

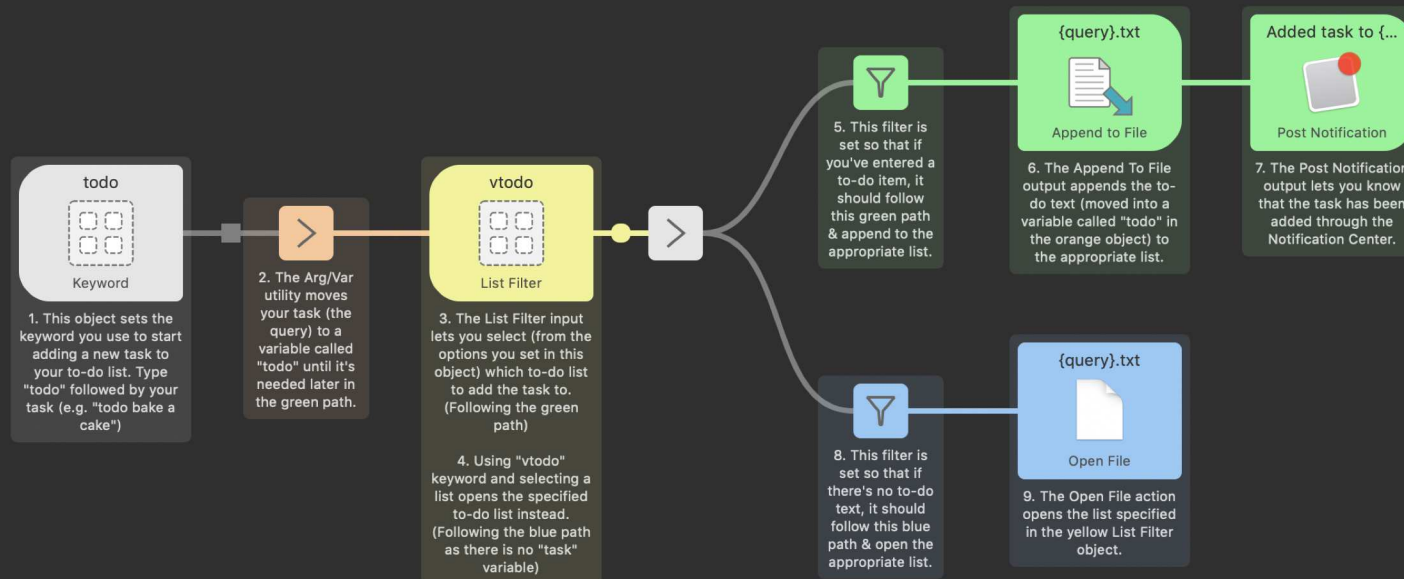
"Hey, IRIs"

Alfred Workflows



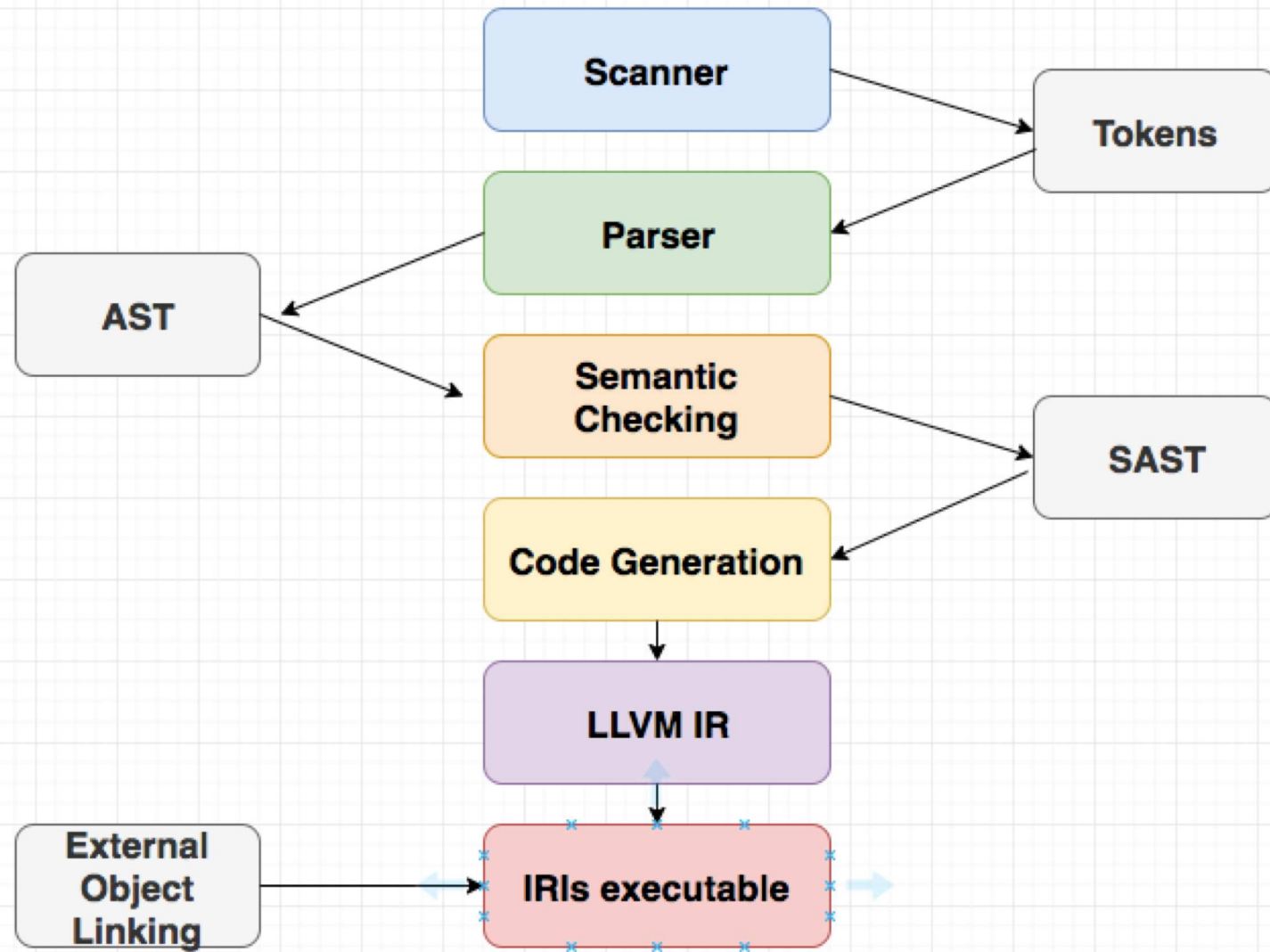
Simple To-Do List

Use "todo" followed by your task, then choose a list to add it to. - v1.0



Team Roles

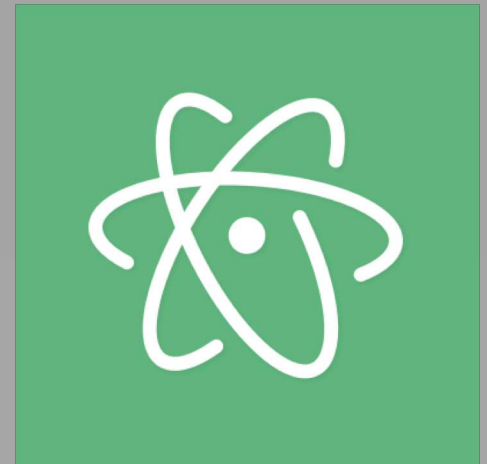




Timeline



Project Environment



SYNTAX

TYPES

int

list

float

string

tuple

bool

|

variable1

|

variable2

variable1

|

function

tuple

|

function

```
int plus2(int a)
    return a + 2
```

Siri

```
int plus1(int a)
    return a + 1
```

Siri

```
int main()
```

0

|

plus1

|

plus2

|

printi

Siri

HIGHLIGHTS

Pipe: "Overload"

Semant + LLVM

Semant: Sassign/SCall

List: C-based Implementation

String: Heap-based Implementation

Tuple: LLVM-based Implementation

Future Works

TESTING

Testing

- Identifiers
- Keywords
- Control flow: if, while
- Types: int, float, string, list, tuple
- Built-in functions: inputint, inputfloat, inputstring, sendmail
- Comments
- Operators
- Variables and function declarations
- Pipe

```
• 1 2 3 | a 1 printi(1) 1,2,3
  | | a | |
  a a a, b, c
    |
    printi
```

Test Suites

- Two test suites , one intended to pass, and one intended to fail.
- test_* , fail_* -> .out , .err
- Over 80 tests in total

```
al@numel:~/Downloads/IRIs$ sh testall.sh
test_calculate_int...OK
test_helloworld...OK
test_pipe...OK
test_print_float...OK
fail_print_int...OK
```

```
int main()
    float a,b,c
    1.234
    |
    a
    |
    b
    |
    c
    |
    printf
```

siri|

```
testall.sh
~/Downloads/IRIs

#!/bin/sh

# Regression testing script for MicroC
# Step through a list of files
# Compile, run, and check the output of each expected-to-work test
# Compile and check the error of each expected-to-fail test

# Path to the LLVM interpreter
LLI="lli"
#LLI="/usr/local/opt/llvm/bin/lli"

# Path to the LLVM compiler
LLC="llc"

# Path to the C compiler
CC="cc"

# Path to the iris compiler. Usually "./iris.native"
# Try "_build/iris.native" if ocamlbuild was unable to create a symbolic link.
IRIS="./src/iris.native"
#IRIS="_build/iris.native"

# Set time limit for all operations
ulimit -t 30

globallog=testall.log
rm -f $globallog
error=0
globalerror=0

keep=0

Usage() {
    echo "Usage: testall.sh [options] [.ir files]"
    echo "-k    Keep intermediate files"
    echo "-h    Print this help"
}
```


Demo

- Calculate Tips
- Binary Search
- Send Email