

An Empirical Study of Highly-Impactful Bugs in Mozilla Projects

Le An, Foutse Khomh
Polytechnique Montréal, Canada

4 August, Vancouver, Canada



POLYTECHNIQUE
MONTRÉAL

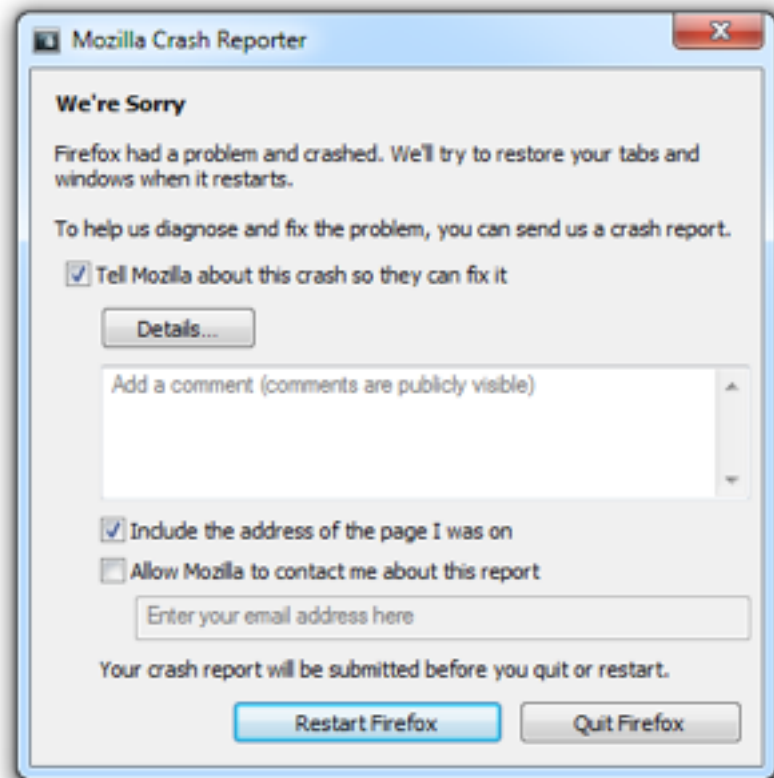
WORLD-CLASS
ENGINEERING



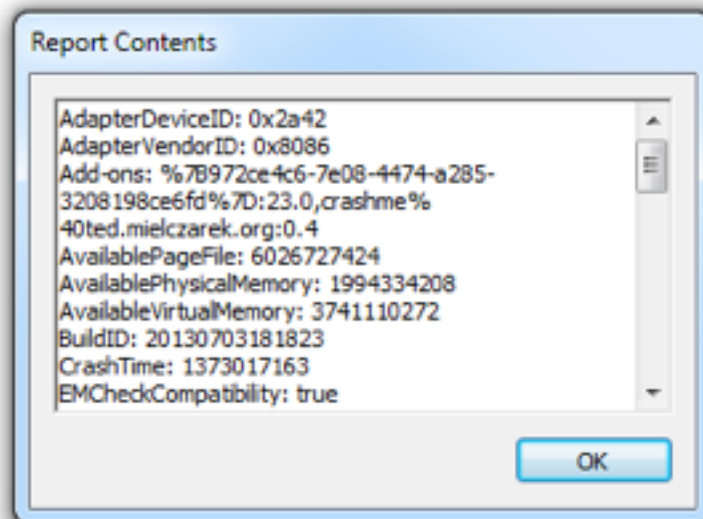
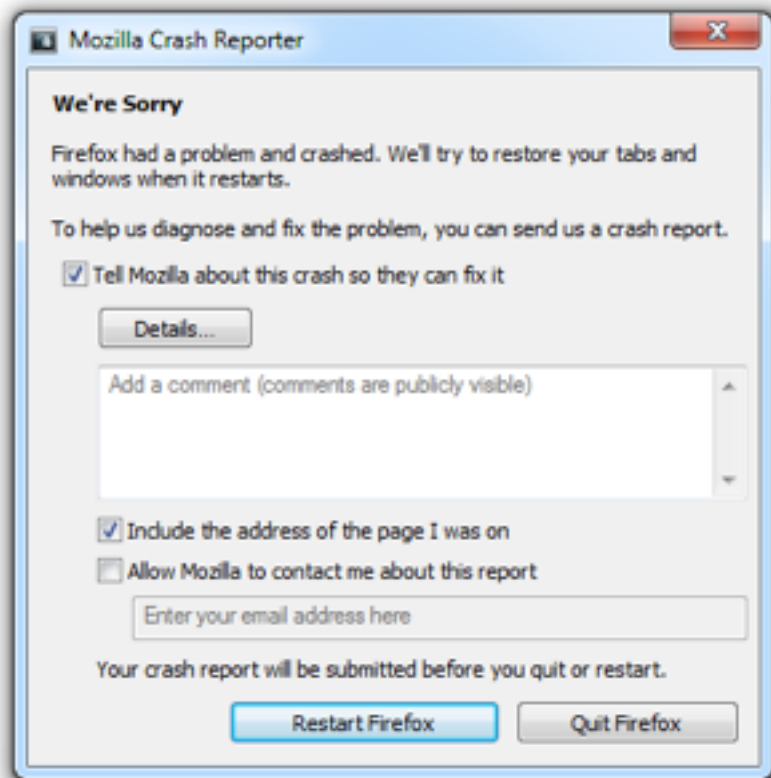
Outline

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

Crash Collecting System

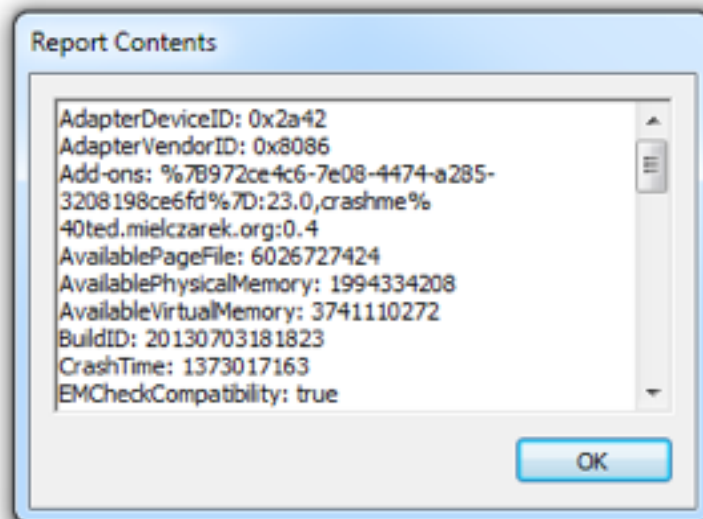
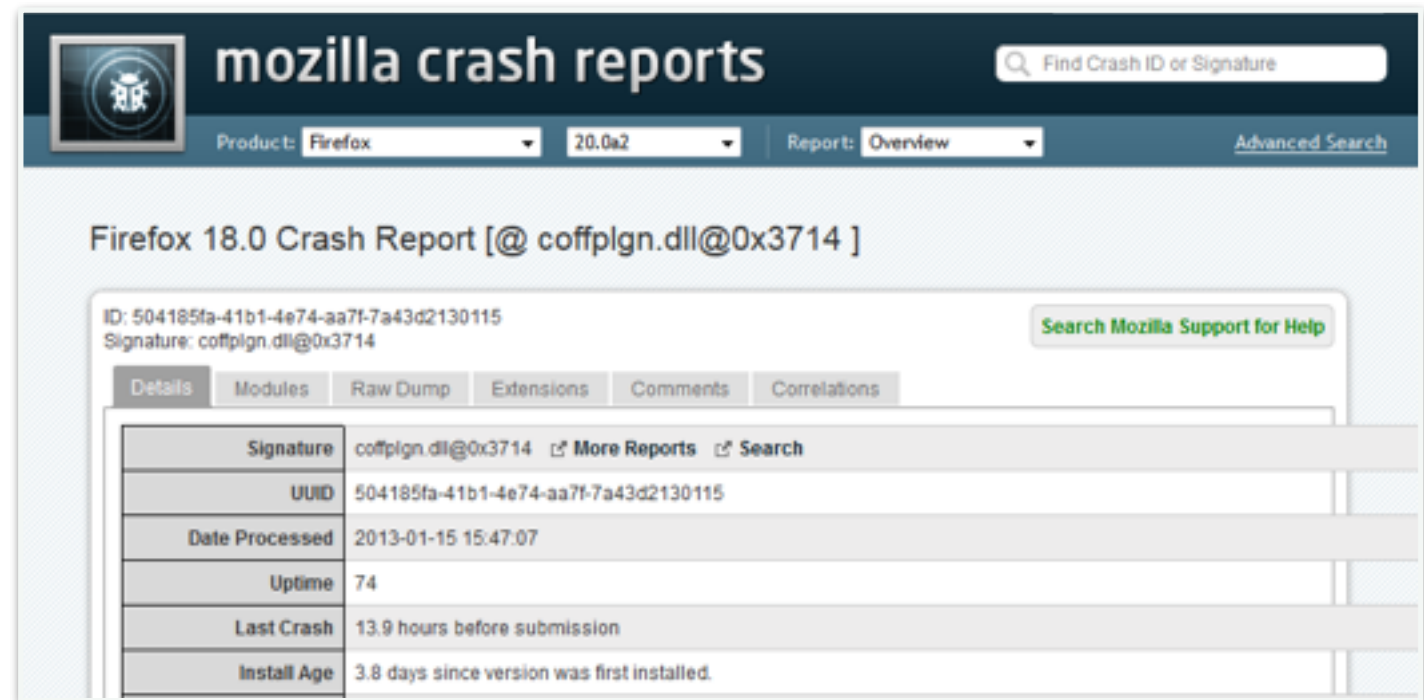
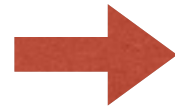
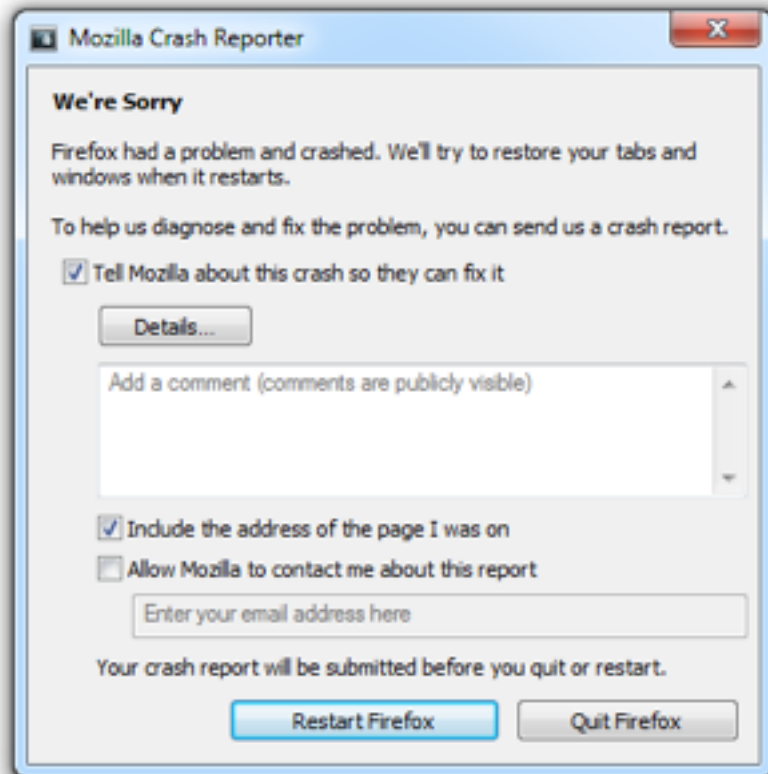


Crash Collecting System



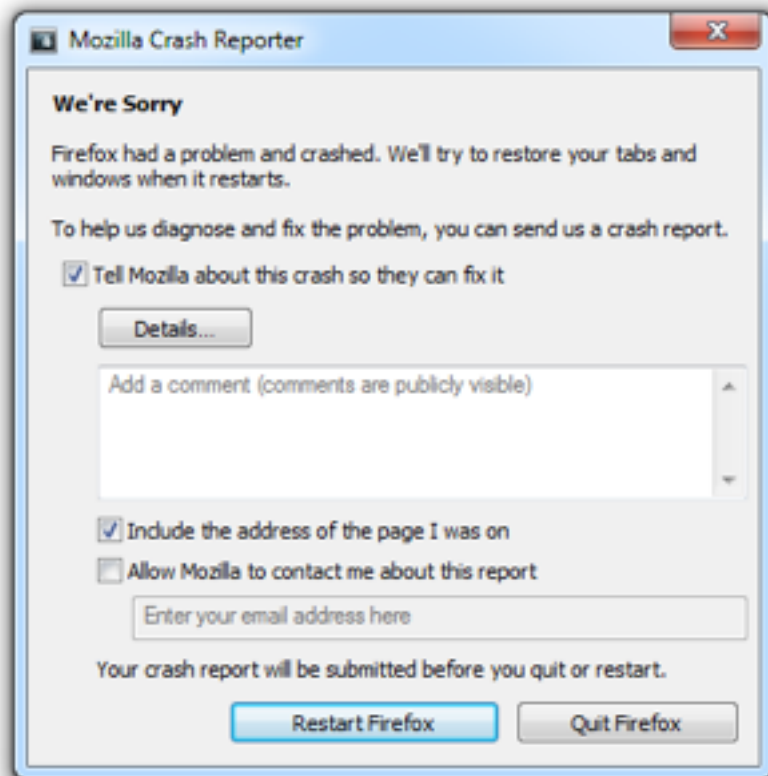
Crash Collecting System

Socorro



Crash Collecting System

Socorro



Mozilla Crash Reporter

We're Sorry

Firefox had a problem and crashed. We'll try to restore your tabs and windows when it restarts.

To help us diagnose and fix the problem, you can send us a crash report.

☒ Tell Mozilla about this crash so they can fix it

[Details...](#)

Add a comment (comments are publicly visible)

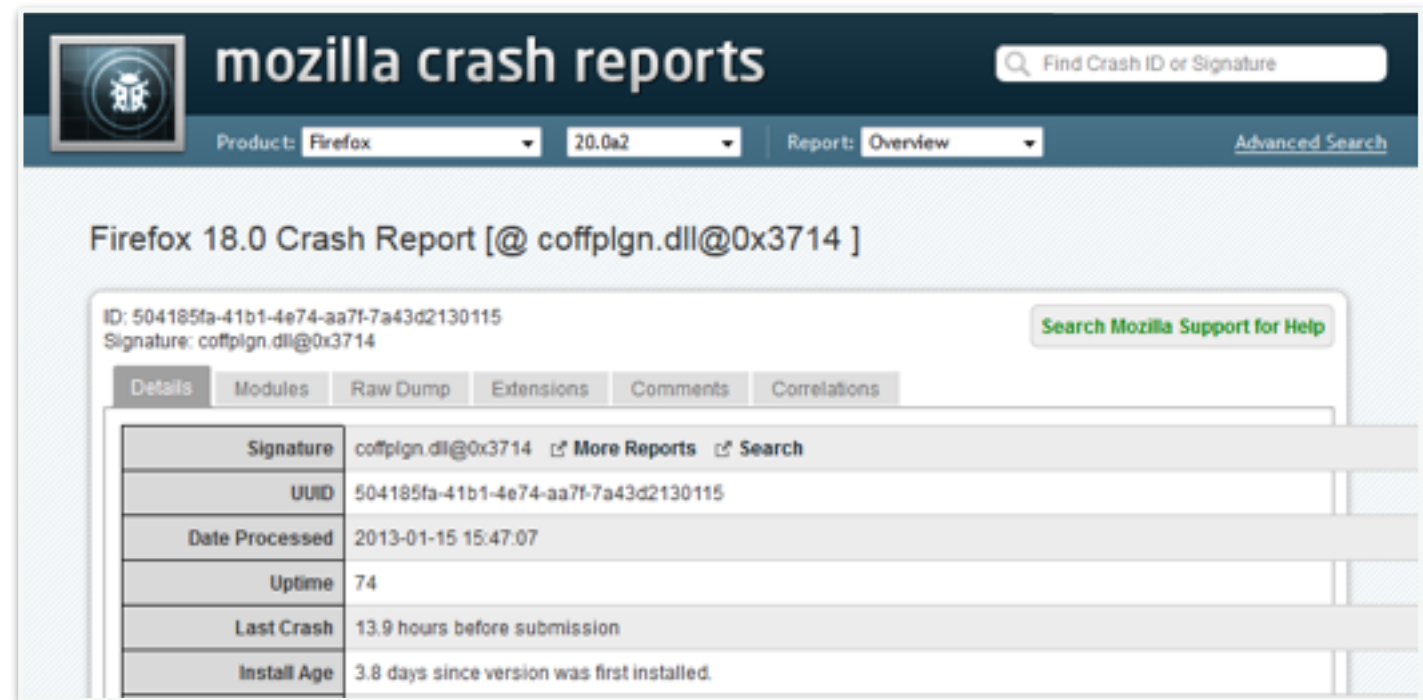
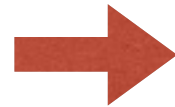
☒ Include the address of the page I was on

☐ Allow Mozilla to contact me about this report

Enter your email address here

Your crash report will be submitted before you quit or restart.

[Restart Firefox](#) [Quit Firefox](#)



mozilla crash reports

Find Crash ID or Signature

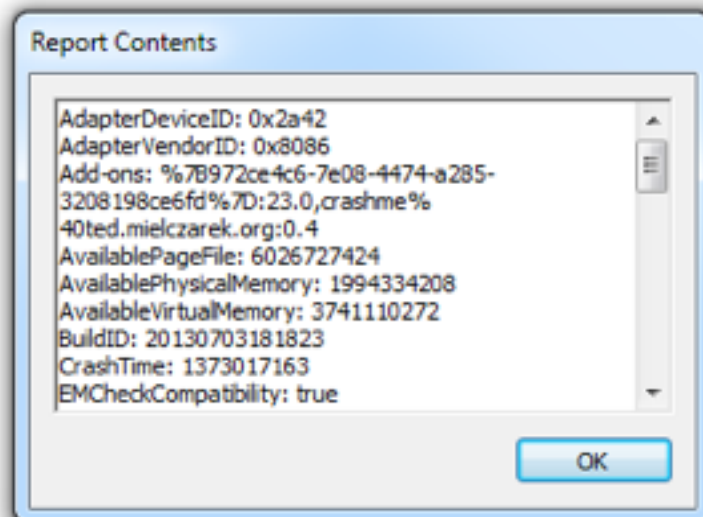
Product: **Firefox** Version: **20.0a2** Report: **Overview** [Advanced Search](#)

Firefox 18.0 Crash Report [@ coffplgn.dll@0x3714]

ID: 504185fa-41b1-4e74-aa7f-7a43d2130115
Signature: coffplgn.dll@0x3714 [Search Mozilla Support for Help](#)

[Details](#) [Modules](#) [Raw Dump](#) [Extensions](#) [Comments](#) [Correlations](#)

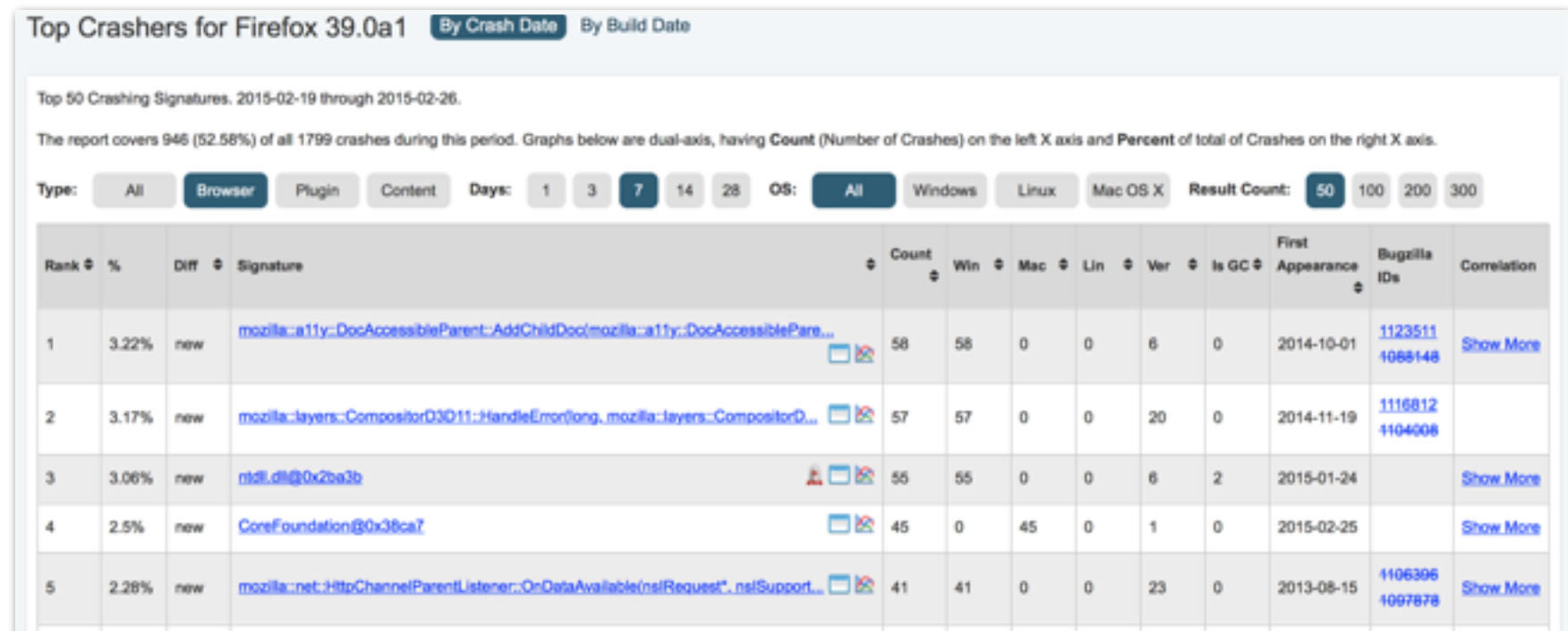
Signature	coffplgn.dll@0x3714 More Reports Search
UUID	504185fa-41b1-4e74-aa7f-7a43d2130115
Date Processed	2013-01-15 15:47:07
Uptime	74
Last Crash	13.9 hours before submission
Install Age	3.8 days since version was first installed.



Report Contents

AdapterDeviceID: 0x2a42
AdapterVendorID: 0x8086
Add-ons: %7B972ce4c6-7e08-4474-a285-3208198ce6fd%7D:23.0,crashme%40ted.mielczarek.org:0.4
AvailablePageFile: 6026727424
AvailablePhysicalMemory: 1994334208
AvailableVirtualMemory: 3741110272
BuildID: 20130703181823
CrashTime: 1373017163
EMCheckCompatibility: true

[OK](#)



Top Crashers for Firefox 39.0a1 [By Crash Date](#) [By Build Date](#)

Top 50 Crashing Signatures, 2015-02-19 through 2015-02-26.

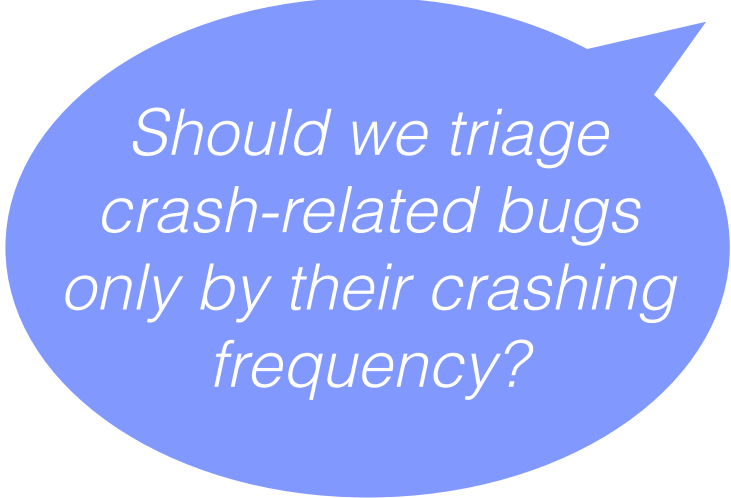
The report covers 946 (52.58%) of all 1799 crashes during this period. Graphs below are dual-axis, having **Count** (Number of Crashes) on the left X axis and **Percent** of total of Crashes on the right X axis.

Type: [All](#) [Browser](#) [Plugin](#) [Content](#) Days: [1](#) [3](#) [7](#) [14](#) [28](#) OS: [All](#) [Windows](#) [Linux](#) [Mac OS X](#) Result Count: [50](#) [100](#) [200](#) [300](#)

Rank	%	Diff	Signature	Count	Win	Mac	Lin	Ver	Is GC	First Appearance	Bugzilla IDs	Correlation
1	3.22%	new	mozilla::a11y::DocAccessibleParent::AddChildDoc(mozilla::a11y::DocAccessibleParent...	58	58	0	0	6	0	2014-10-01	1123511 1088148	Show More
2	3.17%	new	mozilla::layers::CompositorD3D11::HandleError(mozilla::layers::CompositorD...	57	57	0	0	20	0	2014-11-19	1116812 1104908	
3	3.06%	new	ntdll.dll@0x2ba3b	55	55	0	0	6	2	2015-01-24		Show More
4	2.5%	new	CoreFoundation@0x38ca7	45	0	45	0	1	0	2015-02-25		Show More
5	2.28%	new	mozilla::net::HttpChannelParentListener::OnDataAvailable(nsIRequest*, nsISupport...	41	41	0	0	23	0	2013-08-15	1106396 1097878	Show More

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?*



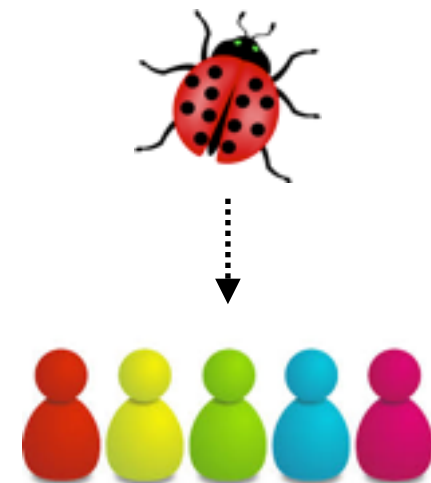
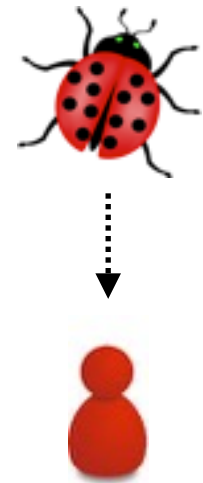
Challenge

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

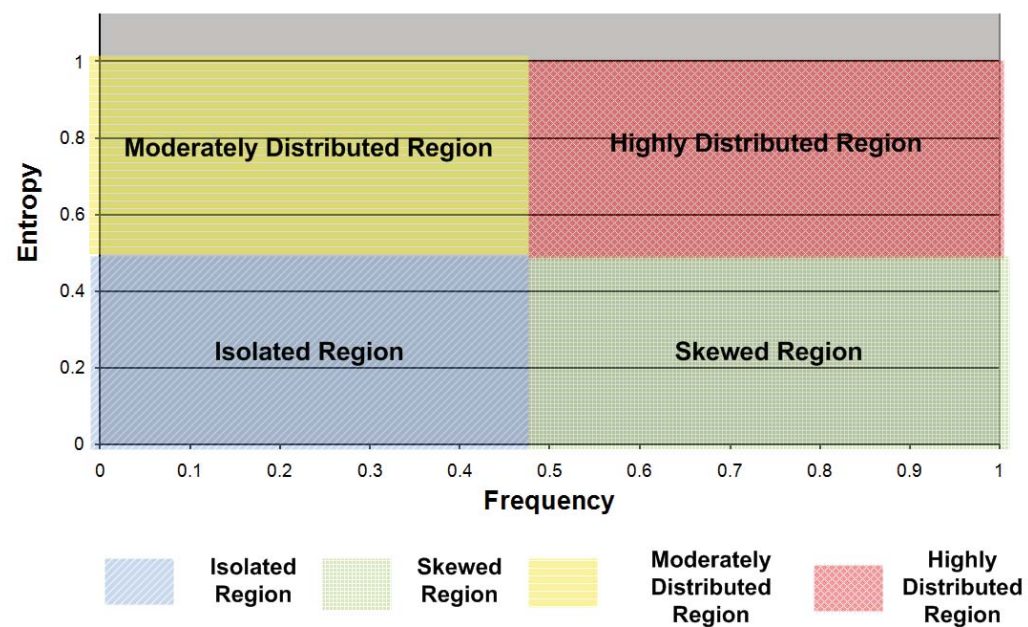


Challenge

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



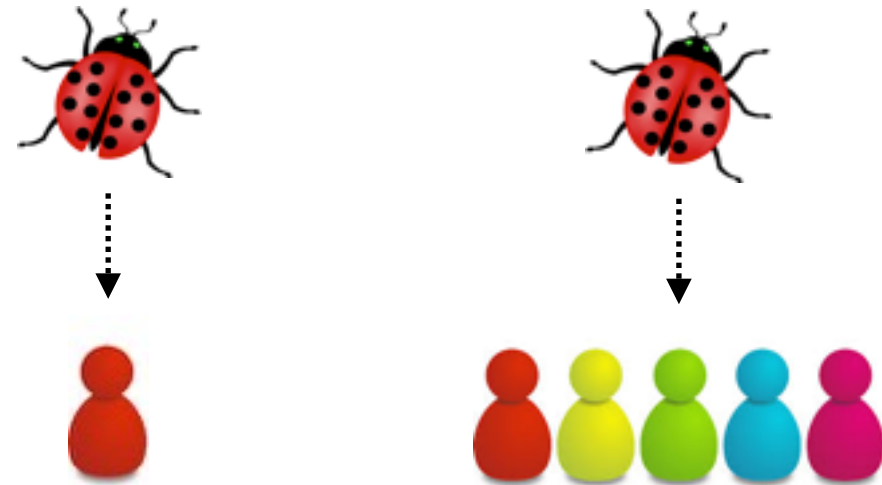
User Entropy Region Graph



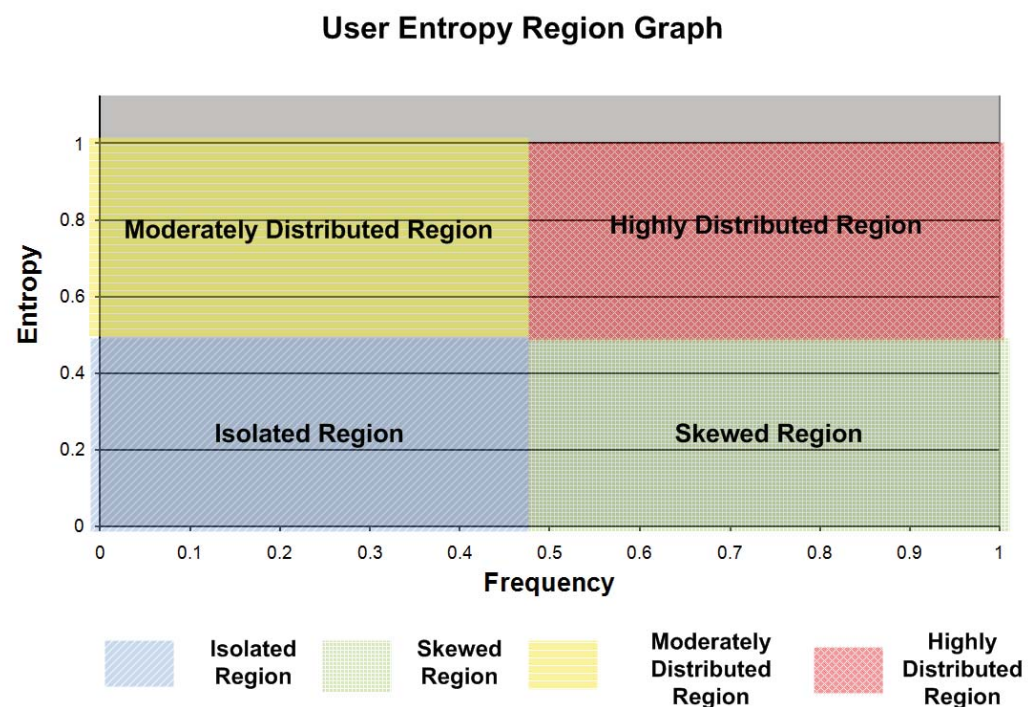
Khomh et al. (2011)

Challenge

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

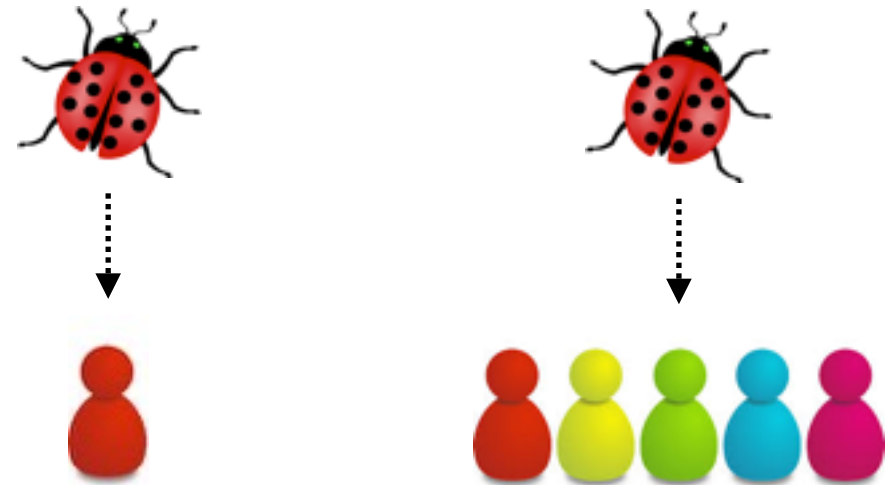


highly-impactful bugs = bugs affecting a large number of users with high crashing frequency

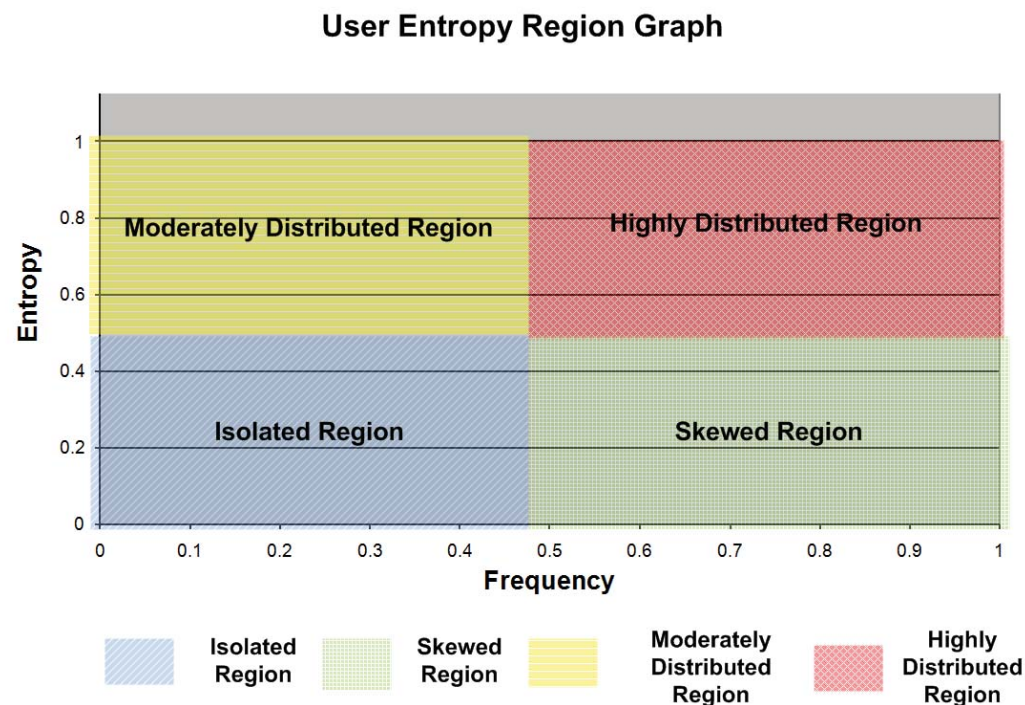


Challenge

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

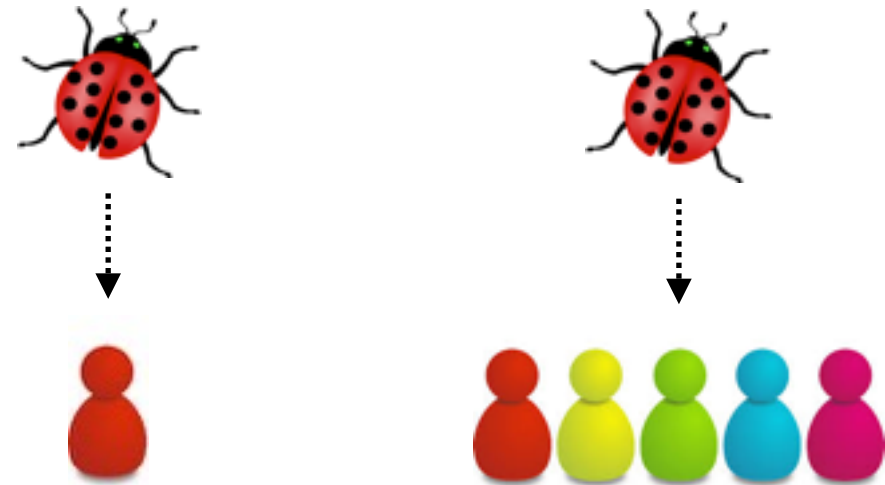


highly-impactful bugs = bugs affecting a large number of users (*entropy*) with high crashing frequency (*occurrences*)

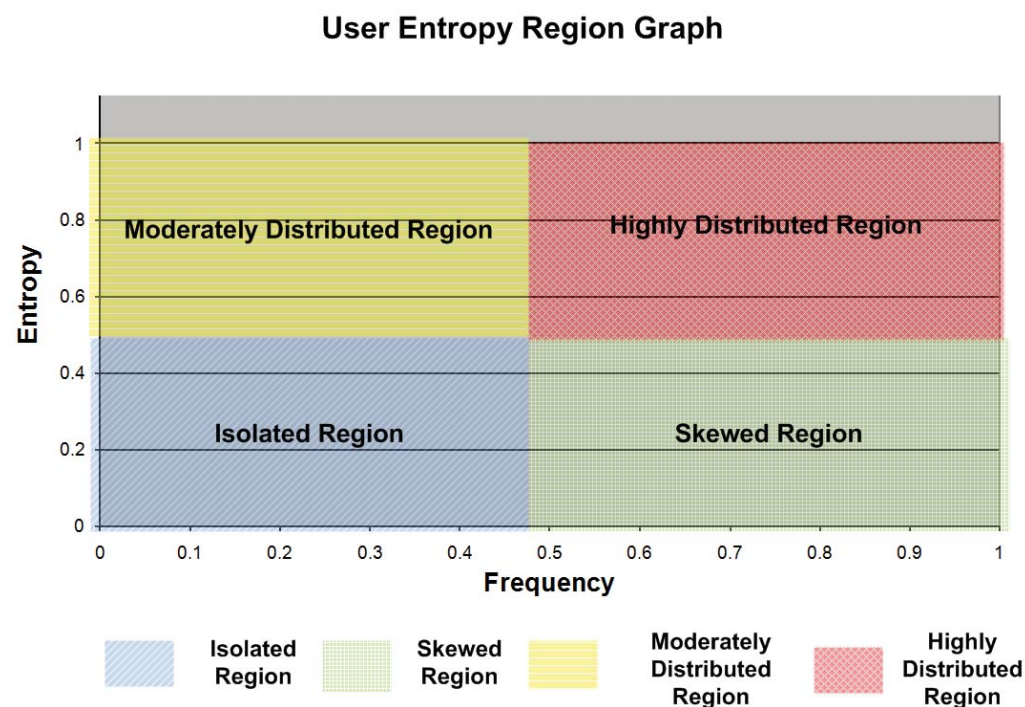


Challenge

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*)



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

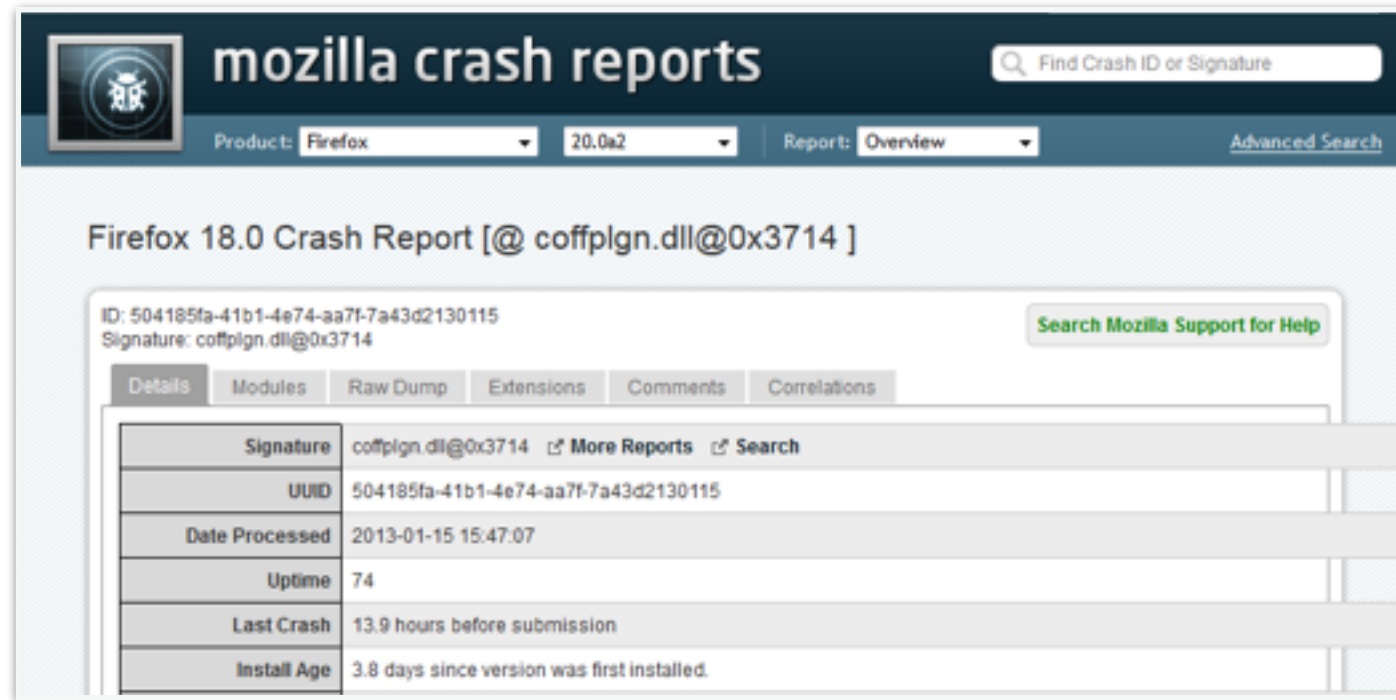


Mozilla Firefox

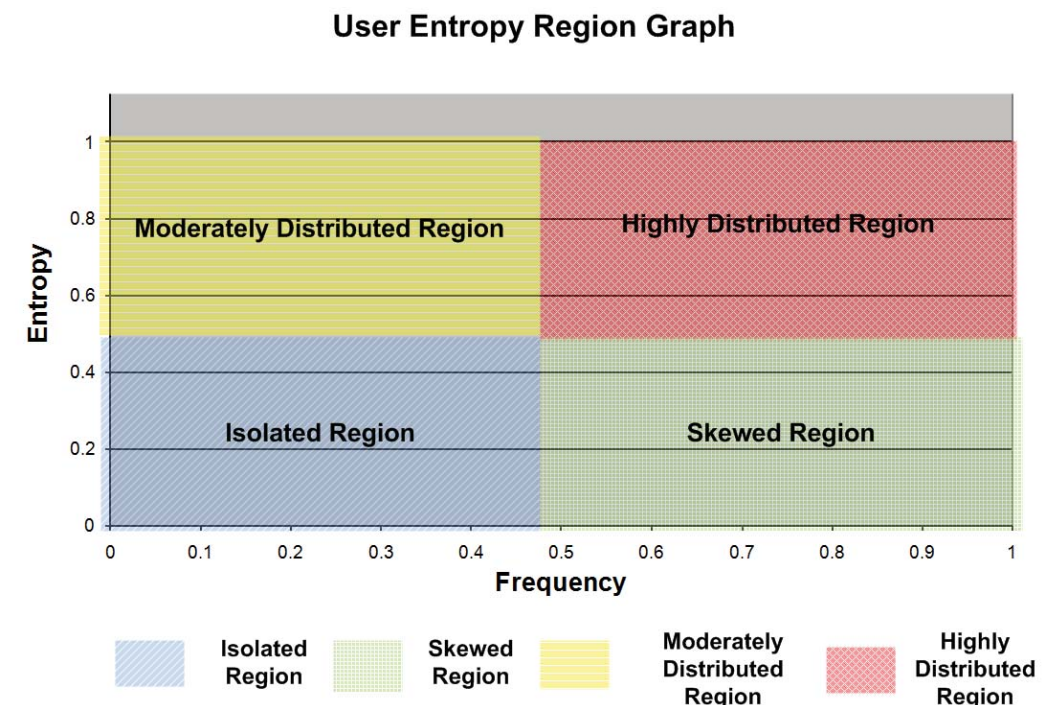


Mozilla Fennec
(for Android)

Identification of Highly-impactful Bugs

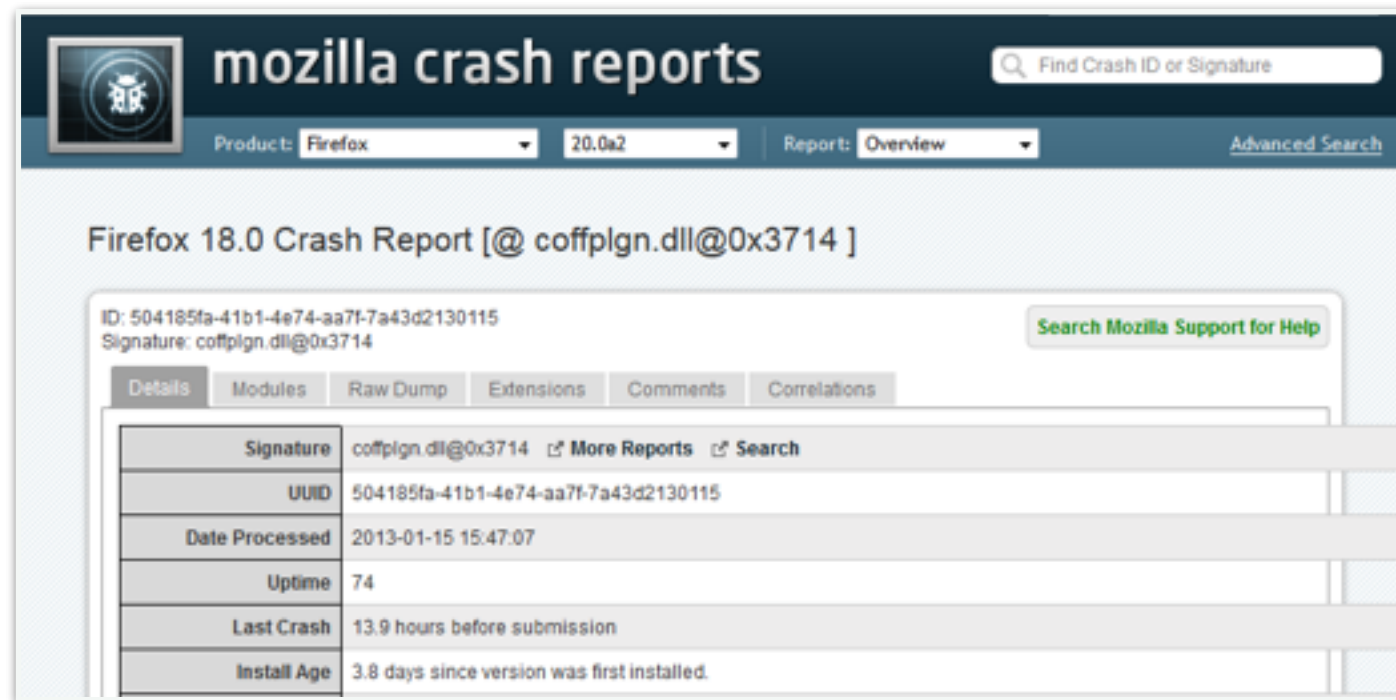


$$H_n(b) = - \sum_{i=1}^n p_i \times \log_n(p_i)$$



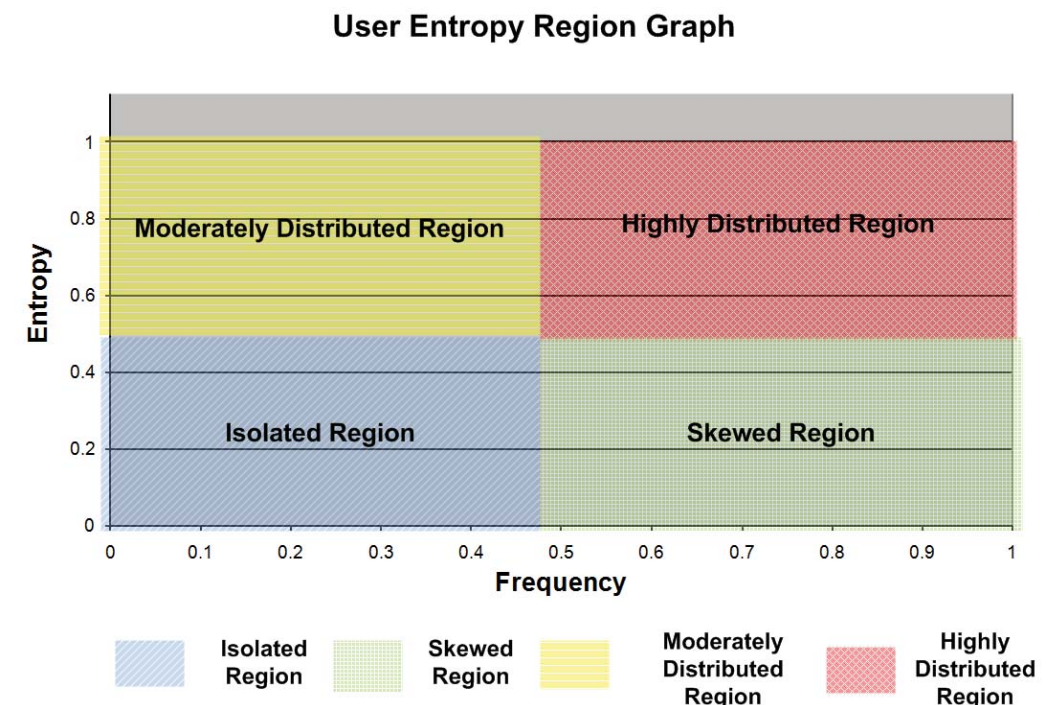
Khomh et al. (2011)

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?



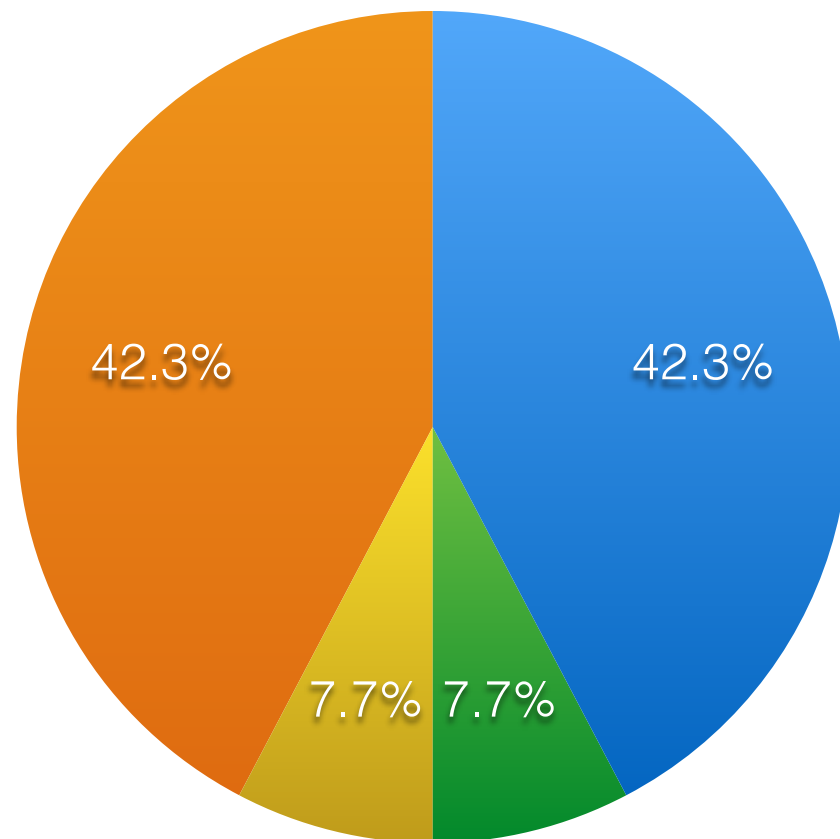
Could we predict highly-impactful bugs?



What are the benefits of applying our early triaging technique?

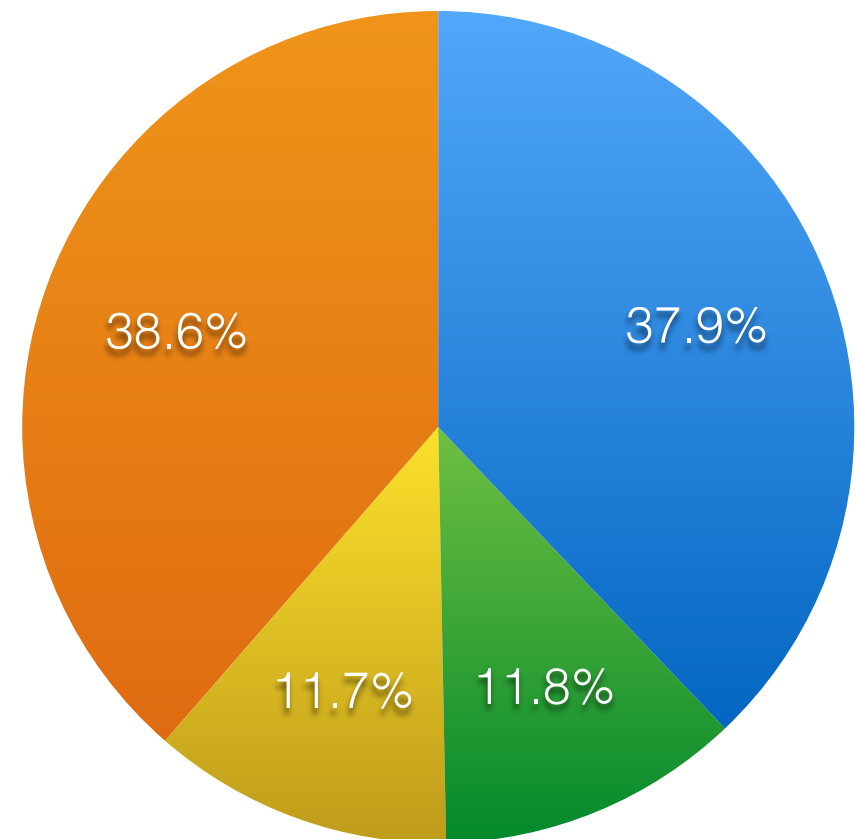
% of Highly-Impactful Bugs

Firefox



● Highly
● Moderately
● Skewed
● Isolated

Fennec for Android



● Highly
● Moderately
● Skewed
● Isolated



What is the percentage of highly-impactful bugs?



What are the characteristics of highly-impactful bugs?



Could we predict highly-impactful bugs?



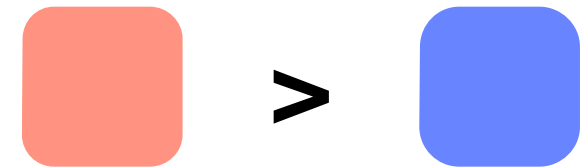
What are the benefits of applying our early triaging technique?

Highly-impactful vs. Other bugs

Highly-impactful
bugs



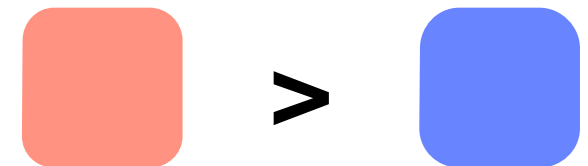
reporter experience



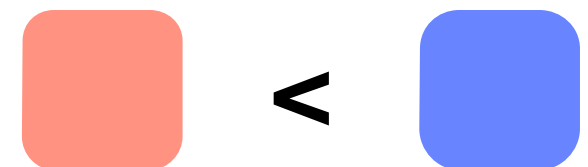
Other bugs



comment length



closed %



Highly-impactful vs. Other bugs

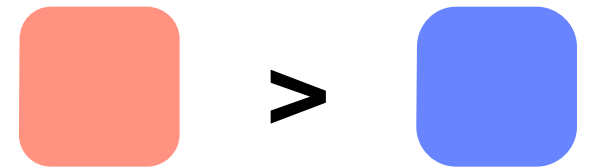
Highly-impactful
bugs

Other bugs

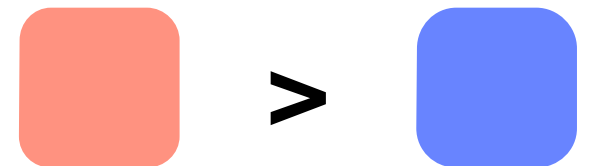
Mozilla quality assurance teams do not prioritise highly-impactful bugs, though they impact a large user base.



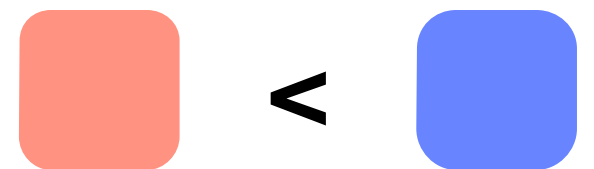
reporter experience



comment length



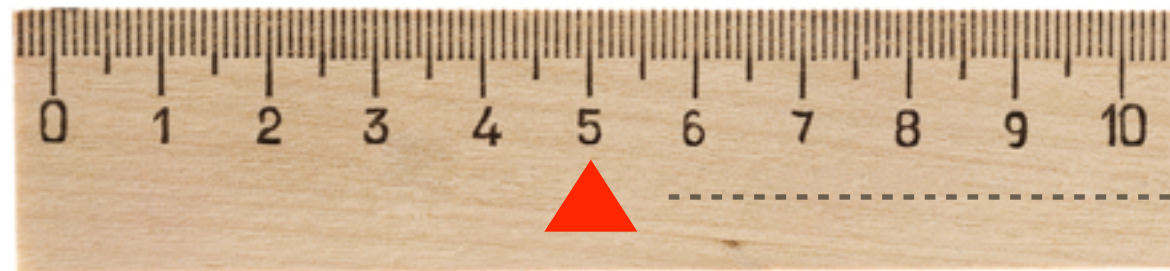
closed %



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



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



50 percentile

V.S.

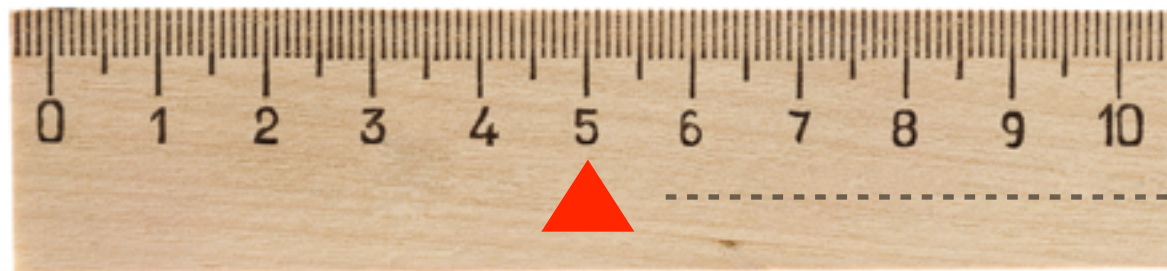


70 percentile



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?

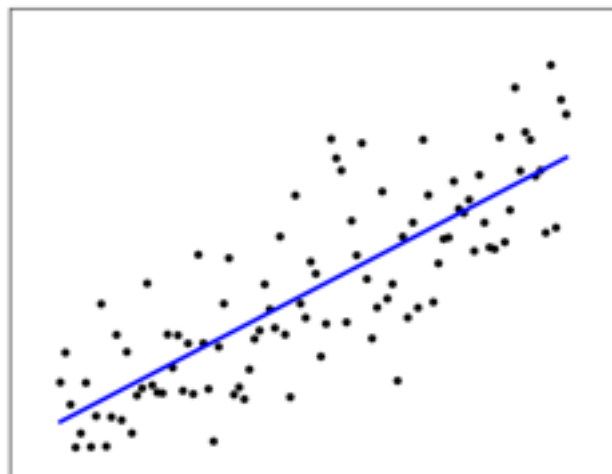


Could we predict highly-impactful bugs?



What are the benefits of applying our early triaging technique?

Predictive Algorithms

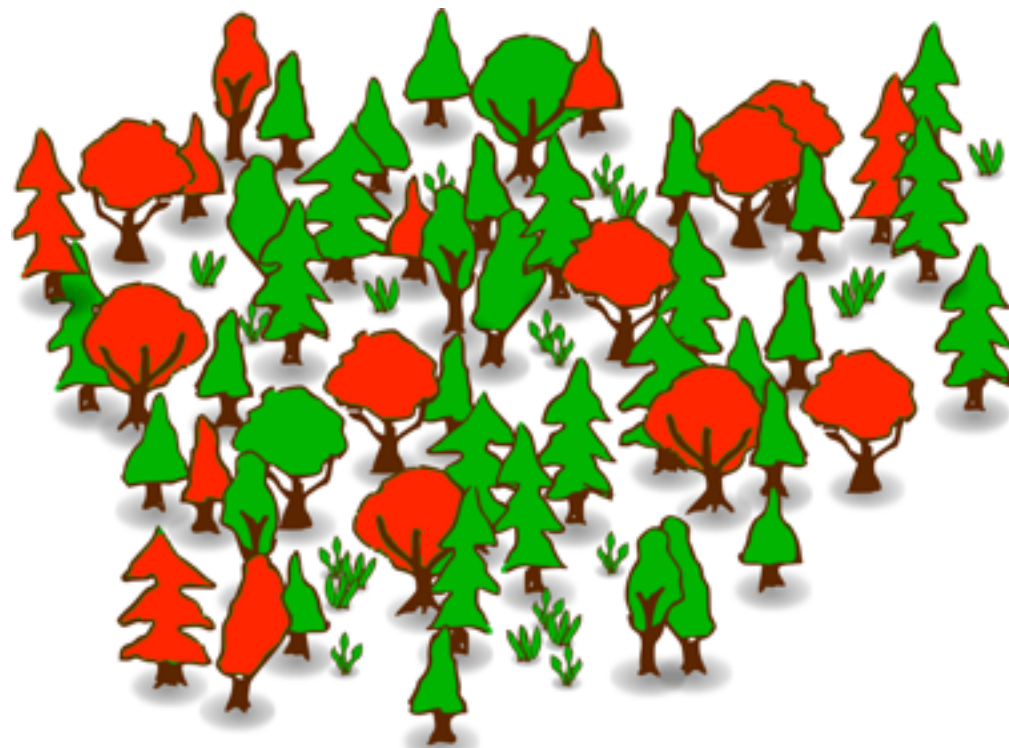


General Linear Model



C5.0

ctree (party)



randomForest

cforest (party)

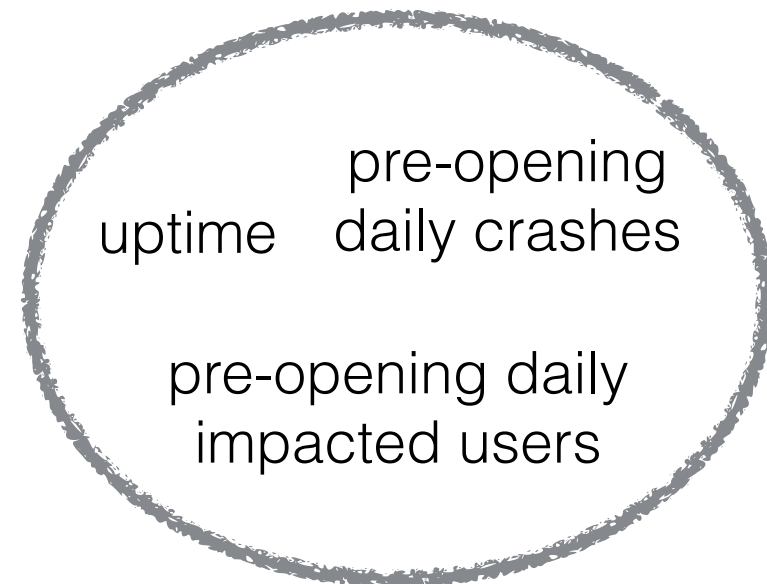


Predictive Metrics and Results

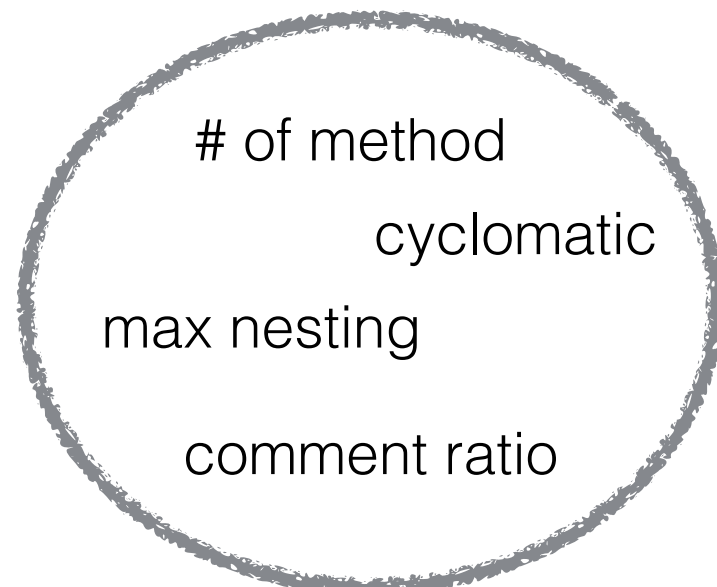
Bug report



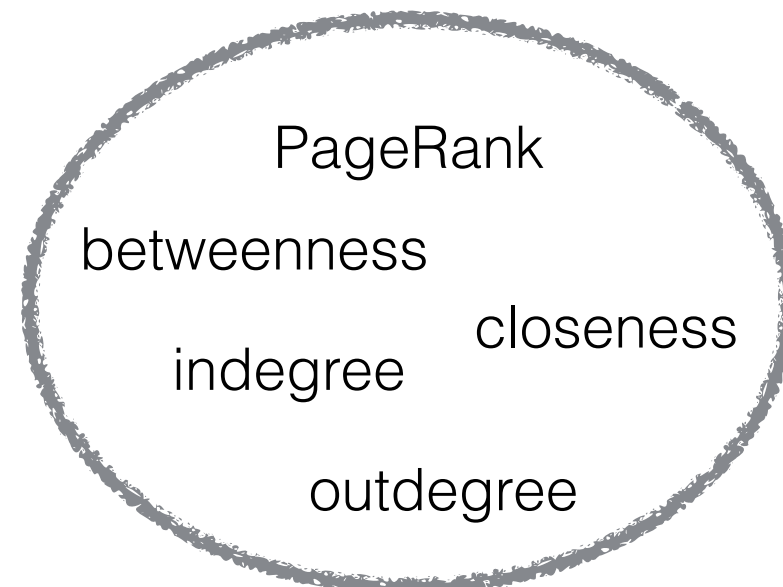
Crash report



Code complexity



Social network analysis



Predictive Metrics and Results

Bug report

day in month
hour weekday
component M
reporter experience

Crash report

pre-opening
uptime 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

Bug report

day in month
hour weekday
component M
reporter experience

**cforest is the best model in
our case study**

Crash report

pre-opening
uptime daily crashes
pre-opening daily
impacted users

Code complexity

of method
cyclomatic
max nesting
comment ratio

**Precision: 64.2%
Recall: 98.3%**

analysis

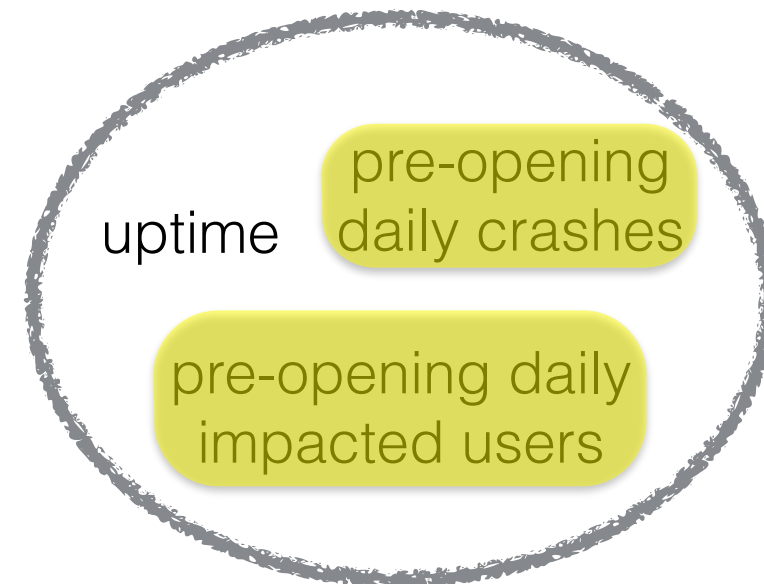
PageRank
betweenness
closeness
indegree
outdegree

Predictive Metrics and Results

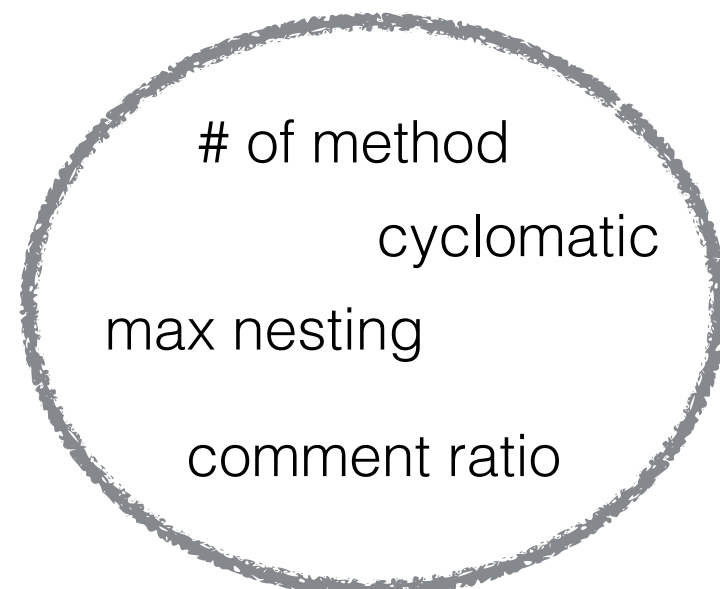
Bug report



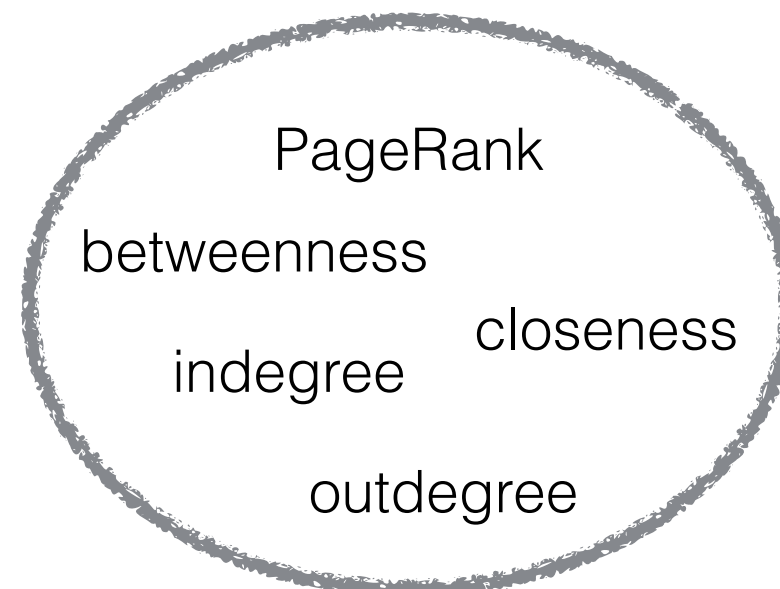
Crash report



Code complexity



Social network analysis





What is the percentage of highly-impactful bugs?



What are the characteristics of highly-impactful bugs?

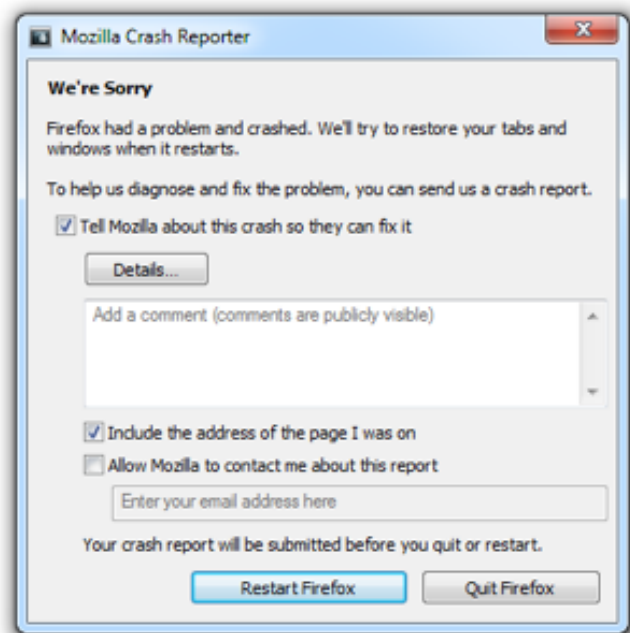


Could we predict highly-impactful bugs?



What are the benefits of applying our early triaging technique?

Benefit Analysis



crash occurrences



Firefox

23%

Fennec

13.4%



unique impacted users



Firefox

28.6%

Fennec

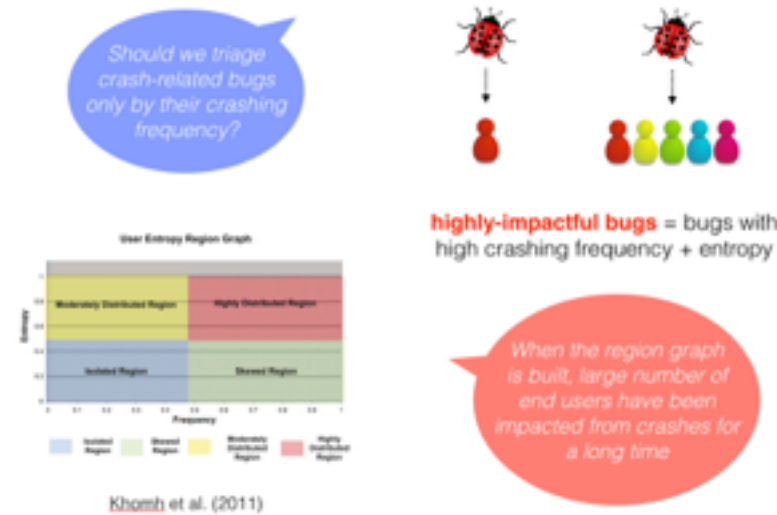
49.4%

Conclusion

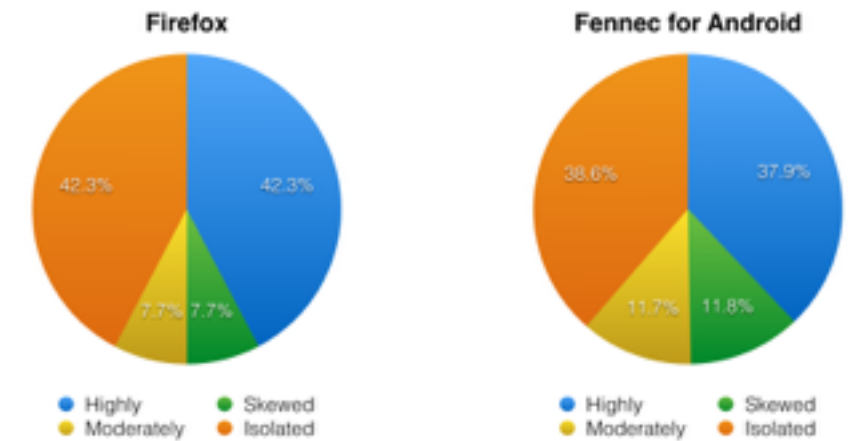
Crash Collecting System



Challenge



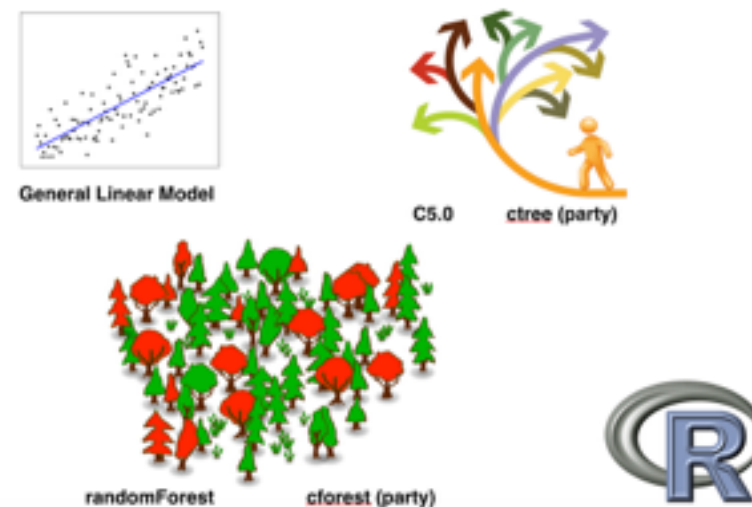
% of highly-impactful bugs



Highly-impactful vs. Other bugs



Predictive algorithms



Benefit Analysis

