On Rapid Releases and Software Testing

Mika Mäntylä Aalto University, Finland

Foutse Khomh Ecole Polytechnique de Montreal, Canada

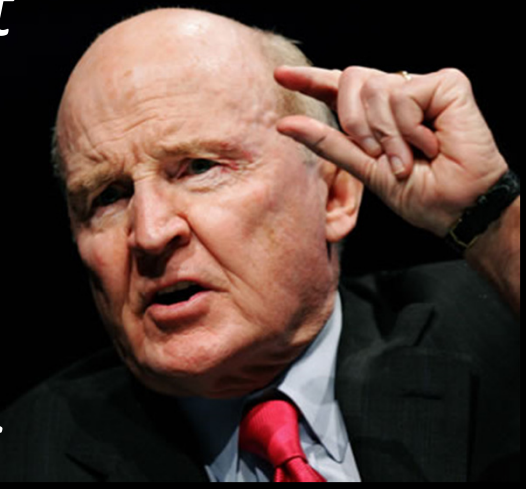
Bram Adams Ecole Polytechnique de Montreal, Canada,

Emelie Engström, Lund University, Sweden

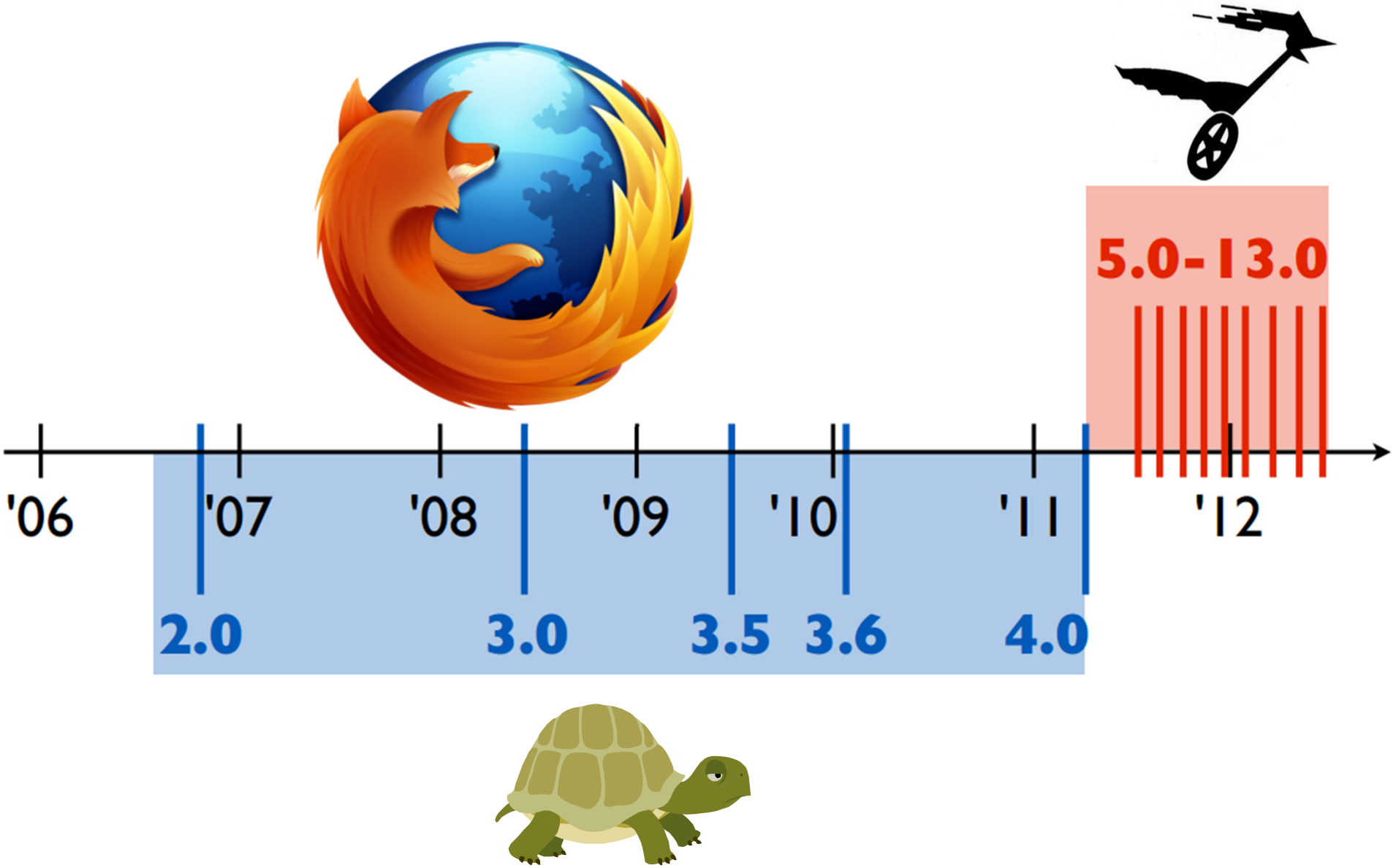
Kai Petersen, Blekinge Institute of Technology, Sweden

Need for rapid releases?

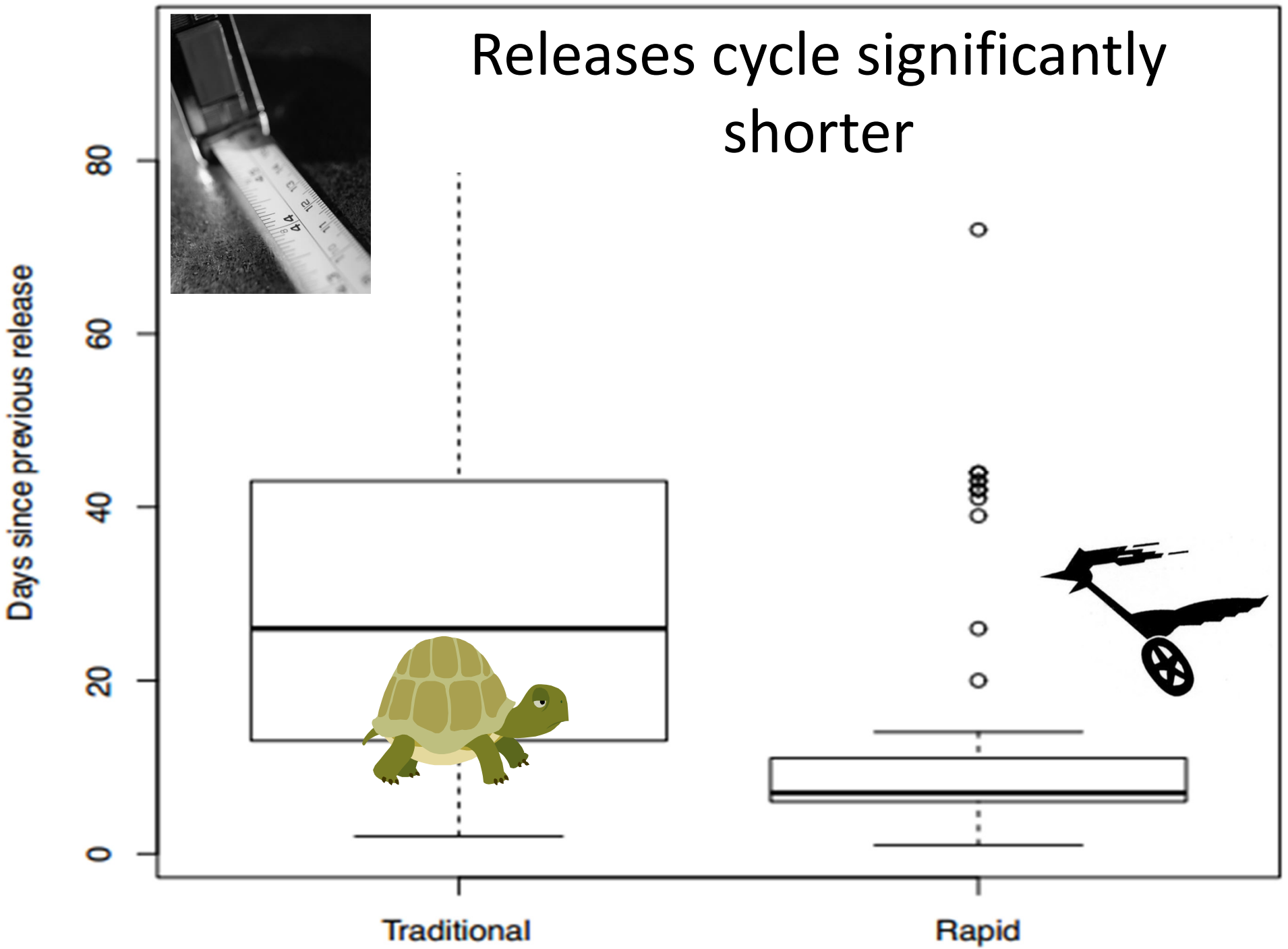
- Jack Welch: *"If you are not moving at the speed of the marketplace you're already dead – you just haven't stopped breathing yet"*
- Jan Bosch: *"Increasing SPEED trumps ANY other improvement software R&D can provide to the company "*



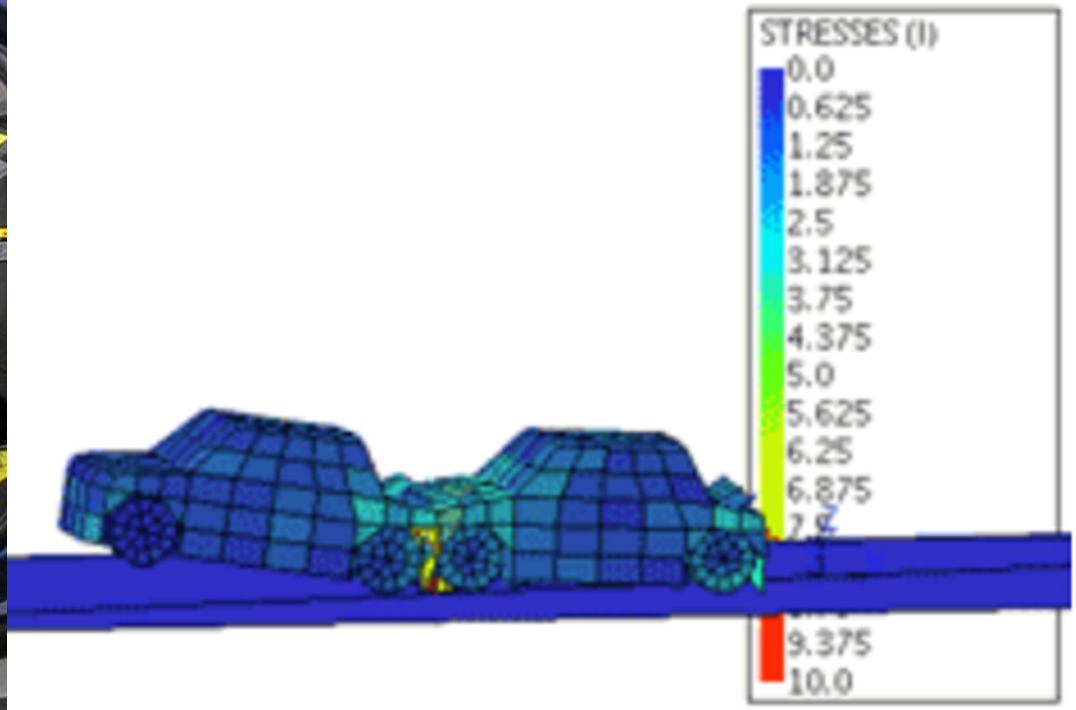
An example: Firefox



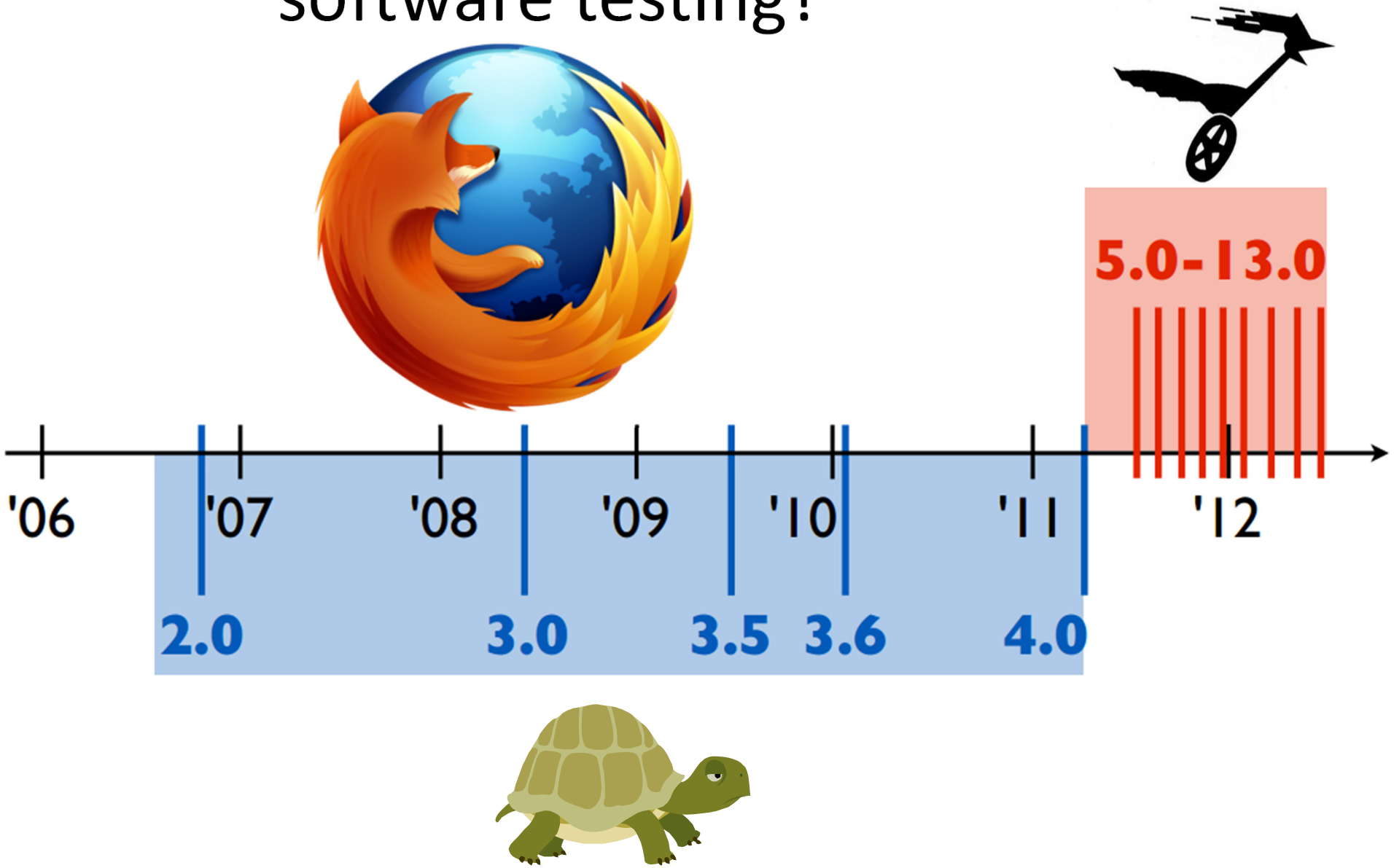
Releases cycle significantly shorter

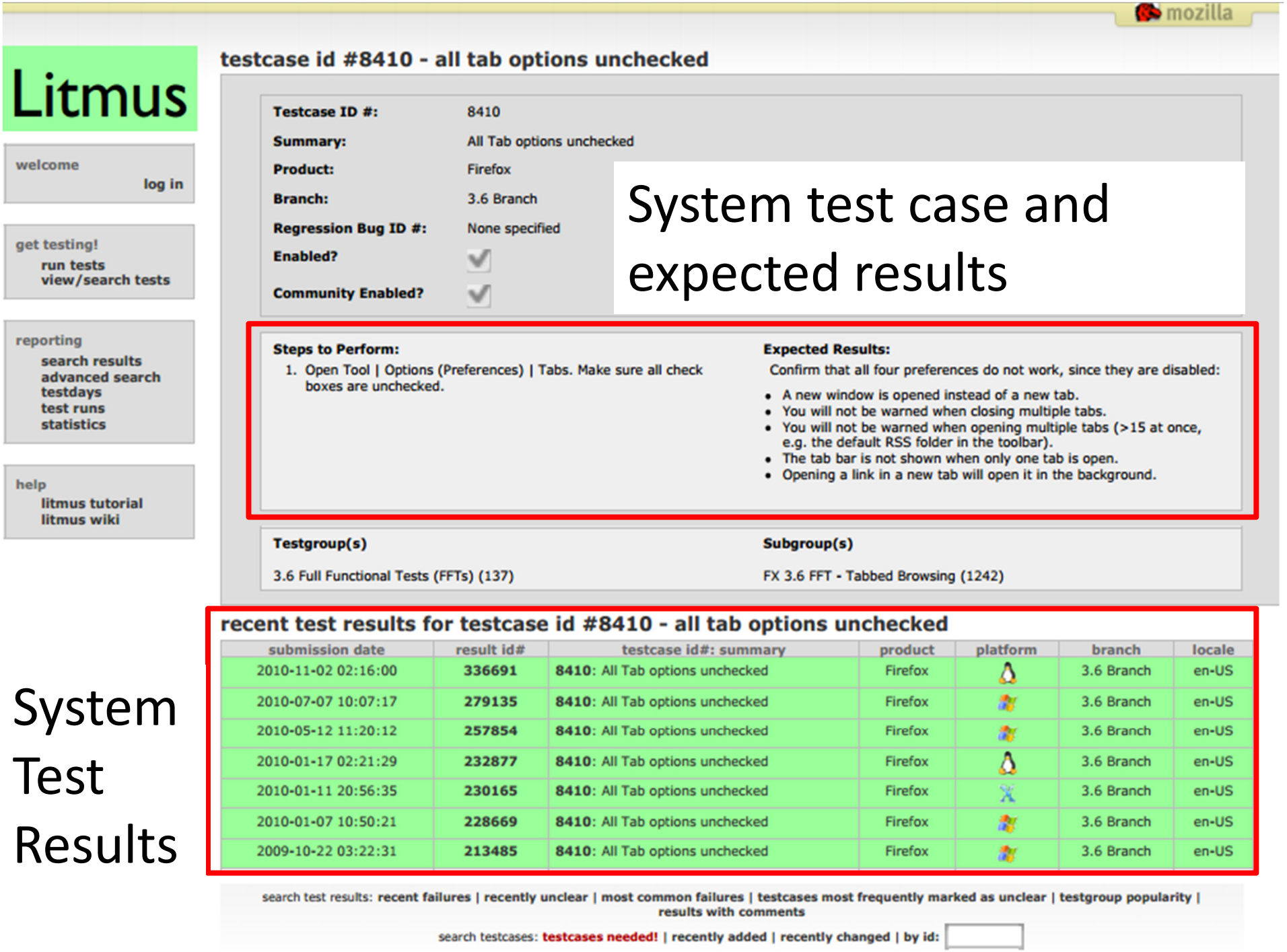


At the expense of …?



Do rapid releases impact software testing?

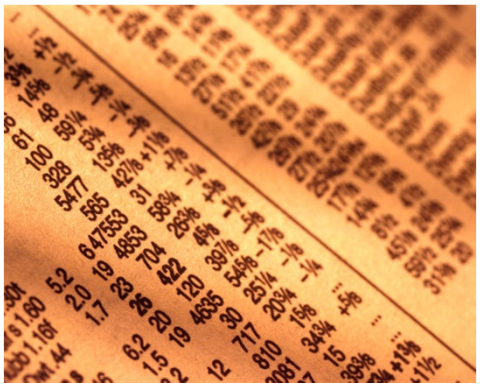




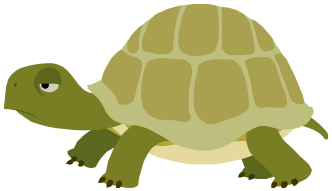
System Test Results

Mining Results

- Descriptive numbers
  - 1,547 unique test cases (roughly 10% of them are automated)
  - 312,502 test case executions
  - 6 years of testing
  - 6,058 individuals
  - 2,009 software builds
  - 22 operating system versions
  - 78 locales



- 5 major TR versions with 147 smaller releases
- 9 major RR versions with 89 smaller releases

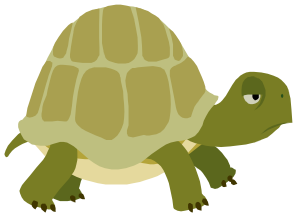


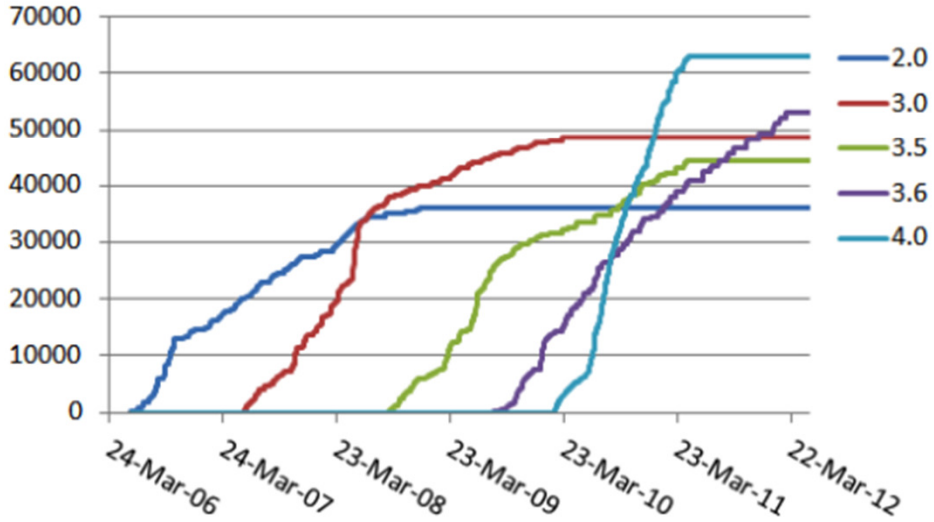
To triangulate:

An email interview with a Mozilla QA engineer who has been working on Firefox QA for 5 years

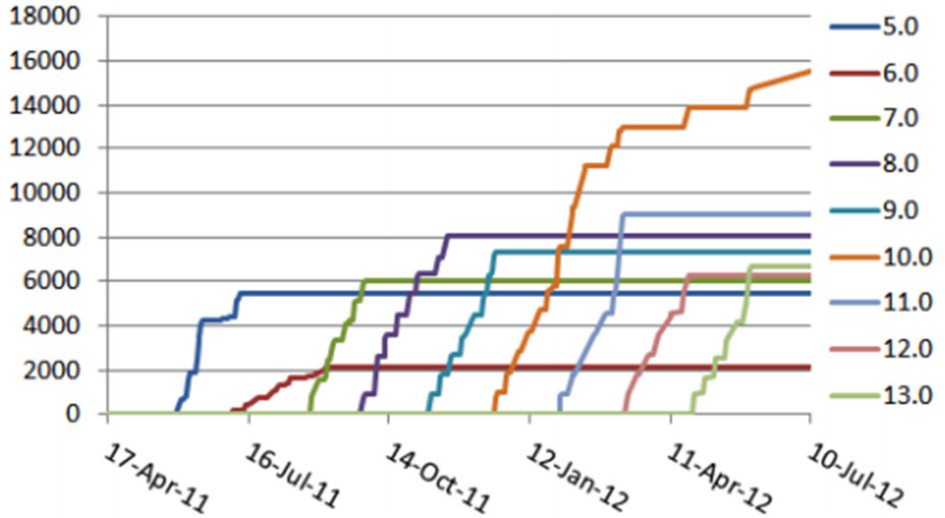
Less test executions

35,000-55,000 executions/release



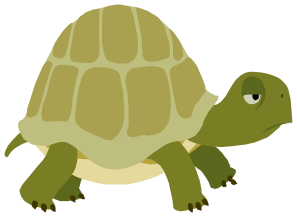


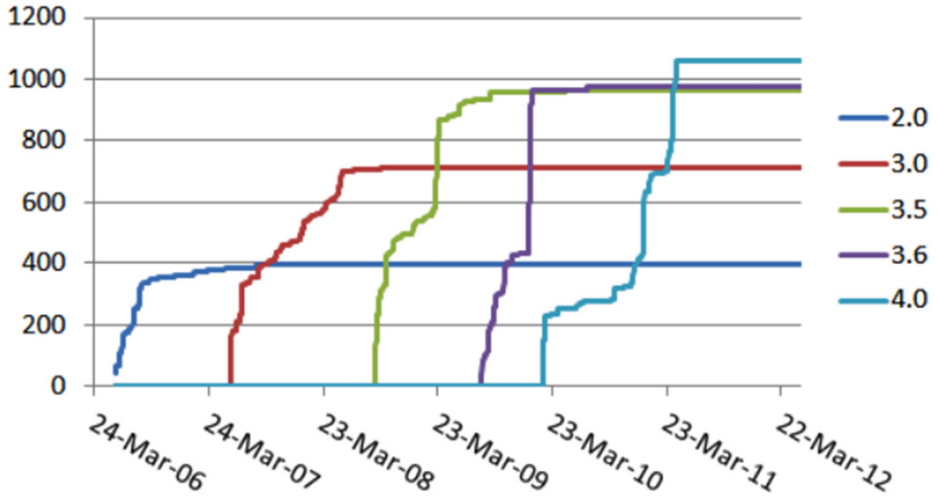
6,000-9,000 executions/release



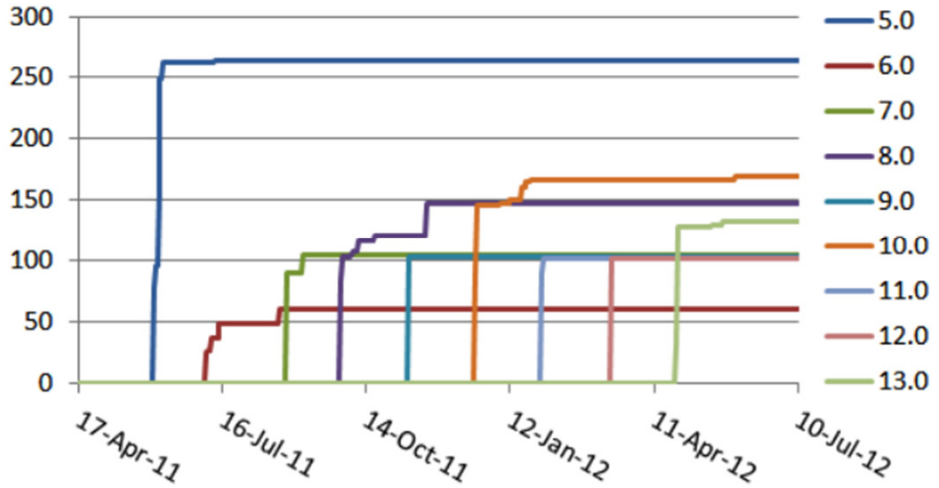
Smaller test coverage

400-1,050 unique test case executions/release

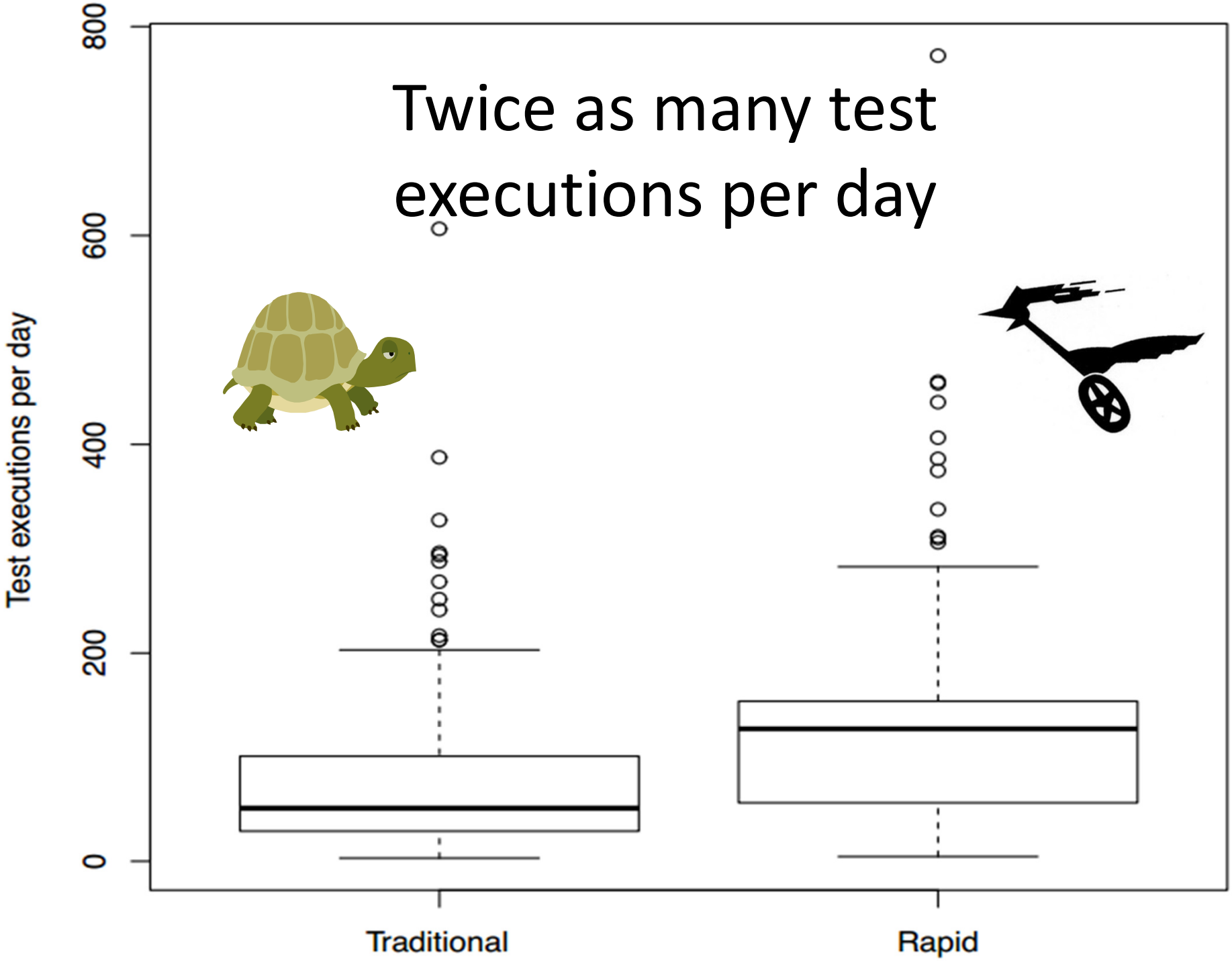




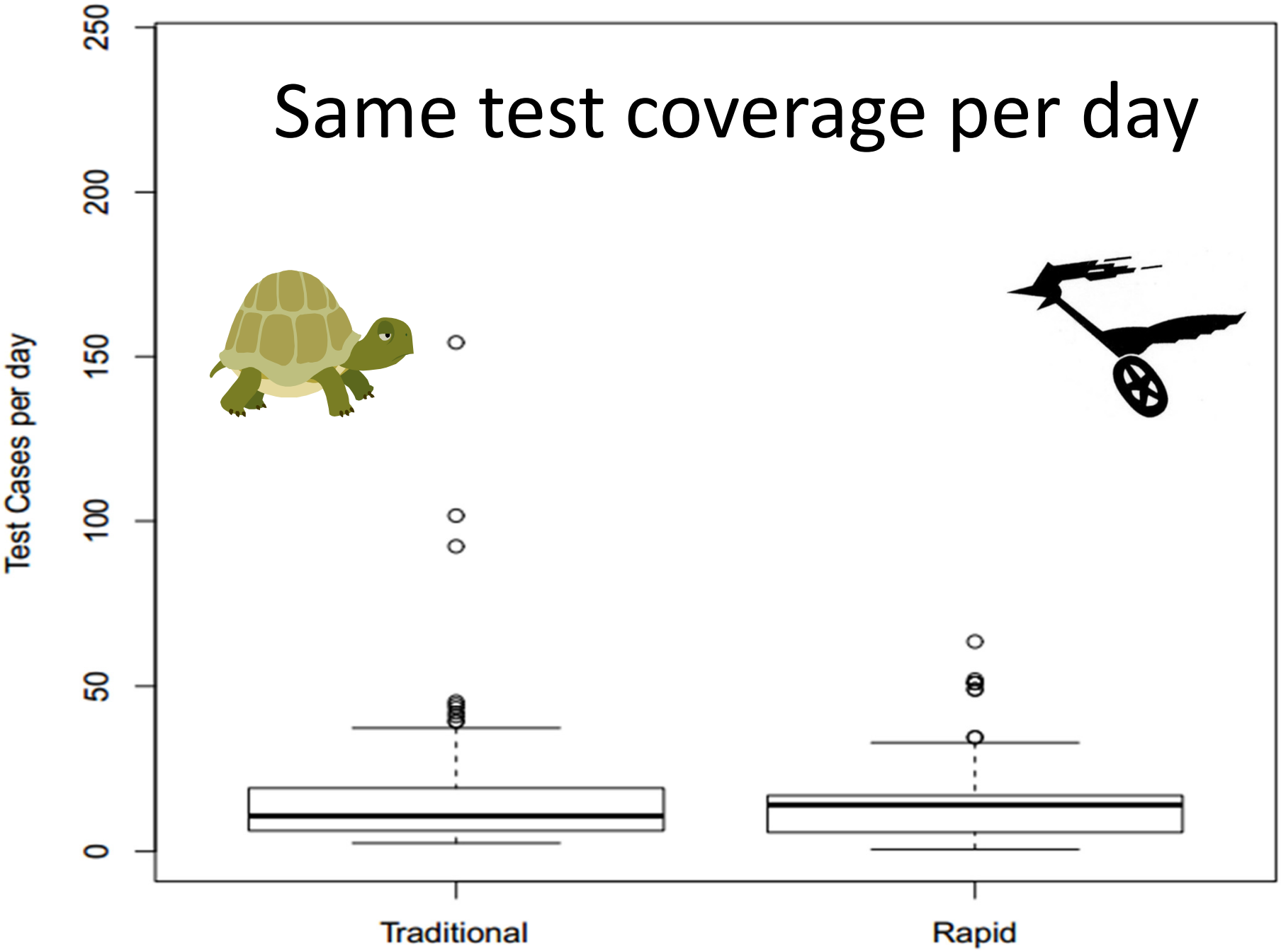
60-260 unique test case executions/release



Twice as many test executions per day



Same test coverage per day



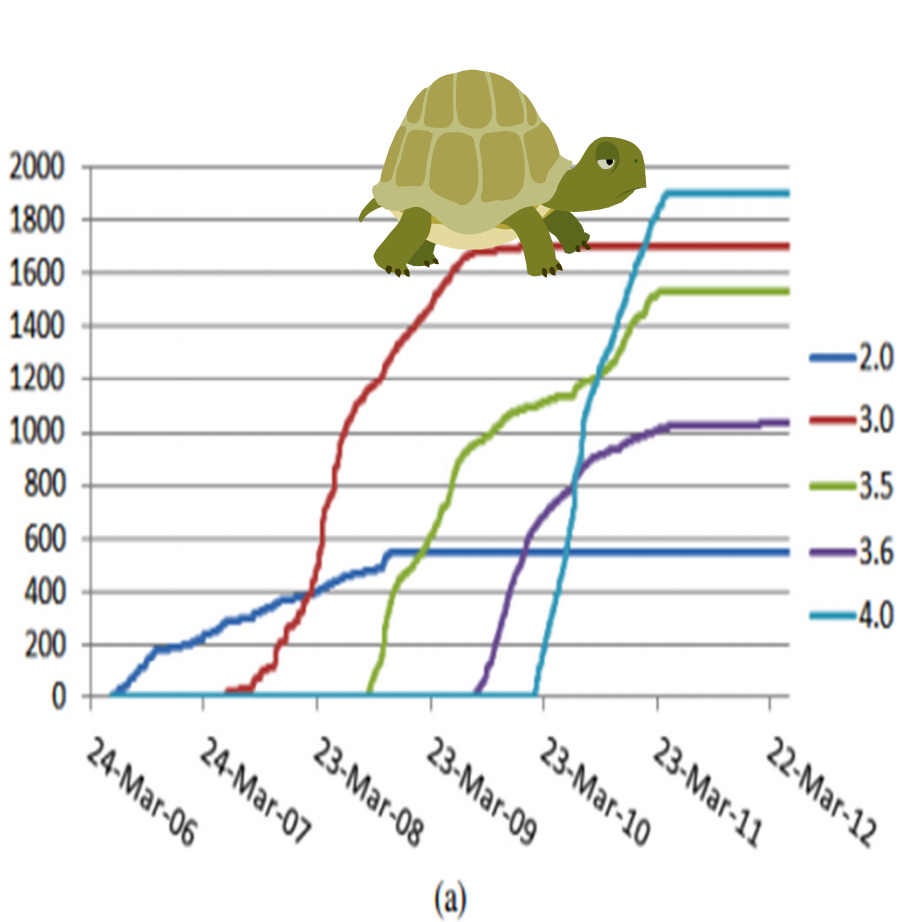
*To survive … we've had to **cut the fat** and focus on … areas … prone to failure, less on … legacy support*

*The greatest strength is that … we can **focus** all of our energy **on deep diving into a few areas***

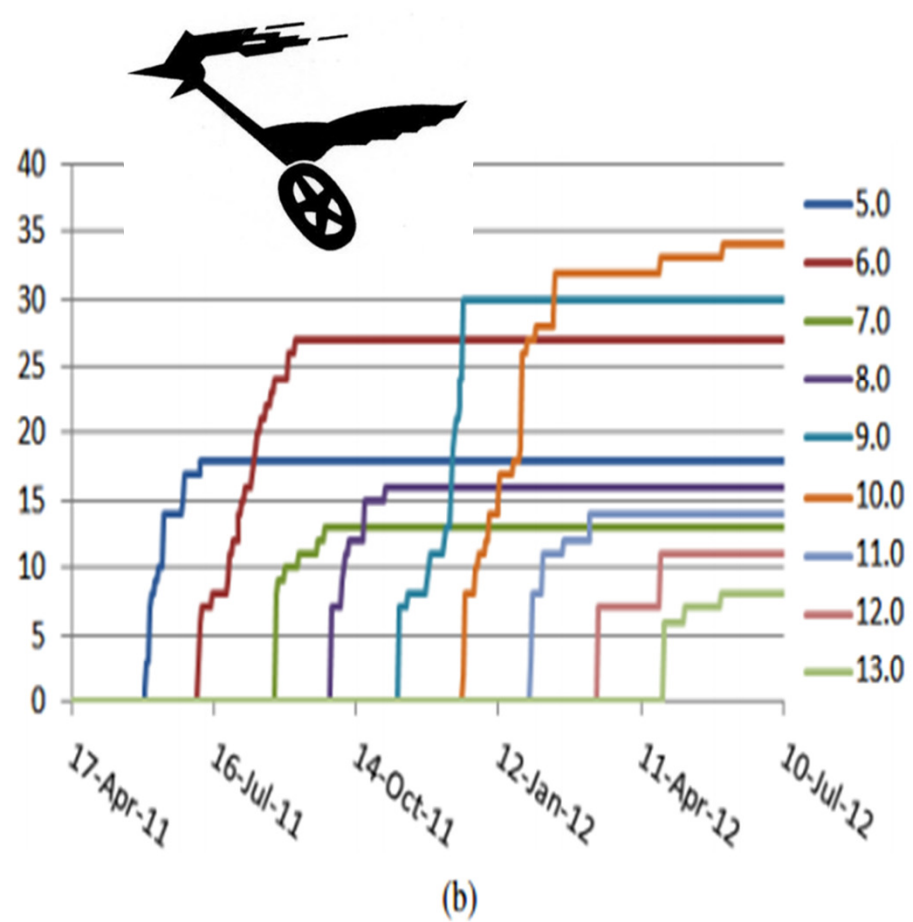
A fixed set of tests for areas prone to regression …dynamically changing set of tests to … high risk features

Less testers

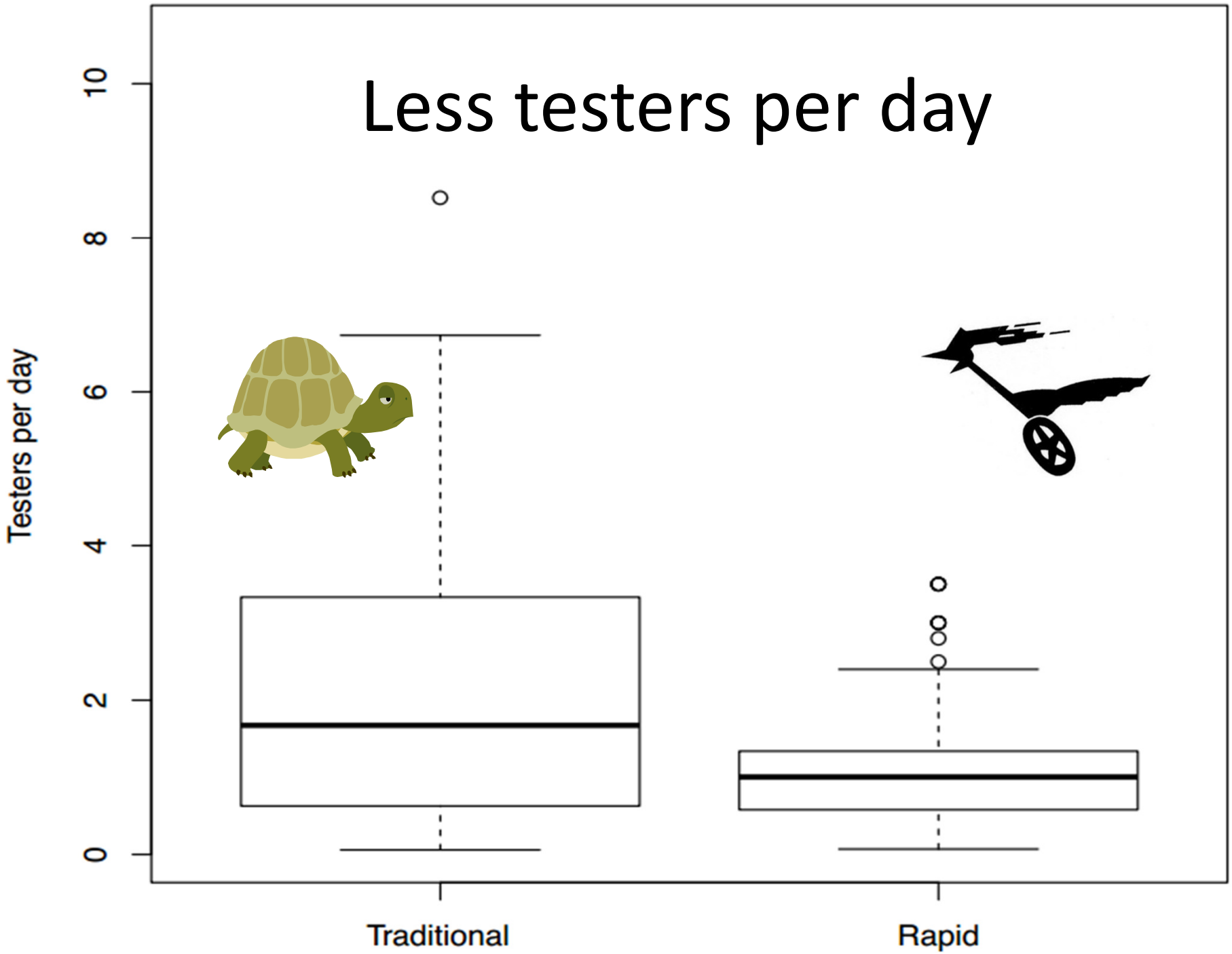
600-1,900 testers/release



7-34 testers / release



Less testers per day

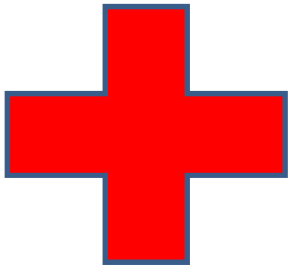


…it's harder to develop a large community

However, in the transition: *1) the core team …unchanged 2) the contract team has nearly doubled . . . We can scale up our team much faster through contractors ….*

fracturing of testing resources to different [Mozilla]projects

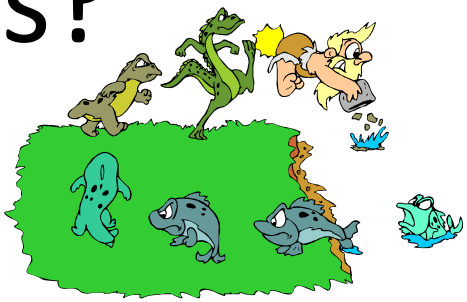
Move to Rapid Releases correlates with



Test executions, Operating systems	**Unique test cases**	**Release length, Testers, Builds, Locales**

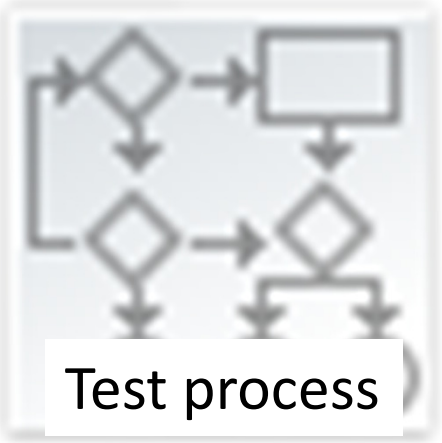
Confounding factors?

Release length



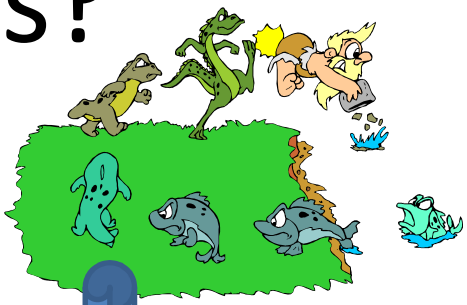
Project evolution

Release model



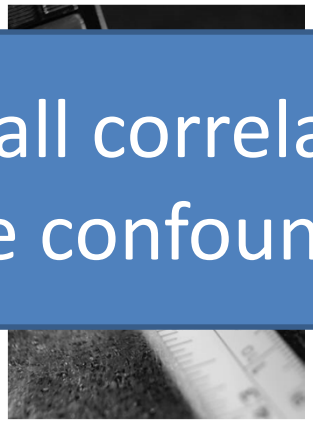
Test process

Confounding factors?



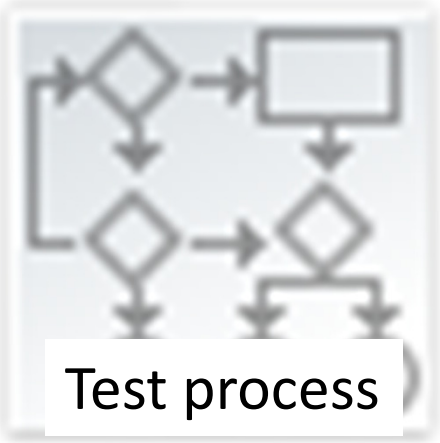
ect evolution

Partial Kendall correlation to control the confounders!



Release length

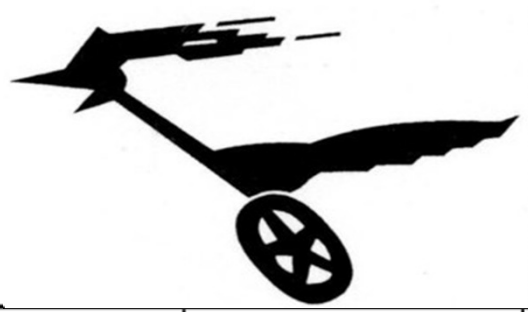
Release model

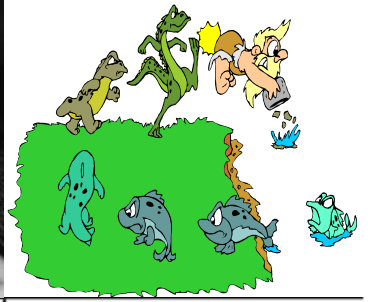


Test process

Shorter release length increases test effort

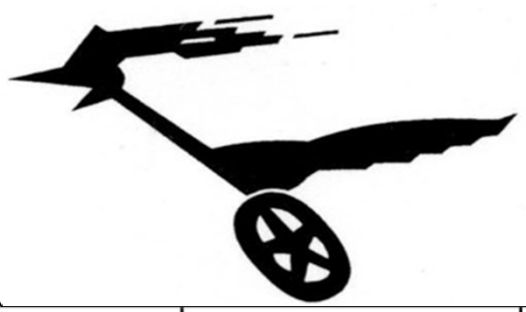
Perhaps due to increased regression testing

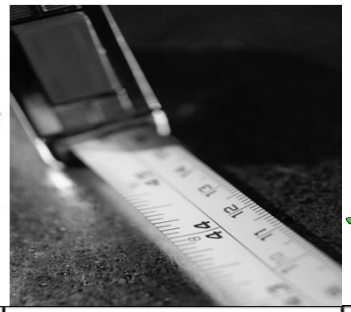


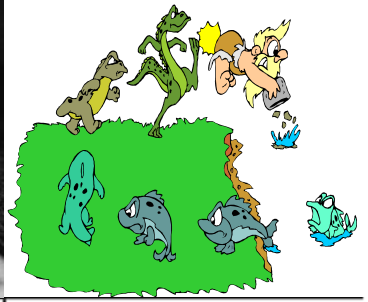


#Test executions per day (RQ1)	0.026	-0.414***	0.048
#Test cases per day (RQ1)	-0.231***	-0.593***	0.017
#Testers per day (RQ2)	-0.224***	-0.331***	-0.069
#Builds per day (RQ3)	-0.105*	-0.258***	-0.176***
#Locales per day (RQ4)	-0.281***	-0.443***	-0.115*
#OSs per day (RQ4)	0.149**	-0.822***	0.130**

Focusing occurs to cope with increased effort

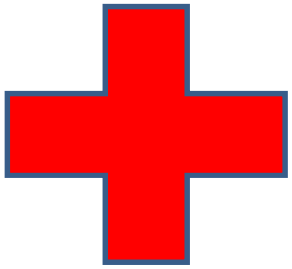






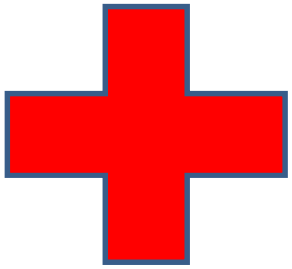
#Test executions per day (RQ1)	0.026	-0.414***	0.048
#Test cases per day (RQ1)	-0.231***	-0.593***	0.017
#Testers per day (RQ2)	-0.224***	-0.331***	-0.069
#Builds per day (RQ3)	-0.105*	-0.258***	-0.176***
#Locales per day (RQ4)	-0.281***	-0.443***	-0.115*
#OSs per day (RQ4)	0.149**	-0.822***	0.130**

Move to Rapid Releases (after controlling confounders) correlates with



~~Test executions~~, Operating systems,	**~~Unique test cases,~~ +Test executions**	**+Unique test cases Testers, Builds, Locales**

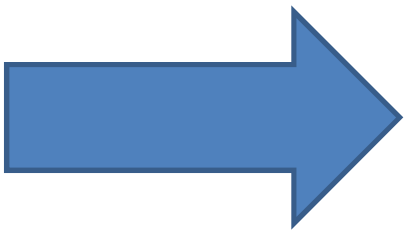
Move to Rapid Releases (after controlling confounders) correlates with



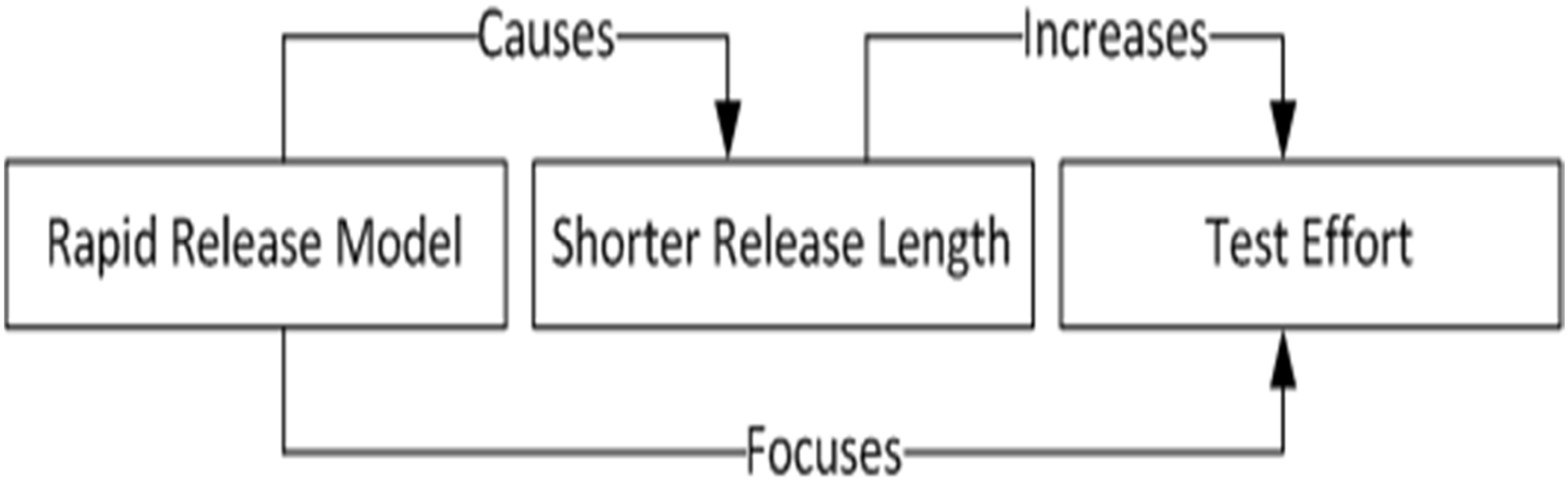
Operating systems,	**Test executions**	**Unique test cases Testers, Builds, Locales**

It's not just the Shorter Cycle Time, but also a Change in Development Process!

- Smaller Scope of Testing
- Smaller Testing Team
  - with more hired resources
- Test Priorities
  - More OSs, less locales, less builds



To summarize

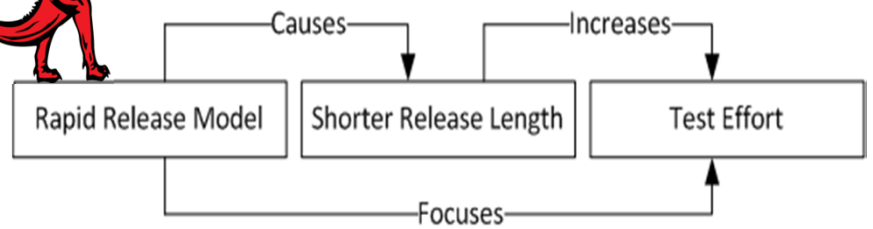


The greatest strength [of rapid releases] is that **the scope of what needs to be tested is narrow so** we can focus all of our energy on deep diving into a few areas

On Rapid Releases and Software Testing

Mika Mäntylä Aalto University, Finland
Foutse Khomh Ecole Polytechnique de Montreal, Canada
Bram Adams Ecole Polytechnique de Montreal, Canada,
Emelie Engström, Lund University, Sweden
Kai Petersen, Blekinge Institute of Technology

To survive under the time constraints of a rapid release we've had to cut the fat and focus on those test areas which are prone to failure, less on ensuring legacy support



Operating systems,	Test executions	Unique test cases Testers, Builds, Locales