ESEC/FSE SRC 2023

Université de Montréal

A Data Set of Extracted Rationale from Linux Kernel Commit Messages

Mouna Dhaouadi

3rd year PhD Candidate

Université de Montréal, Canada

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

Motivation

For a **decision**, software/system developer has **rationale**
'Why' reasoning for their decision

Rationale is *useful information* to :
understand the system,
learn from mistakes,
reuse solutions,
avoid conflicts

Rationale in Commits

‘‘Fix: Added check to prevent null pointer exception.’’

versus

‘‘changes’’

Challenges:

- Implicit or unrecorded
- Ambiguous language
  - Especially past/future tenses
  - Non-native writers
- Need technical understanding

Research Questions

How does rationale appear ? What are its characteristics ?

How can we *structure*, *extract* and *manage* rationale information from code commits?

Proposed Contributions

- An empirical contribution to better understand rationale in open-source software *(FSE SRC 2023, Manuscript under review 2023)*

- An end-to-end rationale extraction and management system *(ASE NIER 2022, MDE-Intelligence 2023)*

Running example: OOM Killer subsystem

Linux kernel:

- Development is through Git commits
- Culture for motivating/describing changes
- Valuable (highest-quality) source of rationale

Out-of-Memory Killer subsystem:

- When Linux kernel runs out of memory , OOM-Killer is called to avoid crashing
- Two broad steps:
    1. Select “best” task to kill
    2. Force task to release memory and exit
- Meaningful heuristics

Research Question 1

How does rationale appear ? What are its characteristics ?

Dataset

Manual labelling of OOM-Killer commit sentences**.**

Procedure:

- Collect 418 commits
- Remove merge commits, filter code sentences
  404 commits / 2234 sentences
- Three annotators label sentences
- Six piloting rounds -> codebook + protocol
- Resolve conflicts in discussion

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.

An example commit

Codebook

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)

- Obtained through piloting rounds and discussions
- **Multiple classifications per sentence**
- In disagreement, take classification union
- Fleiss kappa: Around 0.66 (fair to good agreement)

Labelling example 1

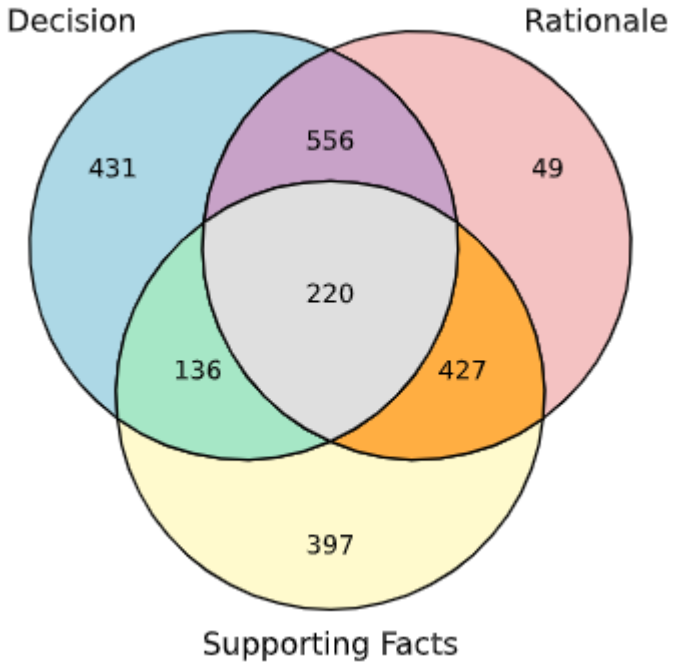
Sentence	Labelling
mm, oom: introduce independent oom killer ratelimit state	Decision
printk_ratelimit() uses the global ratelimit state for all printks	Supporting Facts
The oom killer should not be subjected to this state just because another subsystem or driver may be flooding the kernel log	Rationale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

Labelling example 2

Sentence	Labelling
tlb: mmu_gather: Remove start/end arguments from tlb_gather_mmu()	Decision
The 'start' and 'end' arguments to tlb_gather_mmu() are no longer needed now that there is a separate function for 'fullmm' flushing	Rationale, Supporting Facts
Remove the unused arguments and update all callers.	Decision, Rationale

Labelling insights

Substantial overlap between the categories



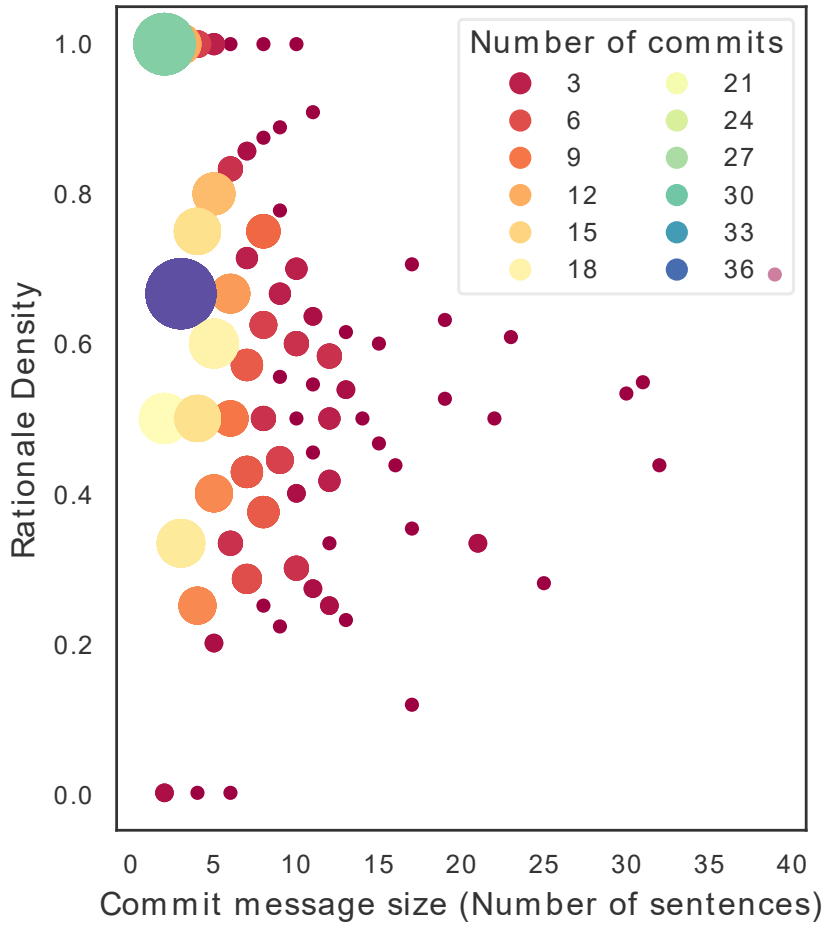
Distribution of the sentences in the OOM dataset

Labelling insights

- **98.9%** of commits contain rationale
- About **60%** of sentences per commit contain rationale

Labelling insights

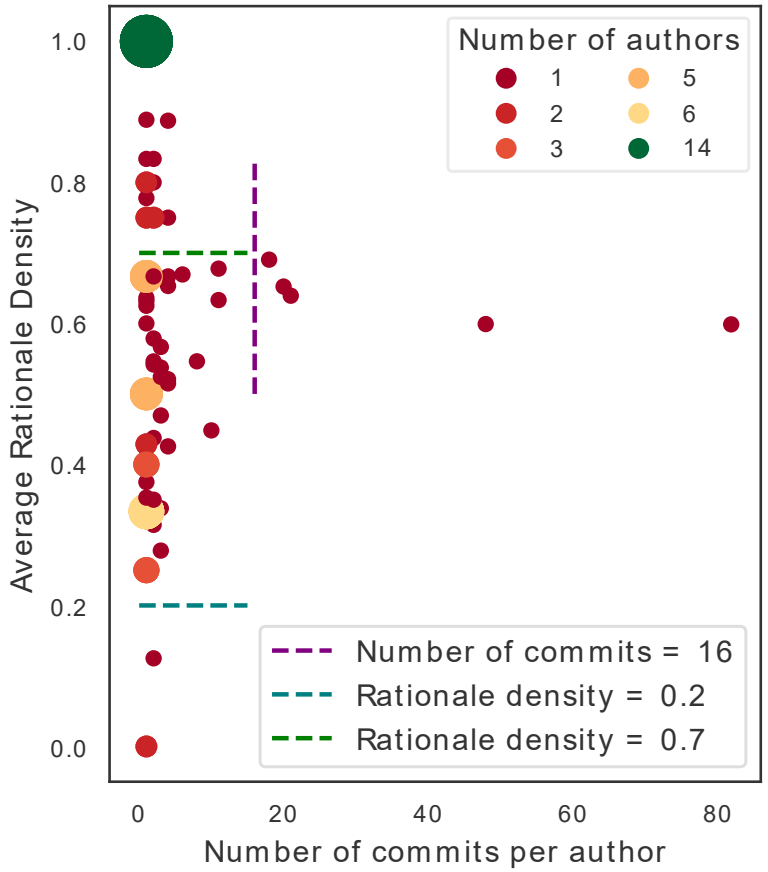
- Most the commits have fewer than 15 sentences
- A lot of the short commits (fewer than 6 sentences) have a high rationale density (>0.6).
- As a commit becomes longer, it tends to have between 40% to 60% of its sentences containing rationale information.



Commit size versus rationale density

Labelling insights

- Only five developers have written more than 16 commits. All the other developers have written fewer than 16 commits, and most of them fewer than 10 commits.
- More experienced developers' commits have a consistent rationale density near 60%.

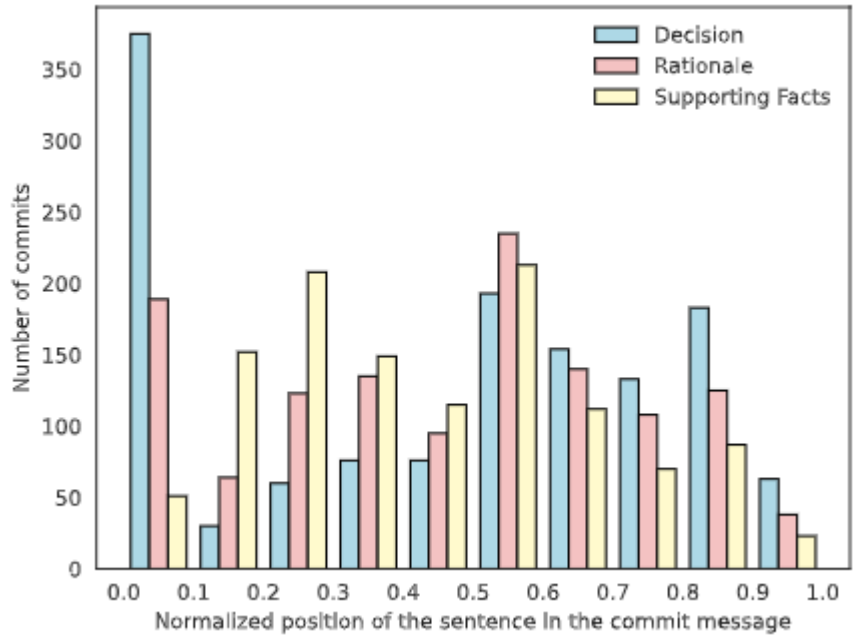


Commits per author versus average rationale density

Labelling insights

- Common Structure:

Decisions -> Supporting Facts -> Rationale -> Decisions



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

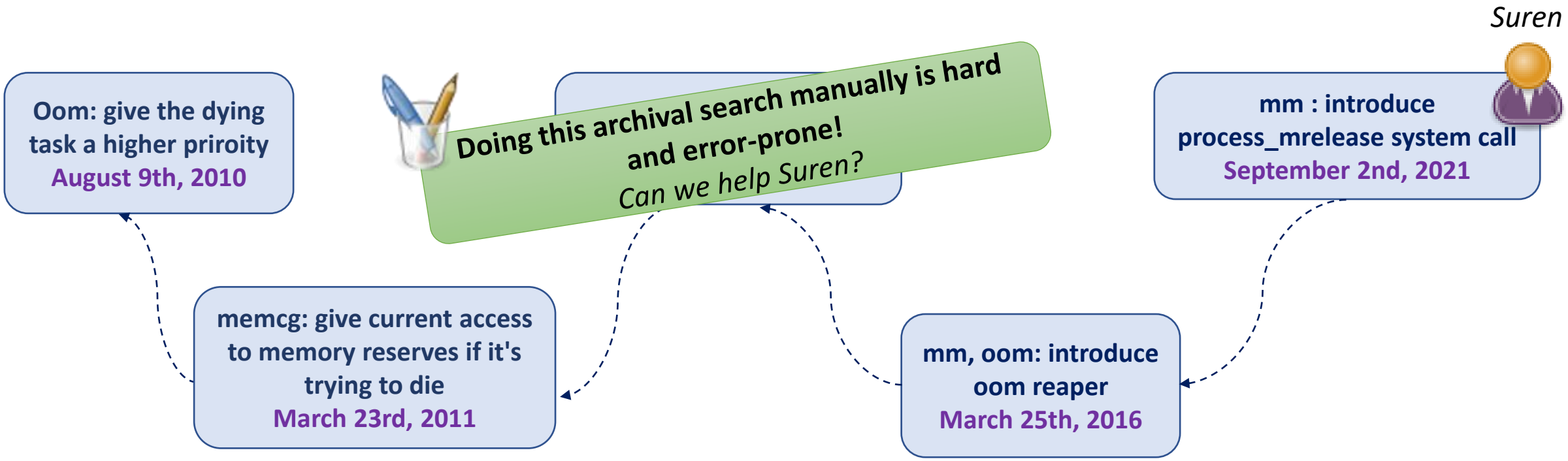
Research Question 2

How can we *structure*, *extract* and *manage* rationale information from code commits?

OOM-Killer example

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

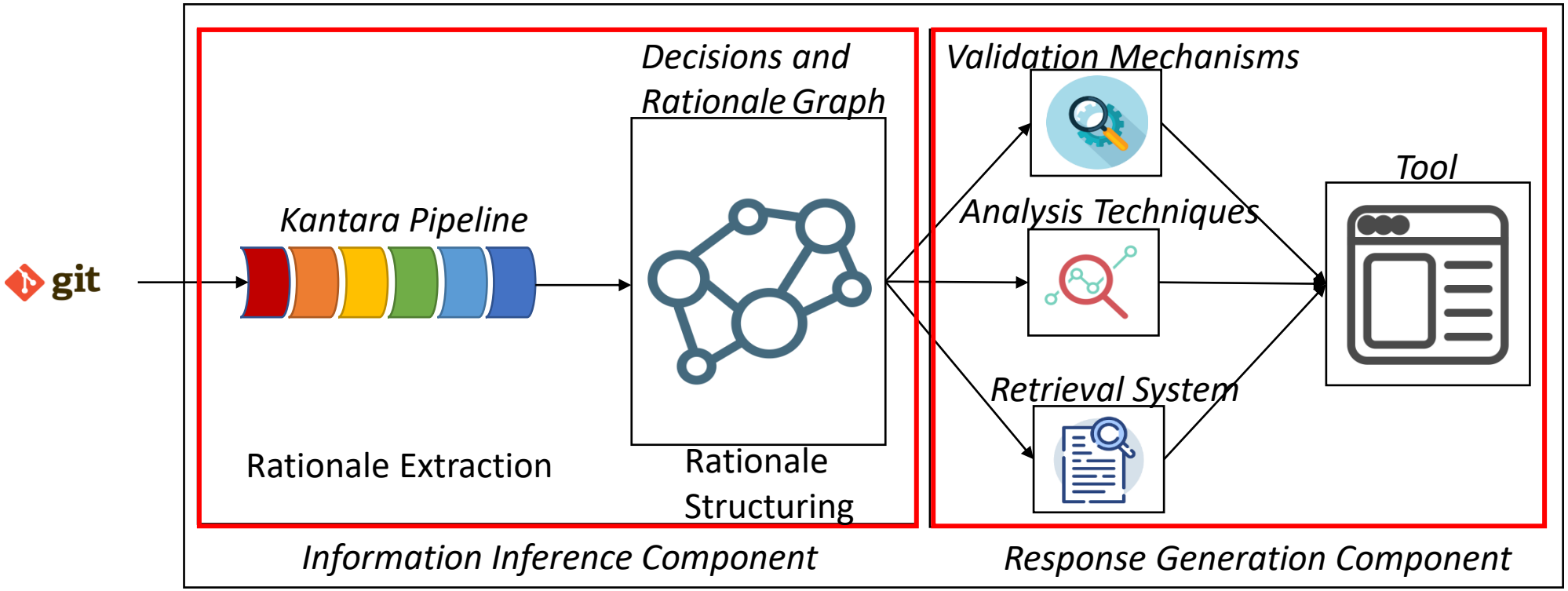
- Manually selected interesting commits from the Git history of OOM-Killer.
- Topic of "*reclaiming used memory from the OOM victim*".



End-to-End Rationale Reconstruction, *Mouna Dhaouadi, Bentley James Oakes, Michalis Famelis.* The IEEE/ACM International Conference on Automated Software Engineering (ASE), Michigan, USA, 2022. *NIER Track*

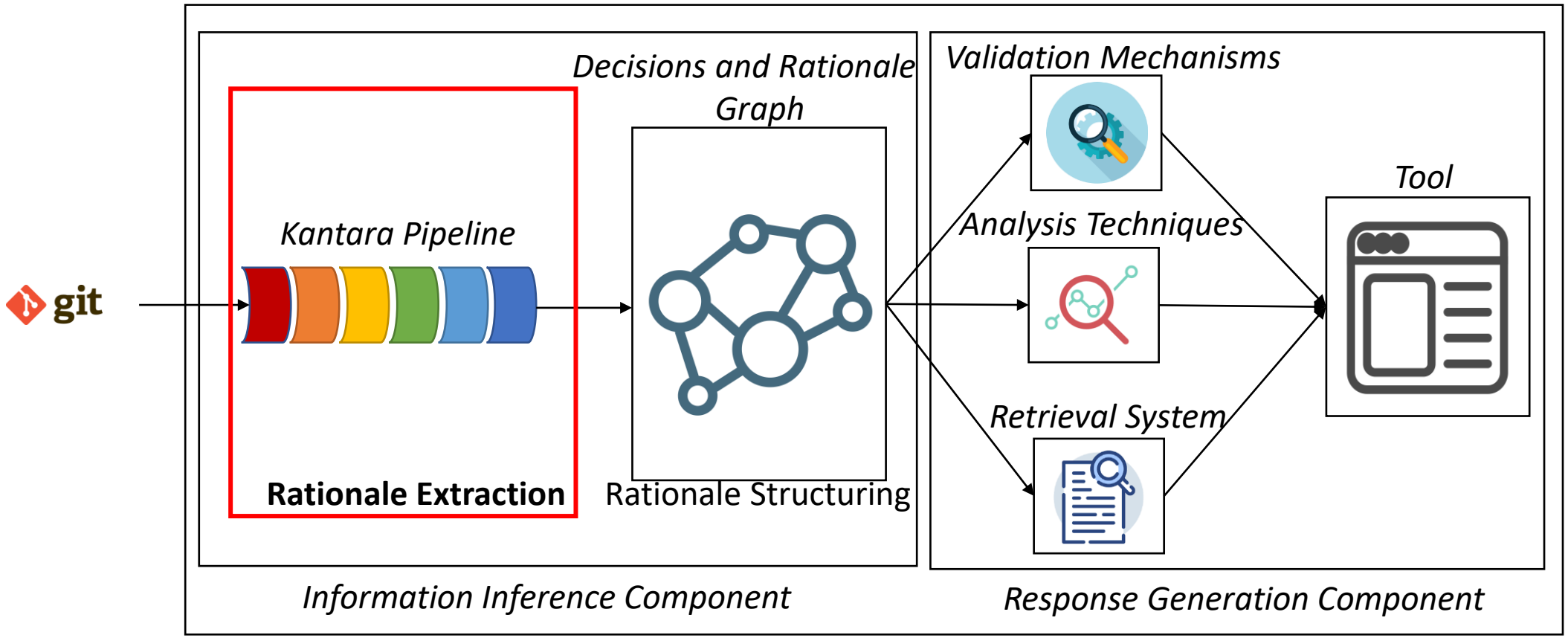
Proposed solution

- A complete rationale extraction and management system

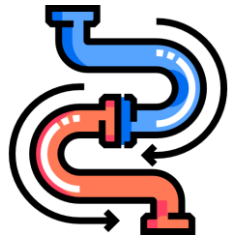


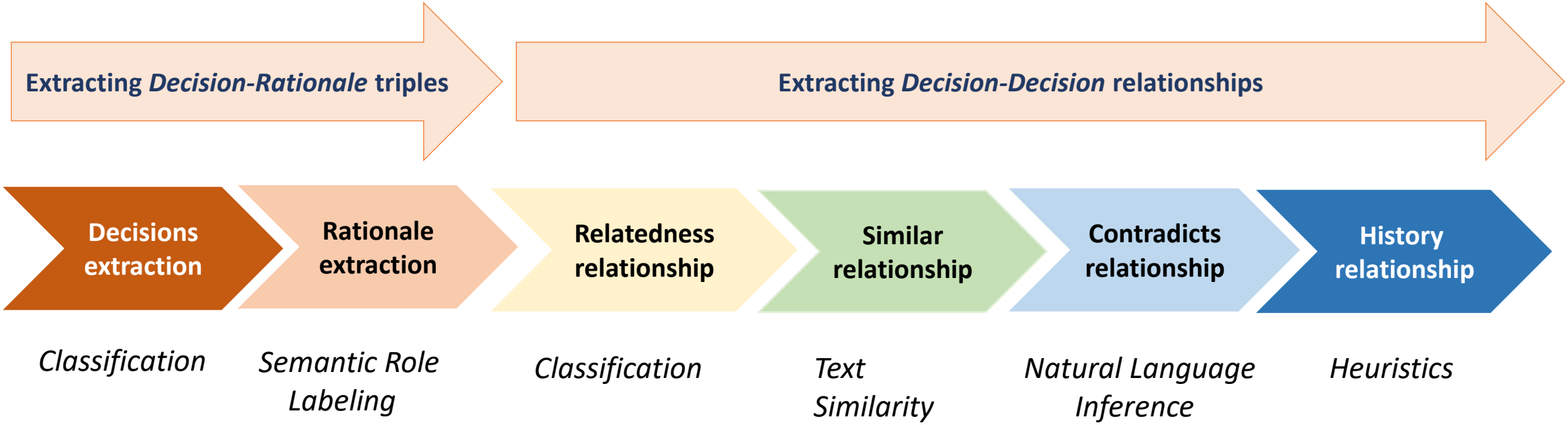
Extraction and Management of Rationale, *Mouna Dhaouadi.* The IEEE/ACM International Conference on Automated Software Engineering (ASE), Michigan, USA, 2022. *Doctoral Symposium*

Automated rationale extraction



Kantara: Information Extraction pipeline

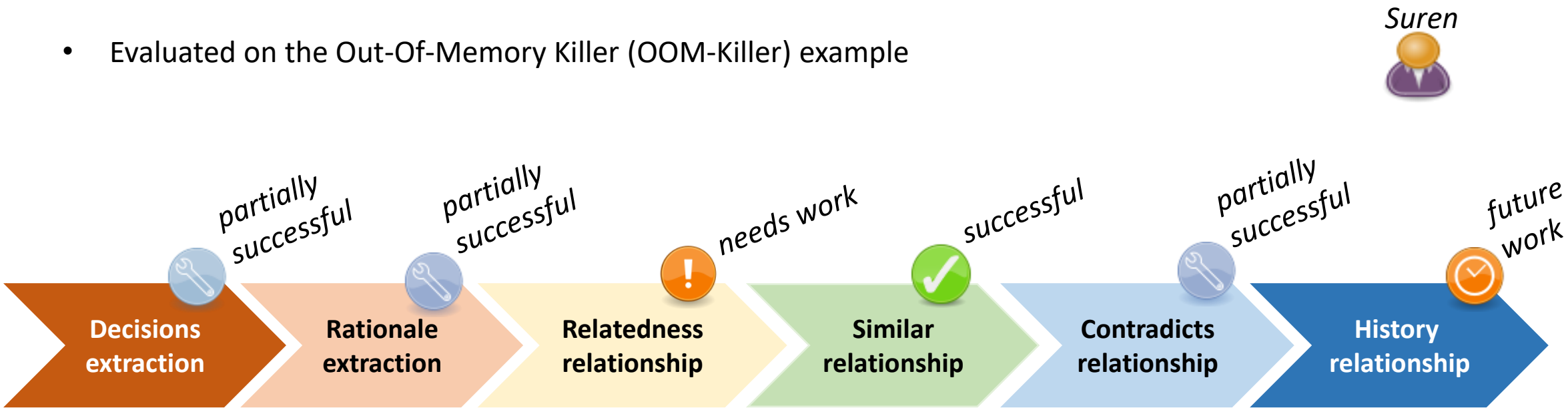




End-to-End Rationale Reconstruction, *Mouna Dhaouadi, Bentley James Oakes, Michalis Famelis.* The IEEE/ACM International Conference on Automated Software Engineering (ASE), Michigan, USA, 2022. *NIER Track*

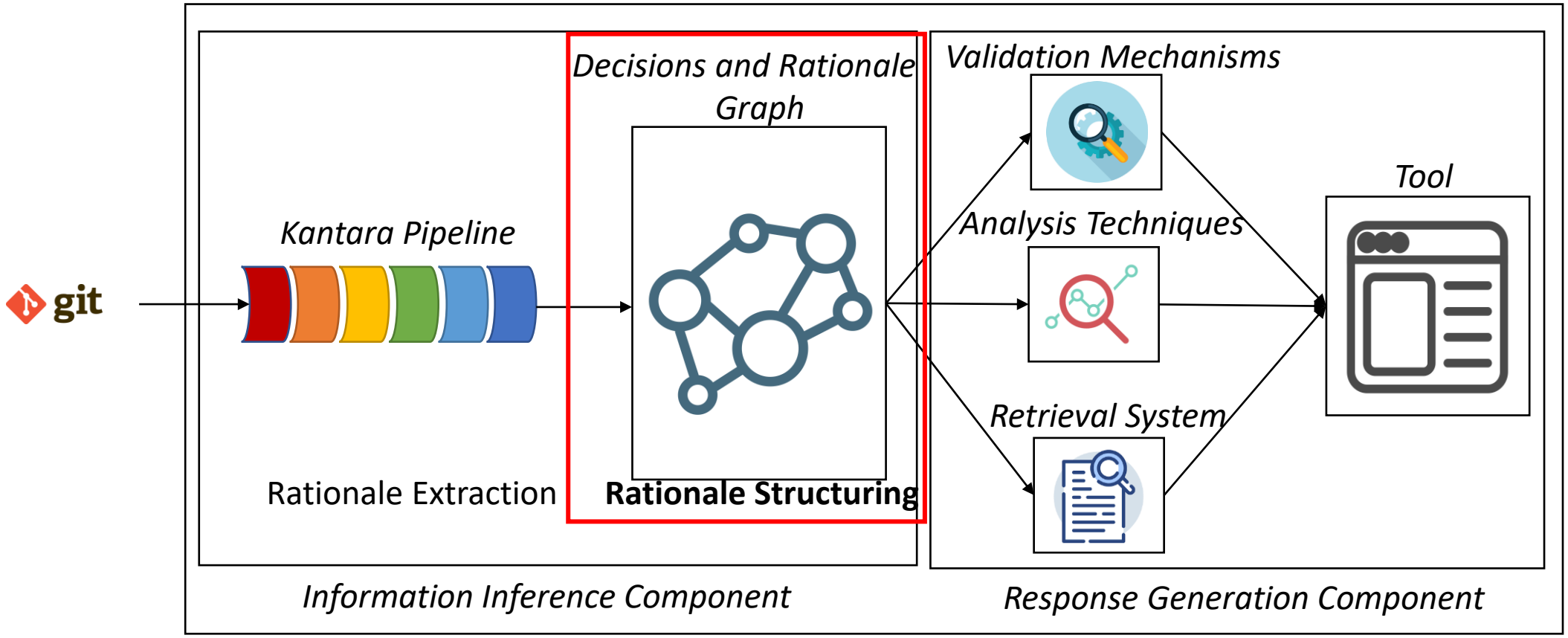
Prototype to evaluate feasibility of Kantara

- Evaluated on the Out-Of-Memory Killer (OOM-Killer) example



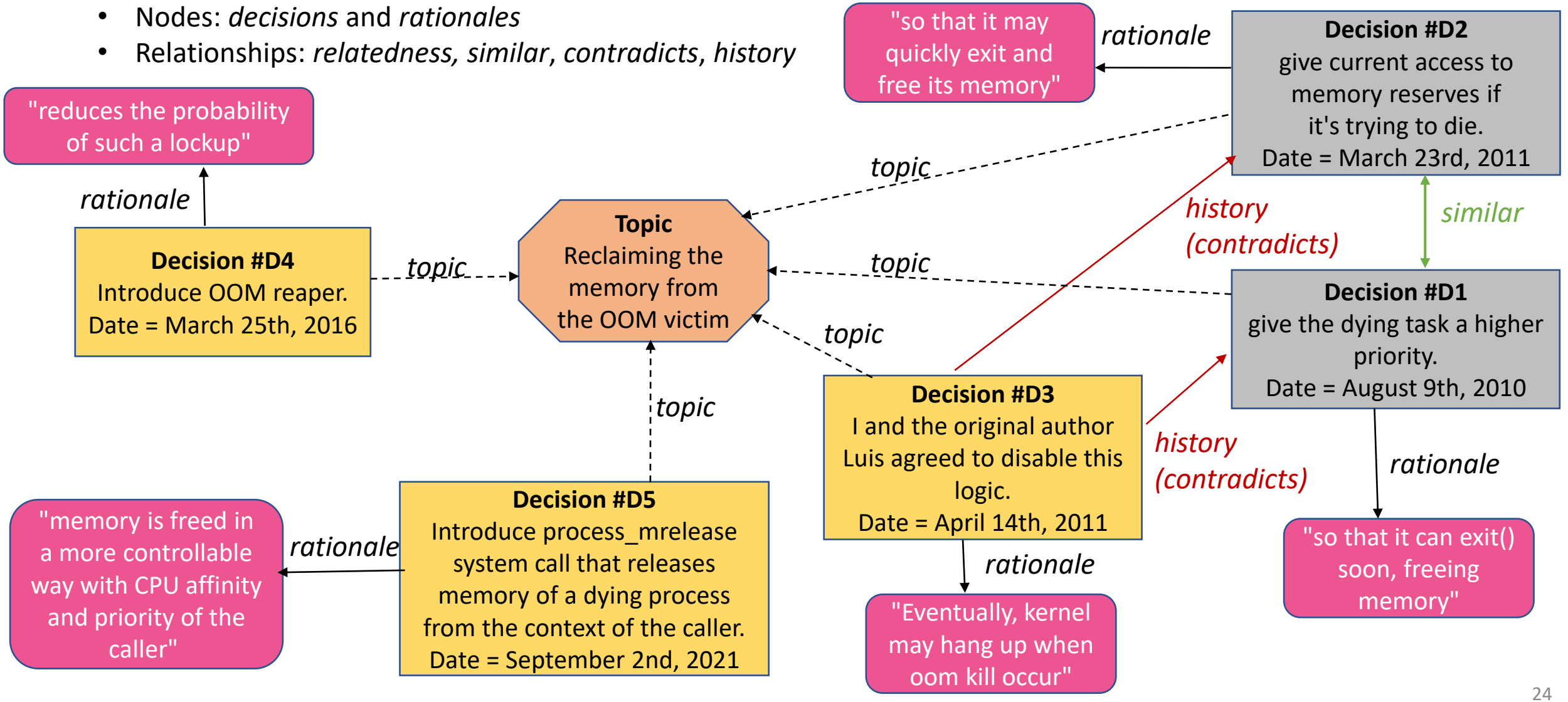
End-to-End Rationale Reconstruction, *Mouna Dhaouadi, Bentley James Oakes, Michalis Famelis.* The IEEE/ACM International Conference on Automated Software Engineering (ASE), Michigan, USA, 2022. *NIER Track*

Rationale structuring



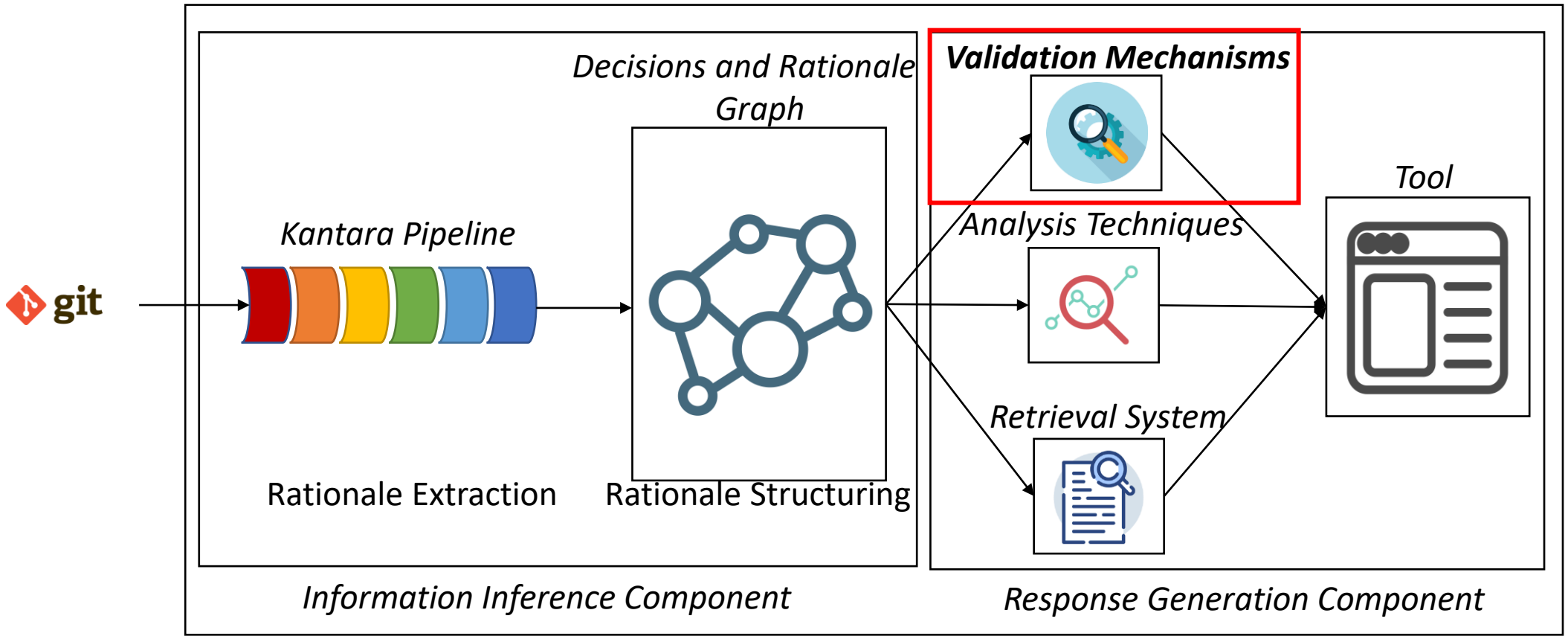
Knowledge Graph

- Nodes: *decisions* and *rationales*
- Relationships: *relatedness, similar*, *contradicts*, *history*



End-to-End Rationale Reconstruction, *Mouna Dhaouadi, Bentley James Oakes, Michalis Famelis.* The IEEE/ACM International Conference on Automated Software Engineering (ASE), Michigan, USA, 2022. *NIER Track*

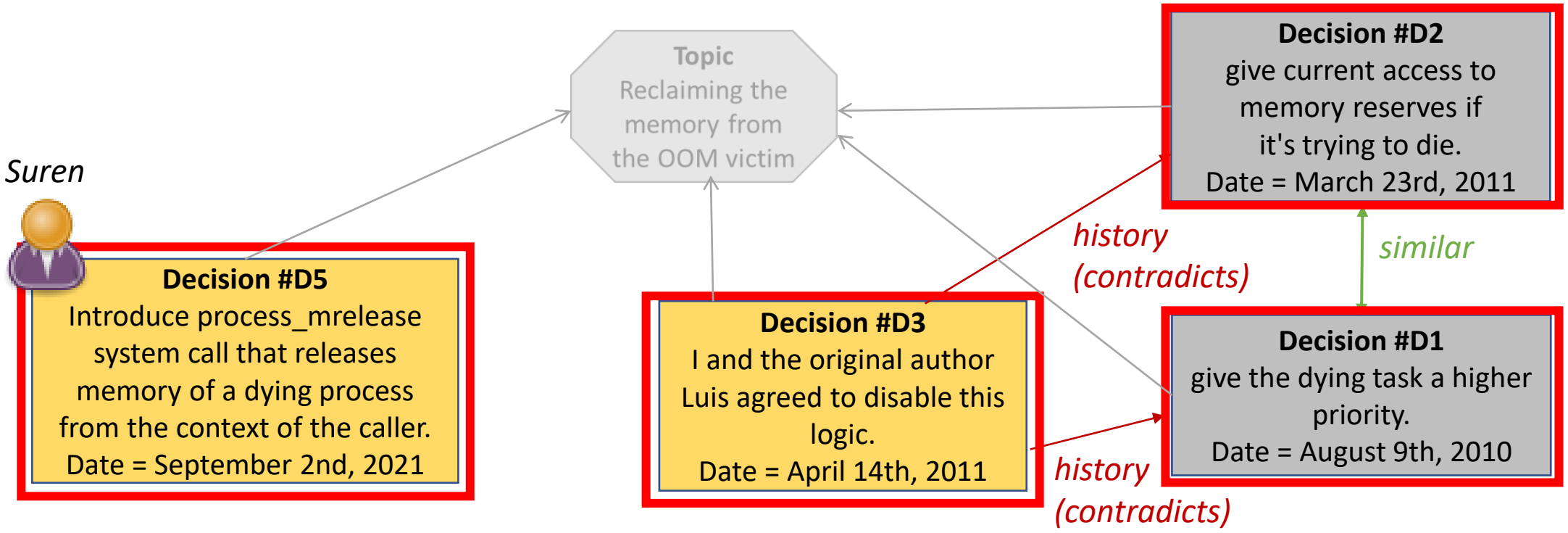
Rationale management



Validation Mechanisms - prototype

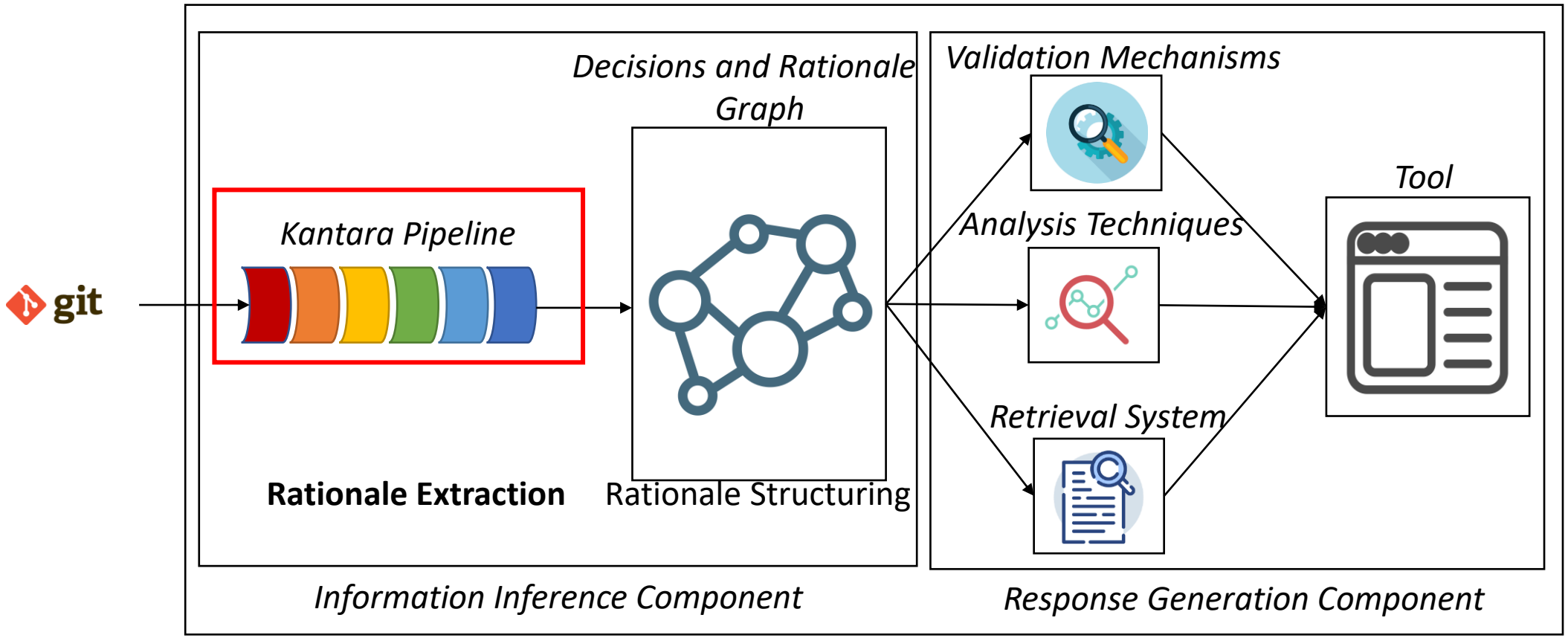
Suren's Challenge:
How does my decision impact previously established decisions? How to make sure I will not cause conflicts with existing rationales?

Look for potential conflicts when a new decision is proposed.



End-to-End Rationale Reconstruction, *Mouna Dhaouadi, Bentley James Oakes, Michalis Famelis.* The IEEE/ACM International Conference on Automated Software Engineering (ASE), Michigan, USA, 2022. *NIER Track*

Automatic rationale extraction



- Need ground truth for the automatic classification

Automatic sentence classification: Initial results

180 commits from the dataset we created (Linux kernel subsystem)

Binary classification: Logistic regression, decision tree, SVM
Multi-label classification: Random Forest, XGBoost, KNN

XGBoost classification evaluation			
Label	Precision	Recall	F1-score
Decision	0.76	0.69	0.72
Rationale	0.62	0.41	0.49
Supporting Facts	0.64	0.68	0.66

Towards Understanding and Analyzing Rationale in Commit Messages using a Knowledge Graph Approach, *Mouna Dhaouadi, Bentley James Oakes, Michalis Famelis. 5th* Workshop on Artificial Intelligence and Model-driven Engineering (MDE-Intelligence), Sweden, 2023.

Conclusion and Future work

Summary:

- An empirical contribution to better understand rationale in-the-wild
- A rationale extraction and management framework

Linux Kernel as object of the study

Future work:

- Improve automatic classification
- Finish implementing the framework
- Provide a tool (e.g., a pull request bot)

This work is partially supported by the ACM-W scholarship program and Google.