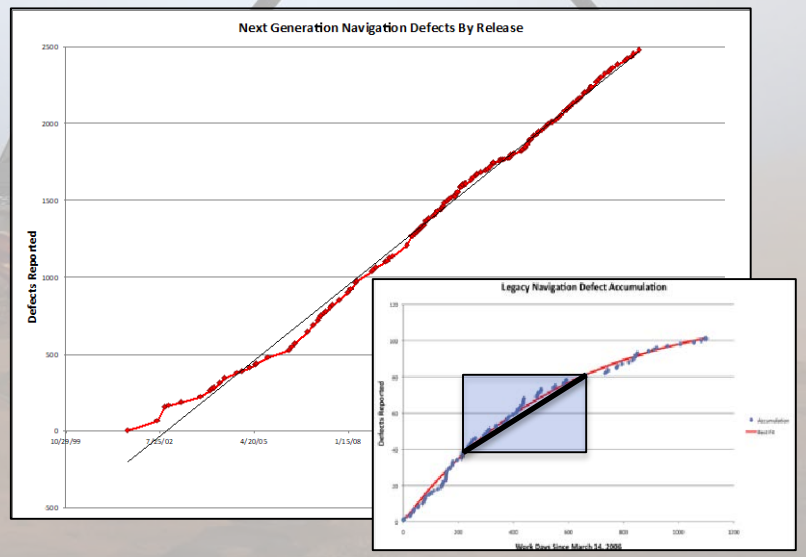
©2016 California Institute of Technology. Government sponsorship acknowledged.

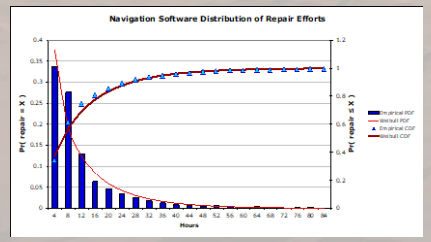
Just Keeping Up

- Issues arrive at some mean rate for a long time

  r issues/day (0.49)
- Mean effort

  E days/issue (1.28)





- k developers resolve issues at a rate

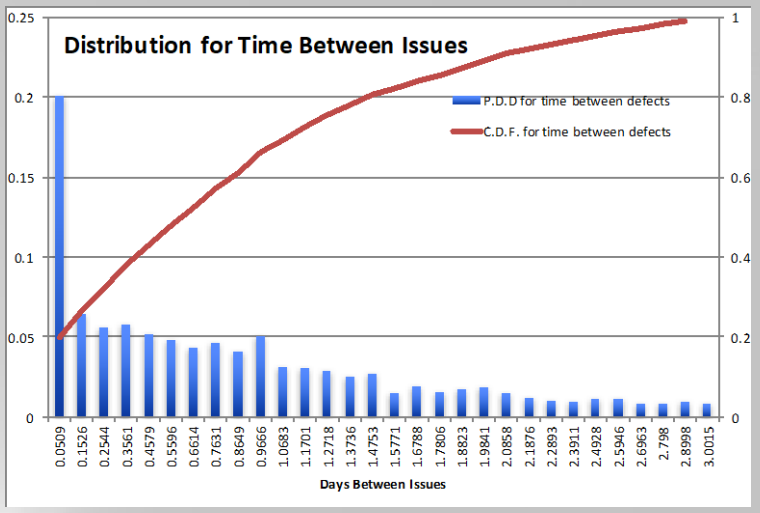
  k/E issues/day

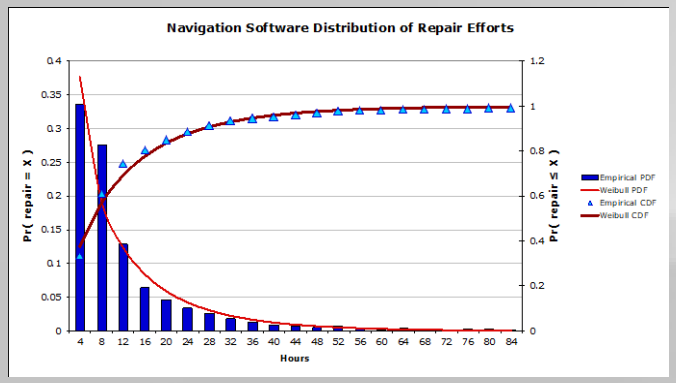
To just keep up $k > rE$ (1.78)

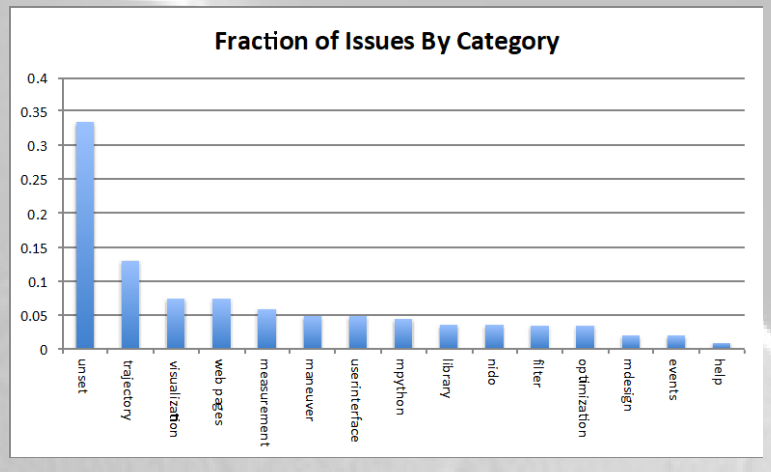
©2016 California Institute of Technology. Government sponsorship acknowledged.

Simulation

- Issue creation distribution
- Effort Resolution
- Areas of expertise
- Learning Model

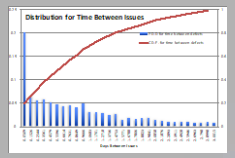






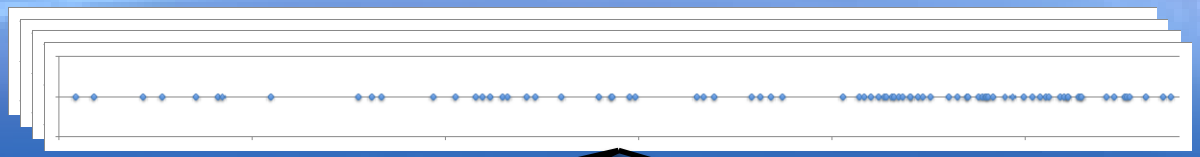
©2016 California Institute of Technology. Government sponsorship acknowledged.

Simulation (2)

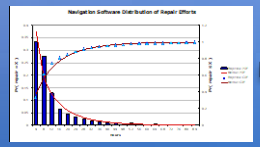


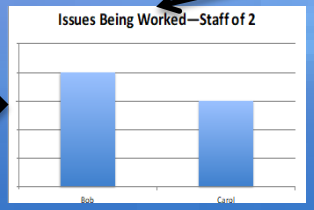
Distribution for times between issues

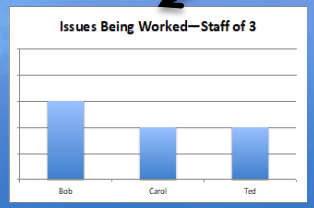
Issue History Generator

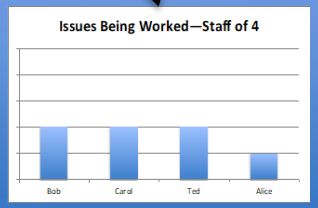


Distribution for Effort to resolve

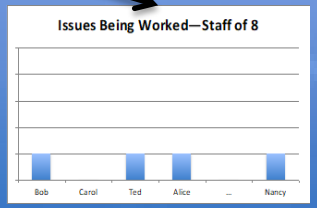




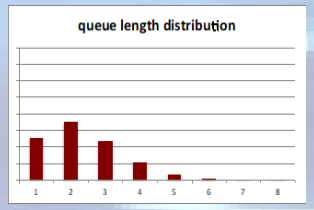


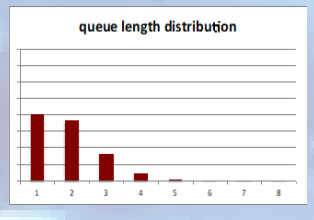


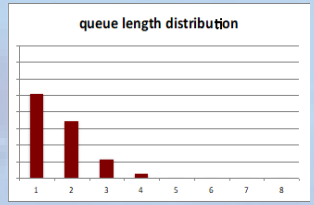
...



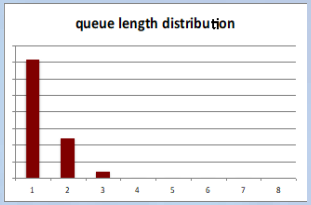
Distributions for Queue sizes, Open times





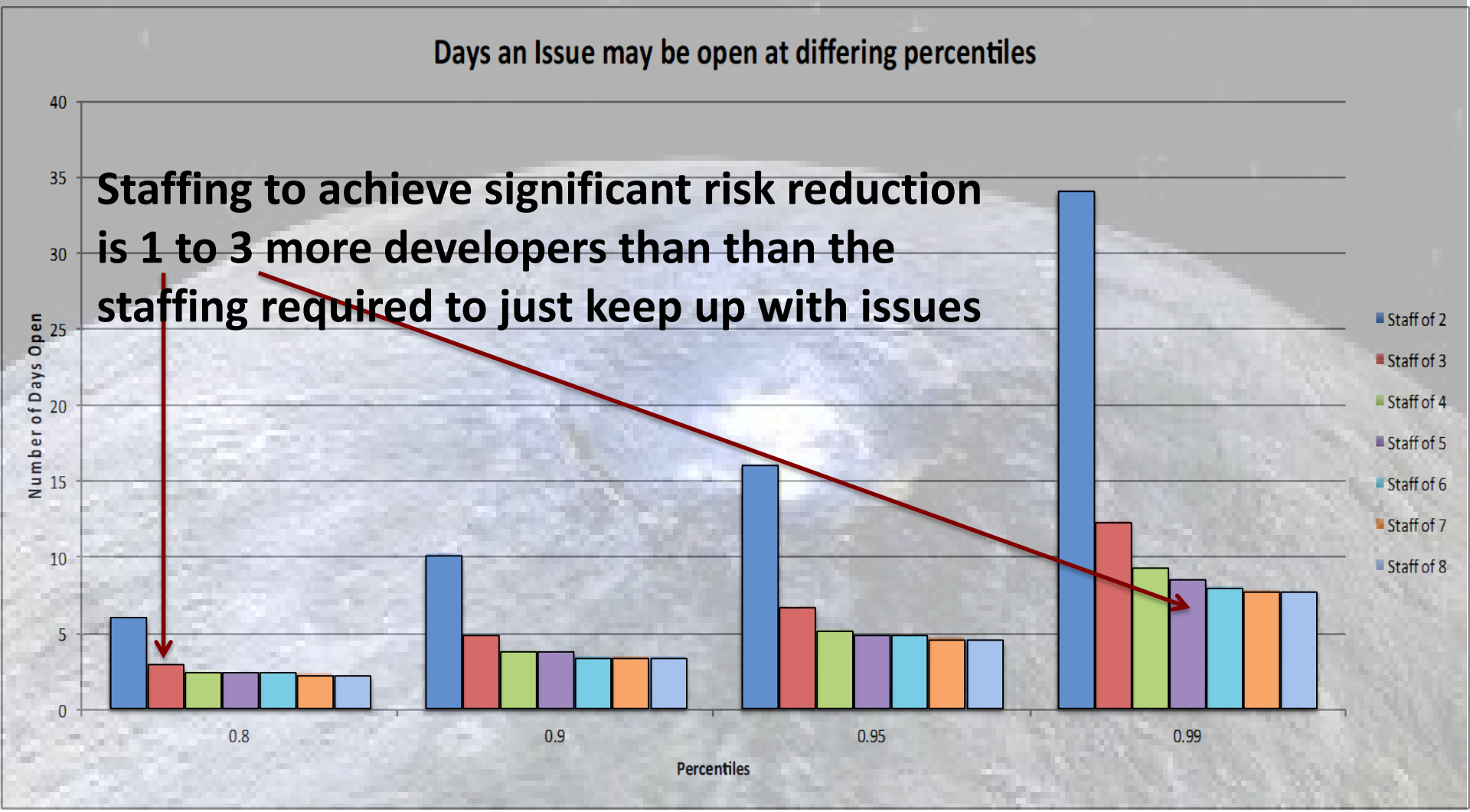


...



©2016 California Institute of Technology. Government sponsorship acknowledged.

Results



©2016 California Institute of Technology. Government sponsorship acknowledged.

Results (2)

- **Given Probability(issue open after t days, staff size k)**
  - **We can determine staffing that successfully resolves issues y % of the time in t days or fewer**
  - **Balance this against the cost of maintaining a staff of a given size.**
- **Staffing discussions focus on institutional risk**
- **Staff does not have to be idle**
  - **New features**
  - **User support**

©2016 California Institute of Technology. Government sponsorship acknowledged.

Next Steps

- How should resources be allocated?
  - We have same maintenance issues
  - Different approach to addressing funding with sponsors
- Advocate for Metrics
  - Models enabled by over a decade of disciplined data collection

©2016 California Institute of Technology. Government sponsorship acknowledged.

Conclusions

- **Maintenance needs to be viewed as a risk mitigation question. Staffing is a function of**
  - **Issue discovery**
  - **Issue response**
  - **Risk comfort**
- **Costs are large**
- **Probabilities are small, but almost certainly larger than you think.**
- **Staffing sizes to lower your risk are are more than the amount it takes to stay even, but not a lot.**

©2016 California Institute of Technology. Government sponsorship acknowledged.