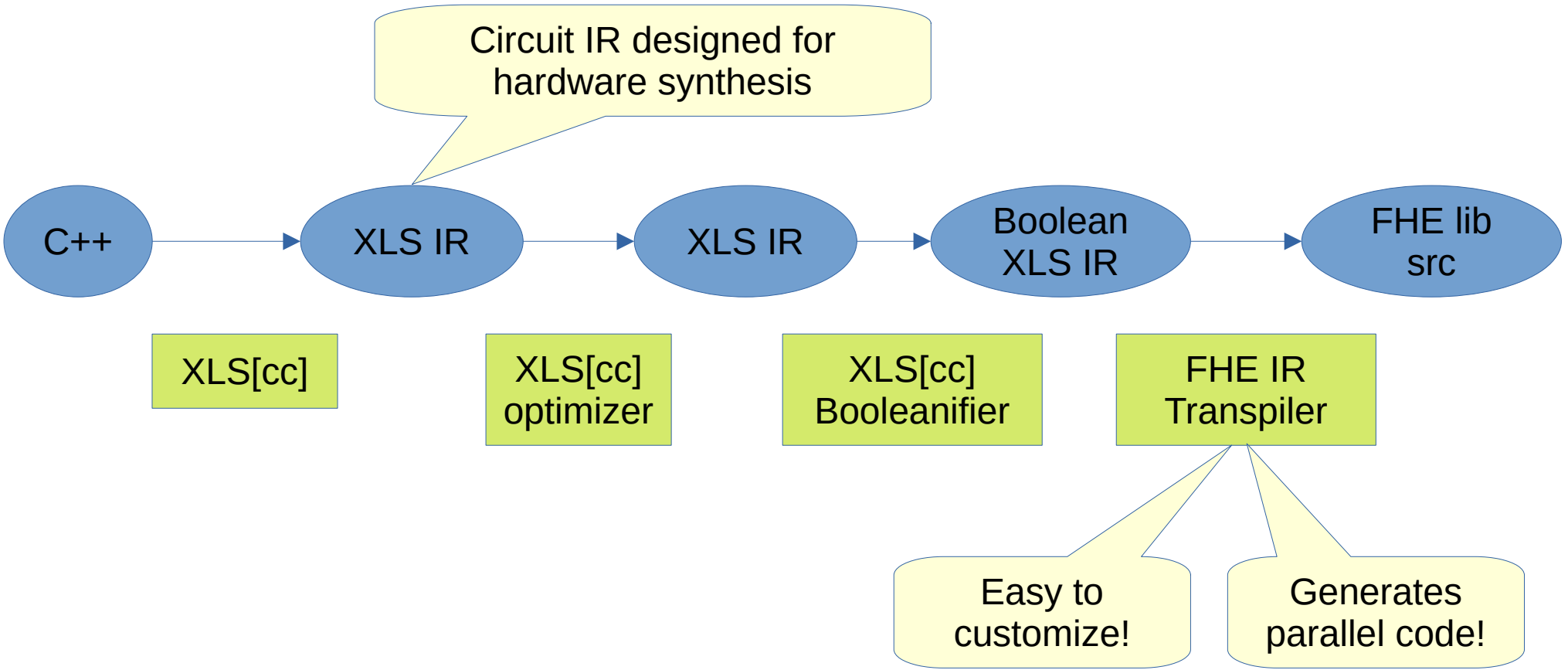
Google’s FHE compiler



Google’s FHE compiler

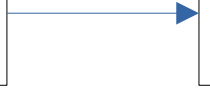
- One of the targets of the transpiler is fhe-rs (a rust FHE library)
- Janmajayamall’s patched it to support phantom-zone:

  https://github.com/Janmajayamall/fully-homomorphic-encryption

Limitations of XLS[cc]

- Variable-length arrays are not supported

```
void foo(int* buf)
```



```
void foo(int buf[N])
```

Limitations of XLS[cc]

- While-loops and for-loops with a variable end-condition are not supported

```
for (int i = 0; i < n; i++) {
    sum += image[i];
}
```

```
for (int i = 0; i < 8; i++) {
    sum += image[i];
    if(i == (length-1)) {
        break;
    }
}
```

Limitations of XLS[cc]

- Pointer arguments in functions are not supported

```
void foo (struct Bar* x) {
    x→baz = 42;
}
```

```
struct Bar foo(struct Bar x) {
    x.baz = 42;
    return x;
}
```

Limitations of XLS[cc]

- Floating-point data types are not supported

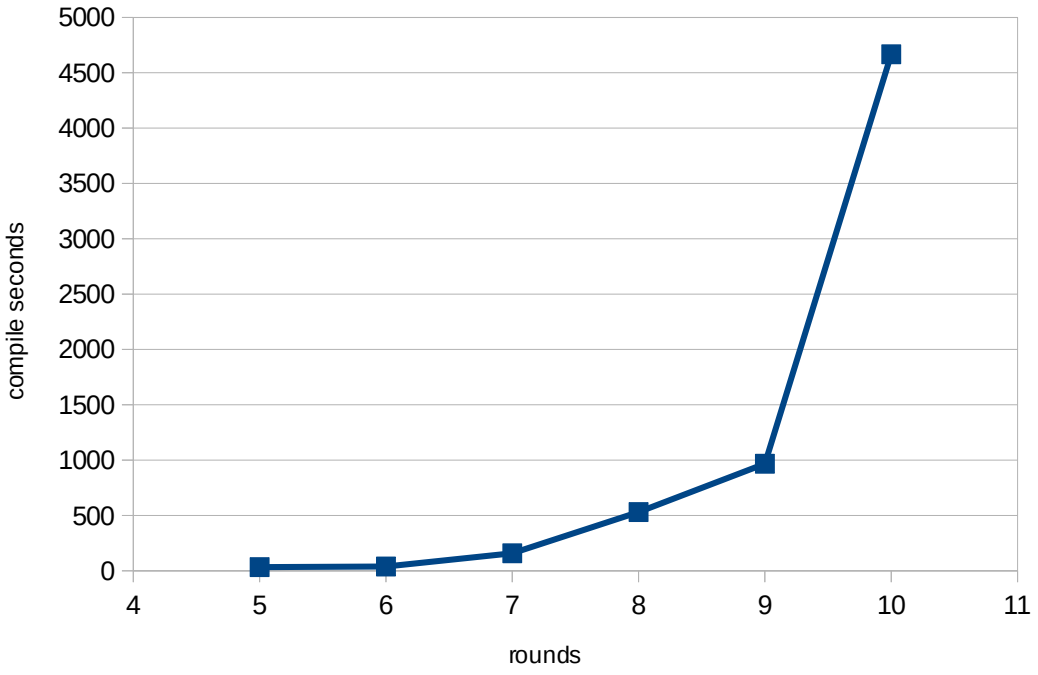
Fibonacci example

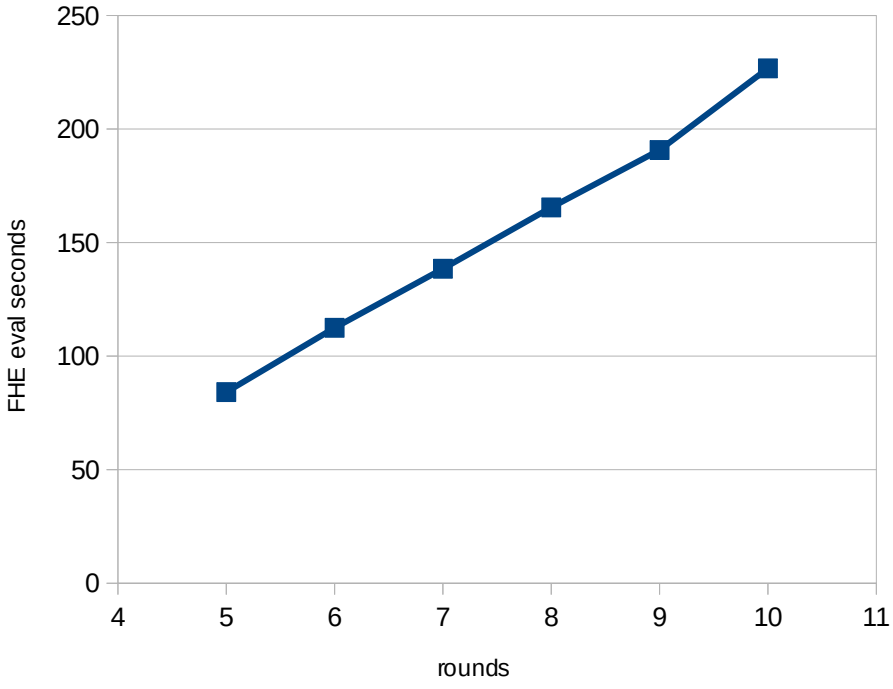
- Let’s check:
  - C source code
  - XLS IR
  - Verilog
  - Rust output

Sha256 benchmark

- Reduced number of rounds
  - 64 rounds (original sha256) takes too long to compile
- Results on 16 vCPU
  - Compile: 33.216s
  - FHE circuit eval: 84.16s
    - Expected for 64 rounds: ~20m

Sha256 benchmark



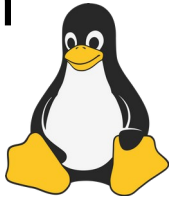


Get your POD!

Try it!

Docker image ready for usage

NOTE: It’s very slow on ARM Macs, use Linux



https://github.com/ed255/fully-homomorphic-encryption/blob/phantom/README-phantom.md

Try it!

Docker image ready for usage

But you can use a shared server for compiling

https://github.com/ed255/fully-homomorphic-encryption/blob/phantom/README-phantom.md