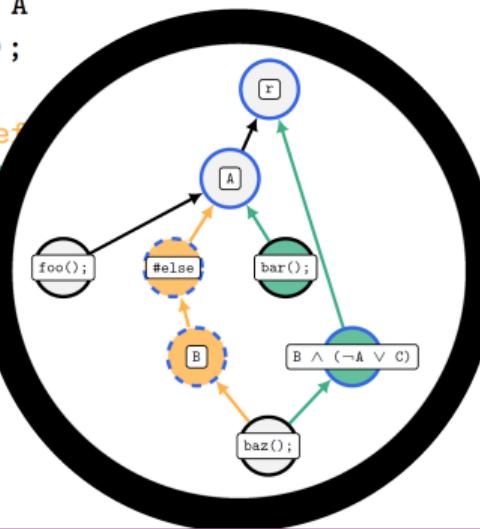


```
#ifdef A
    foo();
#ifndef B
- #else
+ bar()
#endif
+if B
    baz()
- endif
#endif
```



## Variability-Aware Differencing with DiffDetective

Paul Bittner, Alexander Schultheiß, Benjamin Moosberr, Christof Tinnes, Sören Viegner, Timo Kehrer, Thomas Thüm | April 12, 2024

```
#ifdef A
    foo();
#else
    #ifdef B
        baz();
    #endif
#endif
```

```
#ifdef A
    foo();
    bar();
#endif
#ifndef B || (!A && C)
    baz();
#endif
```

```
#ifdef A
    foo();
#else

```

```
    #ifdef B
        baz();
    #endif

```

```
#endif
```

```
#ifdef A
    foo();

```

```
-#else
```

```
-  #ifdef B
+  bar();
+#endif
```

```
+#if B && (!A || C)
```

```
    baz();

```

```
-  #endif

```

```
#endif
```

```
#ifdef A
    foo();

```

```
    bar();

```

```
#endif

```

```
#if B && (!A || C)
    baz();

```

```
#endif
```

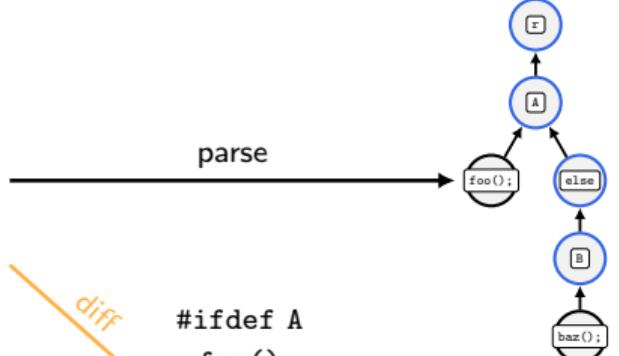
```
#endif
```



```
diff
```



```
#ifdef A
  foo();
#else
  #ifdef B
    baz();
  #endif
#endif
```

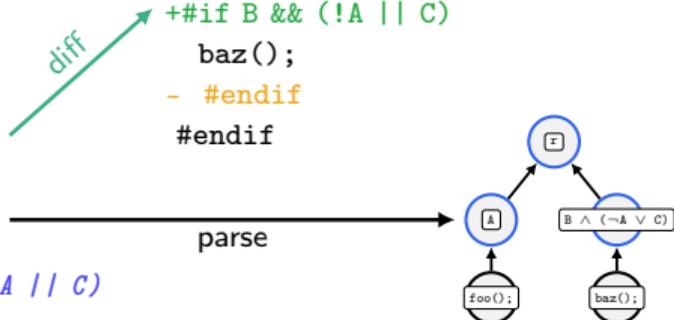


*diff*

```
#ifdef A
  foo();
-#else
```

```
-  #ifdef B
+  bar();
+#endif
```

```
#ifdef A
  foo();
  bar();
#endif
#if B == (!A || C)
  baz();
#endif
```



*diff*

```
+#if B && (!A || C)
  baz();
- #endif
#endif
```

```
#ifdef A
foo();
#else
#endif
```

parse



*diff*

```
#ifdef A
foo();
```

-#else

- #ifdef B

+ bar();

+#endif

+#if B && (!A || C)

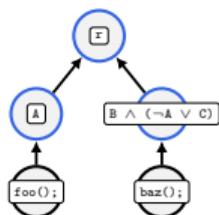
baz();

- #endif

#endif

```
#ifdef A
foo();
bar();
#endif
#if B && (!A || C)
baz();
#endif
```

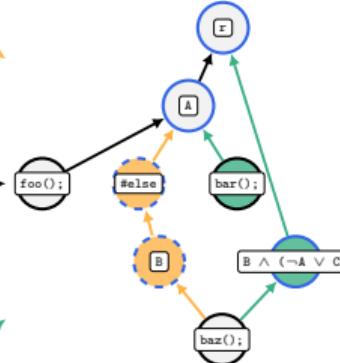
parse



*diff*

*diff*

parse



*diff*

# Why is this interesting?

## Operation Lifting [Bittner'23]

Lift operations on SPLs to operations on *edits* to SPLs (e.g., partial configuration, feature tracing),  
Potential use cases:

- feature-aware commit untangling

- derive patches for variants

- review merge requests

- ...

# Why is this interesting?

## Operation Lifting [Bittner'23]

Lift operations on SPLs to operations on *edits* to SPLs (e.g., partial configuration, feature tracing),  
Potential use cases:

- feature-aware commit untangling
- derive patches for variants
- review merge requests

...

## Edit Explanations [Bittner'22, Güthing'23]

Explain how changes affect the set of variants of code.

*Does this line of code appear in more variants now?*

# Why is this interesting?

## Operation Lifting [Bittner'23]

Lift operations on SPLs to operations on *edits* to SPLs (e.g., partial configuration, feature tracing),  
Potential use cases:

feature-aware commit untangling

derive patches for variants

review merge requests

...

## Edit Explanations [Bittner'22, Güthing'23]

Explain how changes affect the set of variants of code.

*Does this line of code appear in more variants now?*

## Simulate Clone-and-Own [Schult.'22,'23]

Generate evolution histories of single variants.

⇒ clone-and-own datasets à la carte

# Why is this interesting?

## Operation Lifting [Bittner'23]

Lift operations on SPLs to operations on *edits* to SPLs (e.g., partial configuration, feature tracing),  
Potential use cases:

- feature-aware commit untangling
- derive patches for variants
- review merge requests
- ...

## Simulate Clone-and-Own [Schult.'22,'23]

Generate evolution histories of single variants.  
⇒ clone-and-own datasets à la carte

## Edit Explanations [Bittner'22, Güthing'23]

Explain how changes affect the set of variants of code.

*Does this line of code appear in more variants now?*

## Benchmarking Differencers [Moosbrugger'23]

- line-based differencing (Myers/Histogram)
- vs. GumTree [Falleri'14]
- (vs. TrueDiff [Erdweg'21])

# Why is this interesting?

## Operation Lifting [Bittner'23]

Lift operations on SPLs to operations on *edits* to SPLs (e.g., partial configuration, feature tracing),  
Potential use cases:

- feature-aware commit untangling
- derive patches for variants
- review merge requests
- ...

## Simulate Clone-and-Own [Schult.'22,'23]

Generate evolution histories of single variants.  
⇒ clone-and-own datasets à la carte

## Edit Explanations [Bittner'22, Güthing'23]

Explain how changes affect the set of variants of code.

*Does this line of code appear in more variants now?*

## Benchmarking Differencers [Moosbrugger'23]

line-based differencing (Myers/Histogram)  
vs. GumTree [Falleri'14]  
(vs. TrueDiff [Erdweg'21])

## Higher-Order Tree Differencing [TODO]

Diffs of Diffs of Diffs ...



<https://variantsync.github.io/DiffDetective>