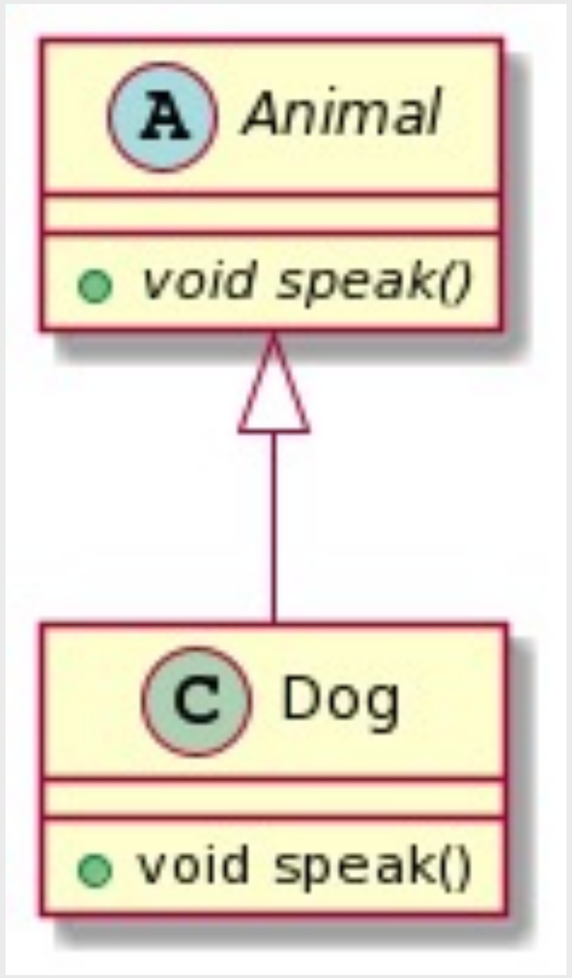
CPSC 5011: Object-Oriented Concepts

Text-to-Diagram Tools
(A Brief Digression)

Text-to-Diagram

- (My term, not industry)
- Like a compiler that generates images, not apps

```
@startuml

abstract class Animal {
{abstract} + void speak()
}

class Dog {
+ void speak()
}

Animal <|-- Dog

@enduml
```

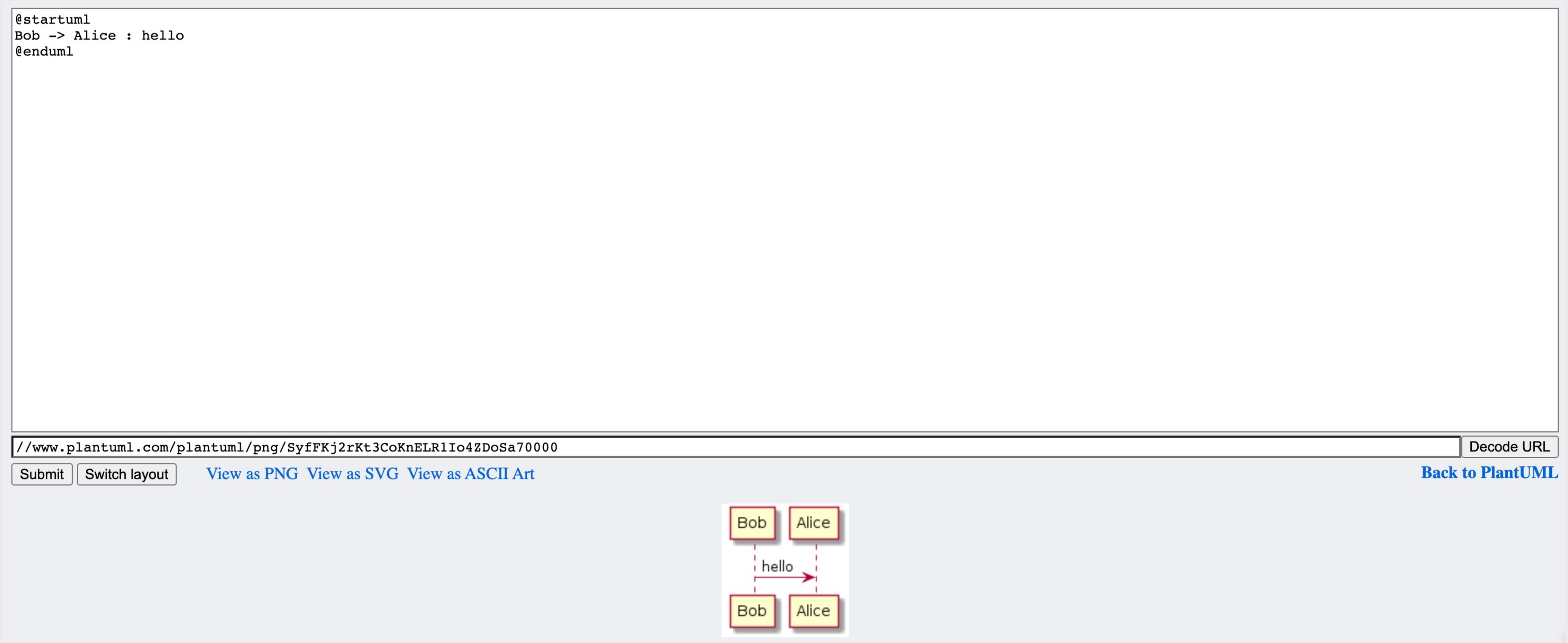


PlantUML

- PlantUML can do all* of UML
- Decent syntax
- Obnoxious runtime (have to run a server) (!!!)
- BUT has good plugins for IntelliJ, vscode, etc.

PlantUML online

- Plantuml.com
- …click on "Online Server" on the left
- Type in code and click "Submit"



```
@startuml
Bob -> Alice : hello
@enduml
```

//www.plantuml.com/plantuml/png/SyfFKj2rKt3CoKnELR1Io4ZDoSa70000

PlantUML for class diagrams

- @startuml
- Class syntax can be simple or very detailed
- Lines and arrows after class declarations
- @enduml

```
@startuml

abstract class Animal {
{abstract} + void speak()
}

class Dog {
+ void speak()
}

Animal <|-- Dog

@enduml
```

Bottom line: doesn’t have to be “good” Java—use this as pseudo code *and* get a diagram!

Code-generating your UML

- Several IntelliJ plugins available for Java --> PlantUML
  - (All have quirks)

- My approach
  - Using a language lib (ANTLR)…
  - …plus a pre-existing Java grammar…
  - Write a program to “hook” parts of the language to extract classes, interfaces, members, etc.

Java grammar

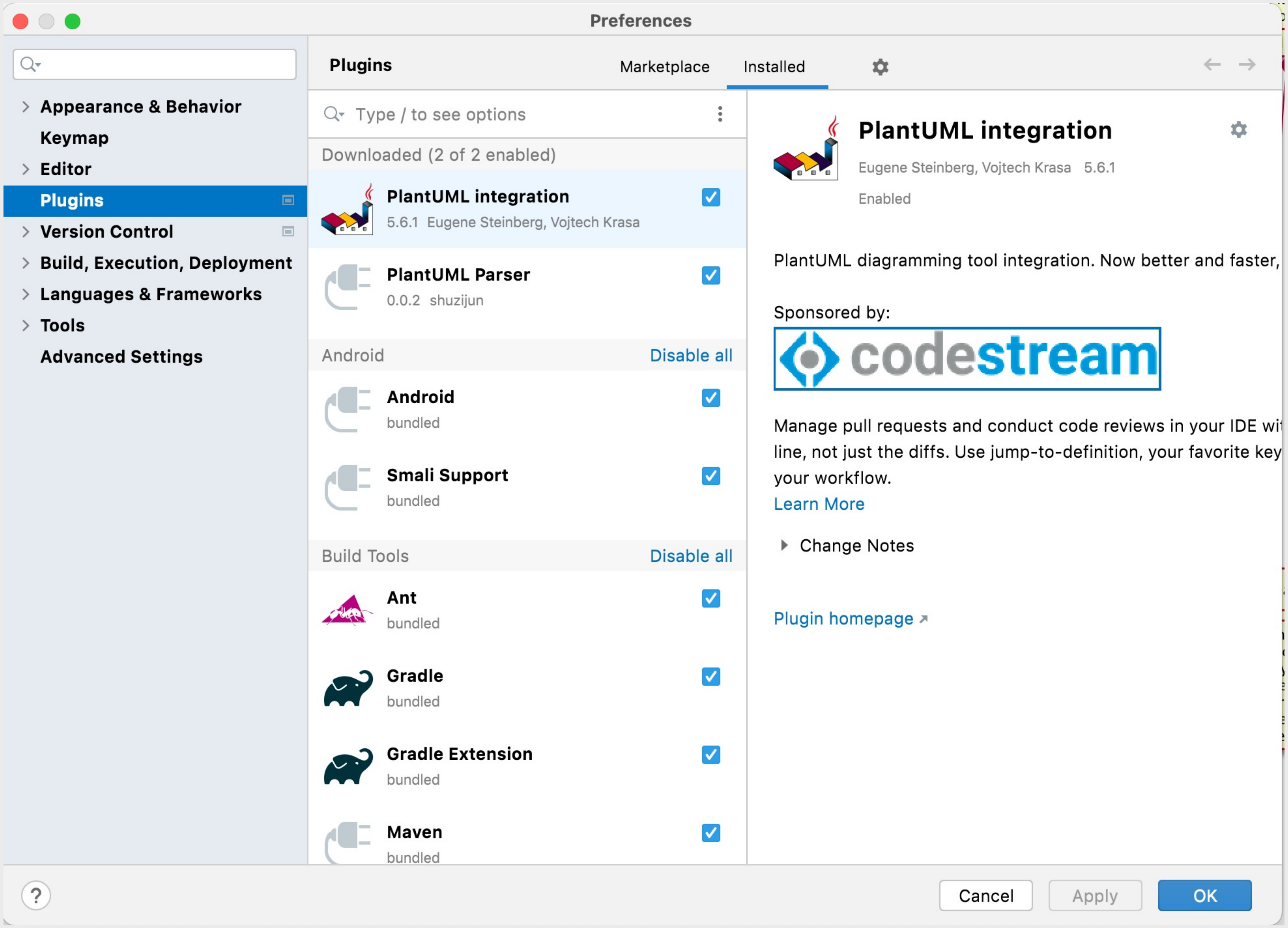
```
normalClassDeclaration
      :     classModifier* 'class' identifier typeParameters?
            superclass? superinterfaces? classBody
      ;

superclass
      :     'extends' classType
      ;

superinterfaces
      :     'implements' interfaceTypeList
      ;

interfaceTypeList
      :     interfaceType (',' interfaceType)*
      ;

classBody
      :     '{' classBodyDeclaration* '}'
      ;

classBodyDeclaration
      :     classMemberDeclaration
      |     instanceInitializer
      |     staticInitializer
      |     constructorDeclaration
      ;
```

Code-generating your UML

ANTLR uses the Visitor pattern so youcan grab just the parts you care about.

```
public class ClassExtractor : Java9ParserBaseVisitor<bool>
{
     public Dictionary<string, ClassDef> ClassDefs { get; } = new();

     public ClassDef CurrentClass { get; set; }

     public override bool VisitNormalClassDeclaration(…) { ... }

     public override bool VisitNormalInterfaceDeclaration(…) { ... }

     public override bool VisitEnumDeclaration(…) { ... }

     public override bool VisitMethodHeader(…) { ... }

     public override bool VisitFieldDeclaration(…) { ... }
}
```

PlantUML plugin for IntelliJ



Preferences

Downloaded (2 of 2 enabled)

PlantUML integration 5.6.1 Eugene Steinberg, Vojtech Krasa ✓

PlantUML Parser 0.0.2 shuzijun ✓

Android — Disable all

Android bundled ✓

Smali Support bundled ✓

Build Tools — Disable all

Ant bundled ✓

Gradle bundled ✓

Gradle Extension bundled ✓

Maven bundled ✓

PlantUML integration

Eugene Steinberg, Vojtech Krasa 5.6.1

Enabled

PlantUML diagramming tool integration. Now better and faster,

Sponsored by:

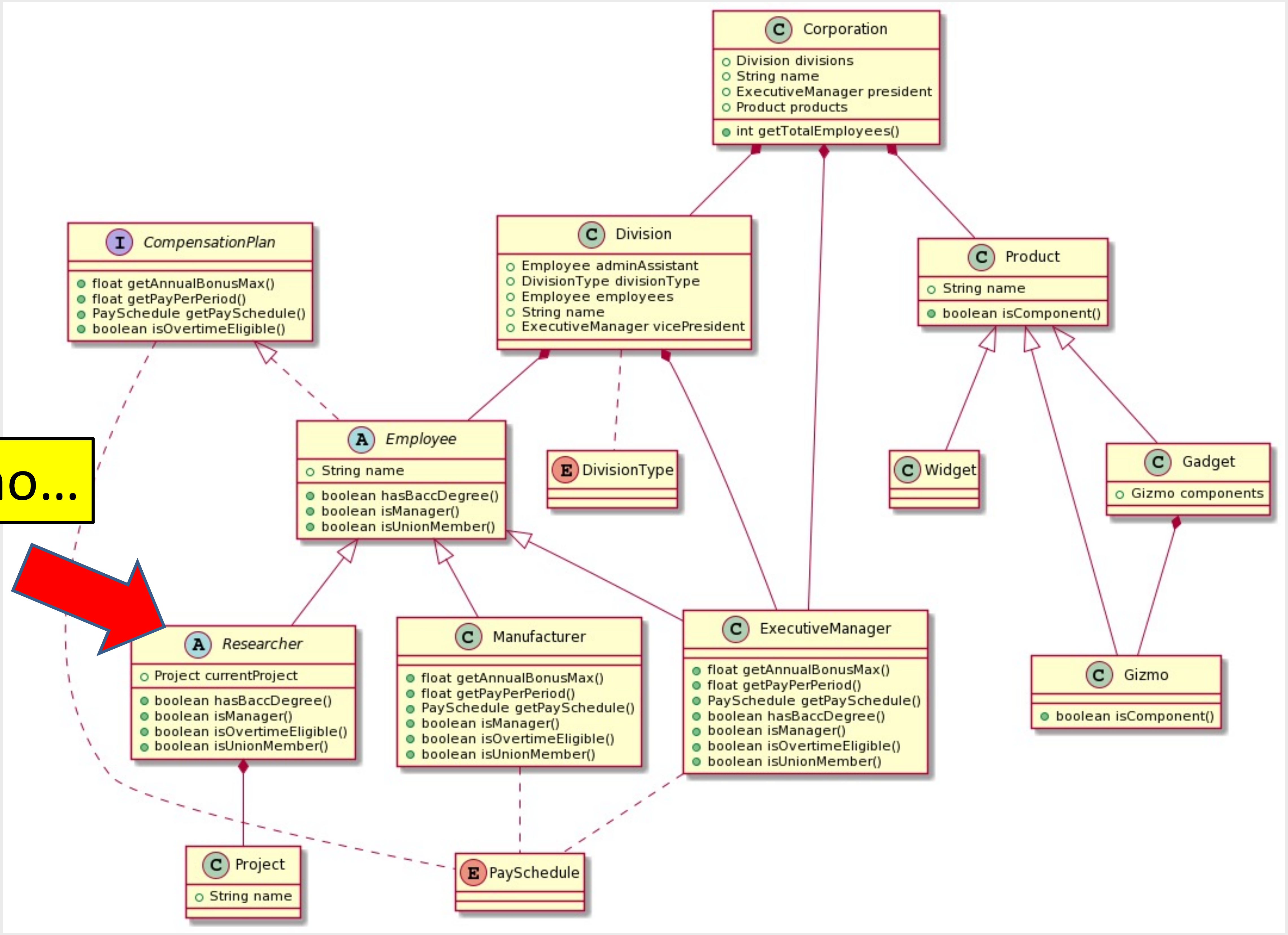
codestream

Manage pull requests and conduct code reviews in your IDE wi
line, not just the diffs. Use jump-to-definition, your favorite key
your workflow.
Learn More

Change Notes

Plugin homepage ↗

Final result



Demo…

JavaPlantUmlGenerator

- Github repo: https://github.com/stephen-riley/JavaPlantUmlGenerator
- ANTLR generates 55,000 lines of code
- Our part is ~200 ☺