A Look at the Practice of Software Logging

Wahab Hamou-Lhadj

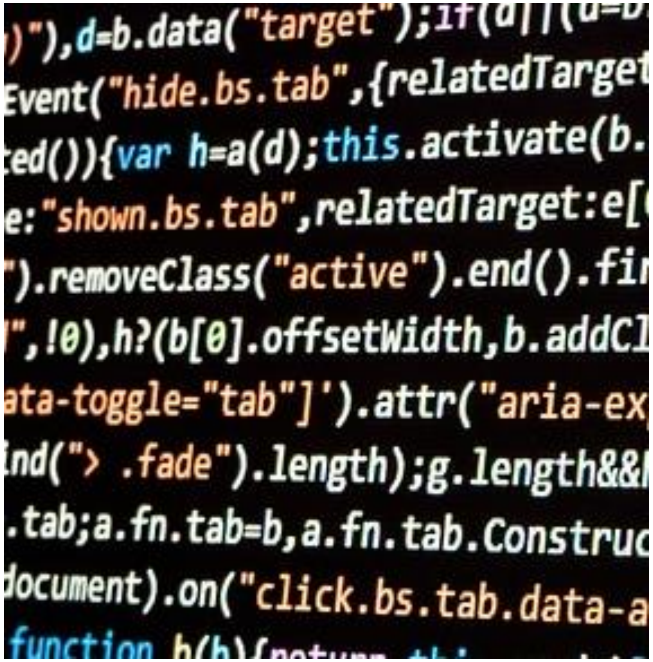
Concordia University
Montréal, QC, Canada

wahab.hamou-lhadj@concordia.ca

IDEAR@ISSRE’18, Memphis, TN
October 15, 2018

Software Logging

- A practice used by developers for debugging, performance analysis, compliance and certification, etc.
- Various types of logs: user-defined, execution traces, profiling data, etc.
- Collectively known as **machine data**
- Contrasted with user data, which focus on user behavioral patterns



```
)"),d=b.data("target");if(d||(d=b
Event("hide.bs.tab",{relatedTarget
ted()){var h=a(d);this.activate(b.
e:"shown.bs.tab",relatedTarget:e[
").removeClass("active").end().fi
",!0),h?(b[0].offsetWidth,b.addCl
ata-toggle="tab"]').attr("aria-ex
ind("> .fade").length);g.length&&
.tab;a.fn.tab=b,a.fn.tab.Construc
document).on("click.bs.tab.data-a
function b(b){return thi
```

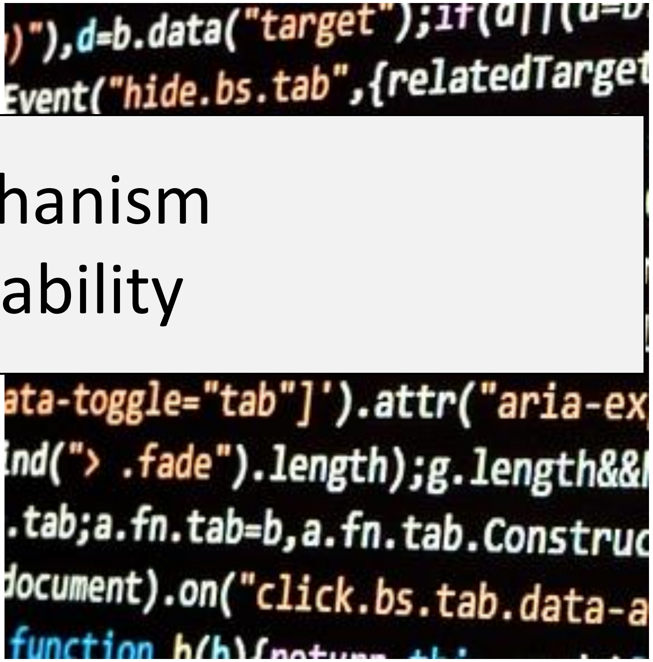
Software Logging

- A practice used by developers for debugging, anomaly detection,

  profiling data, etc.

Logging is a key mechanism
for software observability

- Collectively known as **machine data**
- Contrasted with user data, which focus on user behavioral patterns



```
)"),d=b.data("target");if(d||(d=b
Event("hide.bs.tab",{relatedTarget
ata-toggle="tab"]').attr("aria-ex
ind("> .fade").length);g.length&&
.tab;a.fn.tab=b,a.fn.tab.Construc
document).on("click.bs.tab.data-a
function b(b){return
```

Why is it important? and why now?

- Software ecosystem complexity
- Increase in software-enabled machines
- Device connectivity and advances in IoT
- Emergence of machine intelligence paradigms

Why is it important? and why now?

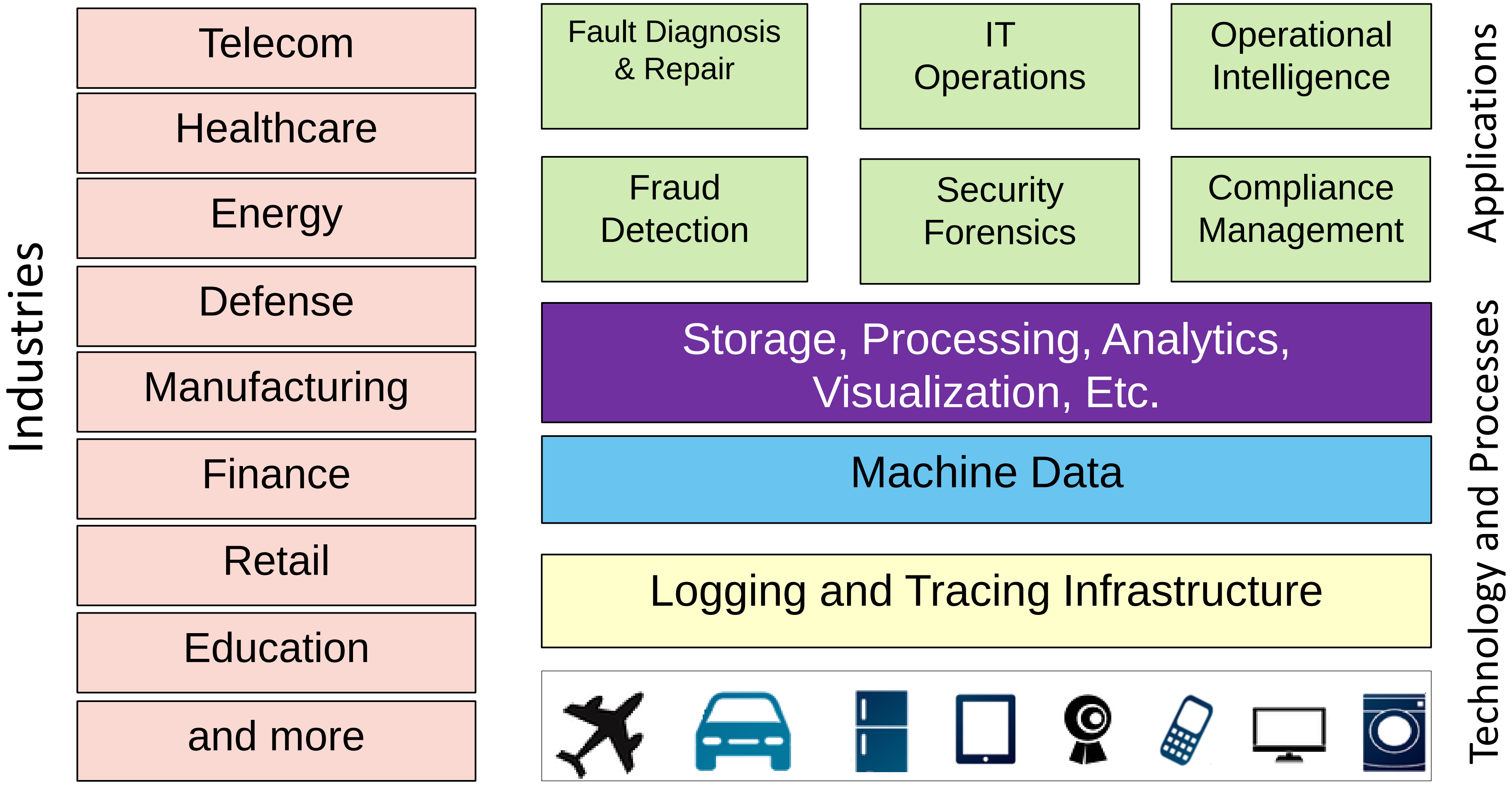
- Astronomical increase in software-enabled machines

complexity

- Emergence of machine intelligence paradigms

As our dependence on machines increases, the need to monitor and observe machines in operation becomes critical.

Listening to machines: The big picture



Industries

- Telecom
- Healthcare
- Energy
- Defense
- Manufacturing
- Finance
- Retail
- Education
- and more

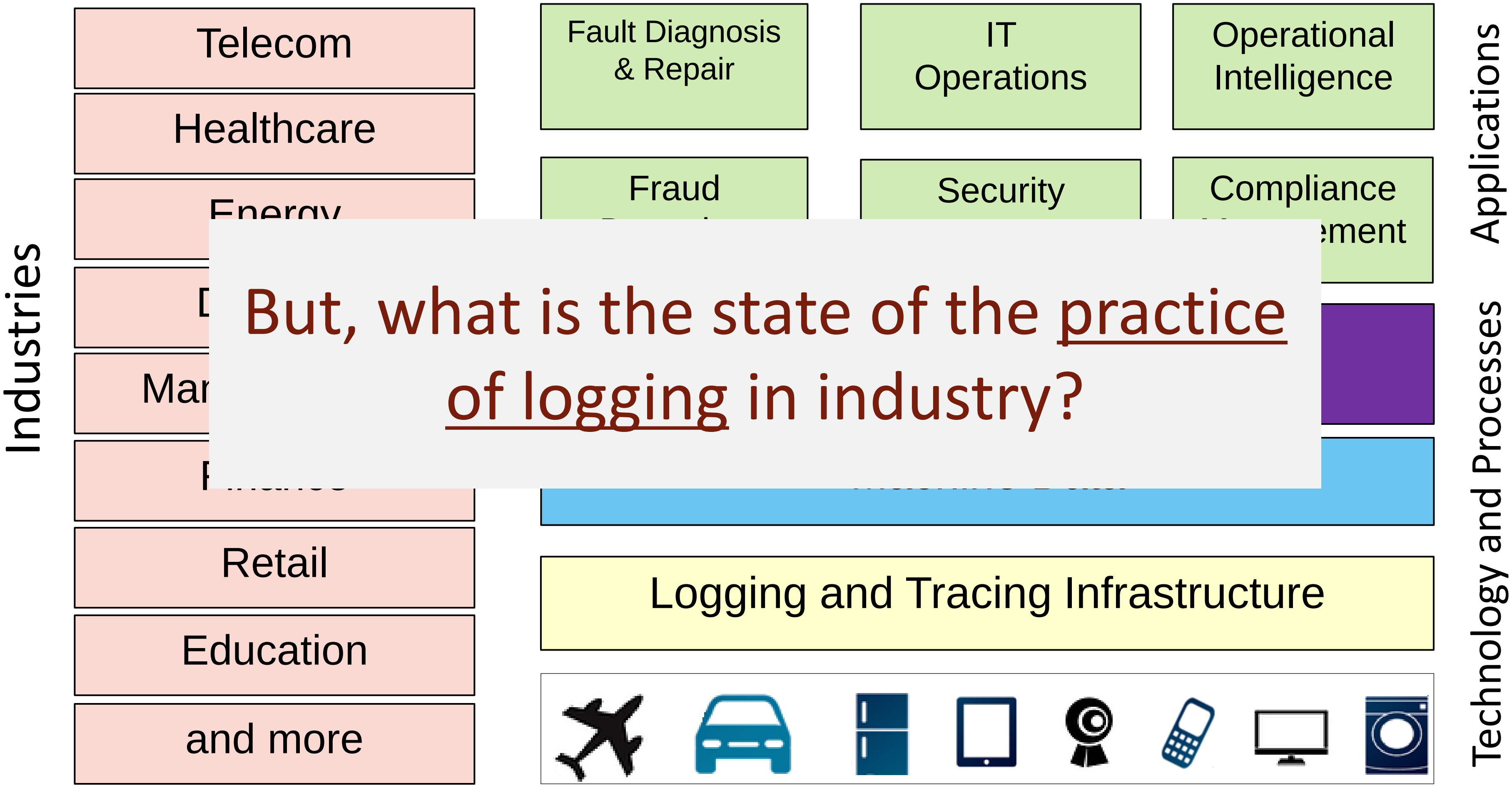
Applications

- Fault Diagnosis & Repair
- IT Operations
- Operational Intelligence
- Fraud Detection
- Security Forensics
- Compliance Management

Technology and Processes

- Storage, Processing, Analytics, Visualization, Etc.
- Machine Data
- Logging and Tracing Infrastructure

Listening to machines: The big picture



Industries

Telecom

Healthcare

Energy

D

Mar

F

Retail

Education

and more

Fault Diagnosis & Repair

IT Operations

Operational Intelligence

Fraud

Security

Compliance ement

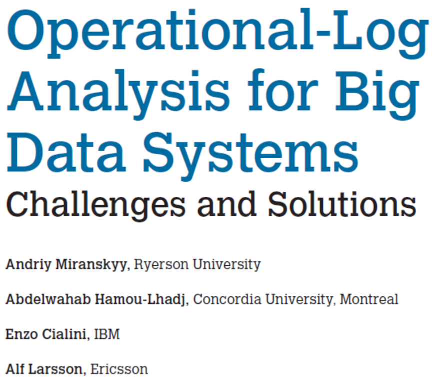
Applications

Logging and Tracing Infrastructure

Technology and Processes

But, what is the state of the practice of logging in industry?

Let’s take a look at the published work



Operational-Log Analysis for Big Data Systems

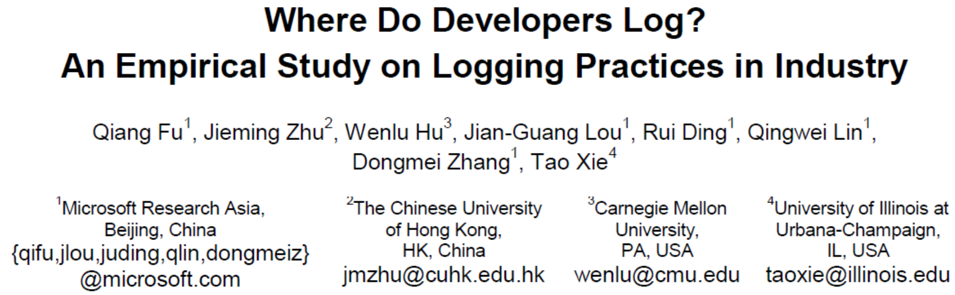
Challenges and Solutions

Andriy Miranskyy, Ryerson University

Abdelwahab Hamou-Lhadj, Concordia University, Montreal

Enzo Cialini, IBM

Alf Larsson, Ericsson



Where Do Developers Log? An Empirical Study on Logging Practices in Industry

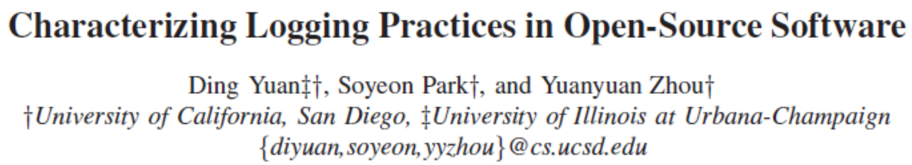
Qiang Fu[1], Jieming Zhu[2], Wenlu Hu[3], Jian-Guang Lou[1], Rui Ding[1], Qingwei Lin[1],
Dongmei Zhang[1], Tao Xie[4]

[1]Microsoft Research Asia,
Beijing, China
{qifu,jlou,juding,qlin,dongmeiz}
@microsoft.com

[2]The Chinese University
of Hong Kong,
HK, China
jmzhu@cuhk.edu.hk

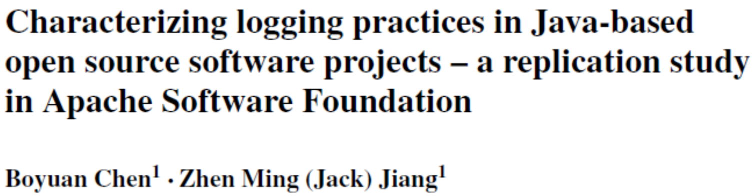
[3]Carnegie Mellon
University,
PA, USA
wenlu@cmu.edu

[4]University of Illinois at
Urbana-Champaign,
IL, USA
taoxie@illinois.edu



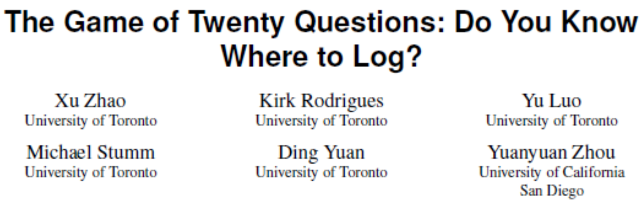
Characterizing Logging Practices in Open-Source Software

Ding Yuan‡†, Soyeon Park†, and Yuanyuan Zhou†
†University of California, San Diego, ‡University of Illinois at Urbana-Champaign
{diyuan, soyeon, yyzhou}@cs.ucsd.edu



Characterizing logging practices in Java-based open source software projects – a replication study in Apache Software Foundation

Boyuan Chen[1] · Zhen Ming (Jack) Jiang[1]



The Game of Twenty Questions: Do You Know Where to Log?

Xu Zhao
University of Toronto

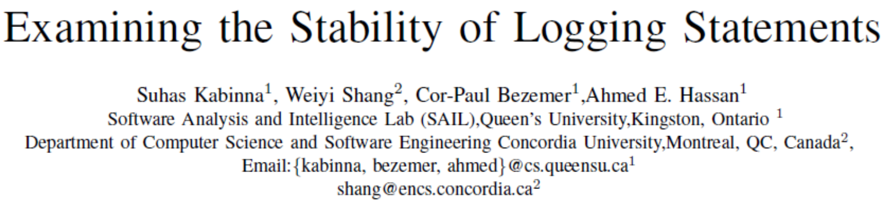
Kirk Rodrigues
University of Toronto

Yu Luo
University of Toronto

Michael Stumm
University of Toronto

Ding Yuan
University of Toronto

Yuanyuan Zhou
University of California
San Diego



Examining the Stability of Logging Statements

Suhas Kabinna[1], Weiyi Shang[2], Cor-Paul Bezemer[1],Ahmed E. Hassan[1]
Software Analysis and Intelligence Lab (SAIL),Queen’s University,Kingston, Ontario [1]
Department of Computer Science and Software Engineering Concordia University,Montreal, QC, Canada[2],
Email:{kabinna, bezemer, ahmed}@cs.queensu.ca[1]
shang@encs.concordia.ca[2]

Key findings

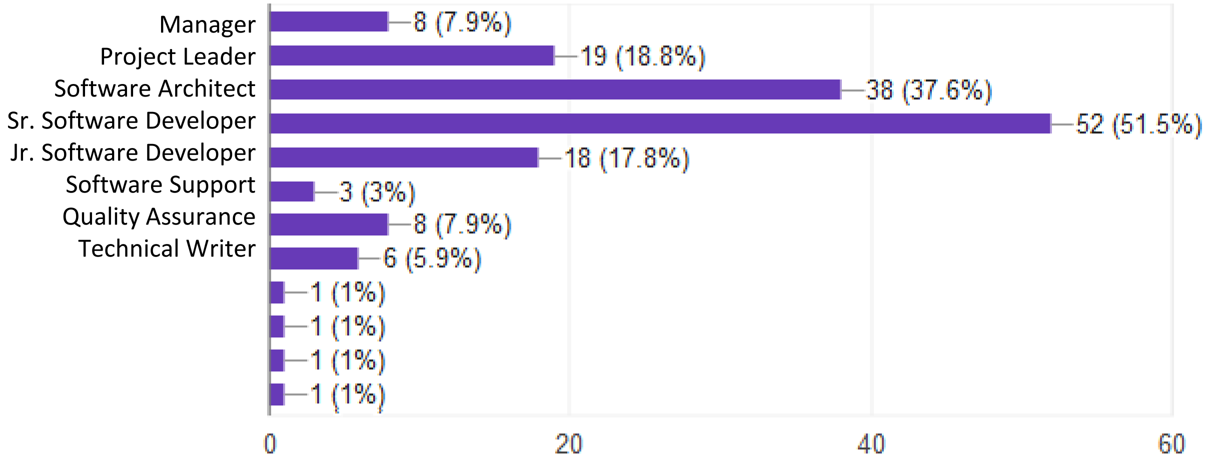
- Every 30 lines of code contains one line of logging code
- The average change rate of logging code is almost two times compared to the entire code
- In contrast to its small density, logging code is modified in a significant number of times
- One third of modifications are after-thoughts

Key findings (cont.)

- Developers seldom delete or move logging code
- Developers spend significant efforts on adjusting the verbosity level of log messages
- Developers are often confused when estimating the cost and benefit of each verbosity level

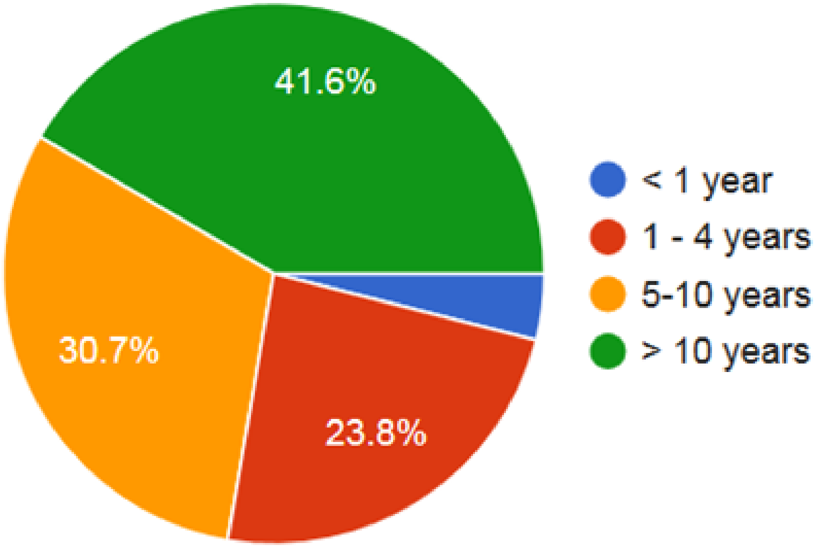
What do developers think about this?

- We conducted an online survey with 105 practitioners from various companies

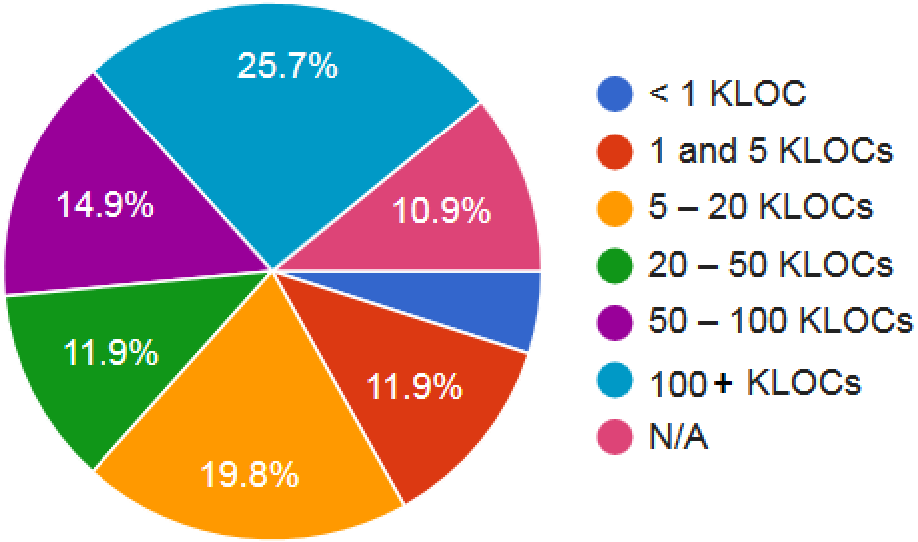


Participants' Background

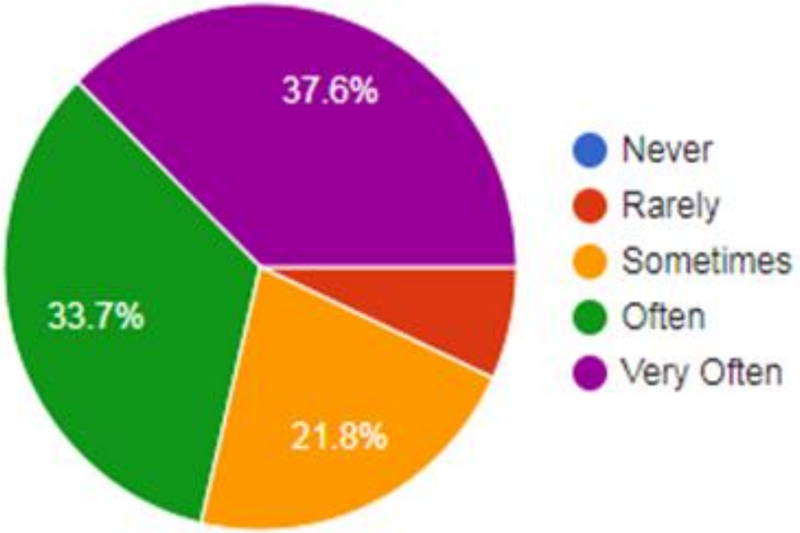
How long have you been working in the software field?



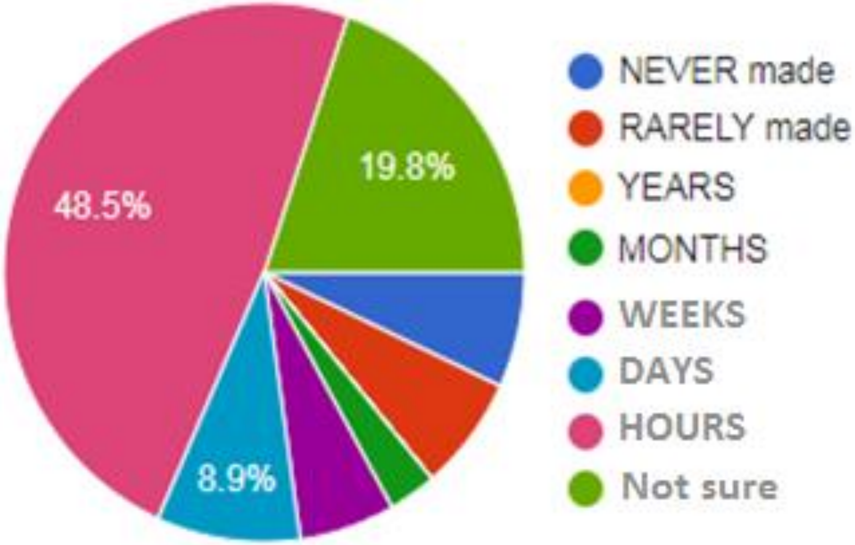
What is the size of your current (or recently completed) project in KLOCs?



How often do you consult the available logs when working on a software system?

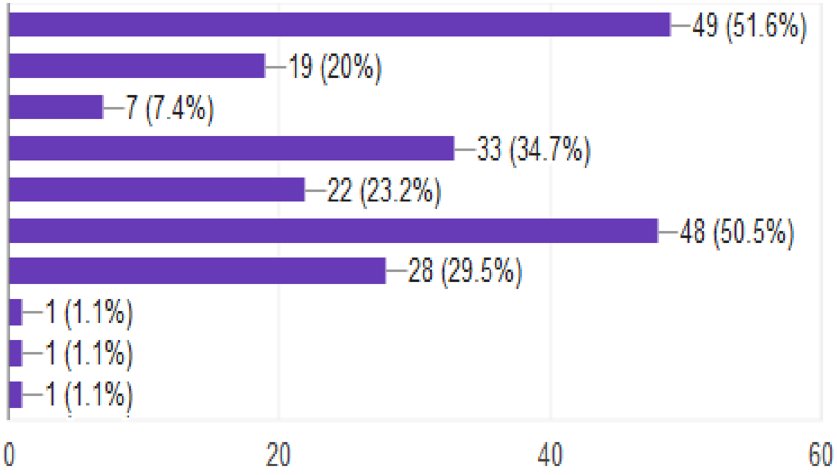


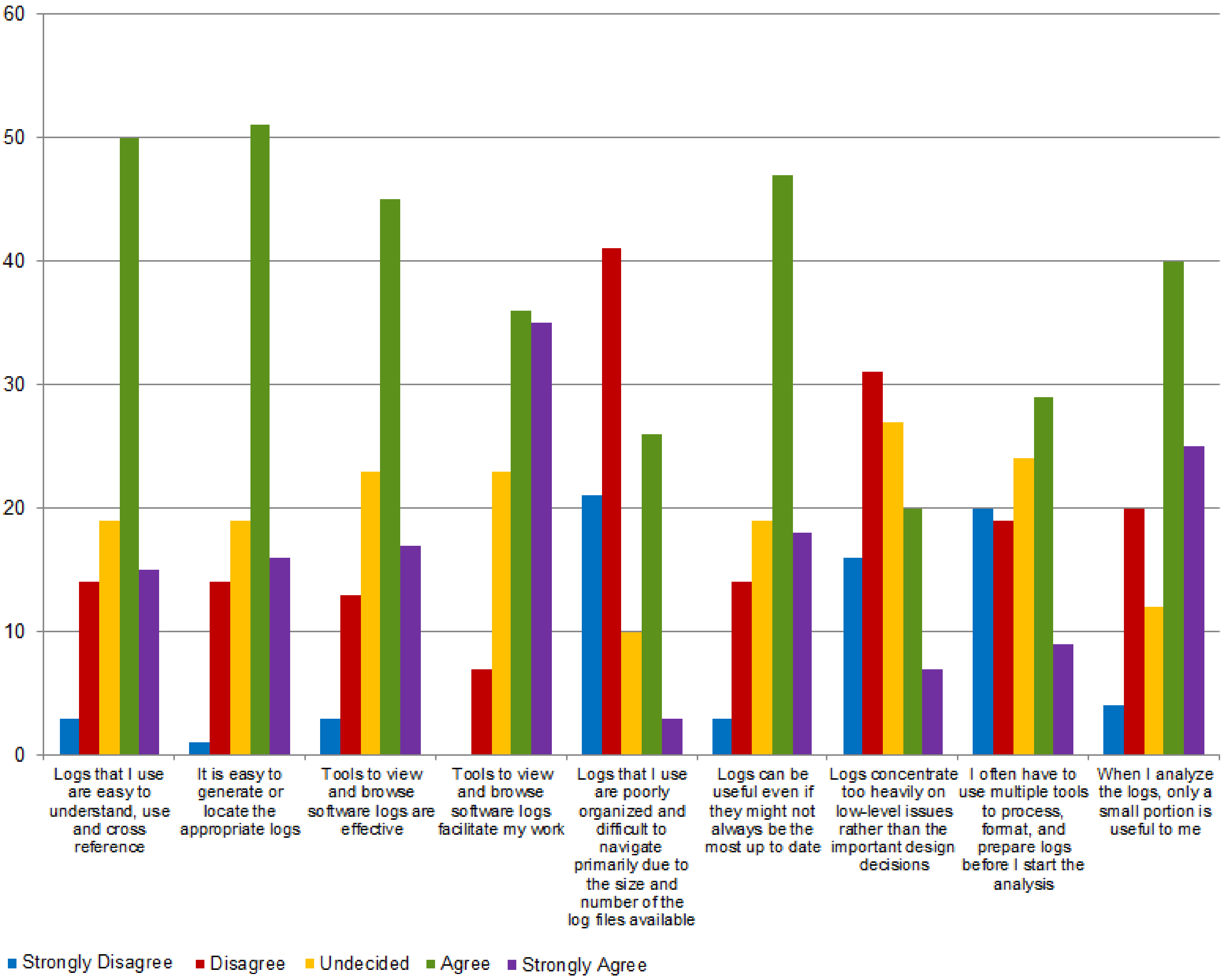
When changes are made to a system, how long does it take for the existing logs to be updated?



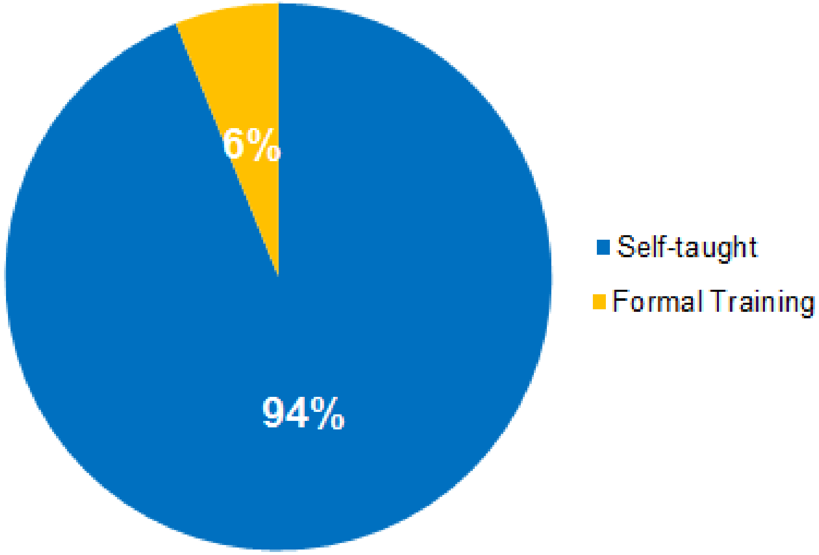
Which factors cause software logs to be out of sync with the system?

Time constraints on developers
Budget constraints on projects
Rapid staff turnover
Rapid changes in requirements
High cost not justifying the benefit
Team members see little benefit in maintaining logs
Lack of tools for maintaining logs

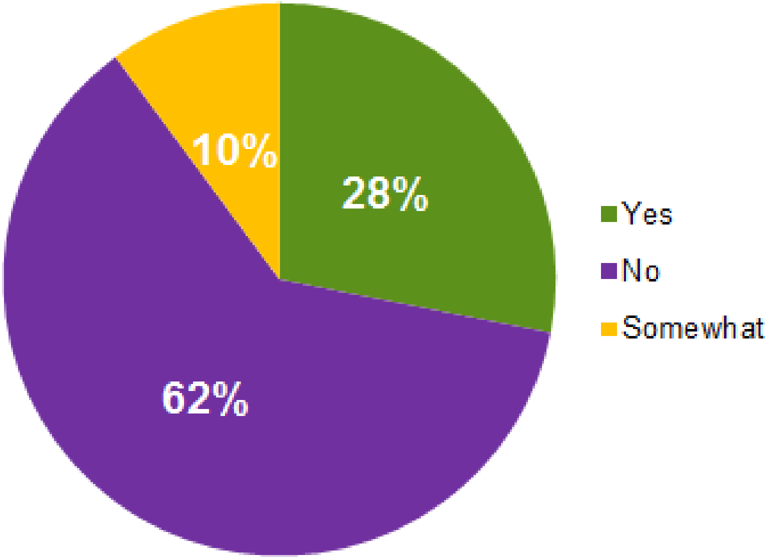



60
50
40
30
20
10
0
Logs that I use are easy to understand, use and cross reference
It is easy to generate or locate the appropriate logs
Tools to view and browse software logs are effective
Tools to view and browse software logs facilitate my work
Logs that I use are poorly organized and difficult to navigate primarily due to the size and number of the log files available
Logs can be useful even if they might not always be the most up to date
Logs concentrate too heavily on low-level issues rather than the important design decisions
I often have to use multiple tools to process, format, and prepare logs before I start the analysis
When I analyze the logs, only a small portion is useful to me
Strongly Disagree
Disagree
Undecided
Agree
Strongly Agree

Where have you learned the practice of logging?



Does your company have guidelines on how to write logs? To what extent they are followed?



Implications

- The current logging practice is ad hoc
  - impacting log quality, hindering the benefits of logging, and increasing log analysis costs

Implications

- The current logging practice is ad hoc
  - impacting log quality, hindering the benefits of logging, and increasing log analysis costs
- Logging statements are not always updated,
  - raising questions as to the validity of analysis

Implications

- The current logging practice is ad hoc
  - impacting log quality, hindering the benefits of logging, and increasing log analysis costs
- Logging statements are not always updated,
  - raising questions as to the validity of analysis
- Logging is thought of during implementation
  - design and architectural decisions are not taken into account.

Implications

- The current logging practice is ad hoc
  - impacting log quality, hindering the benefits of logging, and increasing log analysis costs
- Logging statements are not always updated,
  - raising questions as to the validity of analysis
- Logging is thought of during implementation
  - design and architectural decisions are not taken into account.
- Logging is pervasive in software development
  - adding to the cost of software projects

Implications

- The current logging practice is ad hoc
  - impacting log quality, hindering the benefits of logging, and increasing log analysis costs
- Logging statements are not always updated,
  - raising questions as to the validity of analysis
- Logging is thought of during implementation
  - design and architectural decisions are not taken into account.
- Logging is pervasive in software development
  - adding to the cost of software projects
- Logging takes a significant part of software evolution
  - despite its relatively small presence

Implications

- The current logging practice is ad hoc
  - impacting log quality, hindering the benefits of logging, and increasing log analysis costs
- Logging statements are not always updated,
  - raising questions as to the validity of analysis
- Logging is thought of during implementation
  - design and architectural decisions are not taken into account.
- Logging is pervasive in software development
  - adding to the cost of software projects
- Logging takes a significant part of software evolution
  - despite its relatively small presence
- The current verbosity levels of logging tools are confusing
  - leading to inconsistencies

Where should we go from here?

Awareness

Proven Practices and Standards

Design for Observability

Intelligent Logging

Education

THANK YOU!

CONCORDIA.CA