Rapid API Dialect
The Team

- Nate Brennand
- Benjamin Edelstein
- Brendon Fish
- Dan Schlosser
- Brian (Dong Hee) Shin
Motivation

Rapidly prototype and develop an API server, using strong typing to provide some guarantees about API functionality.
Hello World

println("Hello world");
Hello World via API

```javascript
http () string {
    return "Hello World";
}
```
func gcd(int p, int q) int {
    while (q != 0) {
        int temp = q;
        q = p % q;
        p = temp;
    }
    return p;
}
Let’s Make it a Server!

```cpp
namespace gcd {
    param int a {
        param int b {
            http (int a, int b) int {
                int res = gcd(a, b);
                return res;
            }
        }
    }
}
```
OOP in RAPID

class User {
    int age;
    string name = "Stephen";
    optional int height;

    instance my {
        func is_old() boolean {
            return (my.age >= 30);
        }
        func make_older() {
            my.age = my.age + 1;
        }
    }
}

User stephen = new User(age=29);
println(stephen.age);
stephen.height = 73;
println(stephen.height);

if (stephen.is_old()) {
    println("Stephen is old");
} else {
    println("Stephen is young");
}

stephen.make_older();
if (stephen.is_old()) {
    println("Stephen is old");
}
Types

- Ints
- Floats
- Strings
- Booleans
- Classes
- Lists
Functions

- Instance functions
- Standalone functions
- Multiple return types
- Type checked args
- Optional args with default values
Compiler Pipeline

Source Code ➔ Parser ➔ Translator

AST

Generated Go Code

Go Compiler ⇔ Generator

Semantic AST

Generator ⇔ Semantic Checker (2-pass)

Unsafe Semantic AST
Lessons Learned

- Set smaller milestones and stick to them. Meet more regularly to work.
- Programming a sizeable project in a new programming language takes longer than expected.
- Rigorous code review, and testing are important!
- Focus. Our LRM was ambitious
Demos

- GCD Server
- OOP