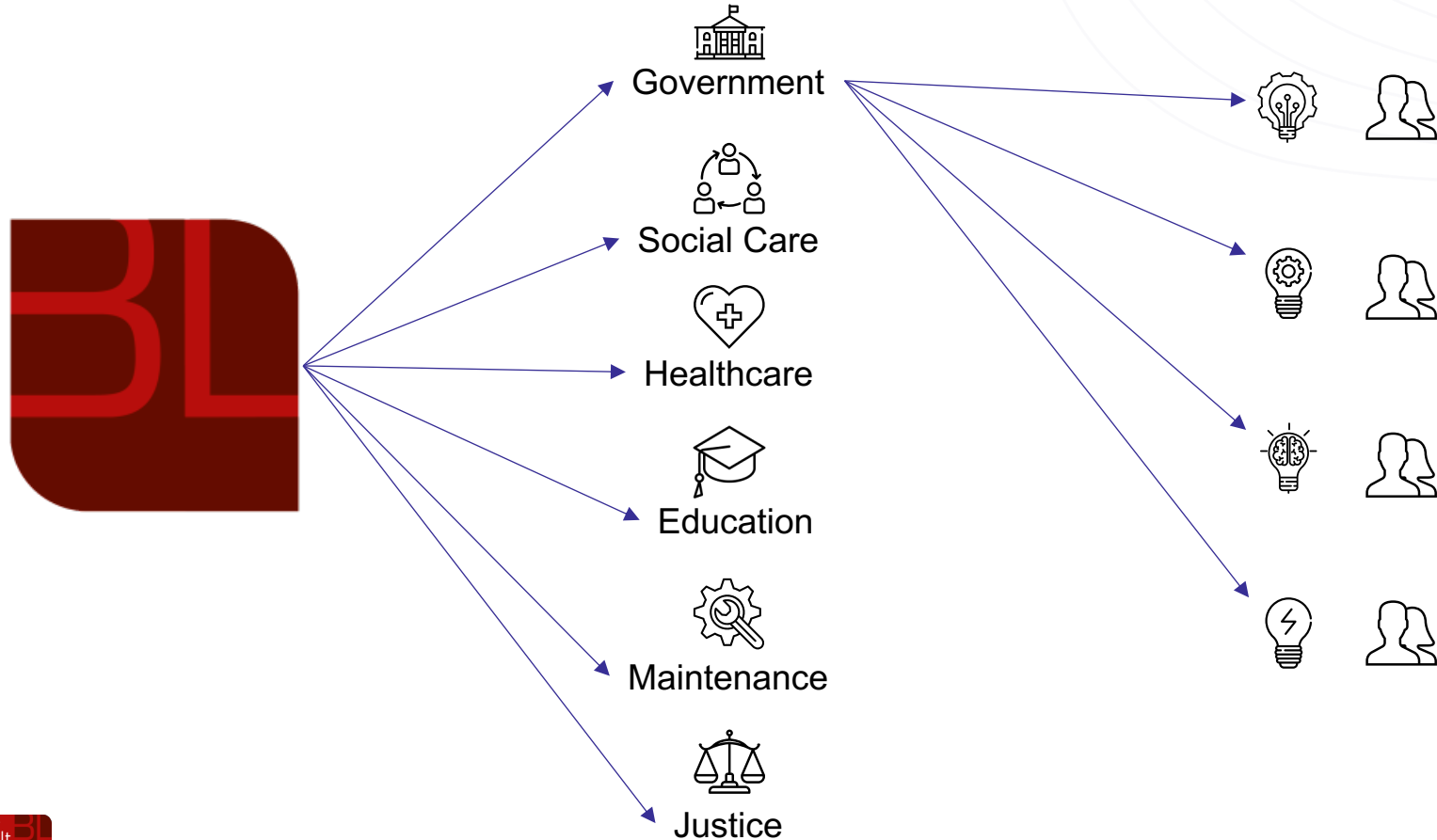


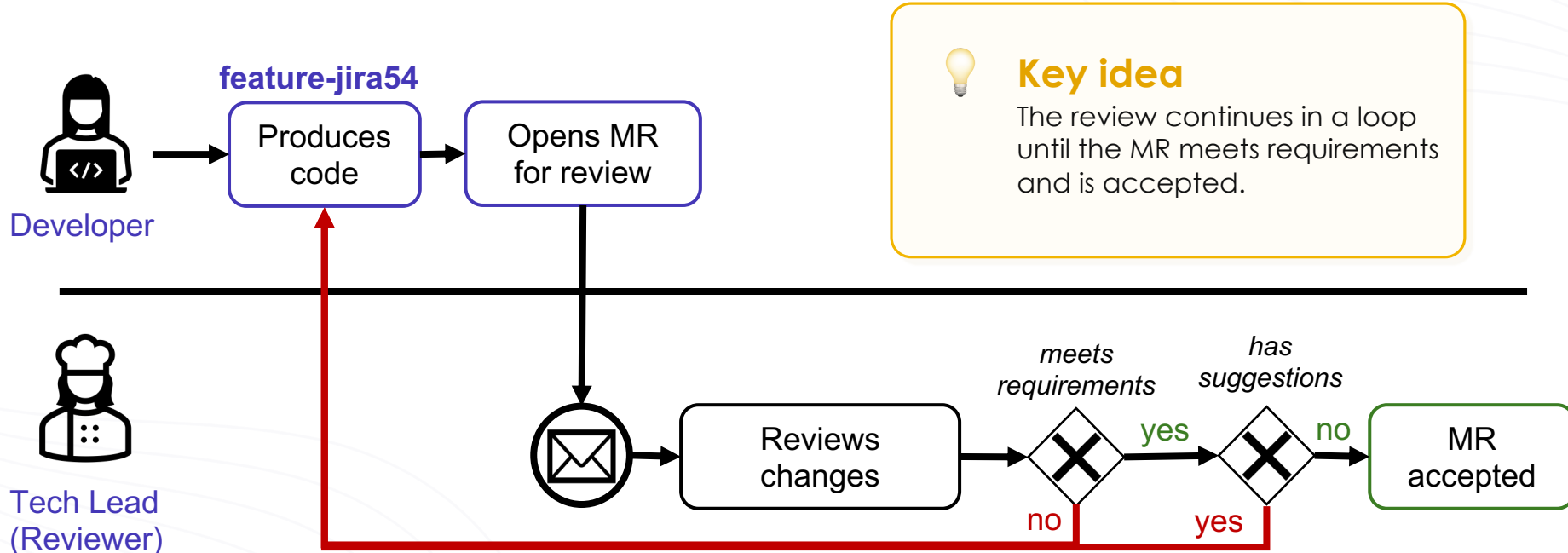
Benoit Verhaeghe



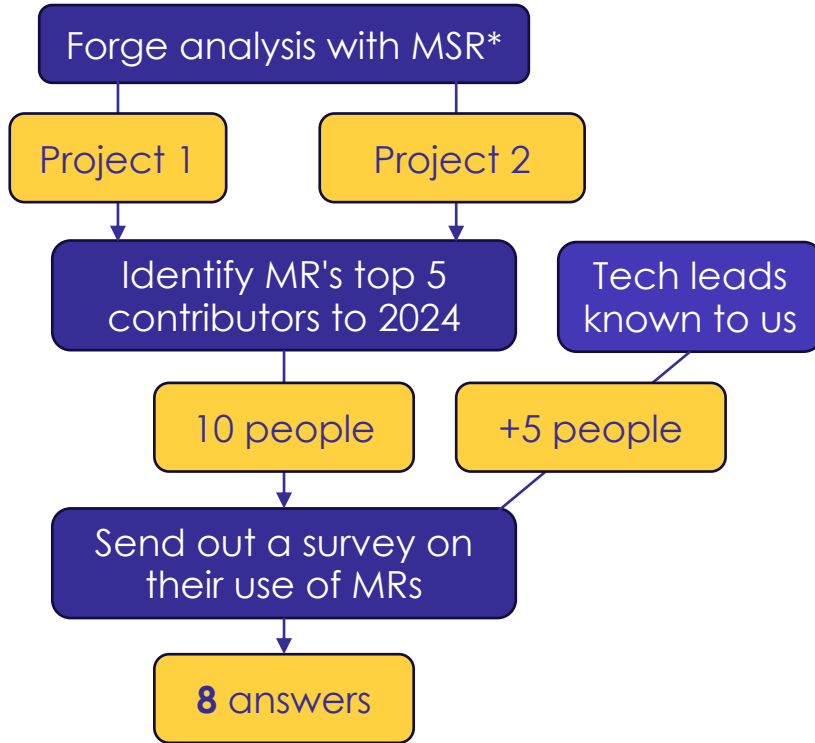
Industrial Context



Process of reviewing a Merge Request (MR)



Motivation



3-6

merge requests reviewed
per week

1-3h

spent on review per week

Recurring difficulty:

Tooling limitations

Opportunity:

**Integrated automated feedback on
code quality, tests, and static analysis**

Static Analysis in Review Workflows

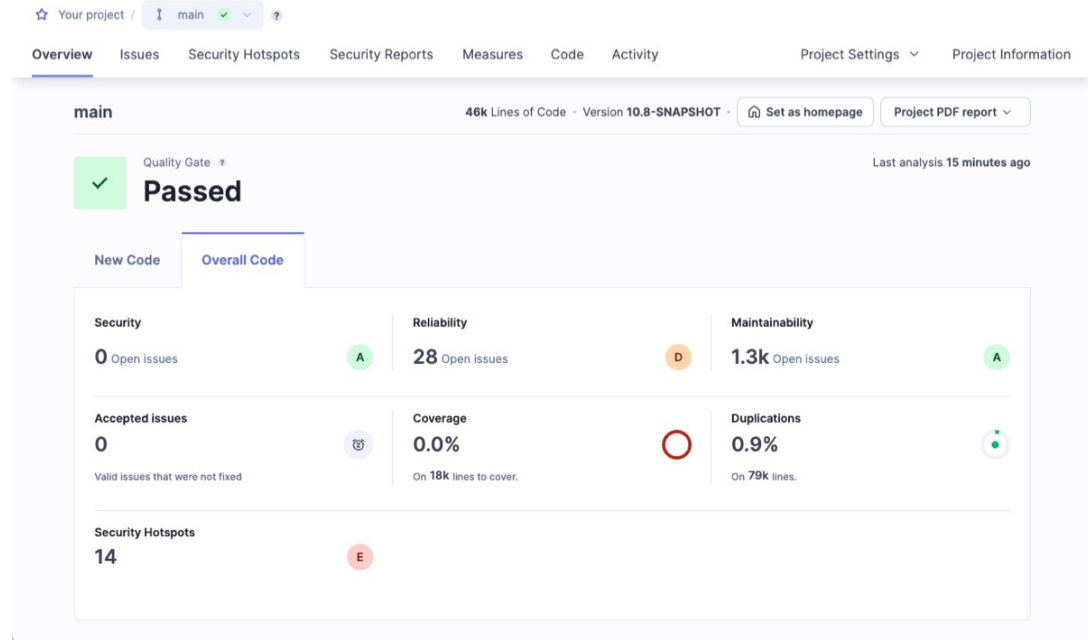
Static analysis checks source code without executing it.

It gives automated feedback before or during merge-request review.

Typical issues detected

- Bugs and reliability problems
- Security vulnerabilities
- Code smells and maintainability issues
- Duplicated code or rule violations

SonarQube



Automated Remediation and Repair Bots

1. Analysis

2. Remediations

3. Delivery

Sorald: Fixes SonarQube violations

"toString()" and "clone()" methods should not return null (Sonar Rule 2225)

For the return statements inside "toString()", this processor replaces the return expression with an empty string.

Note that this processor is incomplete and does not fix null-returning `clone()` methods.

Example:

```
public String toString() {  
    Random r = new Random();  
    if(r.nextInt(10) == r.nextInt(10)){  
        - return null; // Noncompliant  
        + return ""; // Noncompliant  
    }  
    else if(r.nextInt(10) == r.nextInt(10)){  
        return "null";  
    }  
    return "";  
}
```

Criteria

✓ **Multi-language**

✓ **Deterministic fixes**

✓ **Multi-file fixes**

✓ **Native forge suggestions**

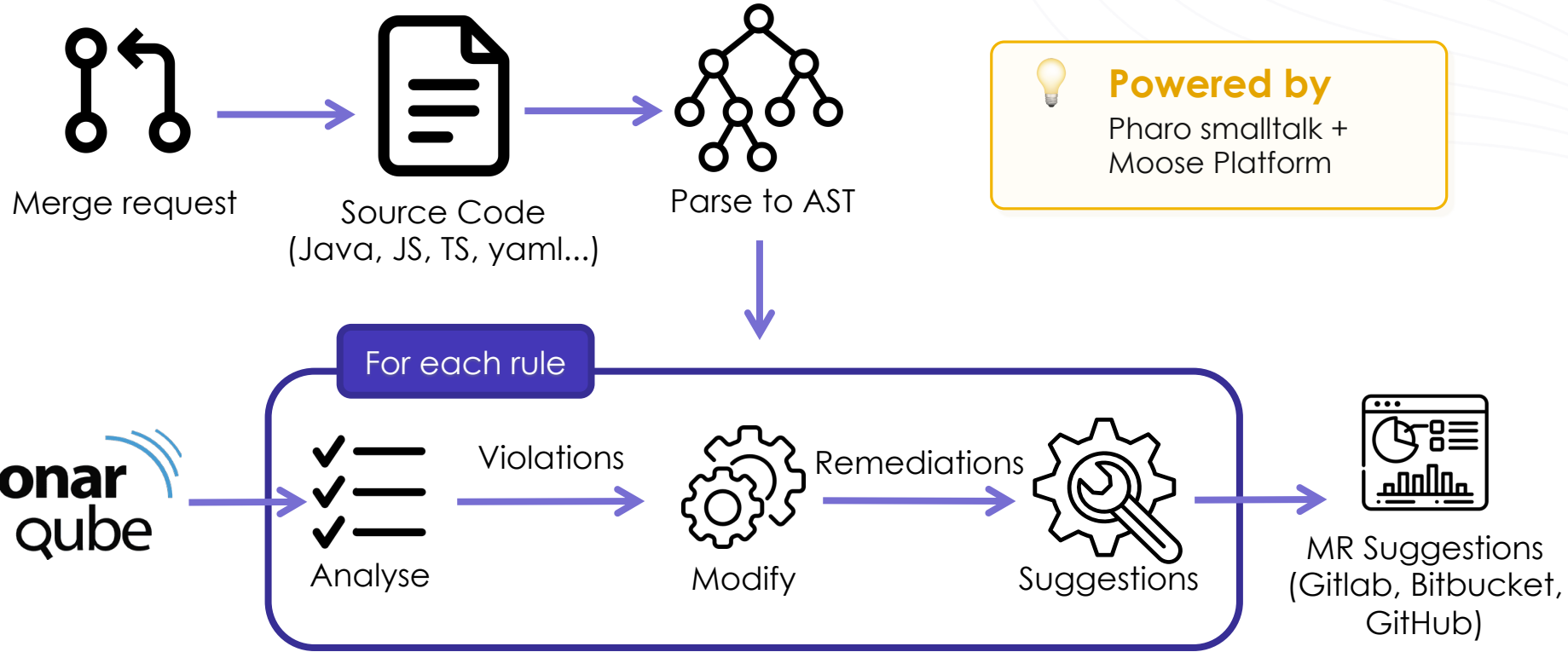
✓ **Custom rules**

✓ **External analyzer input**

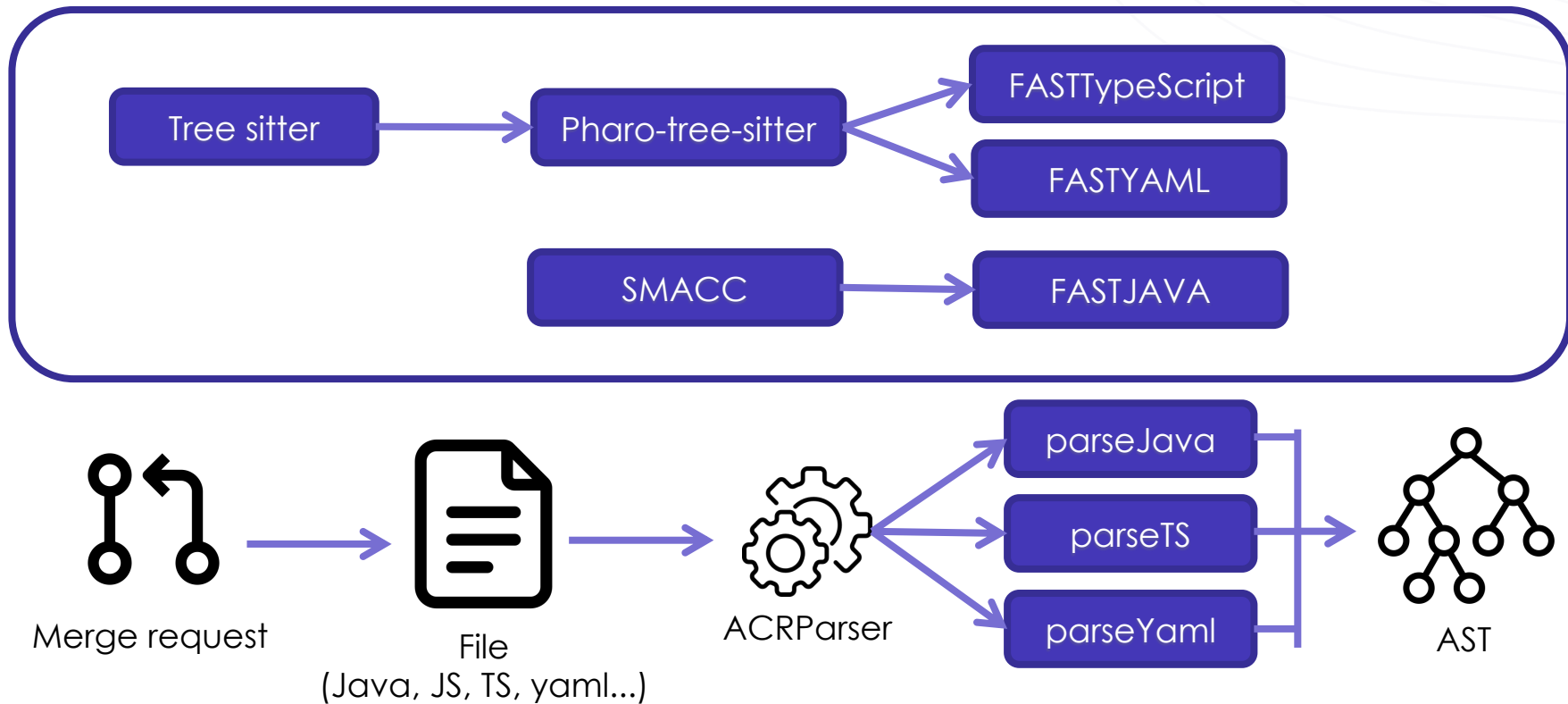
Industrial tools

Tool	Multi-language	Deterministic fixes	Coordinated multi-file fixes	Native forge suggestions	Custom rules	External analyzer input
OPENREWRITE	Yes	Yes	Yes	No	Yes	No
SONARQUBE	Yes	Partial	No	Partial	Yes	No
ESLINT	Partial	Partial	No	No	Yes	No
SEMGREP	Yes	Yes	Partial	Partial	Yes	No

AutoCodeReview Workflow



Parsing layer



Rule layer

Moose-Rule-Engine

Rule

File

Fix

Violation

SonarHelper

Analyse

```
analyse: file
  "take a MREFile and return an Collection of
  MREViolation"

  | sonarHelper |
  sonarHelper := self getHelper: 'sonarqube'.
  sonarHelper ifNil: [ ^ { } ].

  ^ sonarHelper
    violationsOf: self
    inFile: file
    usingRuleId: self ruleId
}
```

SonarqubeHelper

Fix

```
fix: violation
  "take a MREViolation and return a collection of MREFix"

  | entity file variablePattern variable |
  entity := violation entities first.

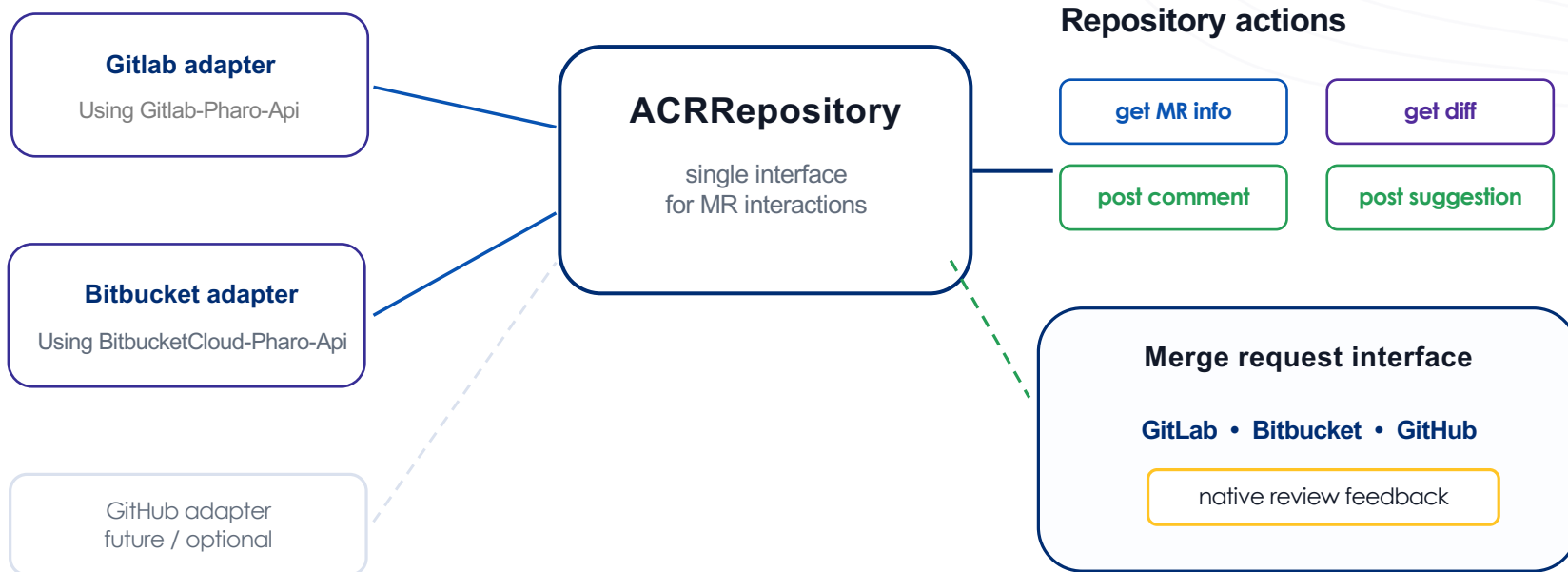
  file := entity file.

  variablePattern := FASTJavaModel
    { ('entities' <=> FASTJavaVariableExpression % {
      (#name <=> entity value).
      (#startPos <=> entity startPos).
      (#endPos <=> entity endPos) } as: #variable)
    }.
  variable := (variablePattern
    collectBinding: #variable
    for: file model) first.

  ^ JAVARenameVariableOperator new
    variable: variable;
    newName: (self camelize: variable sourceCode);
    file: file;
    applyOnFile
```

MoTion Pattern

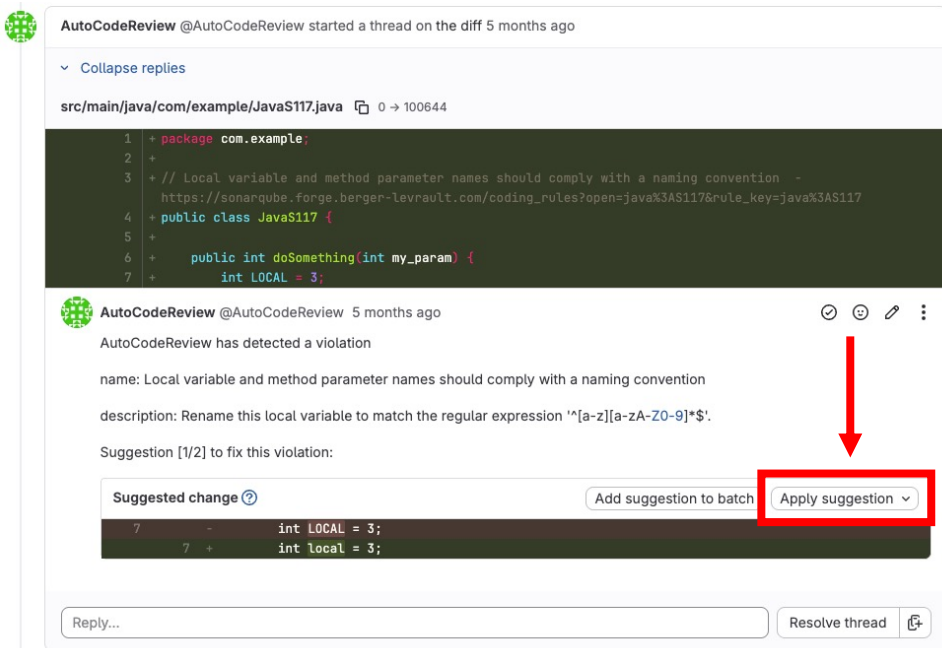
Forge layer



Key idea

AutoCodeReview stays forge-agnostic: ACRRepository hides API differences and exposes only the MR operations we need.

Example of Inline Merge-Request Comments



AutoCodeReview @AutoCodeReview started a thread on the diff 5 months ago

▼ Collapse replies

src/main/java/com/example/JavaS117.java 0 → 100644

```
1 + package com.example;
2 +
3 + // Local variable and method parameter names should comply with a naming convention -
4 + // https://sonarqube.forge.berger-levrault.com/coding_rules?open=java%3AS117&rule_key=java%3AS117
5 + public class JavaS117 {
6 +     public int doSomething(int my_param) {
7 +         int LOCAL = 3;
```

AutoCodeReview @AutoCodeReview 5 months ago

AutoCodeReview has detected a violation

name: Local variable and method parameter names should comply with a naming convention

description: Rename this local variable to match the regular expression `'^[a-z][a-zA-Z0-9]*$'`.

Suggestion [1/2] to fix this violation:

Suggested change ? Add suggestion to batch Apply suggestion ▼

```
7 - int LOCAL = 3;
7 + int local = 3;
```

Reply... Resolve thread

Suggestion Comment



AutoCodeReview ↩ ⋮

17 minutes ago

The automated review has been completed. No issues or modifications were detected according to the current analysis rules.

Simple Comment

Evaluation - Deployment Activity Metrics

- Deployed in **6 projects**
- Evaluated over **39 weeks**

Project	MRs	ACR MRs	Sugg. MRs	Comments	Sugg.	Applied
Atlas/backend	304	33	10	253	223	0
Atlas/frontend	319	44	10	161	108	17
Beacon/backend	312	169	46	567	410	0
Beacon/frontend	775	446	74	1007	493	10
Cobalt/backend	624	359	99	1741	1276	4
Cobalt/frontend	421	327	55	592	197	0
Total	2755	1378	294	4321	2707	31

Evaluation - Developer Acceptance Survey

01

Usefulness

- Improve code quality
- Help respect code convention

02

Real Usage

Consult regularly the outcome of ACR

03

Easy to use

ACR is clear and comprehensible

04

Intention

- Plan to use regularly
- Recommend to other devs

05

Attitude

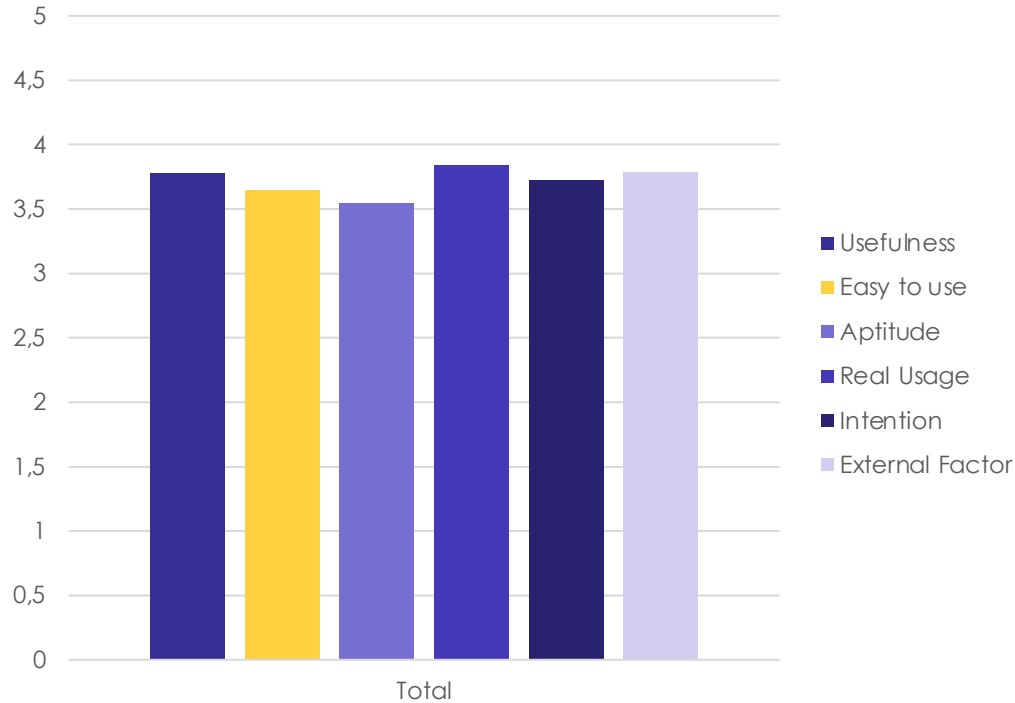
- Easy to apply suggestion
- **Difficult to configure**

06

External Variables

- Integrate well with GitLab
- My team and other devs encourage me to use ACR
- **Missing documentation**

Evaluation - Developer Acceptance Survey



Overall Score of AutoCodeReview



Why only 31 suggestions applied?

What users told us

"I read the suggestion,
then fix it in my IDE"

Why?

Our interpretation

Developers are not yet used
to native MR suggestions

Two likely causes

1

Native suggestions are recent

Gitlab beta in 2023
Bitbucket in 2025

2

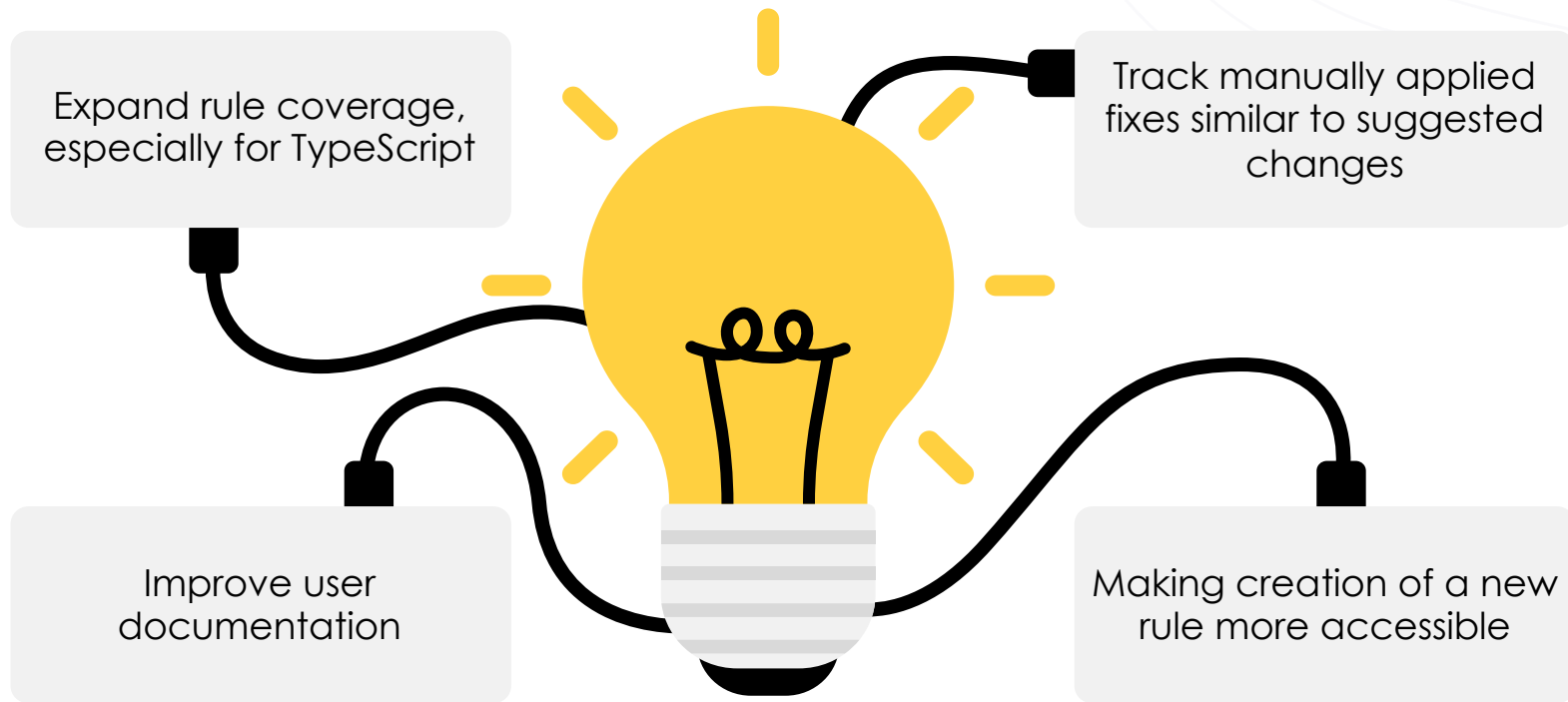
No onboarding

No user documentation for
applying suggestions



Low application may reflect unfamiliar habits, not lack of interest

Future Work



Conclusion

Tool	Multi-language	Deterministic fixes	Coordinated multi-file fixes	Native forge suggestions	Custom rules	External analyzer input
OPENREWRITE	Yes	Yes	Yes	No	Yes	No
SONARQUBE	Yes	Partial	No	Partial	Yes	No
ESLINT	Partial	Partial	No	No	Yes	No
SEMGREP	Yes	Yes	Partial	Partial	Yes	No
AutoCodeReview	Yes	Yes	Yes	Yes	Yes	Yes

- **2 755 MRs** analyzed on **6 projects** over **39 weeks**
- **4 321 comments** including **2 707 suggestions**
- **31 applied suggestions***