UNIVERSITY OF BERN

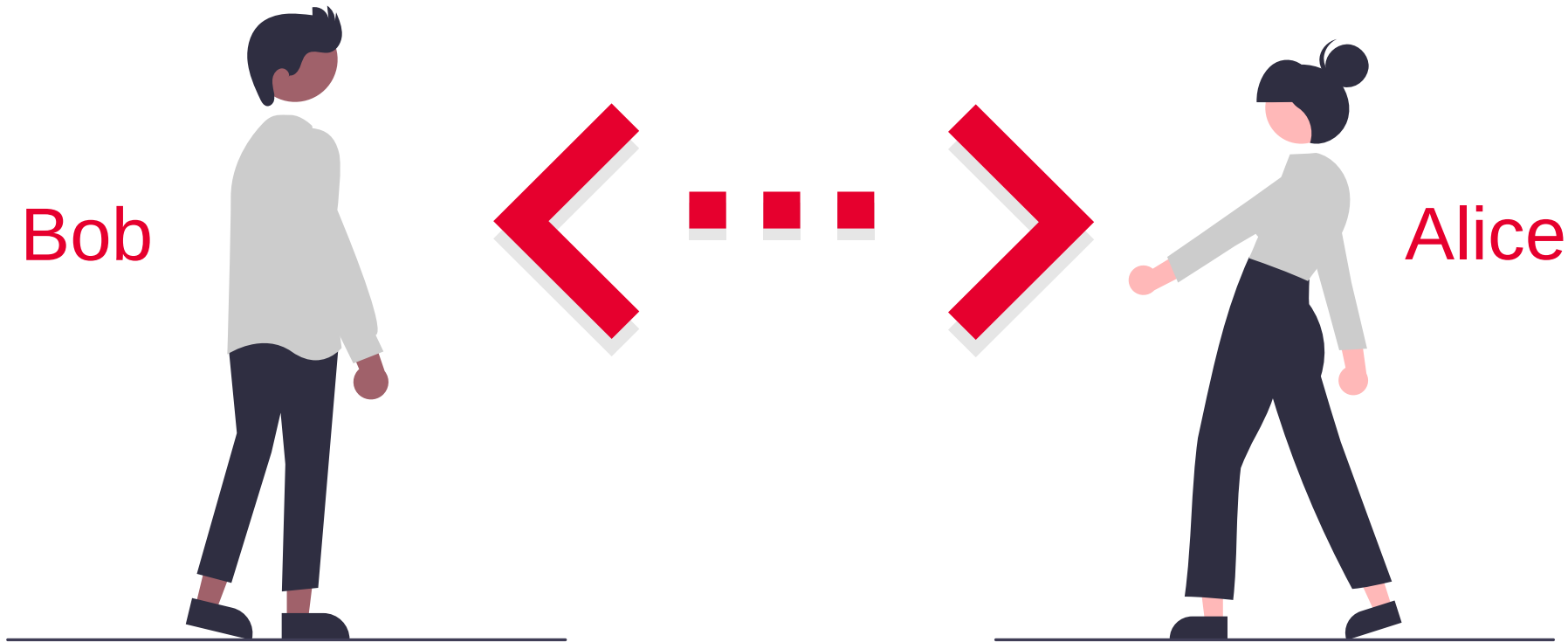
Exploring Platform-independent Code Linting

Noah Bühlmann

Software Composition Seminar

December 7, 2021

Motivation

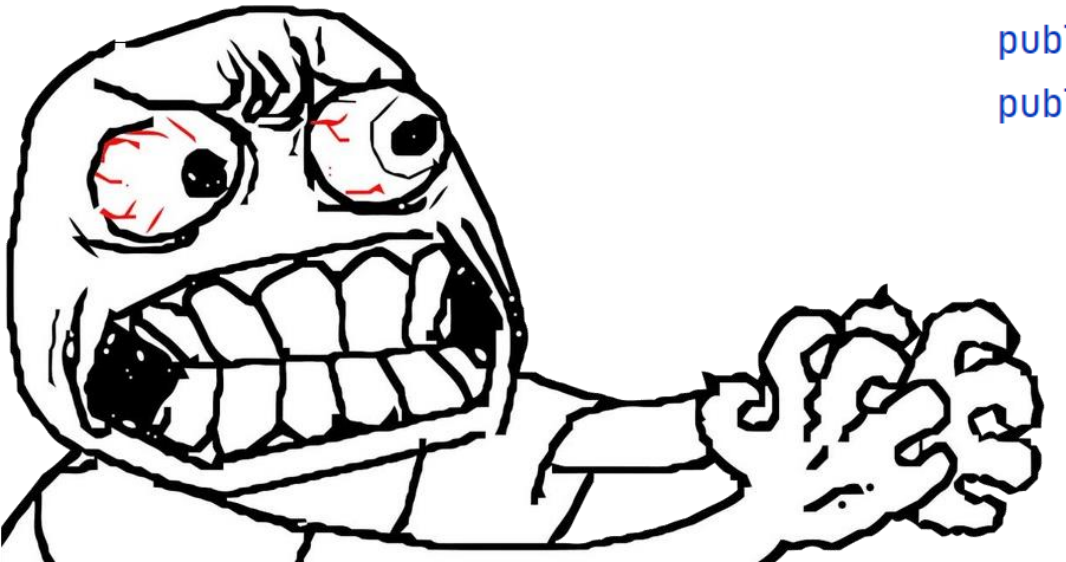


Bob has a meltdown

Bob: «Alice! Didn’t we agree on lowerCamelCase?!»

```java
public class AliceClass {

    public String AliceString = "Hello World!";
    public int AliceInteger = 42;
    public boolean AliceBool = true;
```



Goal: Easy platform-independent code linting

Step 1 - Plug-in installation:

Download and install Jetbrains IntelliJ plug-in

Step 2 - Plug-in configuration:

Create configuration file with linting rules in XML

Syntax is loosely based on srcML

Source control!

Example 1/3 – Literal matcher

```
<linter name="Issue 001: 'Hello World' in production releases" language="Java">
    <note>This linter highlights String literals that match the regex "Hello World".</note>
    <severity>WARNING</severity>
    <matcher type="literal">
        <literal>Hello World</literal>
    </matcher>
</linter>
```

Example 2/3 – Method matcher

```
<linter name="Issue 003: Method highlighting" language="Java">
    <note>This linter highlights methods that match the given return type, name, modifiers and parameter set.</note>
    <severity>INFORMATION</severity>
    <matcher type="method">
        <method returnType="void" name="myMethodName">
            <modifier>static</modifier>
            <modifier>synchronized</modifier>
            <parameter type="int" />
            <parameter type="java.lang.String" />
        </method>
    </matcher>
</linter>
```

Example 3/3 – Class matcher

```
<linter name="Issue 004: Class highlighting" language="Java">
    <note>This linter highlights classes that match the given name, interfaces, and methods.</note>
    <severity>WARNING</severity>
    <matcher type="class">
        <class name="myClass" superClass="mySuperClass">
            <interface name="myInterface1" />
            <interface name="myInterface2" />
            <method returnType="void" name="myMethod1">
                <modifier>synchronized</modifier>
                <parameter type="int" />
            </method>
            <method returnType="void" name="myMethod2"></method>
        </class>
    </matcher>
</linter>
```

Live demo

X-Platform Java Linter

Install

35 Noah Bühlmann

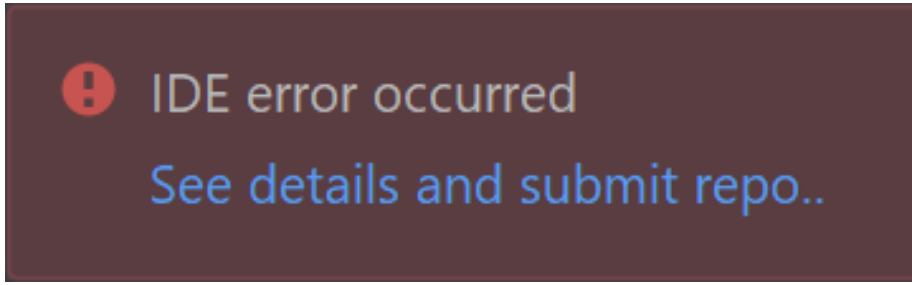
1.1.0 Nov. 26, 2021

Plugin homepage ↗

This IntelliJ-based Plugin allows you to define custom platform-independent linters in a linter.xml config file in the .idea folder of your project. This plugin was built as a seminar project for the Software Composition Group at the University of Bern and serves as proof of concept.

▸ Change Notes

Challenges



IntelliJ

APIs change continuously

SDK dependencies

Lack of documentation

srcML

Mapping between original source code nodes and srcML

Java

Run time Java code manipulation

Future work

Adoption of the entire srcML specification

Custom Annotator implementations

Support for other IDEs

Conclusion

It works and it is customizable...

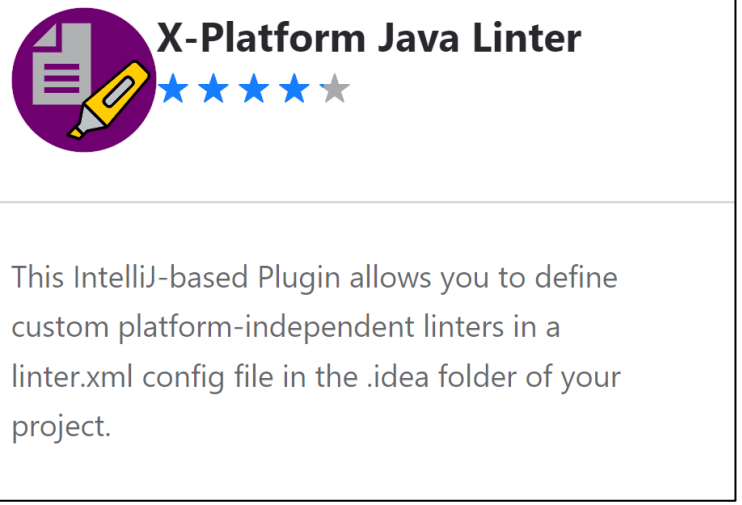
... and IDE plug-in development is hard, but doable.

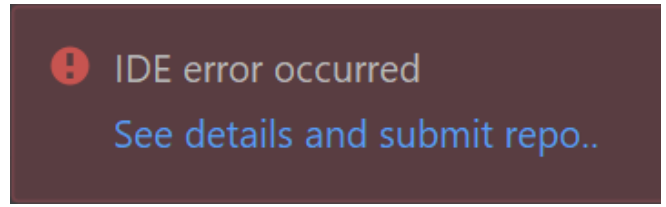
Summary

```java
public class AliceClass {

    public String AliceString = "Hello World!";
    public int AliceInteger = 42;
    public boolean AliceBool = true;
```

Get from Marketplace






```xml
<linter name="Issue 001: 'Hello World' in production releases" language="Java">
    <note>This linter highlights String literals that match the regex "Hello World".</note>
    <severity>WARNING</severity>
    <matcher type="literal">
        <literal>Hello World</literal>
    </matcher>
</linter>
```