Расширение статического анализа кода Java на основе пользовательских аннотаций

Extending Static Analysis of Java Code Based on User Annotations

Выполнил: Пасилецкий Даниил Олегович

Руководитель: Профессор Базовая кафедра «Системное программирование» ИСП РАН, факультета компьютерных наук, Белеванцев Андрей Андреевич

Консультант: Старший лаборант ИСП РАН, Афанасьев Виталий Олегович

Основные термины, понятия и определения

аннотаций

Расширение статического анализа кода Java на основе пользовательских

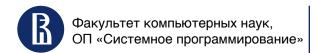
Статический анализ кода – анализ исходного кода на предмет ошибок и недочётов без непосредственного выполнения анализируемых программ.

Java – анализ исходного кода на предмет ошибок и недочётов без непосредственного выполнения анализируемых программ.

Svace – анализ исходного кода на предмет ошибок и недочётов без непосредственного выполнения анализируемых программ.

Java Annotations – это специальная форма синтаксических метаданных, которая может быть добавлена в исходный код.

Моделирование – это специальный подход при котором исходный класс заменяется на его упрощенную модель, с которой умеет работать анализатор



Проблема

Анализатор не всегда может верно работать, иногда из-за ограничений на потребляемые время и память приходится что-то упрощать, а иногда код очень непонятный (либо вообще исходники отсутствуют). И одно из решений - это предоставить пользователю механизм, который подсказывает что-то анализатору.

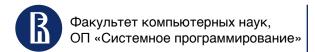
```
public class SalaryCalculator {

public static double calculateAverageDailySalary(double totalSalary, int totalDays) {
    return totalSalary / totalDays;
}

public static void main(String[] args) {
    double dailySalary = calculateAverageDailySalary(50000, Calendar.getWorkingDays());
}

}
```

аннотаций



Постановка задачи

Необходимо расширить возможности статического анализатора Svace, таким образом что бы пользователи могли предоставлять дополнительную информацию анализатору, путем добавления специальных аннотаций в исходном коде анализируемой программы.

```
public class SalaryCalculator {
    public static double calculateAverageInDaySalary(@Range(min = 28, max = 31) int dayInMonths, double salary) {
        return salary / dayInMonths;
    }
}
```

Java на основе пользовательских

аннотаций

Моделирование

Используется специальный класс который перекрывает настоящую реализацию Calendar

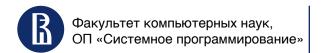
```
public class Calendar {
   public static int getWorkinkDays() {
    int number = Spec.getAnyInt();
   number.range(28,31);
   return number;
```

Основные проблемы

```
public class SalaryCalculator {
       public static double calculateAverageDailySalary(double totalSalary, @Range(min=28, max=32) int totalDays) {
           return totalSalary / totalDays;
       public static void main(String[] args) {
           double dailySalary = calculateAverageDailySalary(50000, 0);
10
```

аннотаций

учитывать их при анализе кода.



План

	Определить набор аннотаций, которые могут быть использованы пользователям
	Улучшить работу статического анализатора с аннотациями
Ш	Разработать пакет с набором аннотаций, которые используются для расширения формации анализатора.

Доработать статический анализатор для реагирование на эти аннотации и

Набор аннотаций

29

аннотаций

Расширение статического анализа кода

Java на основе пользовательских

аннотаций без учета перегрузок

Можно помечать аннотациями:

- Параметры функций
- Поля
- Методы

@Sensitive

@Tainted

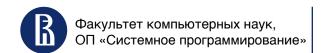
@FunHash

@Leaked

@NotNull

@FunPrintfLike

аннотаций



Как измерить результат

Количество замененных спецификации на пользовательский код

• Количество использований аннотаций пользователями

• Время затрачиваемое на добавление новой аннотации

Расширение статического анализа кода Java на основе пользовательских аннотаций

Extending Static Analysis of Java Code Based on User Annotations

Выполнил: Пасилецкий Даниил Олегович

Руководитель: Профессор Базовая кафедра «Системное программирование» ИСП РАН, факультета компьютерных наук, Белеванцев Андрей Андреевич

Консультант: Старший лаборант ИСП РАН, Афанасьев Виталий Олегович