

Using results of static analysis in the KeY tool

Max Schroeder

Table of Contents

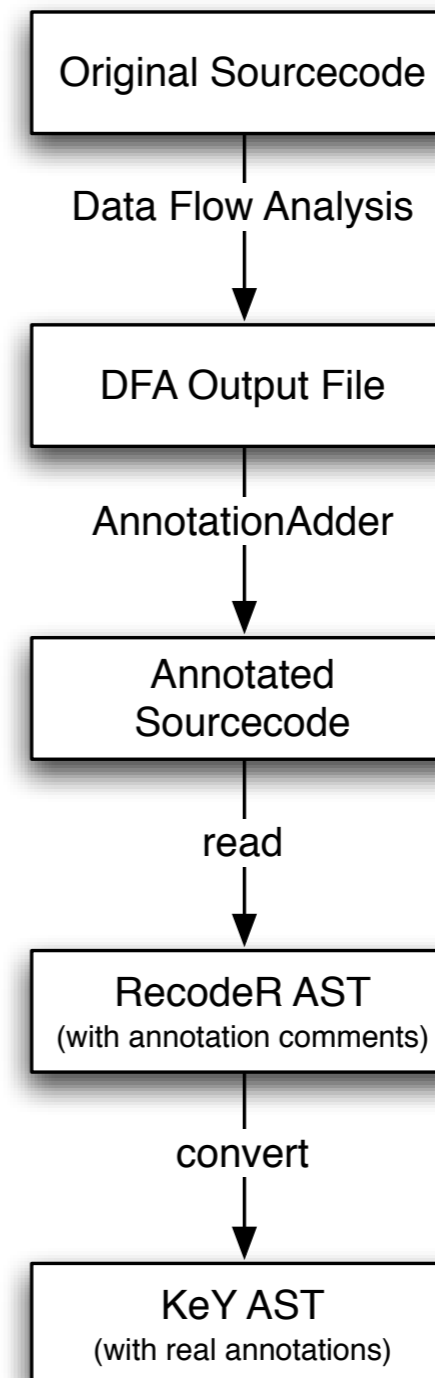
- Introduction
- Get the Information
- Type of Annotations
- How to make use of it
- Notes and Perspectives

Introduction

- Static Analysis is a well known issue
- It gives additional Information about our Java Model
- This information may help avoid unnecessary Branches
- KeY is not able to do static analysis (yet)
- Instead the aicas tool (JamaicaVM) is used

The “interface”

At the moment, annotations are read as comments from the Java source file



Syntax of Annotations

`AnnotationComment ::= '/*DFA' IDENT (, IDENT)* 'DFA*/'`

Examples:

- ▶ `a.o = this./*DFA possible_null DFA*/o;`
- ▶ `i = x[/*DFA non_null,index_in_bounds DFA*/42];`

Supported Program Elements

- Method References

```
c./*DFA non_null DFA*/m()
```

- Field References

```
a./*DFA non_null DFA*/o
```

- Array References

```
x[/*DFA non_null */5]
```

Semantic of Annotations

- `non_null`
 - won't throw a `NullPointerException`
- `index_in_bounds`
 - won't throw a `IndexOutOfBoundsException`
- `possible_null`
 - might throw a `NullPointerException`

Types of Annotation

non required	required
<ul style="list-style-type: none">• non_null• index_in_bounds	<ul style="list-style-type: none">• possible_null
Additional information. Might be skipped	Necessary information. Must not be skipped!

Example of application in a tactic rule

```
\find (\modality{#allnormalass}  
{..#v.#a=#se;...}\endmodality(post))  
\sameUpdateLevel  
\varcond(\not \static(#a))
```

```
"Normal Execution (#v != null)":  
\replacewith({#v.#a := #se}\modality{#allnormalass}{.. ..})  
\endmodality(post))  
\add (==>(#v=null));
```

```
"Null Reference (#v = null)":  
\replacewith(\modality{#allnormalass}{..throw new  
java.lang.NullPointerException();...}\endmodality(post))  
\add (#v=null ==>)
```

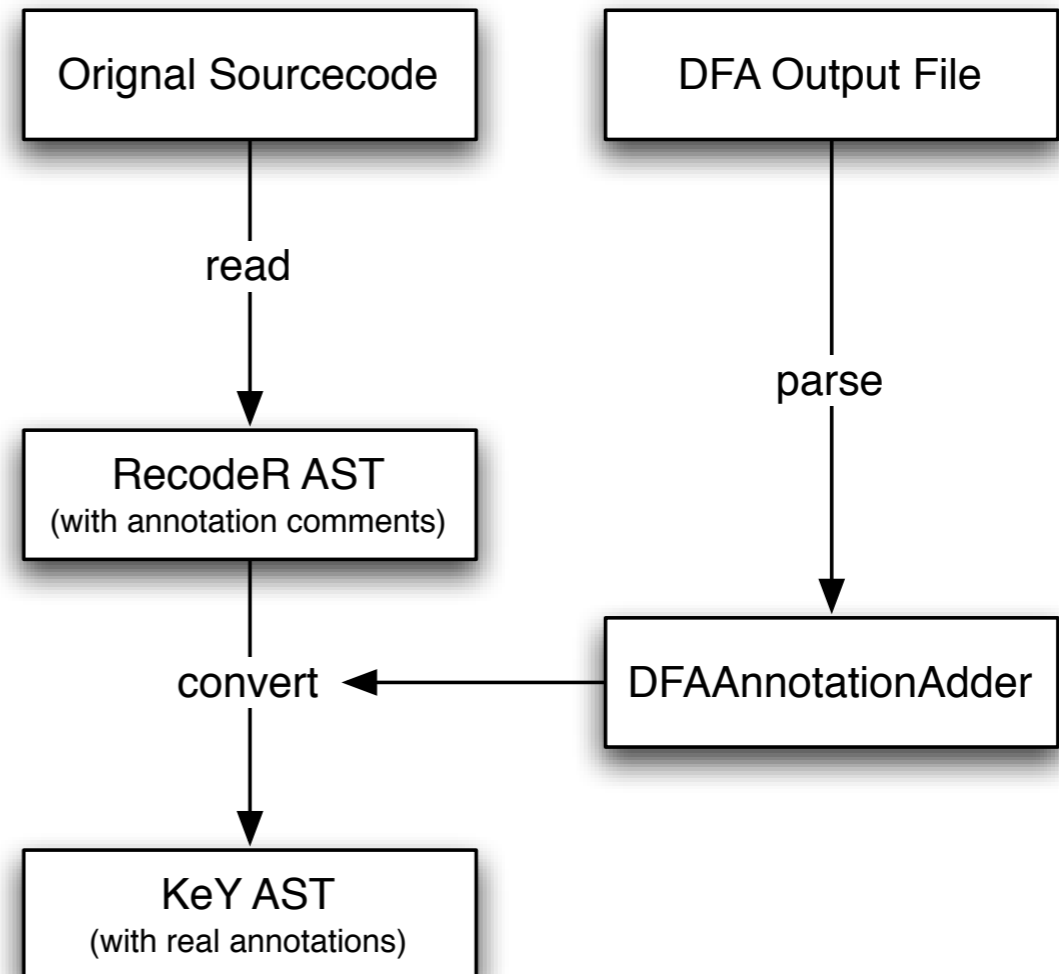
```
\heuristics(simplify_prog, simplify_prog_subset)  
\displayname "assignment";
```


Notes

- Reading comments is still - and will possibly always be - a problem (with RecodeR)
- There aren't rules that make use of DFA-Annotations in every case it would make sense. That would lead to a huge number of additional Rules
- Right now, only a small ruleset which makes use of DFA-Annotations is included

Perspectives

- Do no longer use comments for annotations
- Define a standardized input format for DFA-Information (possible as XML)
- Find method for adding more rules, that use DFA-Annotations



Demo