

Principled Syntactic Code Completion using Placeholders

Eduardo Souza Sebastian Erdweg Guido Wachsmuth Eelco Visser



Unsound!

```
public class Main {
    public static void main(String[] args) {
    }
}
```

arrayadd - add an element
arraymerge - merge two arrays
cast - dynamic cast
catch - catch block
do - do while statement
else - else block

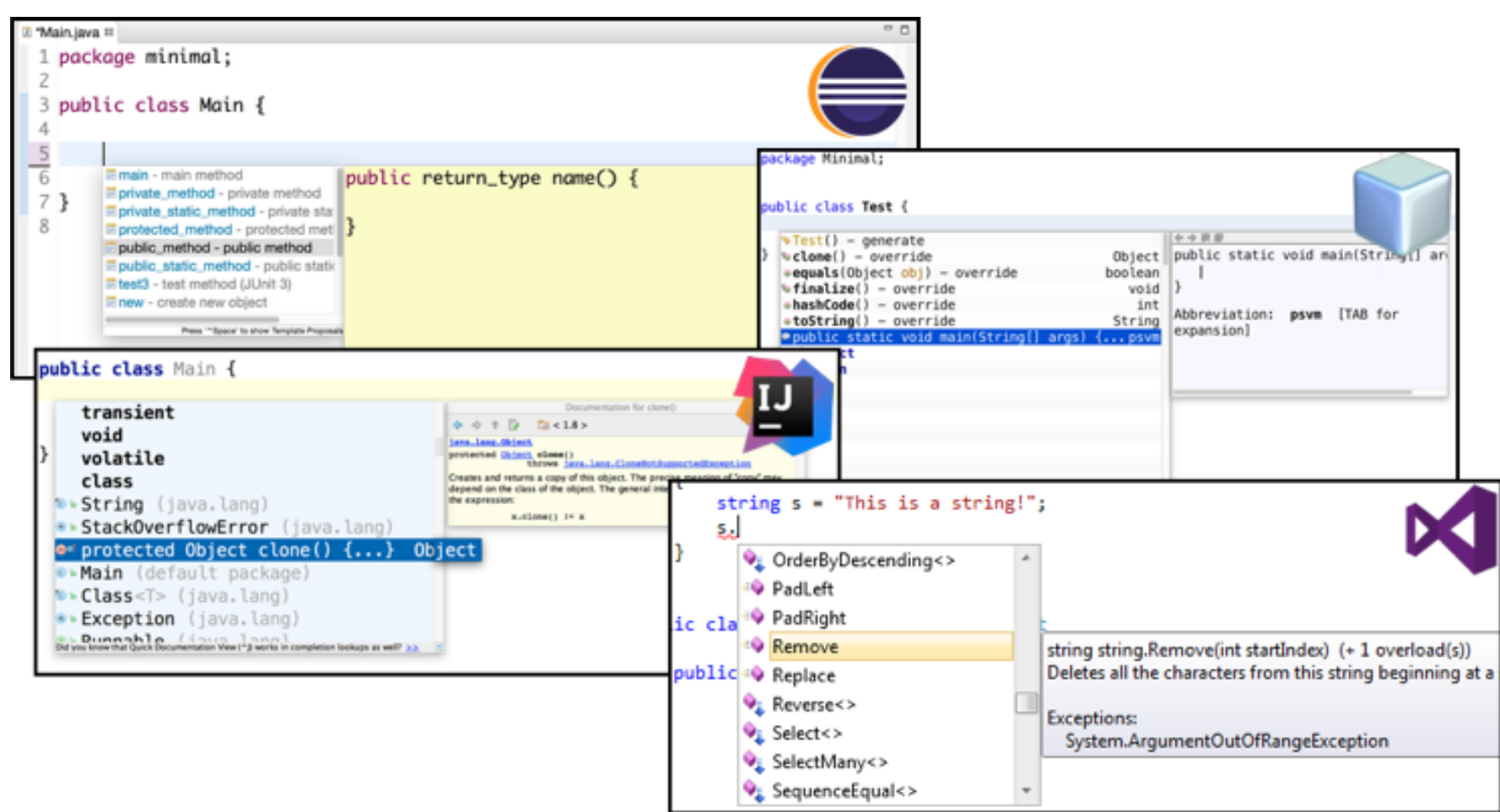
Incomplete!

```
public class Main {
    public static void main(String[] args) {
    }
}
```

arrayadd - add an element
arraymerge - merge two arrays
cast - dynamic cast
catch - catch block
do - do while statement
else - else block

String[] args2
System.arraycopy
args2[args.length]

Ad-hoc!



Incomplete programs

```
class A {
    public int m() {
        int x;
        x = $Exp;
        return $Exp + $Exp;
    }
}
```

```
class A {
    public int m() {
        int x;
        x = $Exp + $Exp;
        return $Exp + $Exp;
    }
}
```

```
class A {
    public int m() {
        int x;
        x = 21 + $Exp;
        return x;
    }
}
```

```
class A {
    public int m() {
        int x;
        x = 21 + 21;
        return x;
    }
}
```

SDF3

context-free syntax
//regular template rule

Statement.If = <
if(<Exp>) <Statement>
else <Statement>>

Parse Placeholders

context-free syntax

Statement.Statement-Plhdr = <\$Statement>

Expand Placeholders

rules

rewrite-placeholder:
Statement-Plhdr() -> If(Exp-Plhdr(), Statement-Plhdr(), Statement-Plhdr())

Format Expansions

```
class A {
    public int m() {
        int x;
        if(x < 0) x = 0;
        else $Statement
        return x;
    }
}
```

if(\$Exp) \$Statement
else \$Statement

Complete programs

```
class A {
    public int m() {
        int x;
        x = 21 + 21;
        return x;
    }
}
```

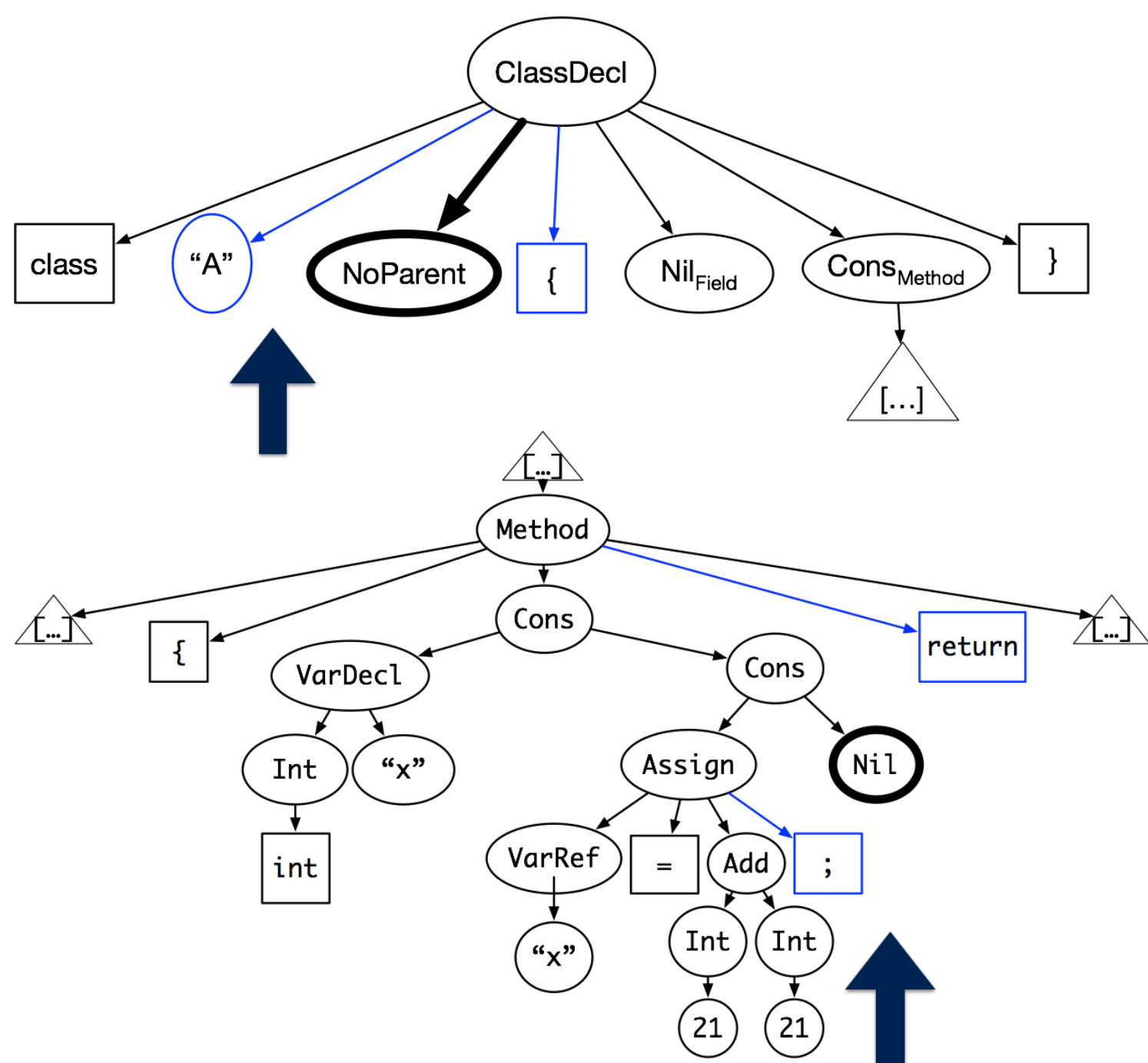
```
class A extends $ID {
    public int m() {
        int x;
        x = 21 + 21;
        return x;
    }
}
```

```
class A {
    public int m() {
        int x;
        x = 21 + 21;
        return x;
    }
}
```

```
class A {
    public int m() {
        int x;
        x = 21 + 21;
        return x;
    }
}
```

```
class A {
    public int m() {
        int x;
        x = 21 + 21;
        return x;
    }
}
```

```
class A {
    public int m() {
        int x;
        x = 21 + 21;
        return x;
    }
}
```



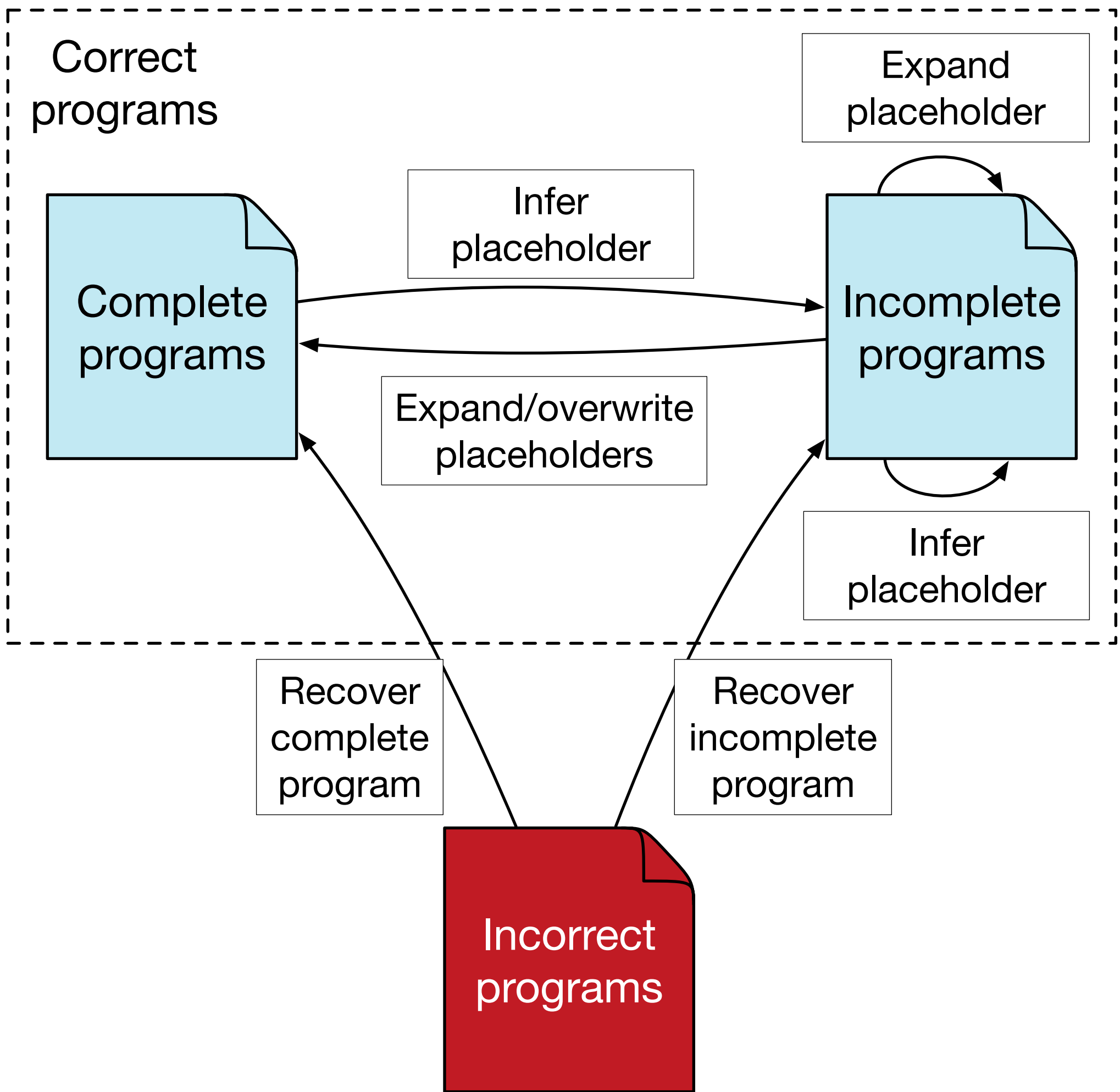
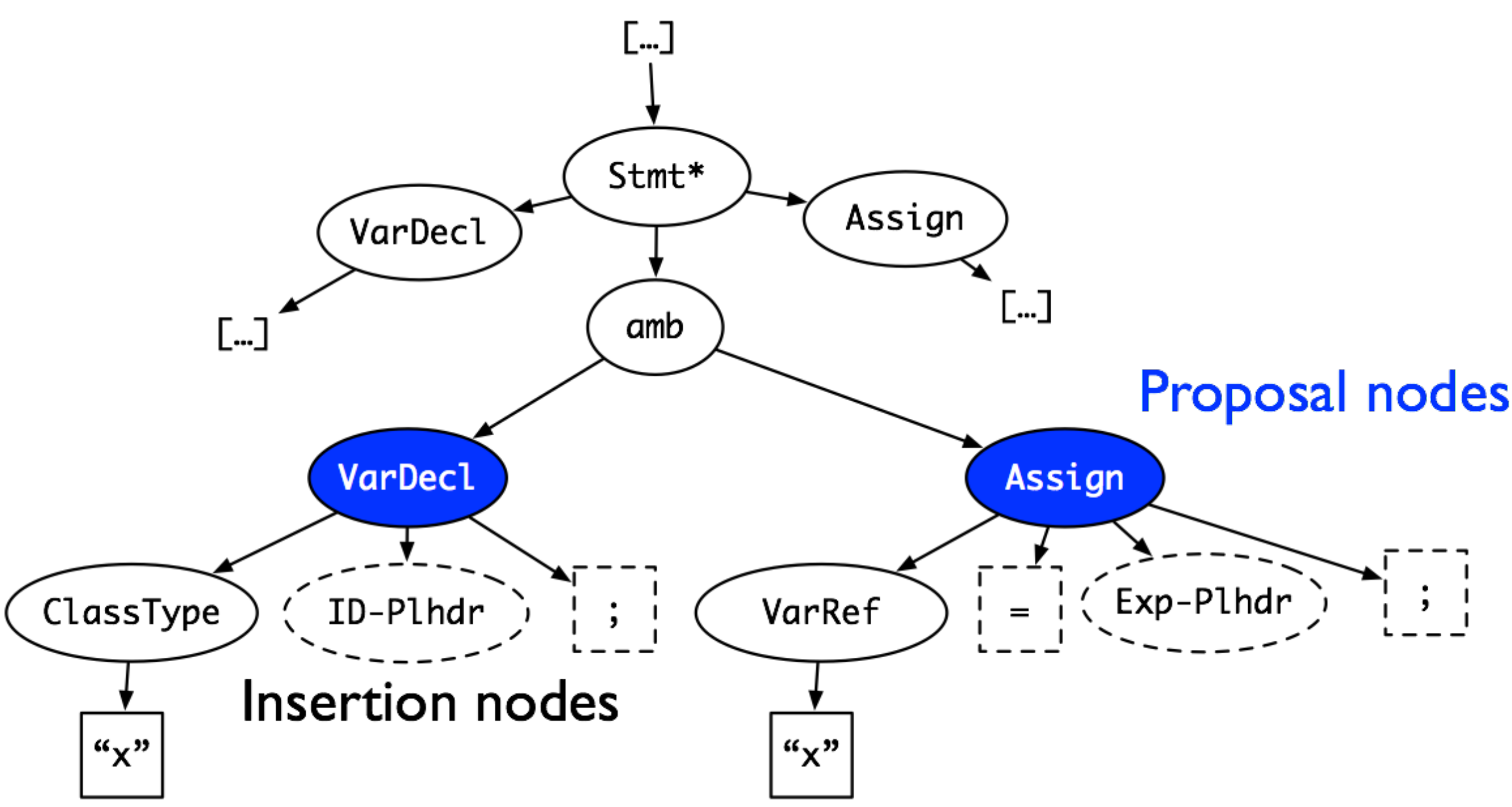
Incorrect programs

```
class A {
    public int m() {
        int x;
        x = $ID;
        return x;
    }
}
```

```
class A {
    public int m() {
        int x;
        x = $ID;
        return x;
    }
}
```

```
class A {
    public int m() {
        int x;
        x = $Exp;
        return x;
    }
}
```

```
class A {
    public int m() {
        int x;
        x = $Exp;
        return x;
    }
}
```



Sound!
Complete!

{S}poofax

