

Fuzion — Safety through Simplicity



flang.dev
github.com/fridis/fuzion

F. Siebert, Tokiwa Software GmbH

Ada Europe 2021 (Online)
WiP Session, 14:15 9th June `21



Motivation

Many languages overloaded with concepts like classes, methods, interfaces, constructors, traits, records, structs, packages, values, ...

⇒ Fuzion has one concept: a feature

Compilers and tools are more powerful

⇒ Tools make better decisions

Systems are safety-critical

⇒ we need to ensure correctness



Fuzion Unifies Language Concepts

A feature unifies concepts found in other languages:

⇒ class, record / struct, interface, trait

⇒ method, constructor, function

⇒ field, argument, local variable

⇒ package, name space

⇒ product or union type

Implementation details are the compiler's concern, not the developer's.

Feature Declaration

- ⇒ an outer feature
- ⇒ a name
- ⇒ type args
- ⇒ formal args, result type
- ⇒ parent features
- ⇒ a contract
- ⇒ an implementation
 - inner features

```
myUtils is
  myStack<T>(maxSize i64)
    : lifo<T>, List<T>
  pre
    maxSize > 0
  post
    isEmpty
is
  size u64 => 0
  isEmpty => size = 0
  push(x T) myStack<T>
  is ...
```

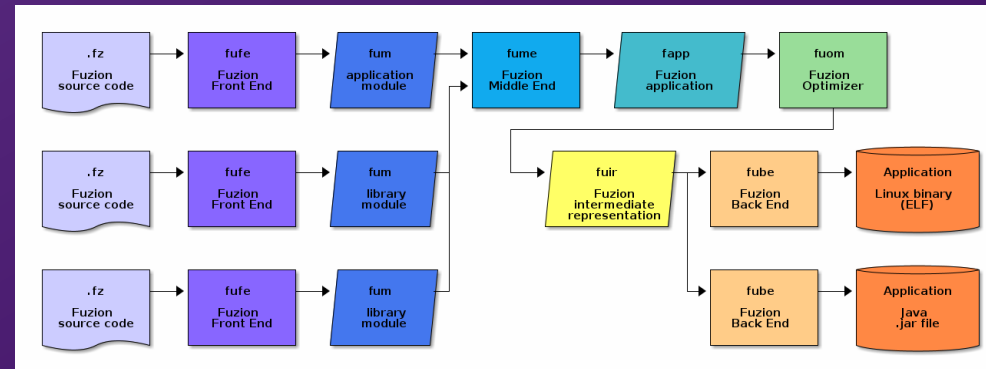


Toolchain

Based on

- ⇒ simple intermediate Code
- ⇒ processed by different tools

- static analysers
- optimizers
- compilers
- etc.



More Information

⇒ come to WP 4 Poster

⇒ on the web

→ flang.dev — tutorial, design, examples, ...

→ github.com/fridis/fuzion

→ tokiwa.software

