An Empirical Study of Highly-Impactful Bugs in Mozilla Projects

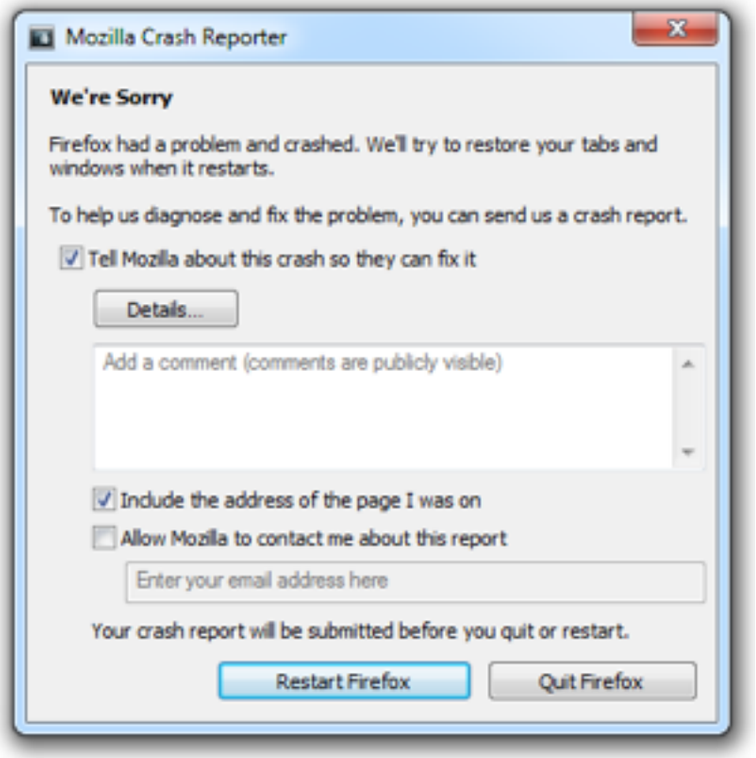
Le An, Foutse Khomh
Polytechnique Montréal, Canada

4 August, Vancouver, Canada

Outline

- Background (Mozilla crash collecting system)
- Challenge
- Research questions
- Study design
- Case study results
- Future work
- Conclusion

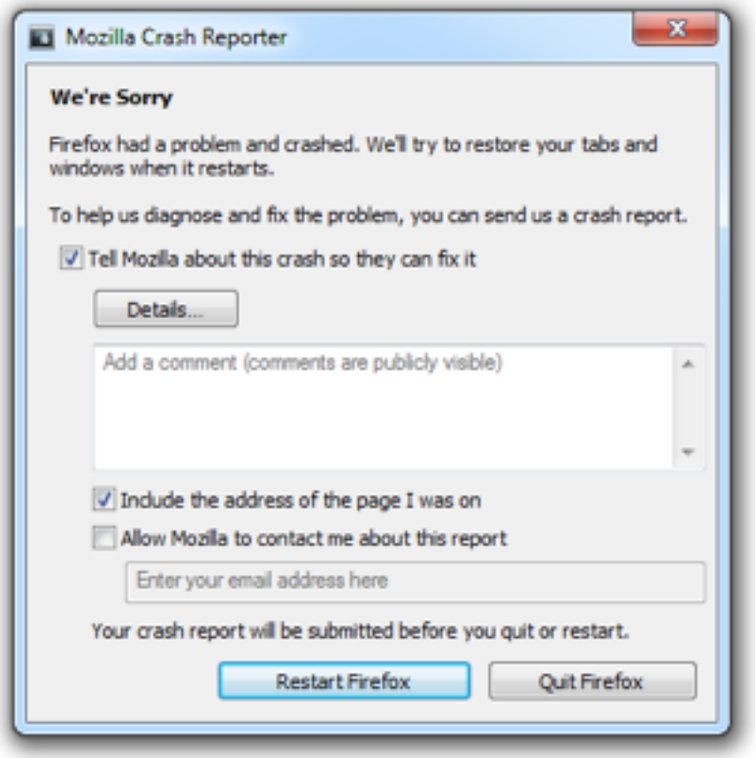
Crash Collecting System

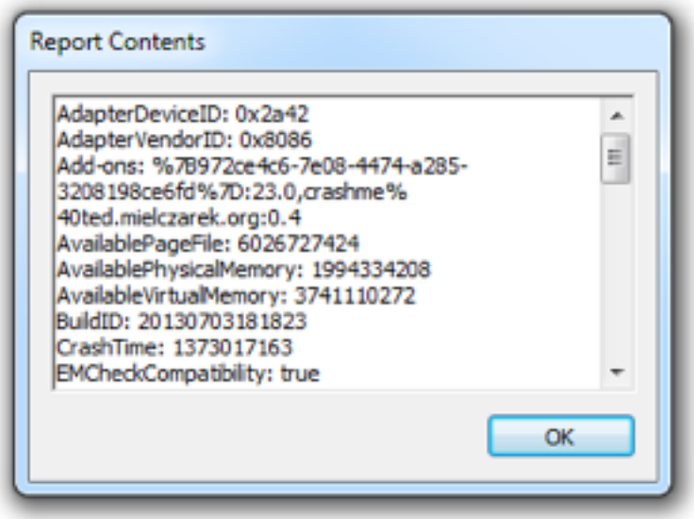


https://support.mozilla.org/en-US/kb/mozillacrashreporter

https://crash-stats.mozilla.com/

Crash Collecting System



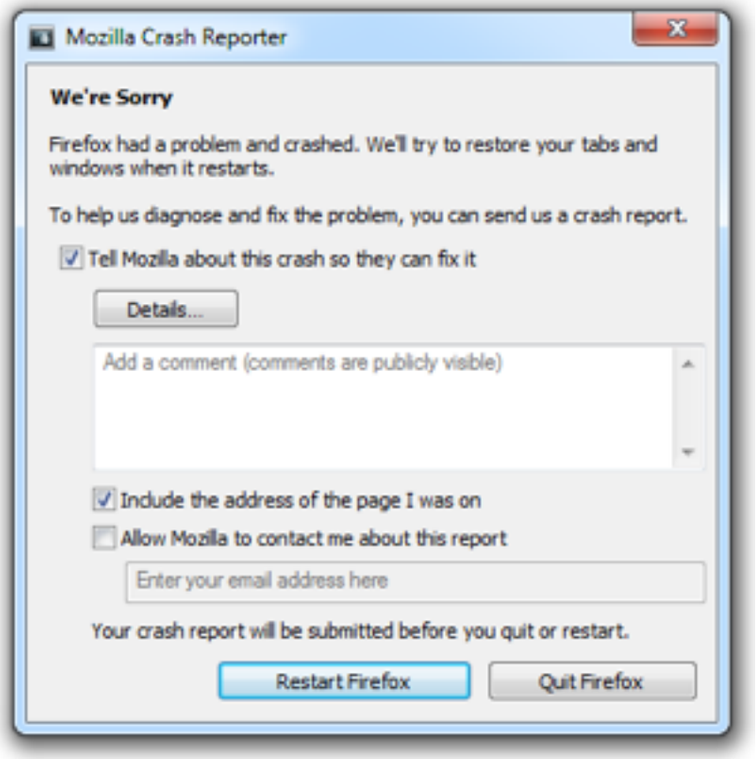


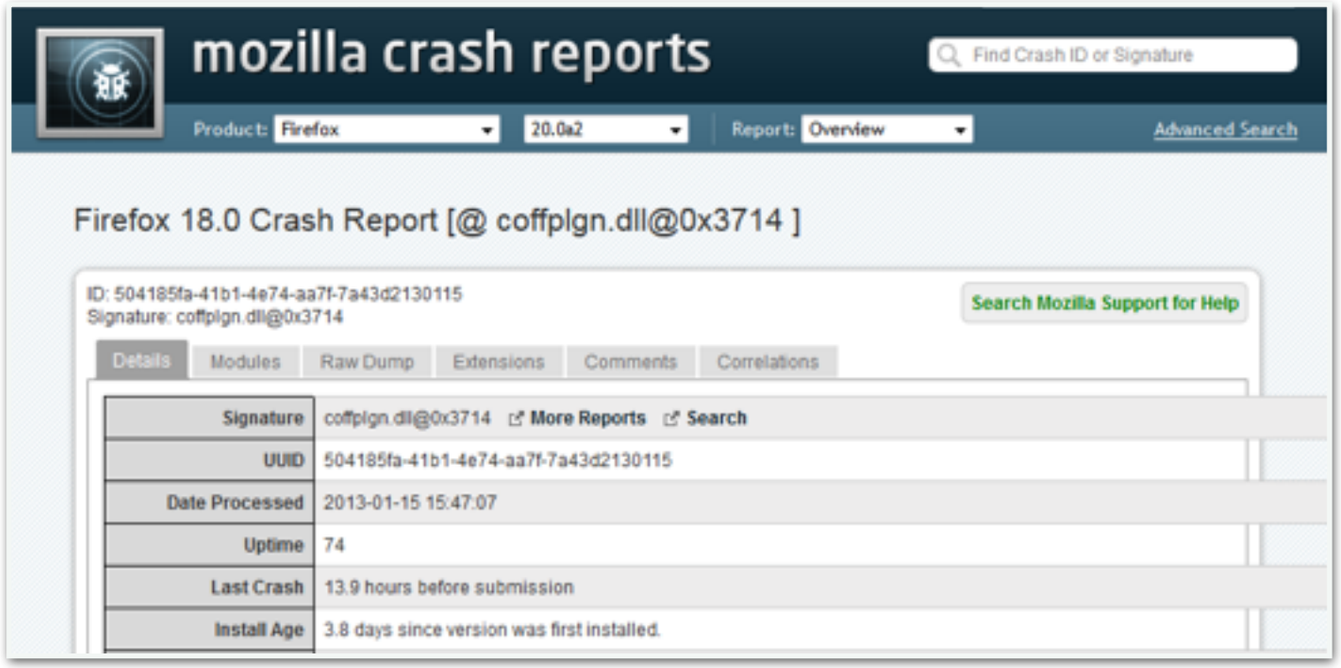
https://support.mozilla.org/en-US/kb/mozillacrashreporter

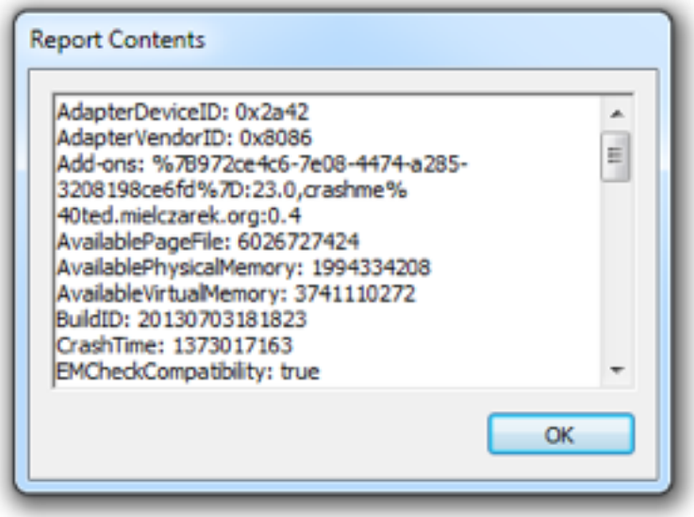
https://crash-stats.mozilla.com/

Crash Collecting System

Socorro



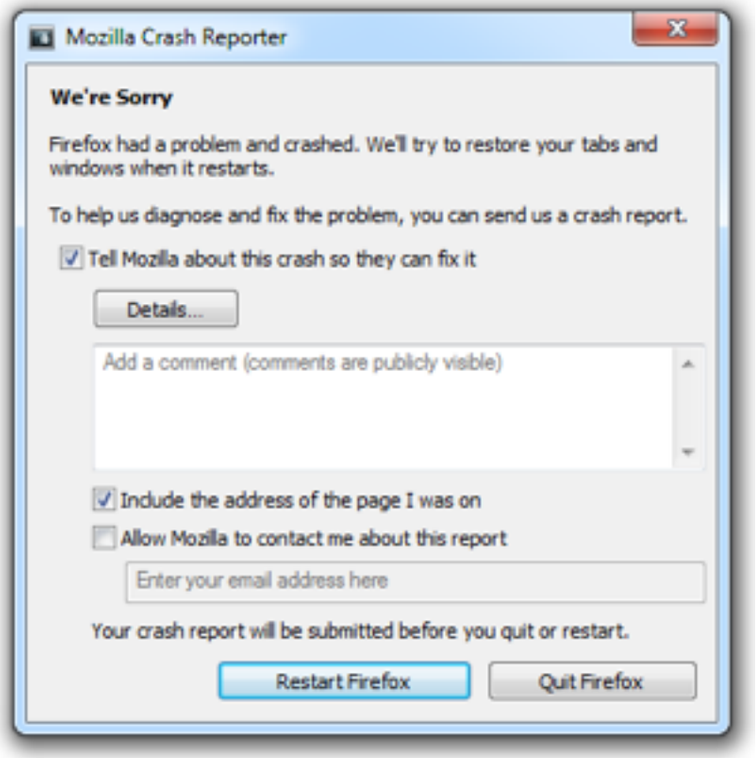


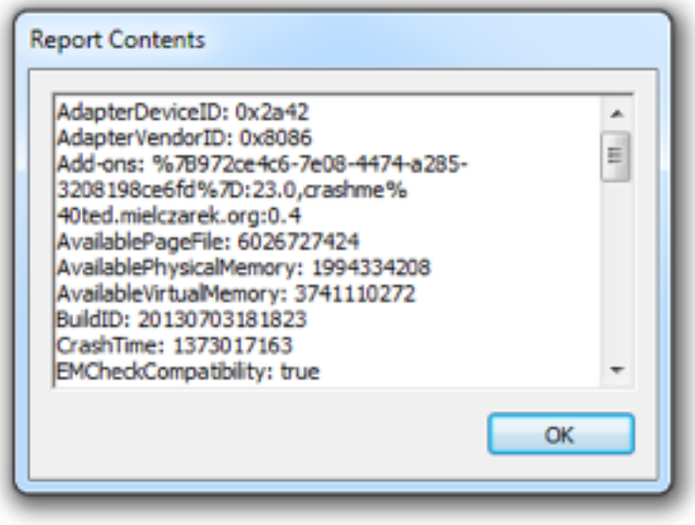


https://support.mozilla.org/en-US/kb/mozillacrashreporter

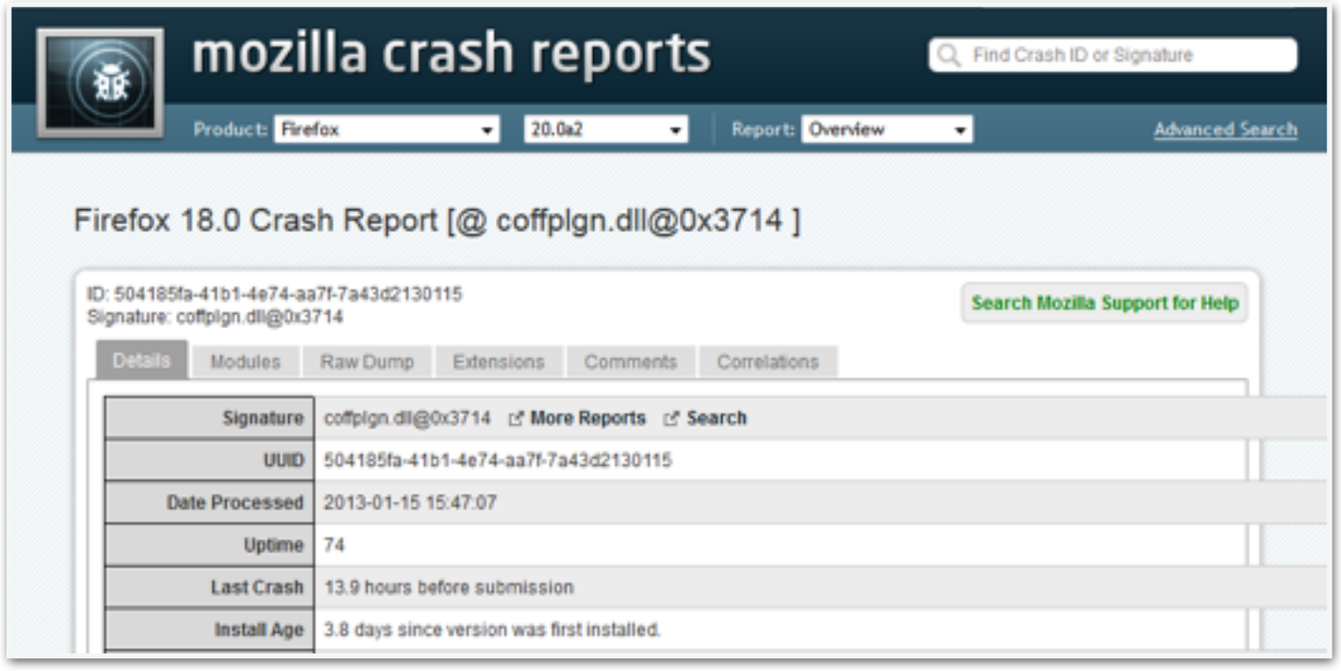
https://crash-stats.mozilla.com/

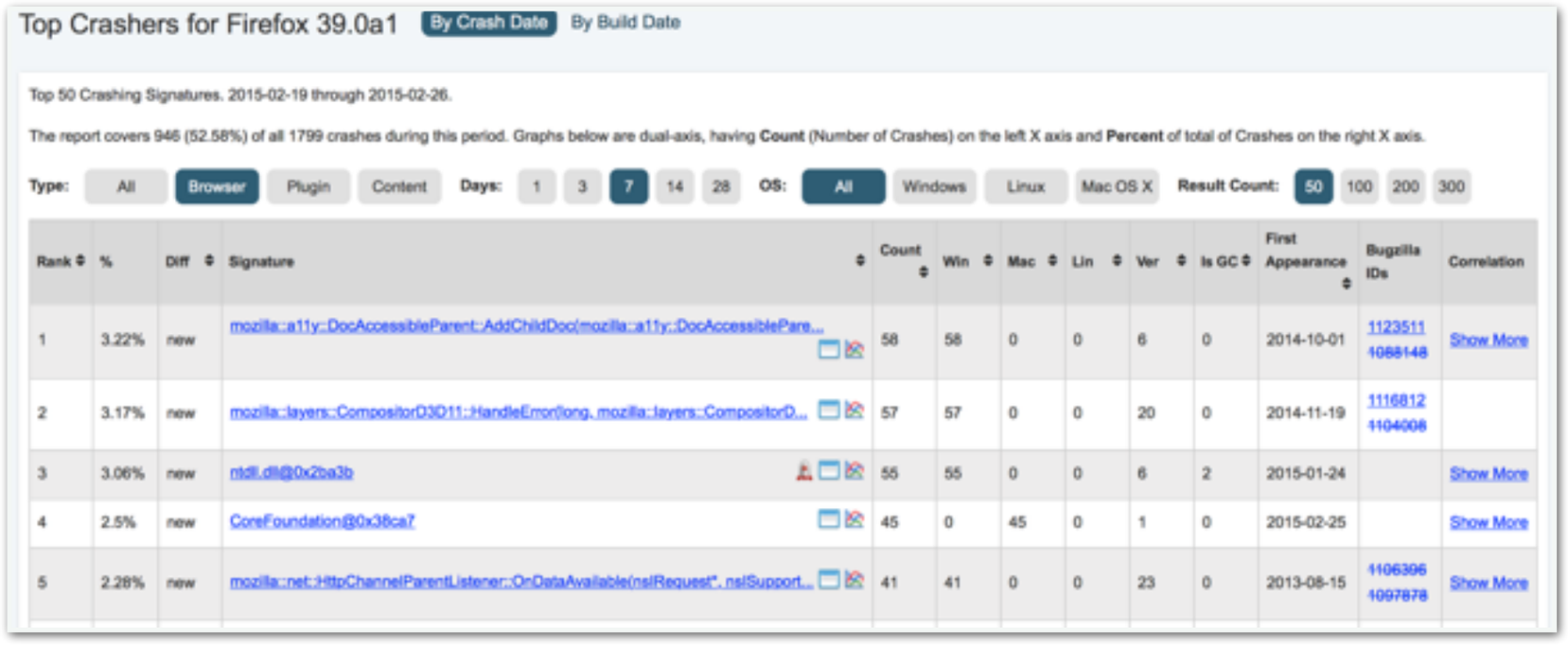
Crash Collecting System





Socorro





https://support.mozilla.org/en-US/kb/mozillacrashreporter

https://crash-stats.mozilla.com/

Challenge

Challenge

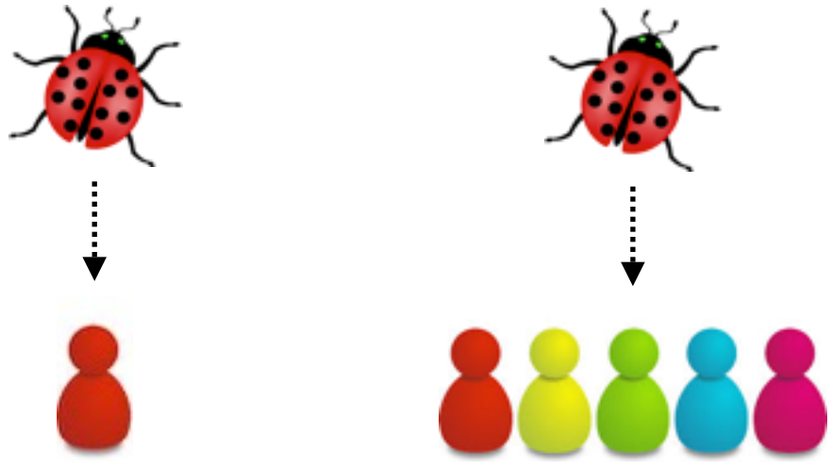
Should we triage crash-related bugs only by their crashing frequency?

Challenge

Should we triage crash-related bugs only by their crashing frequency?

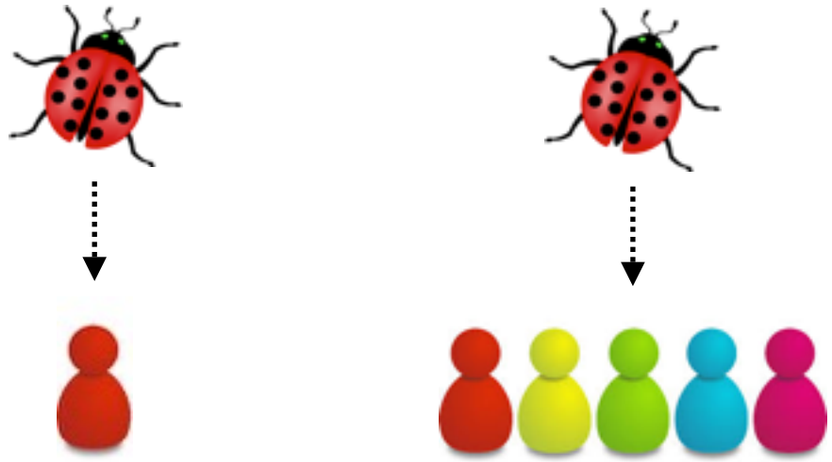
Challenge

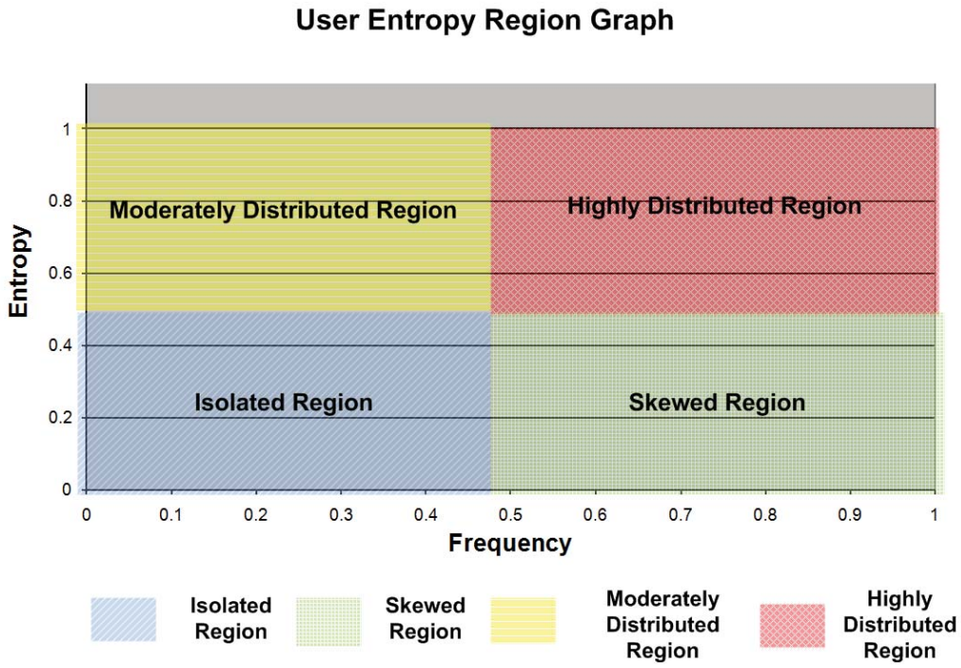
Should we triage crash-related bugs only by their crashing frequency?



Challenge

Should we triage crash-related bugs only by their crashing frequency?

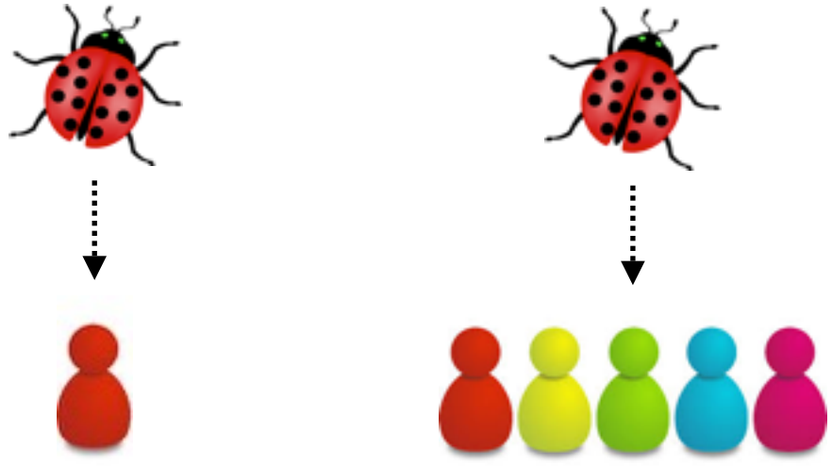




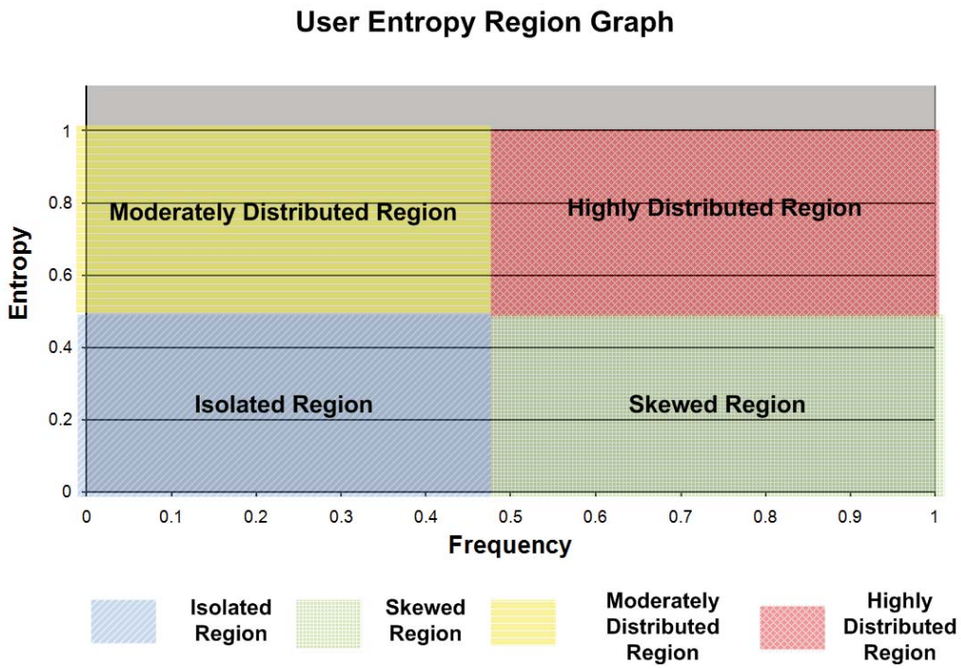
Khomh et al. (2011)

Challenge

Should we triage crash-related bugs only by their crashing frequency?



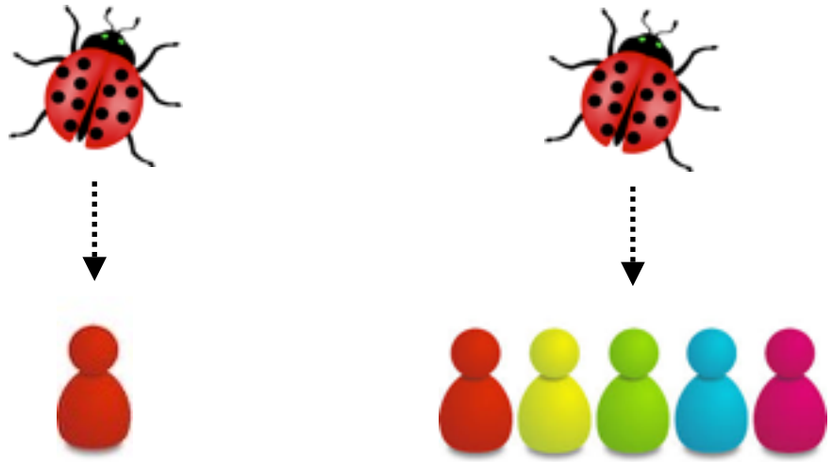
highly-impactful bugs = bugs affecting a large number of users with high crashing frequency



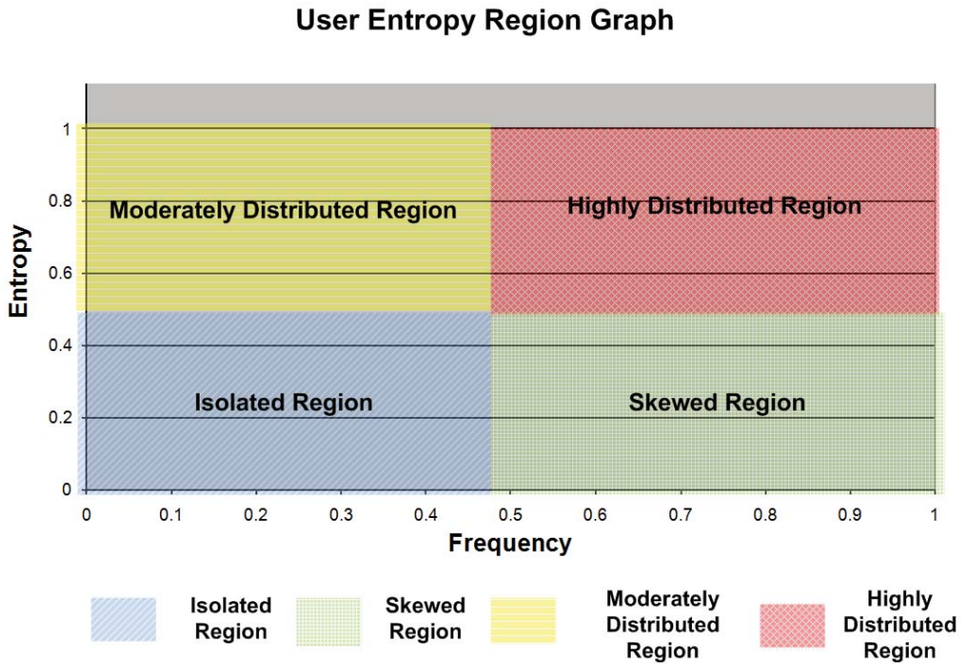
Khomh et al. (2011)

Challenge

Should we triage crash-related bugs only by their crashing frequency?



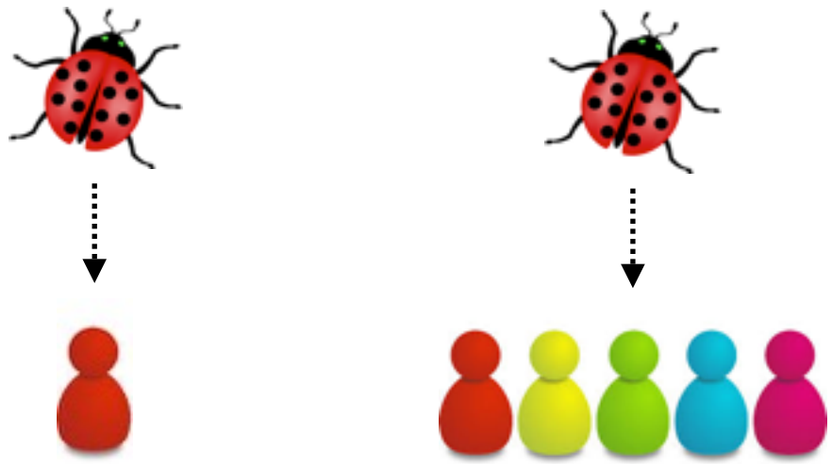
highly-impactful bugs = bugs affecting a large number of users (*entropy*) with high crashing frequency (*occurrences*)



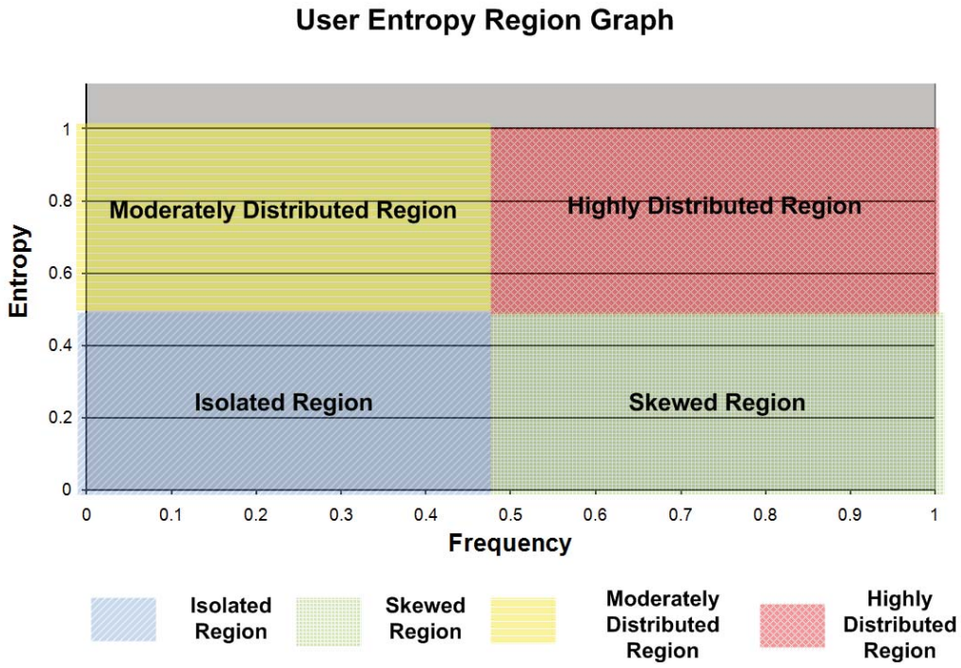
Khomh et al. (2011)

Challenge

Should we triage crash-related bugs only by their crashing frequency?



highly-impactful bugs = bugs affecting a large number of users (*entropy*) with high crashing frequency (*occurrences*)



Khomh et al. (2011)

When the region graph is built, large number of end users have been impacted from crashes for a long time

Research Questions?

What is the percentage of highly-impactful bugs?

What are the characteristics of highly-impactful bugs?

Could we predict highly-impactful bugs?

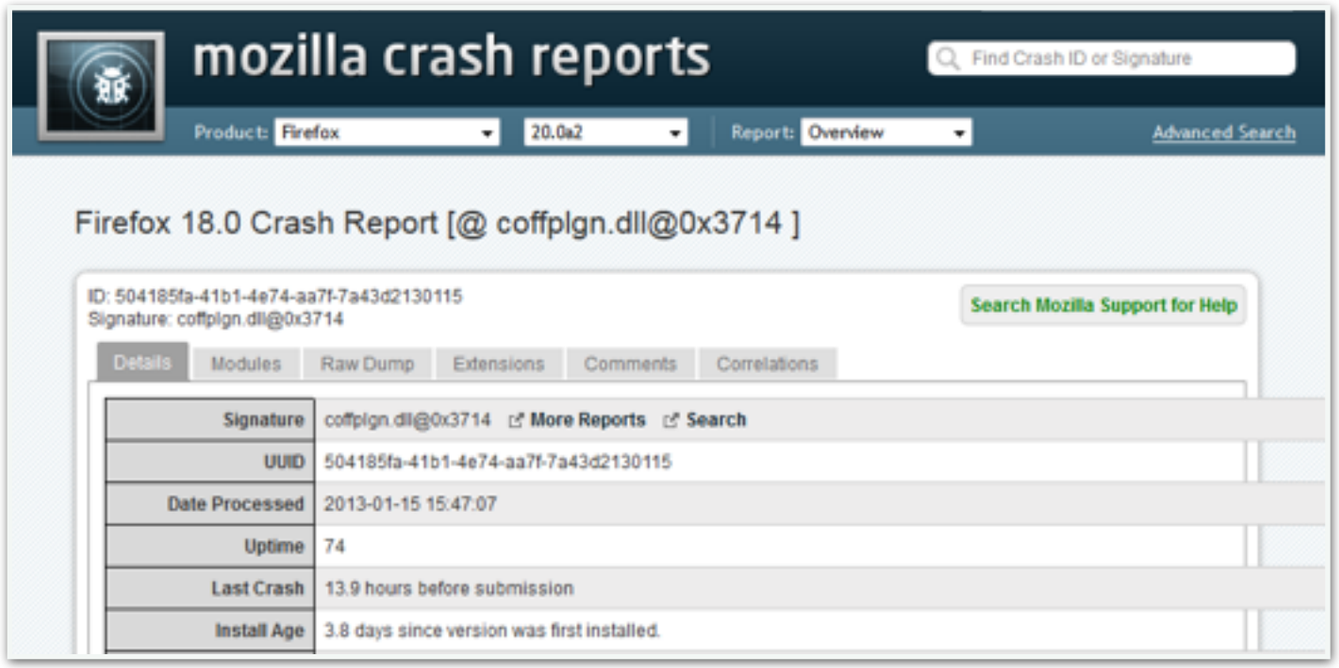
What are the benefits of applying our early triaging technique?

Subject Systems

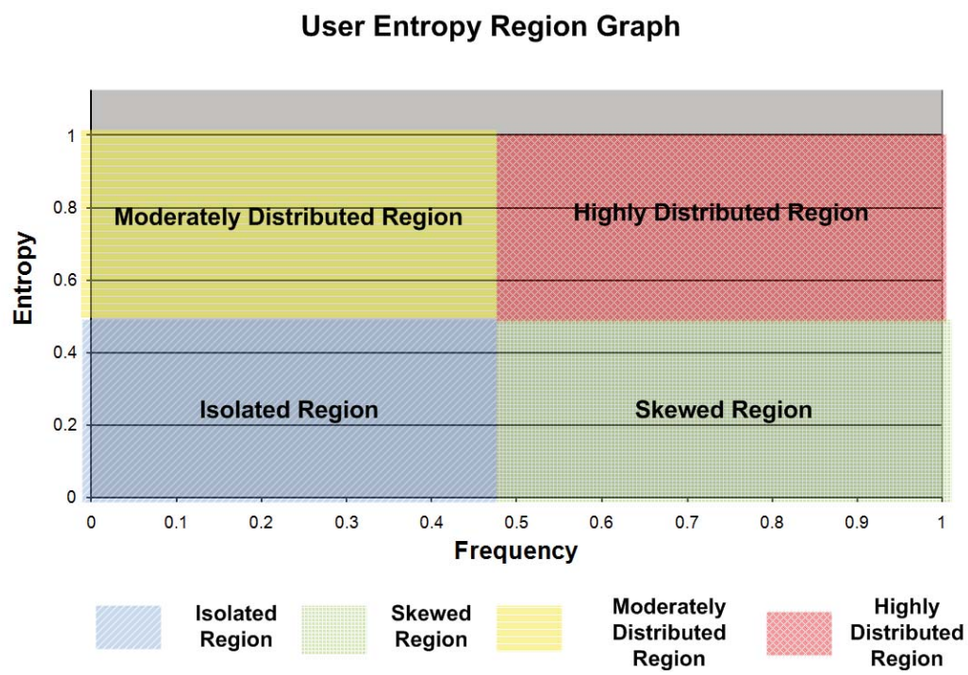
Mozilla Firefox

Mozilla Fennec
(for Android)

Identification of Highly-impactful Bugs

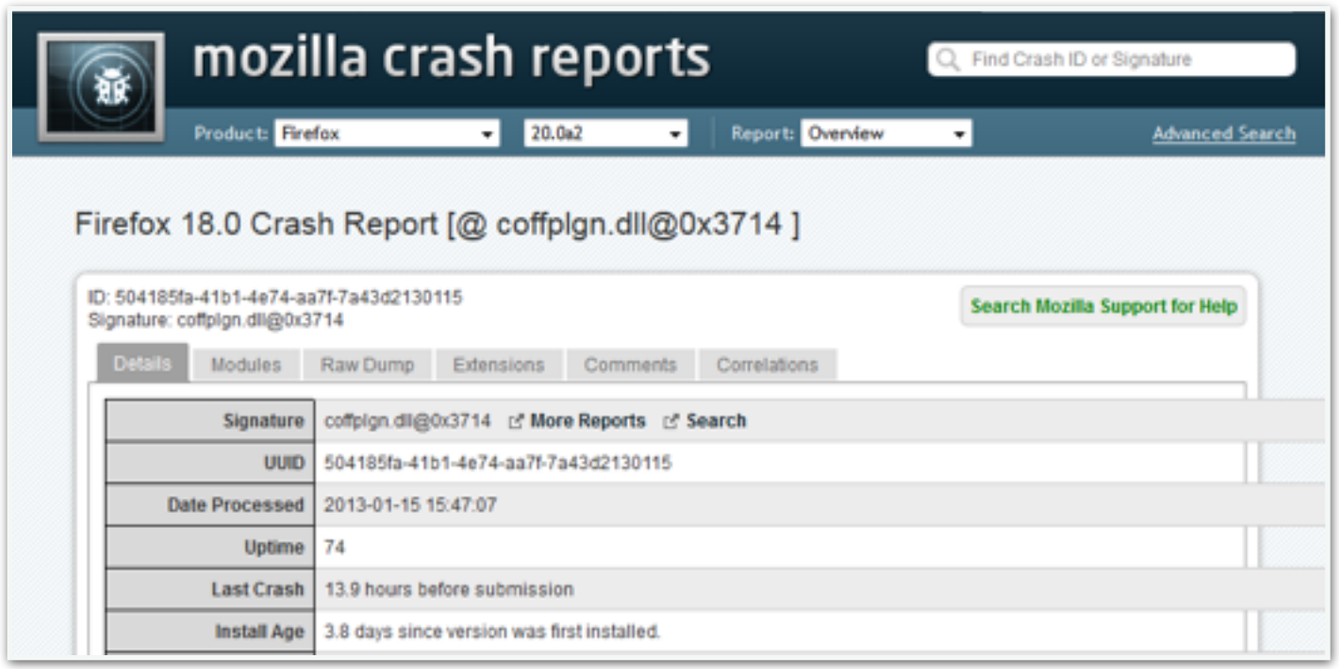


$$H_n(b) = -\sum_{i=1}^{n} p_i \times log_n(p_i)$$



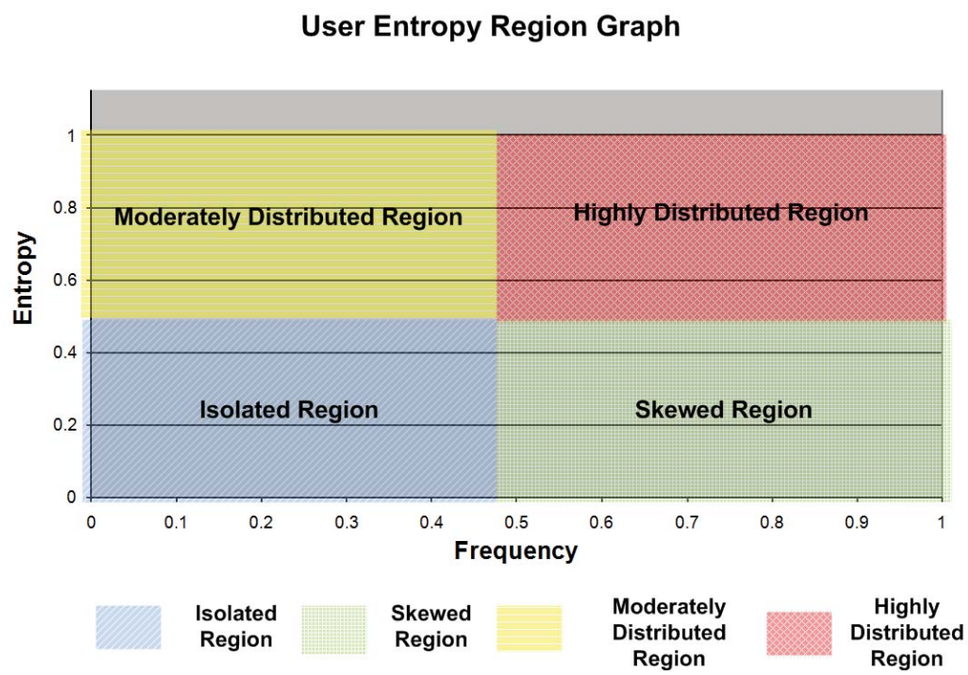
Khomh et al. (2011)

Identification of Highly-impactful Bugs



$$H_n(b) = -\sum_{i=1}^{n} p_i \times log_n(p_i)$$

machine profile (CPU, OS, OS version)



Khomh et al. (2011)

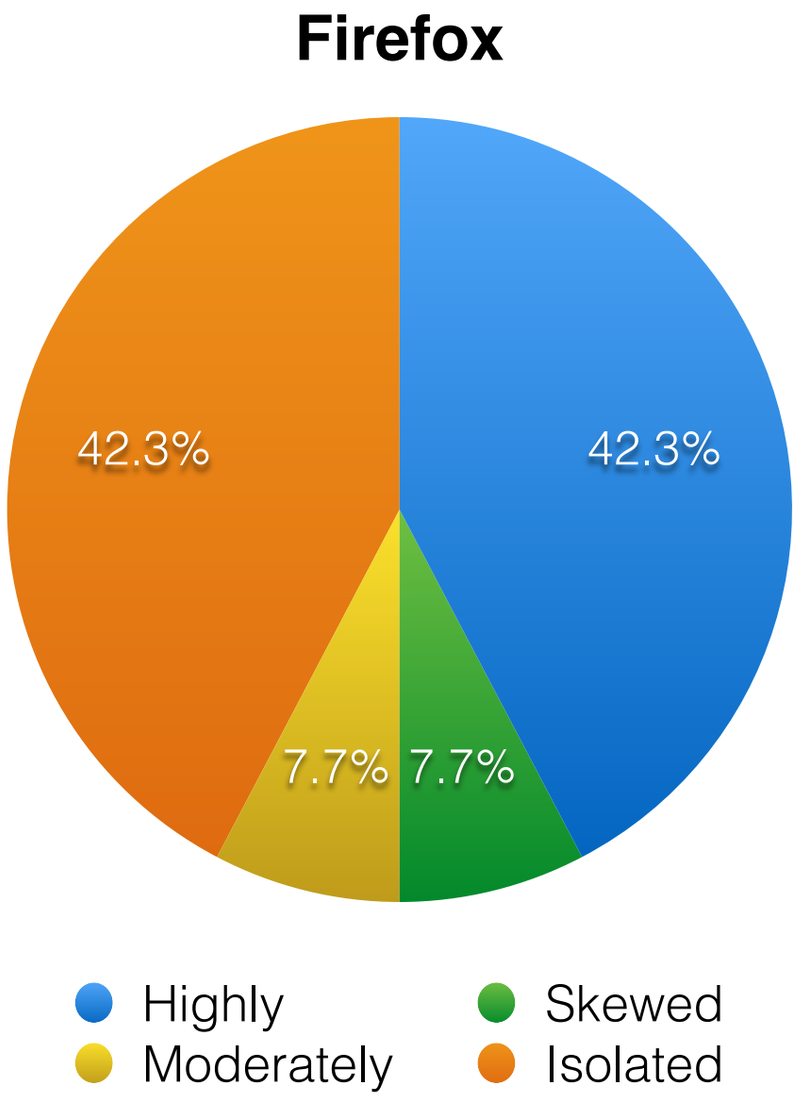
What is the percentage of highly-impactful bugs?

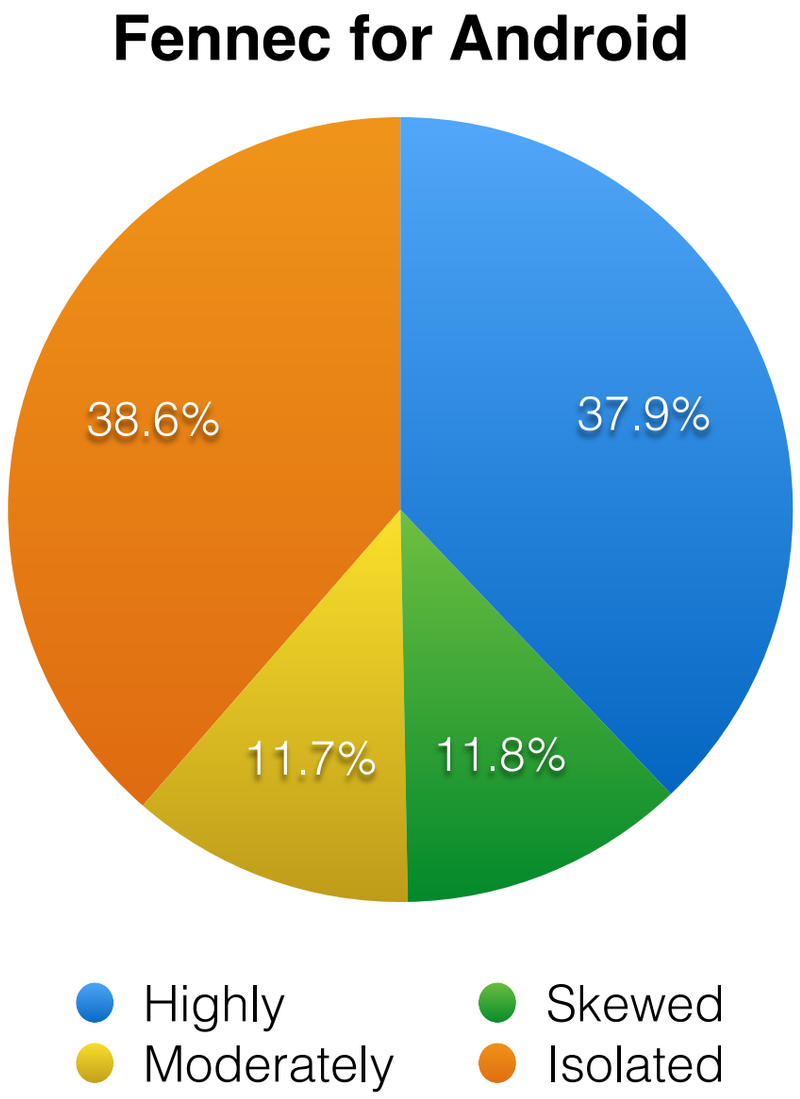
What are the characteristics of highly-impactful bugs?

Could we predict highly-impactful bugs?

What are the benefits of applying our early triaging technique?

% of Highly-Impactful Bugs





What is the percentage of highly-impactful bugs?

What are the characteristics of highly-impactful bugs?

Could we predict highly-impactful bugs?

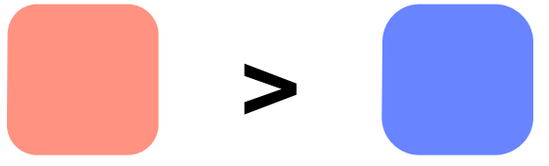
What are the benefits of applying our early triaging technique?

Highly-impactful vs. Other bugs

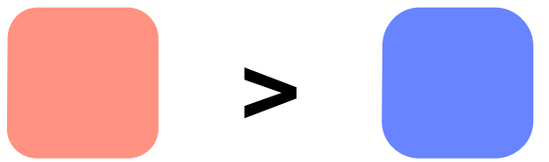
Highly-impactful bugs

Other bugs

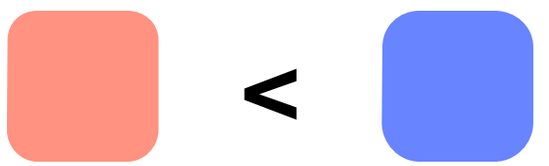
reporter experience



comment length



closed %



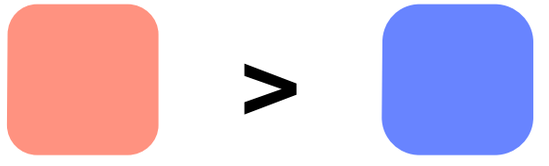
Highly-impactful vs. Other bugs

Highly-impactful bugs

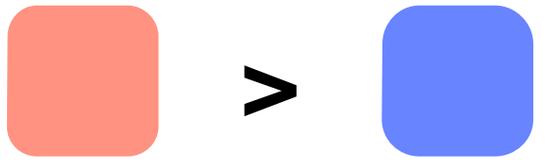
Other bugs

Mozilla quality assurance teams do not prioritise highly-impactful bugs, though they impact a large user base.

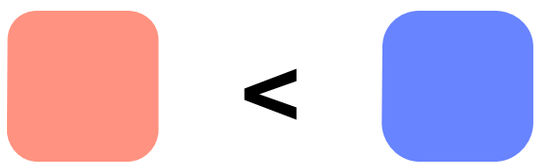
reporter experience



comment length



closed %



Identifying Highly-Impactful Bugs with Different Thresholds of Frequency & Entropy

50 percentile

70 percentile

90 percentile

Identifying Highly-Impactful Bugs with Different Thresholds of Frequency & Entropy

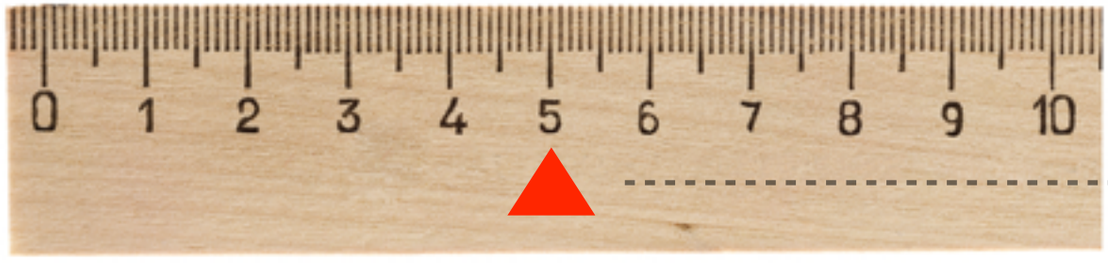
50 percentile

V.S.

70 percentile

90 percentile

Identifying Highly-Impactful Bugs with Different Thresholds of Frequency & Entropy



50 percentile

V.S.

70 percentile

90 percentile

- Only trivial changes could be observed from the results of 70/90 percentile.
- Very few highly-impactful bugs are identified if we increase the thresholds.
- Different thresholds will not affect our conclusion.

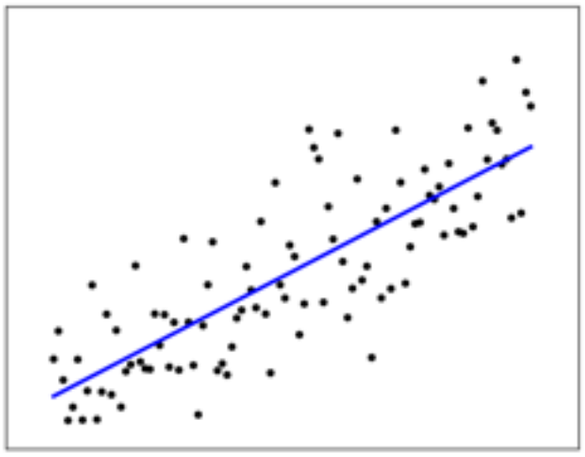
What is the percentage of highly-impactful bugs?

What are the characteristics of highly-impactful bugs?

Could we predict highly-impactful bugs?

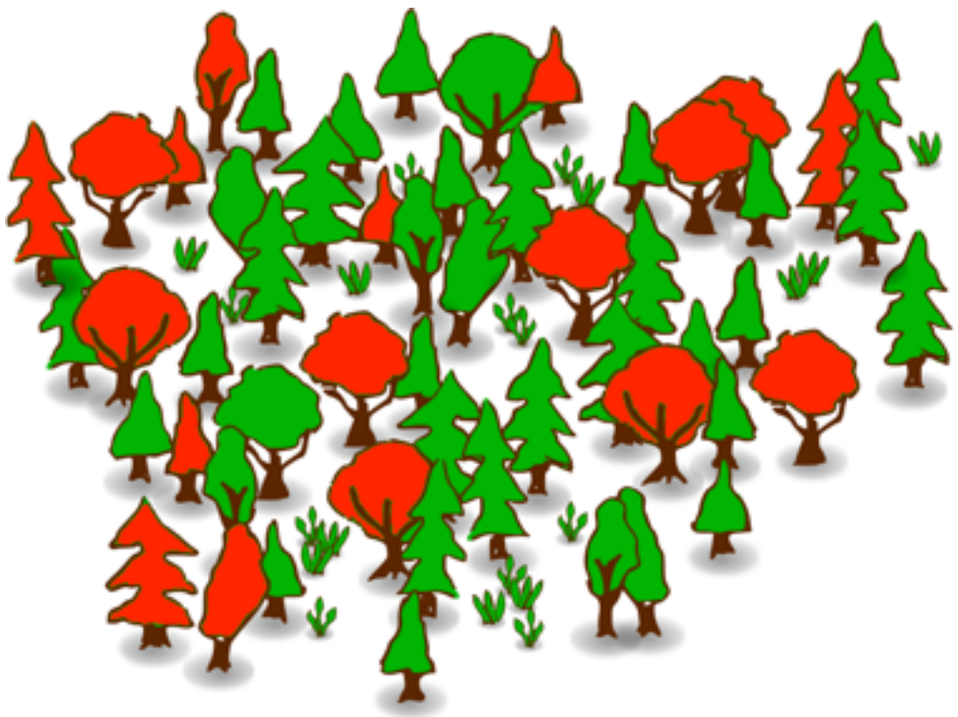
What are the benefits of applying our early triaging technique?

Predictive Algorithms



General Linear Model

C5.0 **ctree (party)**

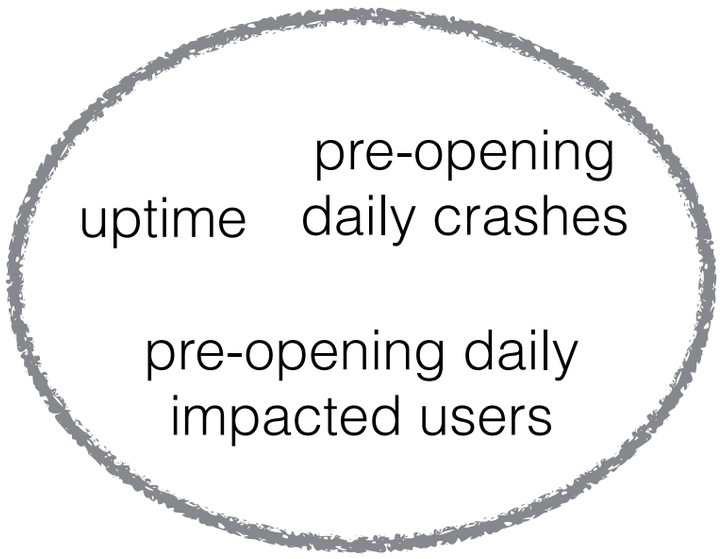


randomForest **cforest (party)**

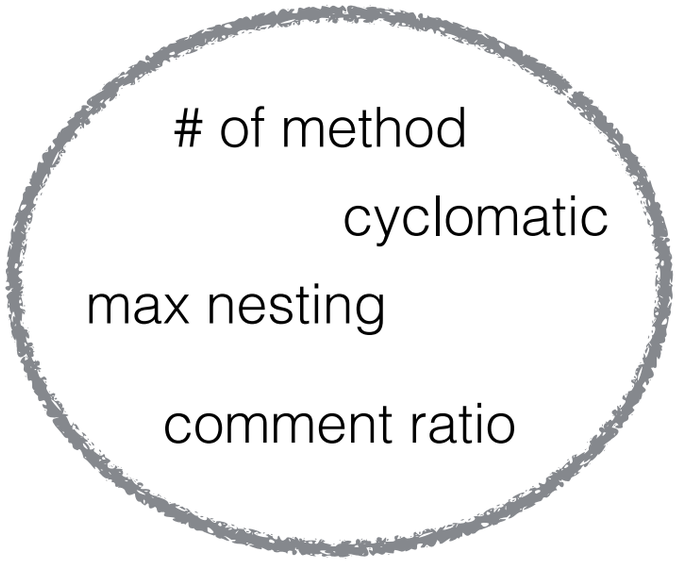
Predictive Metrics and Results

Bug report

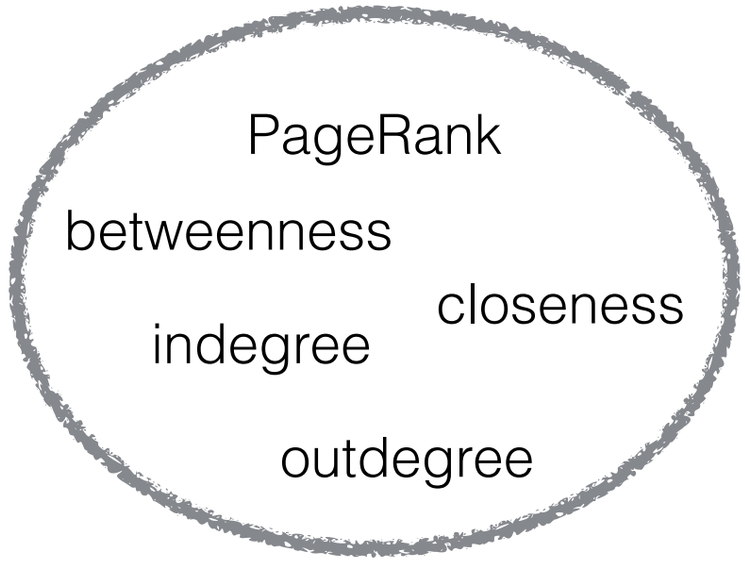
Crash report



Code complexity



Social network analysis



Predictive Metrics and Results

Bug report

day in month

hour

weekday

component

M

reporter experience

Crash report

uptime

pre-opening daily crashes

pre-opening daily impacted users

cforest is the best model in our case study

Code complexity

# of method

cyclomatic

max nesting

comment ratio

Social network analysis

PageRank

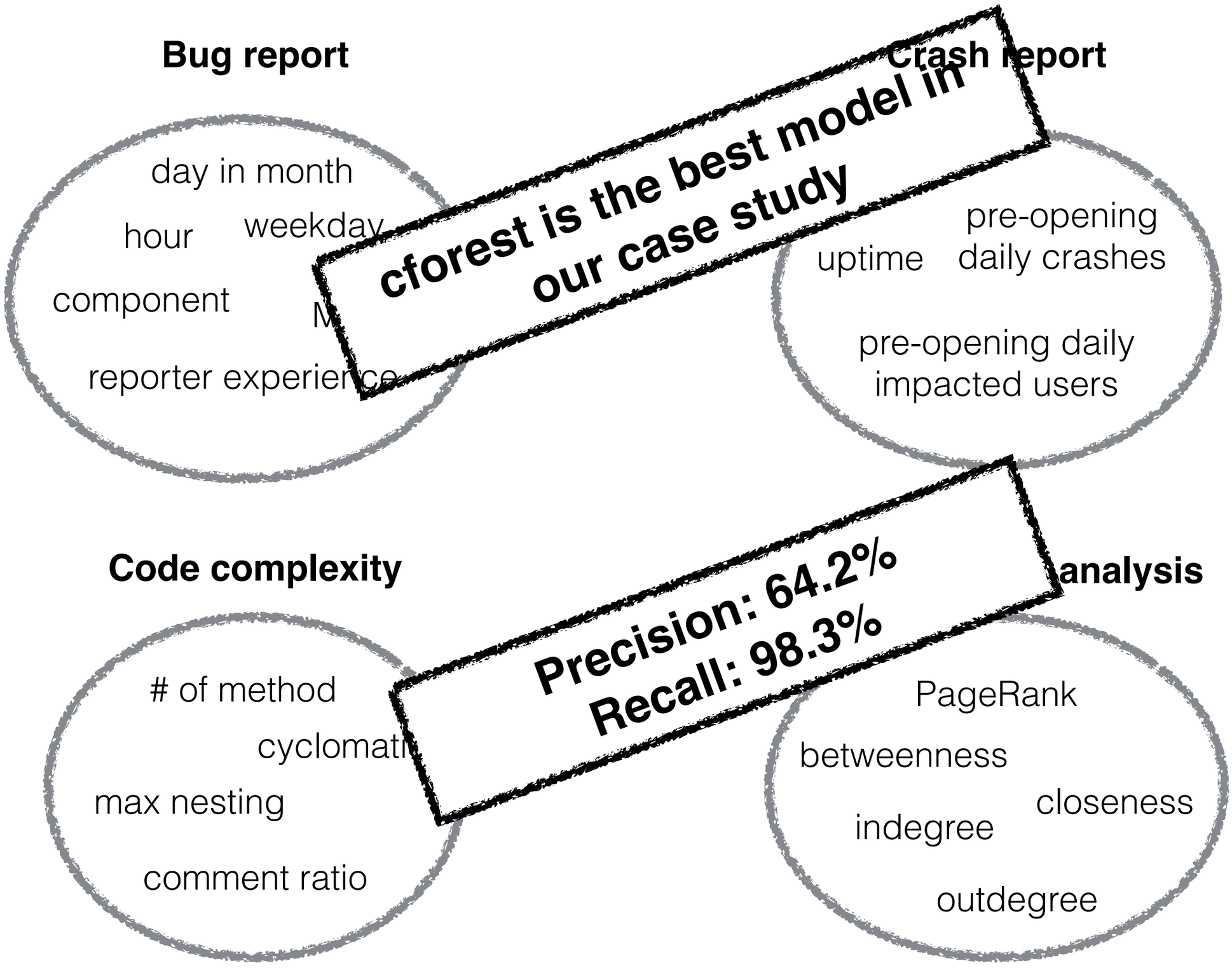
betweenness

closeness

indegree

outdegree

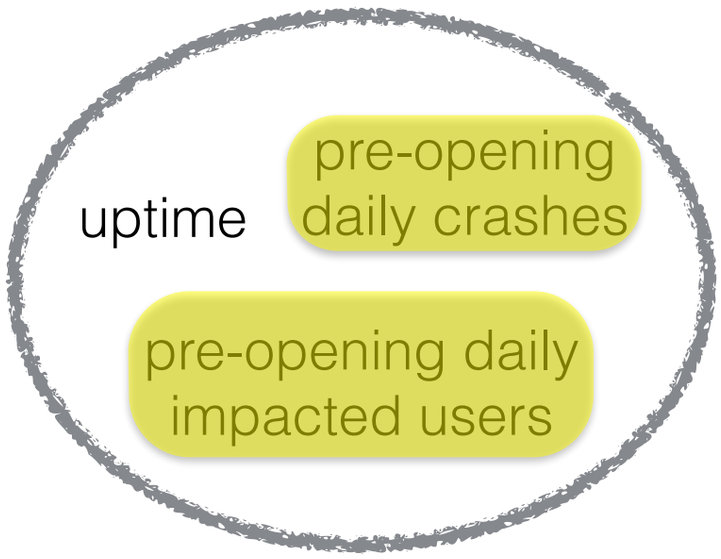
Predictive Metrics and Results



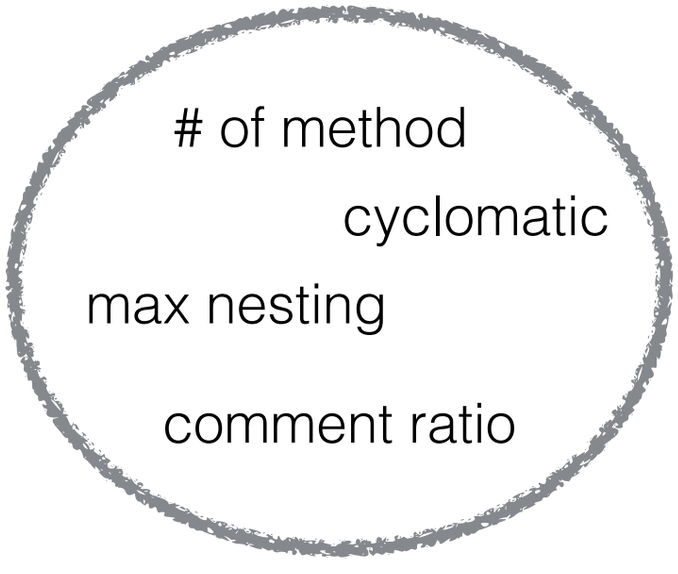
Predictive Metrics and Results

Bug report

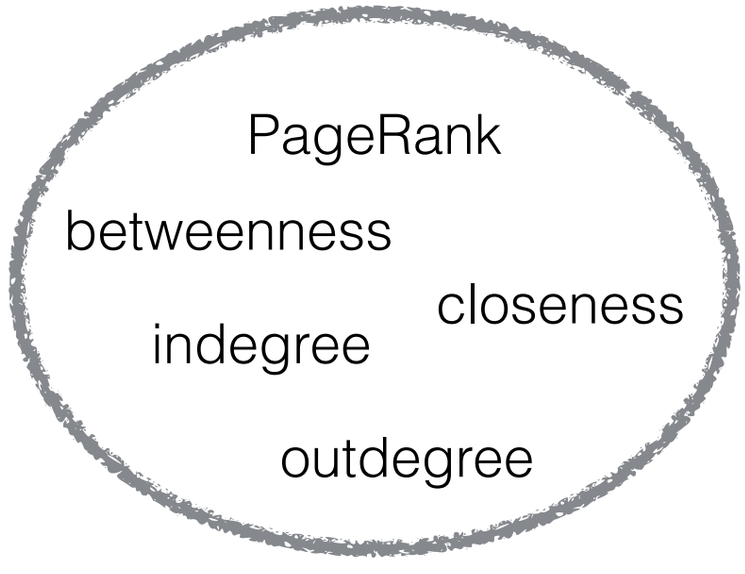
Crash report



Code complexity



Social network analysis



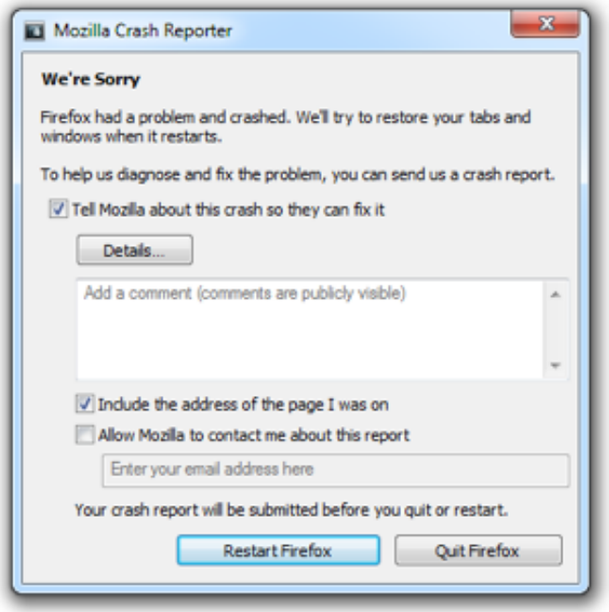
What is the percentage of highly-impactful bugs?

What are the characteristics of highly-impactful bugs?

Could we predict highly-impactful bugs?

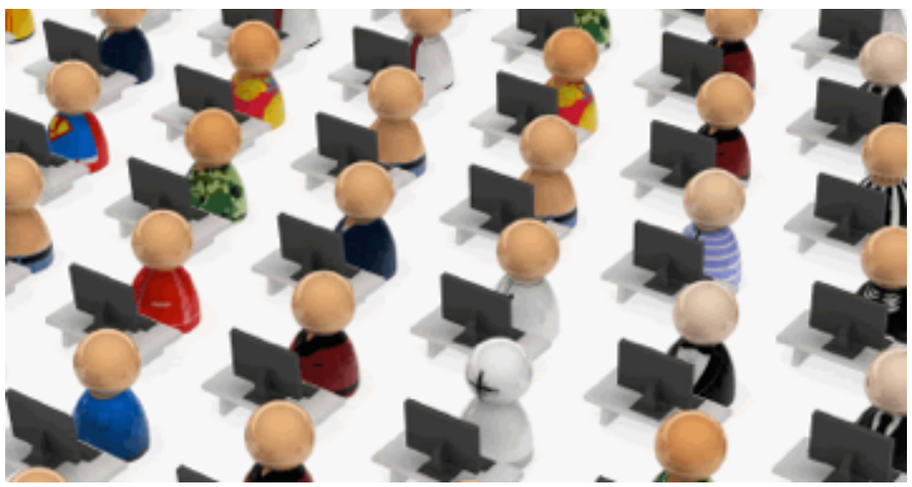
What are the benefits of applying our early triaging technique?

Benefit Analysis



crash occurrences

Firefox	Fennec
23%	13.4%



unique impacted users

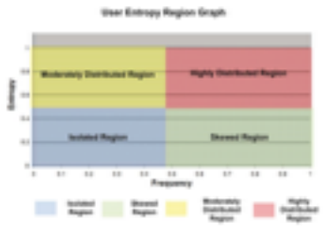
Firefox	Fennec
28.6%	49.4%

Conclusion

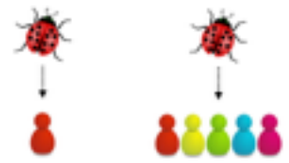
Crash Collecting System

Challenge

Should we triage crash-related bugs only by their crashing frequency?



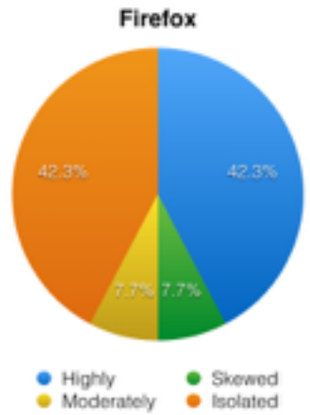
Khomh et al. (2011)

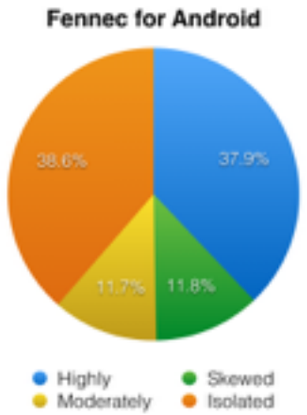


highly-impactful bugs = bugs with high crashing frequency + entropy

When the region graph is built, large number of end users have been impacted from crashes for a long time

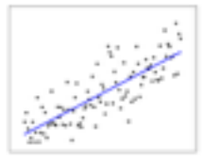
% of highly-impactful bugs





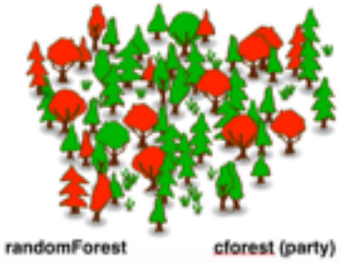
Highly-impactful vs. Other bugs

Predictive algorithms



General Linear Model

C5.0 ctree (party)

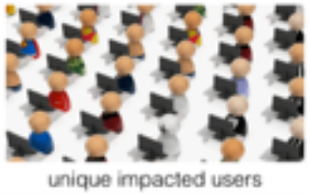


randomForest cforest (party)

Benefit Analysis

crash occurrences

Firefox	Fennec
23%	13.4%



unique impacted users

Firefox	Fennec
28.6%	49.4%