POLYTECHNIQUE MONTRÉAL

Université de Montréal

Dataset and Analysis for the Commit Messages of the Linux Kernel Out-of-Memory Killer

Mouna Dhaouadi

PhD Candidate

Université de Montréal, Canada

http://www-labs.iro.umontreal.ca/~dhaouadm/

Bentley James Oakes

Assistant Professor

Polytechnique Montréal, Canada

https://bentleyjoakes.github.io/

Michalis Famelis

Associate Professor

Université de Montréal, Canada

https://michalis.famelis.info/

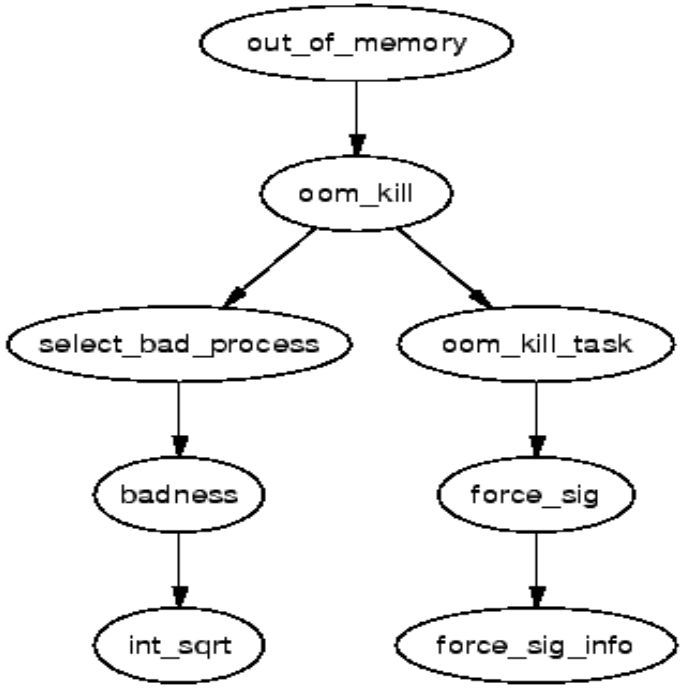
Linux Out-of-Memory Killer

Linux kernel:

- Development is through Git commits
- Culture for motivating/describing changes

Out-of-Memory Killer subsystem:

- When Linux runs out of memory, it calls OOM-Killer to avoid crashes
- Two broad steps:
  - Select “best” task to kill using heuristics
  - Force task to release memory and exit



https://www.kernel.org/doc/gorman/html/understand/understand016.html

What is the impact of changes?

- Dev works on "*reclaiming used memory from the OOM victim*".
- Find interesting commits from the Git history of OOM-Killer.

Suren's Challenge:
How does my decision impact previously established decisions? How to make sure I will not cause conflicts with existing rationales?

Suren

Manually uncovering why the system is the way it is can be hard!

Oom: give the dying task a higher priroity
August 9th, 2010

memcg: give current access to memory reserves if it's trying to die
March 23rd, 2011

_dying_task_prio()
April 14th, 2011

mm, oom: introduce oom reaper
March 25th, 2016

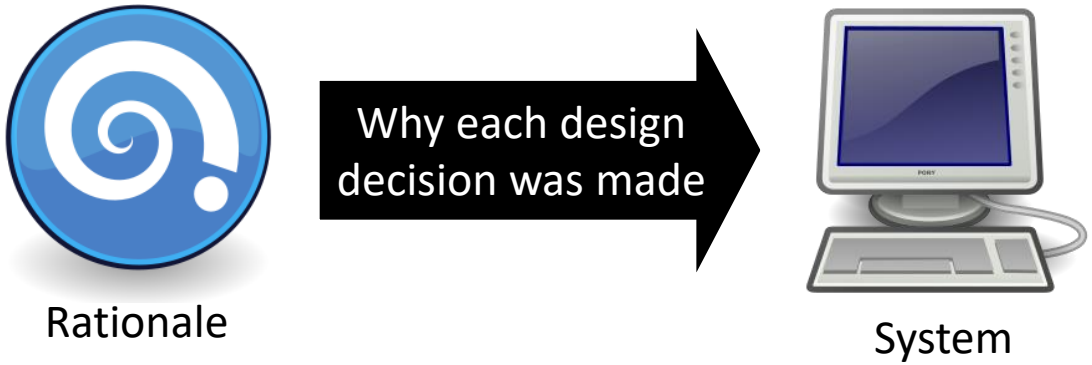
mm : introduce process_mrelease system call
September 2nd, 2021

Software Rationale

Big corpus of work on representing, structuring, extracting rationale

Useful to:

- understand the system
- learn from mistakes
- reuse solutions
- avoid conflicts



Little (Alkhadi'18, Sharma'21) about its characteristics in **real world systems**

No prior work on developer's rationale in **code commit messages** of OSS

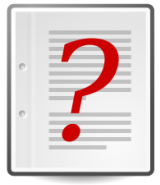
Alkadhi, Nonnenmacher, Guzman, and Bruegge. "How Do Developers Discuss Rationale?" SANER 2018
Sharma, Savarimuthu and Stanger. "Extracting Rationale for Open Source Software Development Decisions — A Study of Python Email Archives" ICSE 2021

Rationale in the OOM Killer Commit History

- Is rationale information present in commit messages?
- What are the factors that impact it?

- How does it evolve over time?
- How is it structured in commit messages?

Rationale in the OOM Killer Commits



1. Motivation

2. Dataset creation

3. Analysis

4. Conclusions

Rationale in the OOM Killer Commits

1. Motivation

2. Dataset creation

3. Analysis

4. Conclusions

Dataset Creation: Labelling

An example commit

Sentence
signal: Use SEND_SIG_PRIV not SEND_SIG_FORCED with SIGKILL and SIGSTOP
Now that siginfo is never allocated for SIGKILL and SIGSTOP there is no difference between SEND_SIG_PRIV and SEND_SIG_FORCED for SIGKILLand SIGSTOP.
This makes SEND_SIG_FORCED unnecessary and redundant in the presence of SIGKILL and SIGSTOP.
Therefore change users of SEND_SIG_FORCED that are sending SIGKILL or SIGSTOP to use SEND_SIG_PRIV instead.
This removes the last users of SEND_SIG_FORCED.

- Collect 418 commits
- Remove merge commits, filter code sentences
- 404 commits / 2234 sentences
- 3 annotators label sentences

Piloting

- 6 rounds → codebook + protocol
  - Resolve conflicts by discussion
- “Decision”, “Rationale”, “Supporting Facts”, “Inapplicable”

Batch annotations of the rest of the sentences

- Multiple classifications per sentence
- In disagreement, take classification union
- Fleiss kappa: Around 0.66 (fair to good agreement)

Codebook definitions

Label	Meaning
Decision	An action or a change that has been made, including a description of the patch behaviour
Rationale	Reason for a decision or value judgment
Supporting Facts	A narration of facts used to support a decision
Inapplicable	Pre-processing error or bad sentences (i.e., does not contain English sentences)

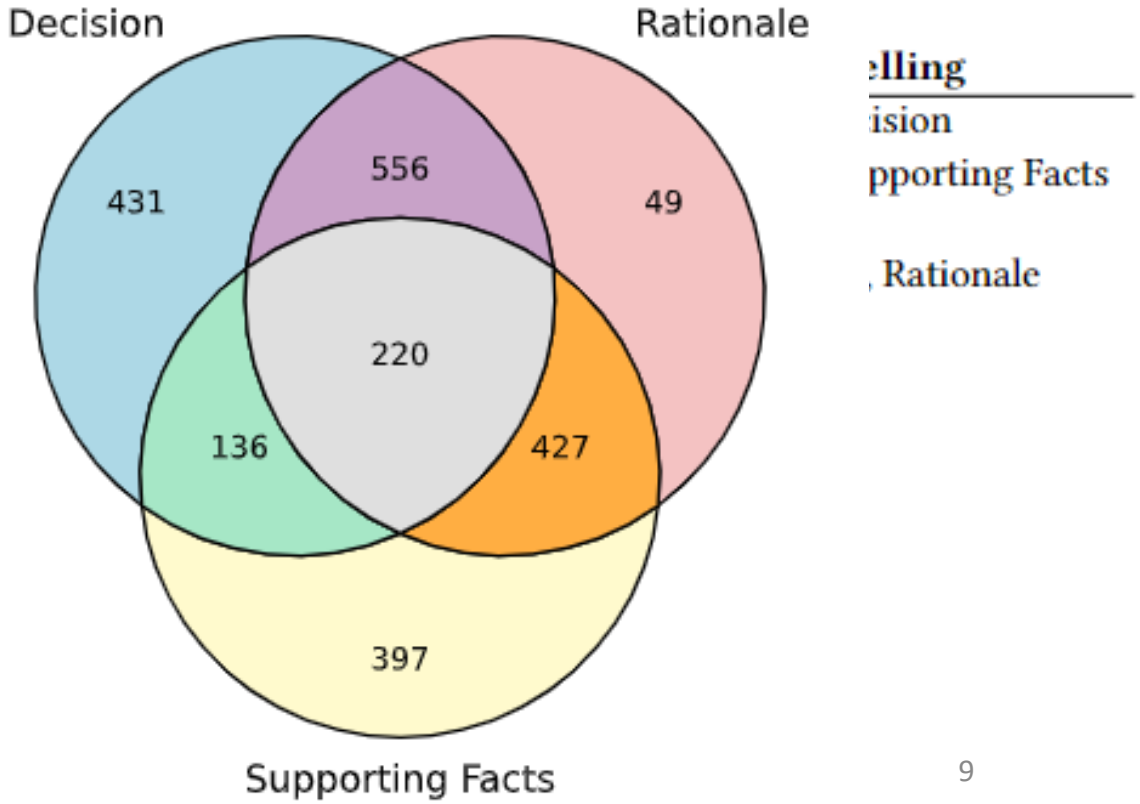
Dataset: https://zenodo.org/records/10063089

Examples

Sentence	Labelling
mm, oom: introduce independent oom killer ratelimit state	Decision
printk_ratelimit() uses the global ratelimit state for all printks	Supporting Facts
The oom killer should not be subjected to this state just because another subsystem or driver may be flooding the kernel log	Rationale
This patch introduces printk ratelimiting specifically for the oom killer.	Decision

Sentence	elling
tlb: mmu_gather: Remove start/end arguments from tlb_gather_mmu()	ision
The 'start' and 'end' arguments to tlb_gather_mmu() are no longer needed r	pporting Facts
'fullmm' flushing	
Remove the unused arguments and update all callers.	, Rationale

Substantial label overlap:



Rationale in the OOM Killer Commits

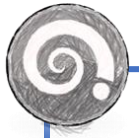
1. Motivation

2. Dataset creation

3. Analysis

4. Conclusions

Dataset Analyses and Research Questions



Presence of Rationale

- RQ1. How **many** commits contain rationale?
- RQ2. How **much** of the commit contains rationale?

Factors impacting Rationale

- RQ3. Does the quantity of rationale reported depend on the commit message **size**?
- RQ4. Does the quantity of rationale reported depend on the developer **experience**?

Evolution of rationale over time

- RQ5. How does rationale **evolve** over time?
- RQ6. How does rationale evolve over time for the five **core** contributors?

Structure of commit messages

- RQ7. In what **order** do the categories mostly appear?

Dataset Analyses: Presence of Rationale

Presence of Rationale

- RQ1. How **many** commits contain rationale?
- RQ2. How **much** of the commit contains rationale?

$$rationale\ density\% = \frac{\text{number of commits that contain rationale}}{\text{total number of commits}}$$

- **98.9%** of commits contain rationale

$$average\ rationale\ density = \frac{\sum_{commits} \text{rationale density}}{\text{number of commits that contain rationale}}$$

- About **60%** of sentences per commit contain rationale

Dataset Analyses: Factors impacting Rationale

Factors impacting Rationale

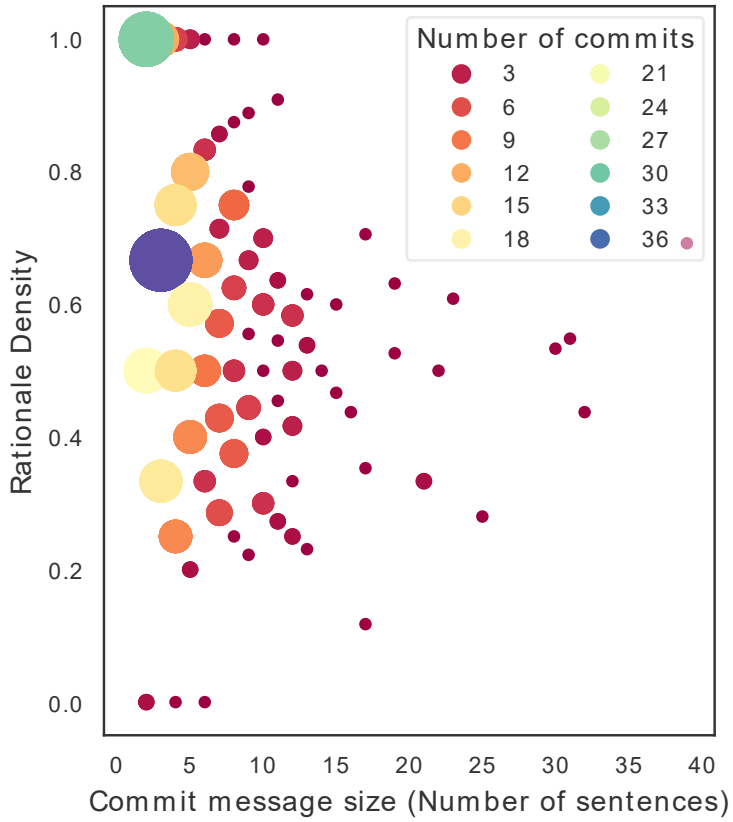
- RQ3. Does the quantity of rationale reported depend on the commit message **size**?
- RQ4. Does the quantity of rationale reported depend on the developer **experience**?

Most the commits have fewer than 15 sentences

No statistically significant correlation.

Observations:

- A lot of the **short** commits (fewer than 6 sentences) have a **high** rationale density (> 60%).
- As a commit becomes **longer**, the tendency is between **40% to 60%** of sentences to contain rationale information.



Commit size versus rationale density

Dataset Analyses: Factors impacting Rationale

Factors impacting Rationale

- RQ3. Does the quantity of rationale reported depend on the commit message size?

- RQ4. Does the quantity of rationale reported depend on the developer **experience**?

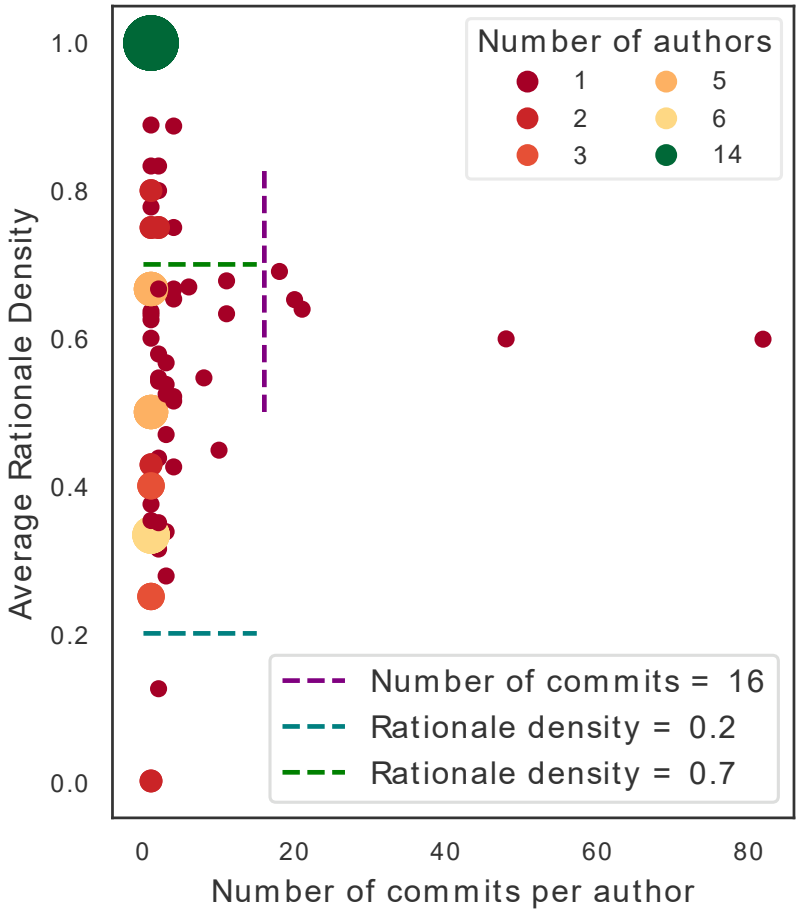
Only 5 developers wrote more than 16 commits.

All the other developers wrote fewer than 16 commits; most of them, fewer than 10 commits.

No statistically significant correlation.

Observation:

- More experienced developers' commits have a **consistent** rationale density near 60%.

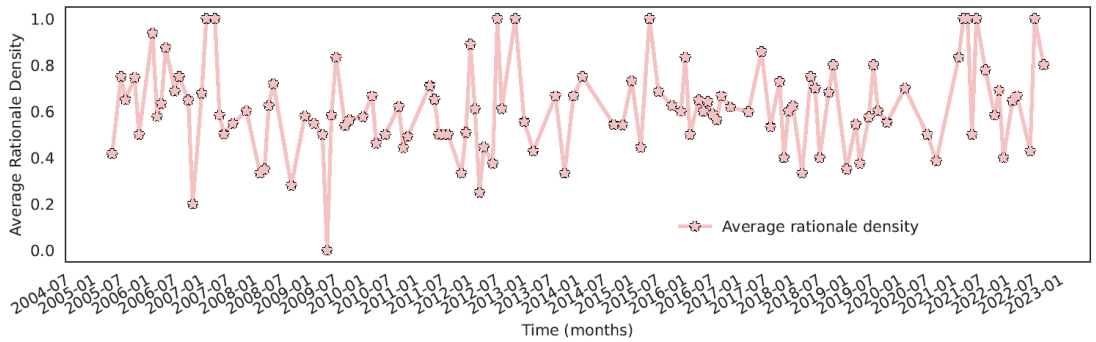


Commits per author versus average rationale density

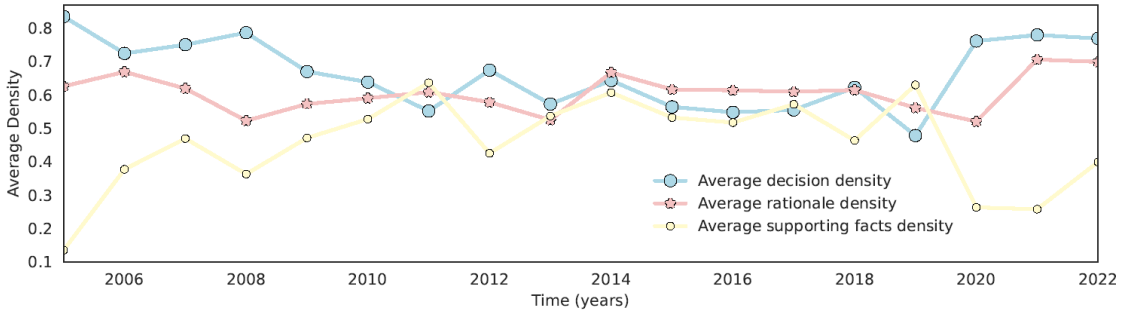
Dataset Analyses: Evolution of rationale over time

Evolution of rationale over time

- RQ5. How does rationale **evolve** over time?
- RQ6. How does rationale evolve over time for the five **core** contributors?



Monthly evolution of the average rationale density



Yearly evolution of the average rationale density, the average decision density and the average supporting facts

Rationale density consistently high at around 0.6.

Decision density consistently **high** (> 0.5).

Supporting facts density typically **low** (< 0.6).

In early and late years **decision** density **> rationale** density **>> supporting facts** density.

In middle years, all converge at around 0.55, supporting facts density always at bottom

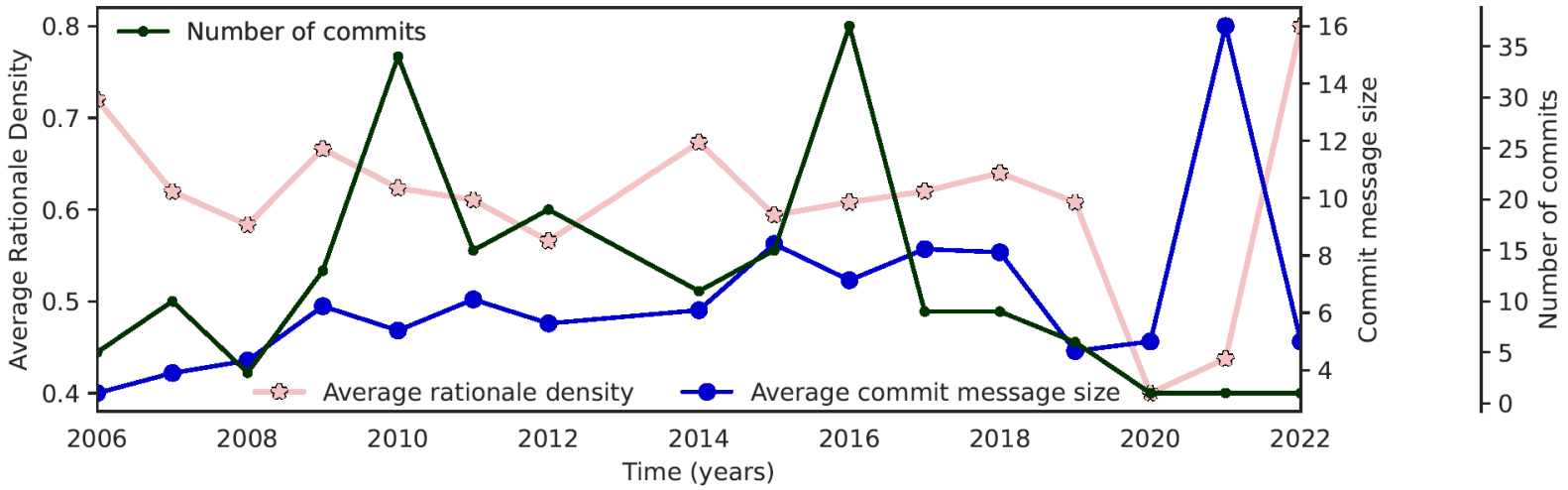
Dataset Analyses: Evolution of rationale over time

Evolution of rationale over time

- RQ5. How does rationale **evolve** over time?

- RQ6. How does rationale evolve over time for the five **core** contributors?

Five contributors wrote 189 commits **~half** of the studied commits.



Evolution of average rationale density, average commit message size, and number of commits for top 5 contributors

Rationale density was consistent around 0.6 for all the years before 2020, but it dropped to around 0.4 in **2020 and 2021** and went up to 0.8 in 2022.

The number of commits varies considerably each year

Usually, the top contributors write short commits (< 8 sentences)

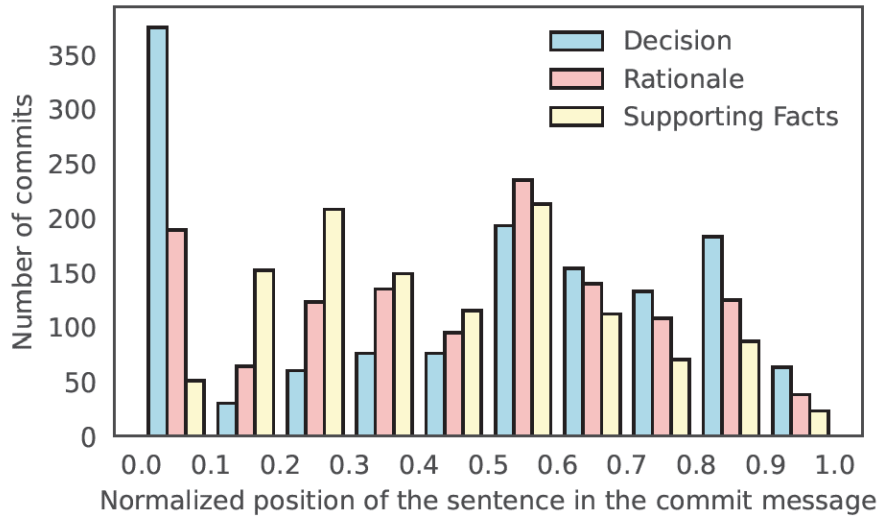
Dataset Analyses: Structure of commit messages

Structure of commit messages

- RQ7. In what **order** do the categories mostly appear?

Common Structure:

1. Decisions
2. Supporting Facts
3. Rationale
4. Decisions



Distribution of the categories over the normalized positions of the sentences of the commit messages

Rationale in the OOM Killer Commits

1. Motivation

2. Dataset creation

3. Analysis

4. Conclusions

Rationale in the OOM Killer

Presence

Commit messages **almost always** contain rationale information.

On average, around **60% of the message** contains rationale information

Impacting factors

The **quantity** of rationale information reported does **not** depend on the commit message **size** or developer **experience**.

Experienced developers have a rationale density around **60%**.

Evolution over time

Rationale density is **consistent** (~0.6).
Decision density is always **high** (> 0.5).
Supporting facts density is **lower** (< 0.6).

More experienced developers write **short** commit messages (fewer than eight sentences).

Structure of commit messages

Developers tend to **start and end** their commit messages with Decisions.

Rationale and Supporting Facts appear in the **middle** of the commit, with Supporting Facts usually preceding Rationale sentences

Dataset and Analysis for the Commit Messages of the Linux Kernel Out-of-Memory Killer

An empirical contribution to better understand rationale in-the-wild.

Université de Montréal

POLYTECHNIQUE MONTRÉAL

Mouna Dhaouadi

Bentley James Oakes

Michalis Famelis

Software Rationale

Big corpus of work on representing, structuring, extracting rationale

Useful to:
- understand the system
- learn from mistakes
- reuse solutions
- avoid conflicts

Little (Alkhadi'18, Sharma'21) about its characteristics in **real world systems**

No prior work on developer's rationale in **code commit messages** of OSS

Rationale in the OOM Killer

Presence

Commit messages **almost always** contain rationale information.

On average, around **60% of the message** contains rationale information

Impacting factors

The **quantity** of rationale information reported does **not** depend on the commit message **size** or developer **experience**.

Experienced developers have a rationale density around **60%**.

Evolution over time

Rationale density is **consistent** (~0.6).
Decision density is always **high** (> 0.5).
Supporting facts density is **lower** (< 0.6).

More experienced developers write **short** commit messages (fewer than eight sentences).

Structure of commit messages

Developers tend to **start and end** their commit messages with Decisions.

Rationale and Supporting Facts appear in the **middle** of the commit, with Supporting Facts usually preceding Rationale sentences

- Is rationale information present in commit messages?
- What are the factors that impact it?
- How does it evolve over time?
- How is it structured in commit messages?

Next Steps:
- Improve dataset quality and richness
- Compare with other Linux modules, other OSS projects
- Automate rationale classification

Dataset: https://zenodo.org/records/10063089
Paper: https://arxiv.org/abs/2403.18832