Mining and Integrating Reusable Methods

{ Yu Feng, Yuepeng Wang, Ruben Martins, Arati Kaushik, Isil Dillig }@UT Austin

PLINY

Objectives

Hunter is a next generation code reuse tool that finds, adapts and synthesizes common programs in *large corpus*.

Advantages:

1. ***Increase productivity:*** progammers can focus on more creative tasks.
2. ***Decrease buggy code:*** code reuse reduces the likehihood of buggy implementation.

Key ideas behind Hunter:

- Code search
- Interface alignment
- Synthesis of wrapper code

Alignment & Synthesis

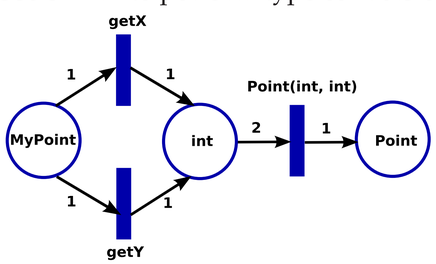
Available code is often *not* the desired code:

- Align parameters based on type similarity using ILP (Integer Linear Programming)

```
int binsearch(int elem, Vector<Integer> vec)
int binarysearch(int[] array, int element)
```

Synthesize wrapper code for type conversion:

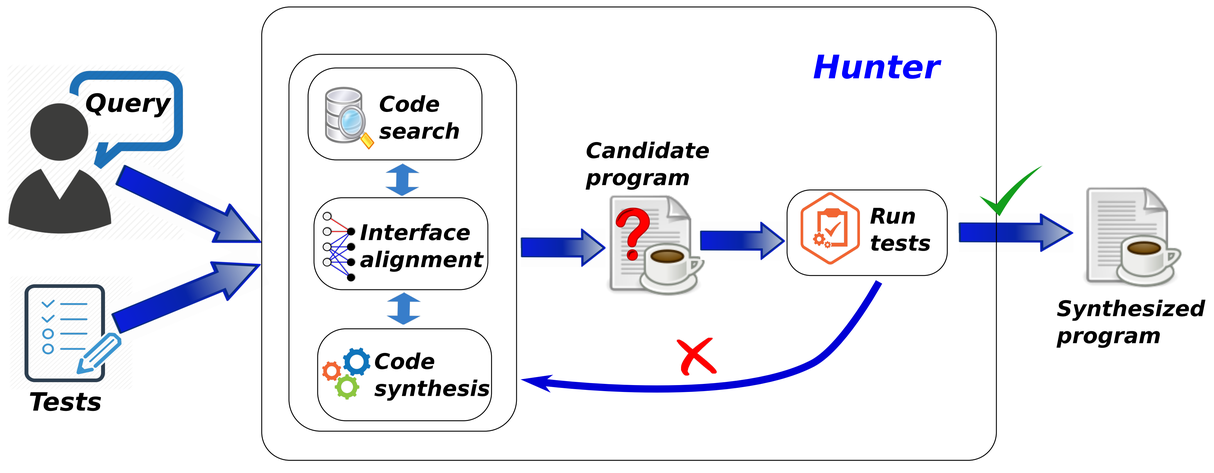
- Use SyPet to perform type conversion



Acknowledgments

This material is based on research sponsored by the Air Force Research Laboratory, under agreement number FA8750-14-2-0270.

Overview



Example

1. Write the natural language description, desired method signature and test cases

```
//@Remove duplicates from an array
public int[] removeDuplicates(int[] nums) {
   // code is expected to appear here
}
```

2. Run Hunter as an Eclipse plugin.
   Search and try out all candidate automatically

```
public int[] removeDuplicates(int[] nums)
public String[] removeDuplicateStrings(String[] strs) {
   // from org.springframework.util.StringUtils
   // method body
}
```

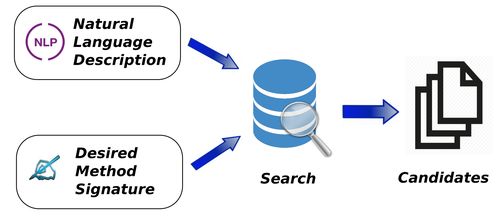
3. Get all your code ready!

```
//@Remove duplicates from an array
public int[] removeDuplicates(int[] nums) {
  String[] hunter_var1 = new String[nums.length];
  int hunter_var3 = 0;
  for (int hunter_var2 : nums) {
    hunter_var1[hunter_var3] = String.valueOf(hunter_var2);
    hunter_var3++;
  }
  String[] ret = org.springframework.util.StringUtils
                    .removeDuplicateStrings(hunter_var1);
  int[] hunter_var4 = new int[ret.length];
  int hunter_var6 = 0;
  for (String hunter_var5 : ret) {
    hunter_var4[hunter_var6] = Integer.parseInt(hunter_var5);
    hunter_var6++;
  }
  return hunter_var4;
}
```

Search

Candidates are ranked using the metrics:

- Jaccard similarity coefficient for *types*
- Tf-idf weighting, edit distance for *method signature* and *comments*



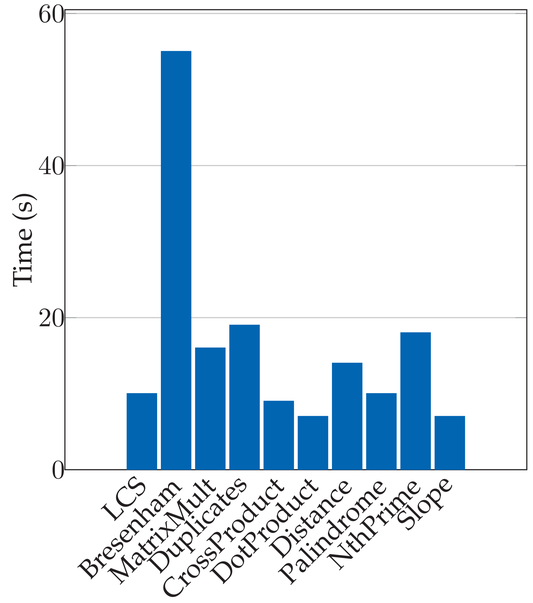
Corpus statistics:

- # Projects: 66,341
- # Java Files: 5,129,942
- # Classes: 5,896,806
- # Methods: 51,690,524

Results

Hunter is an Eclipse plugin:

- Fully *automatic*
- Synthesizes a *large variety* of programs
- Synthesizes programs in a *few seconds*
- Guaranteed to *pass all test cases*



For more examples come to the demo session!