

PM D TOOL

GROUP
N
- PRESENTATION -



MEMBERS

SE172607 - Nguyễn Nhật Phát

SE170526 - Trịnh Quốc Thái

SE172565 - Bùi Thanh Tú

SE172605 - Hoàng Văn Anh Nghĩa

SE170092 - Nguyễn Việt Nam Khánh

-OVERVIEW-

WHAT IS PMD TOOL?

PMD IS AN OPEN-SOURCE STATIC CODE ANALYSIS TOOL PRIMARILY USED FOR IDENTIFYING POTENTIAL ISSUES IN JAVA CODEBASES. IT OFFERS CUSTOMIZABLE RULE SETS, INTEGRATION WITH VARIOUS BUILD TOOLS AND IDES, AND SUPPORTS MULTIPLE PROGRAMMING LANGUAGES BEYOND JAVA, INCLUDING JAVASCRIPT, XML, SQL,...

-OVERVIEW-

WHY IS PMD TOOL?

Code Quality Improvement

Early Issue Detection

Consistency

Automated Feedback

Integration with Development Workflow

-OVERVIEW-

WHO IS CREATED PMD TOOL?



**Name:
TOM
COPELAND**

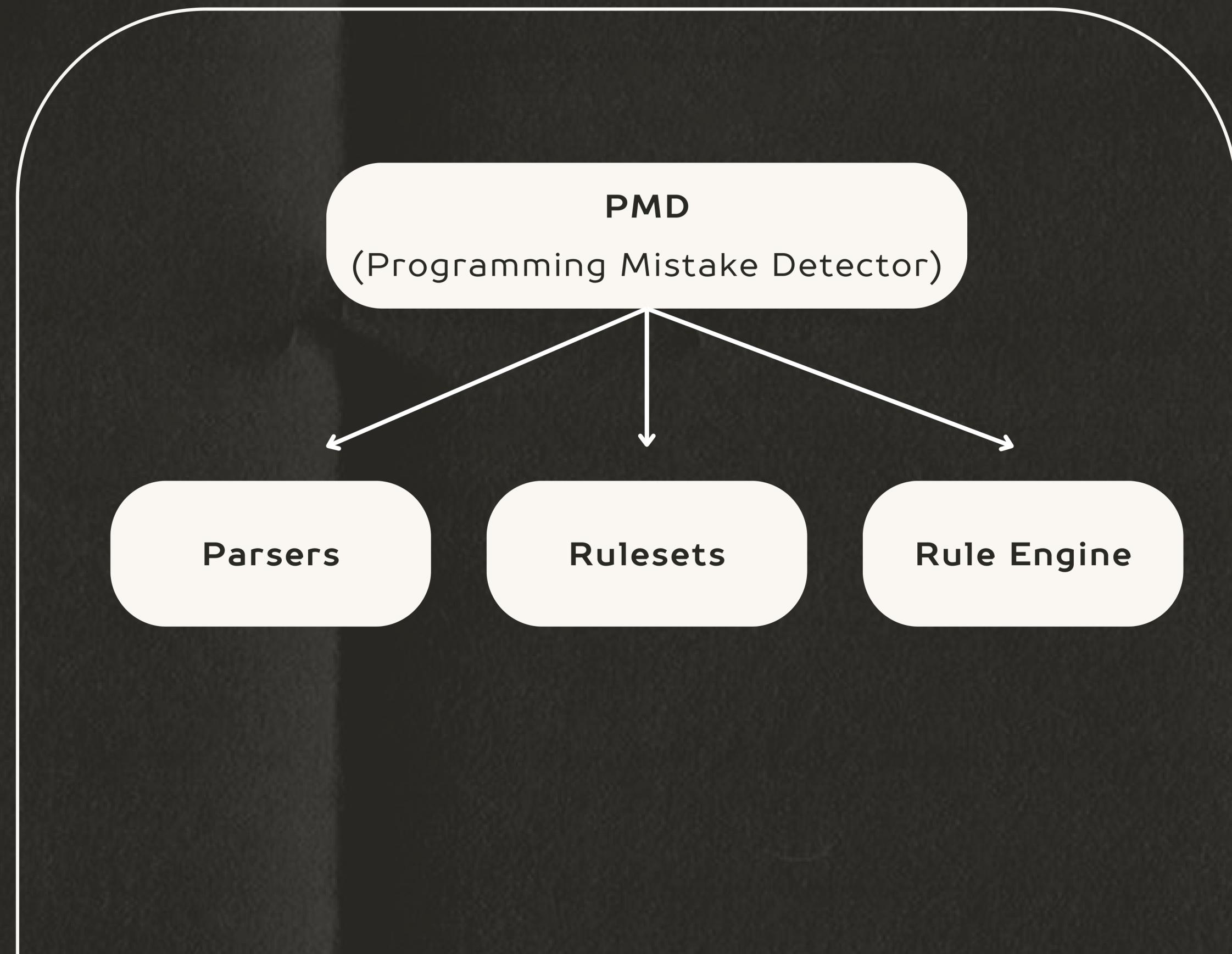
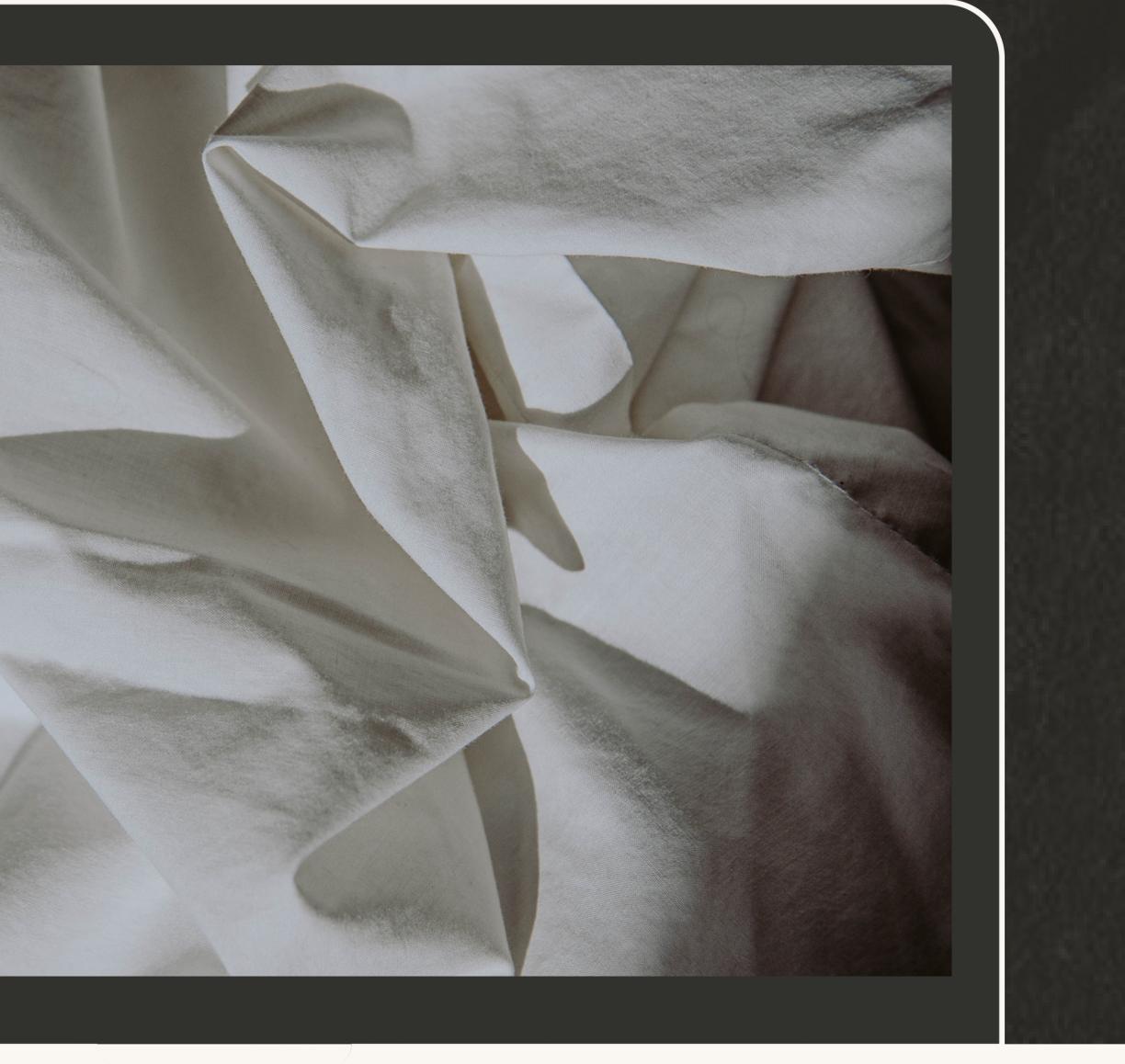
PMD was created by a team of software developers, including Tom Copeland, Xavier Le Vourch, and Ryan Gustafson. Initially, Tom Copeland was the primary developer and maintainer of the PMD project. However, the project has received contributions from a large community of programmers and continues to be developed and maintained by the open-source community.

-OVERVIEW-

CUSTOMER REFERENCE

- Software Developers
- Project Managers
- Testers
- Open Source Community

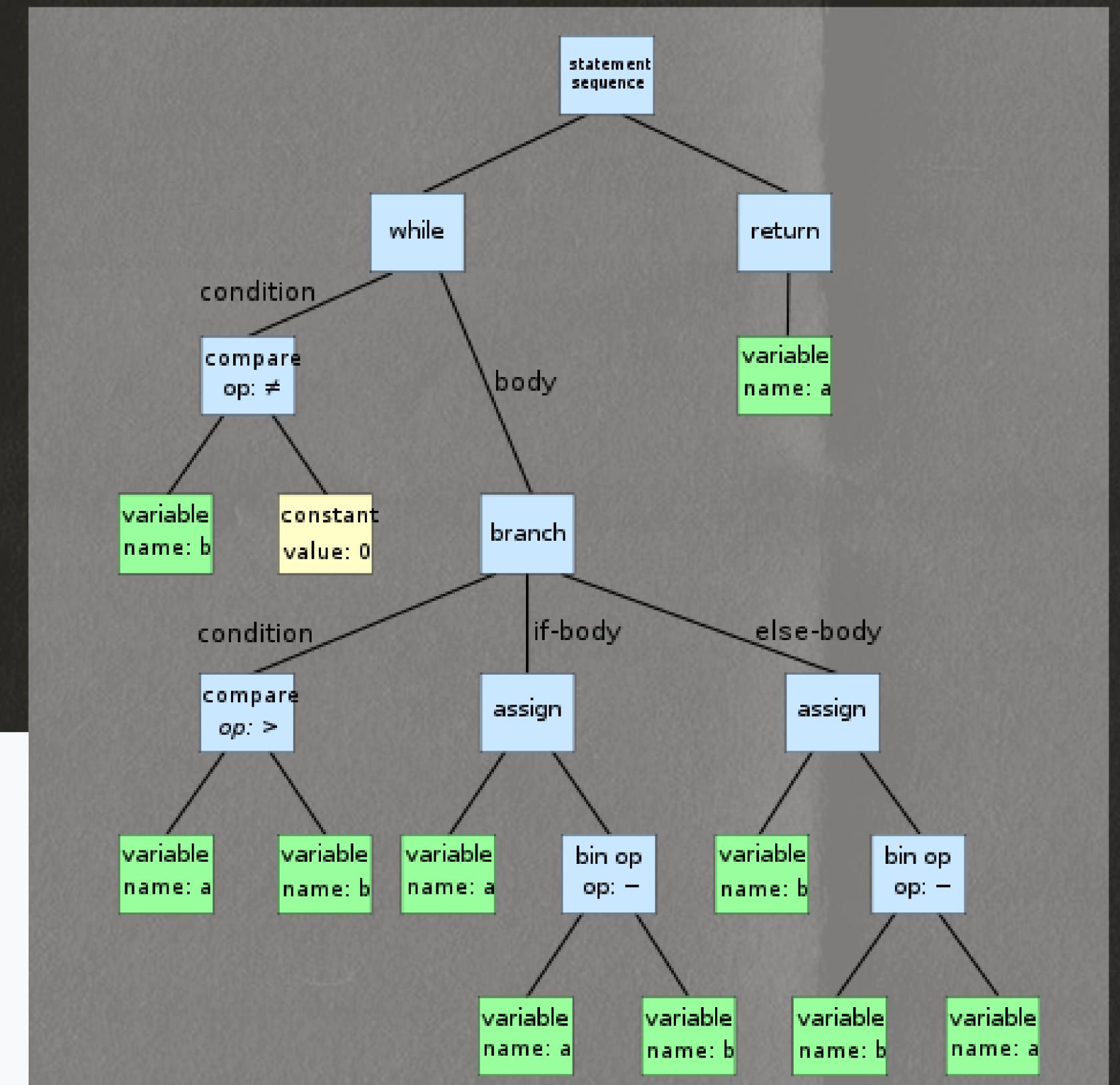
ARCHITECTURE



PARSERS

- Analyzing the source code ==> creating an abstract syntax tree (**AST**).
- The AST represents the structure of the code.
- PMD uses the AST to perform rule checks and identify potential issues.

```
while b != 0:  
    if a > b:  
        a := a - b  
    else:  
        b := b - a  
return a
```



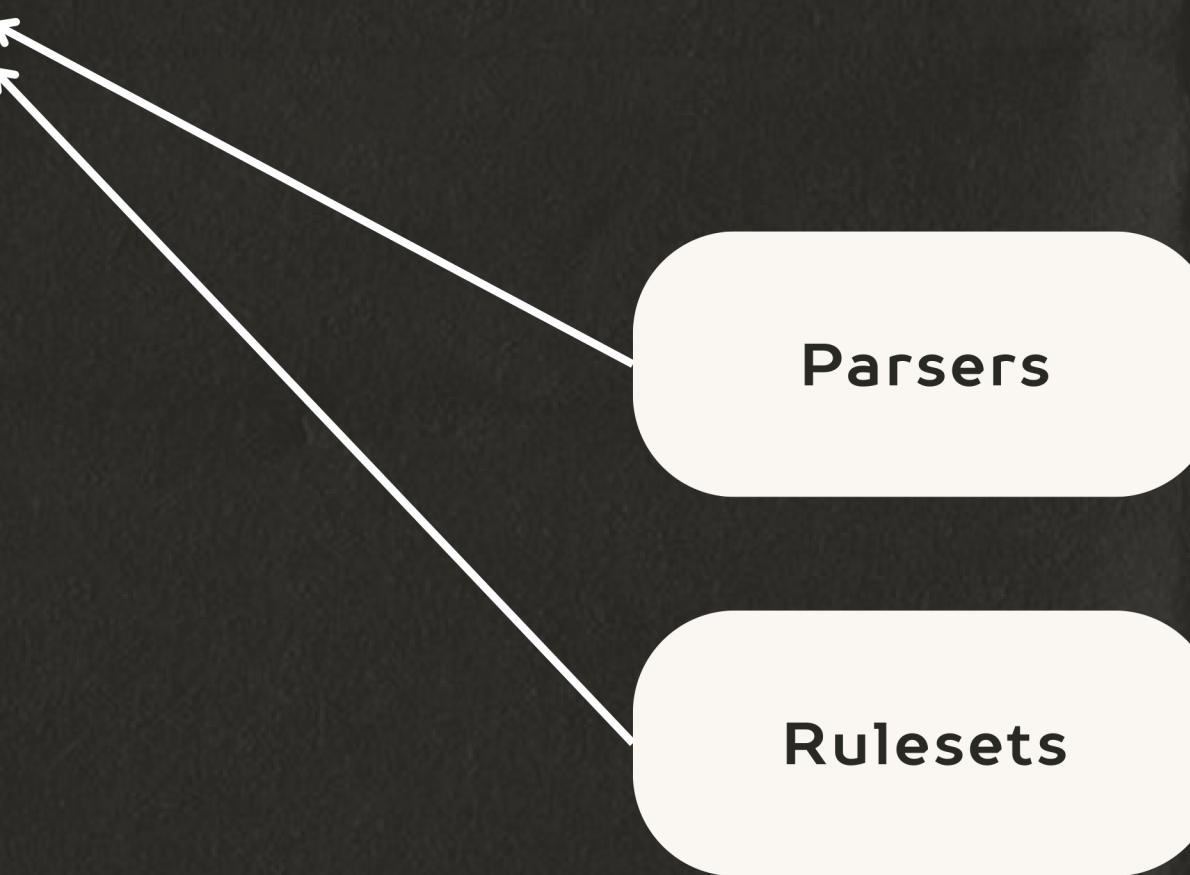
RULESETS

- A ruleset is a collection of individual rules that define specific coding standards or best practices.
- PMD allows you to configure which rules to apply by specifying a ruleset.
- You can create custom rulesets tailored to your project's requirements.

- `basic.xml`
- `empty.xml`
- `imports.xml`
- `unnecessary.xml`
- `unusedcode.xml`

RULE ENGINE

- The Rule Engine is the heart of PMD.
- It processes the Abstract Syntax Tree (AST) generated from the source code.
- Its primary function is to apply the rules defined in the selected rulesets to the AST



HIGHLIGHT OF PMD

- Code Quality Analysis

identify common programming flaws, such as unused variables, empty catch blocks, and unnecessary object creation.

- Cross-Language Support

including Java, JavaScript, Salesforce.com Apex, PLSQL, and more.

- Customizability

create custom rules

PROS

- Automated Code Review
- Integration with Build Process
- Community and Open Source

CONS

- False Positives
- Learning Curve

DETAILS OF FEATURES

+ Static Code Analysis

+ Rule-Based Analysis

+ Support for Java

+ Detects Code Smells

+ Customizable Rules

+ Integration with Build Tools

+ Command-Line and IDE Integration

+ Reports and Output Formats

+ Continuous Integration Support

+ Active Community and Updates



DEMO 1

PMD Tool With Maven Project

DEMO 2

PMD Tool With Java Project

BORCELL
E
ORCELLE

THANK YOU

PMD TOOL

- GROUP 2 -