

Sublimate

ridiculously fast full stack Swift prototyping with

Vapor & Sourcery

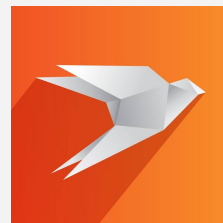
Vapor

Sourcery

Sublime

Why Server Side Swift?

- ❤️ Swift ❤️
- Great performances



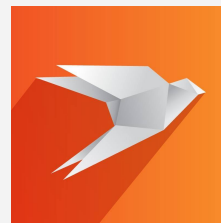
Vapor

Sourcery

Sublime

Several frameworks

- Vapor
- Perfect
- Kitura



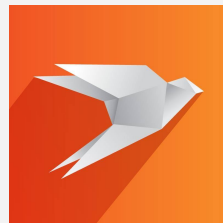
Vapor

Sourcery

Sublime

Why Vapor

- Very popular
- Great community
- Swift NIO
- Very simple



very simple

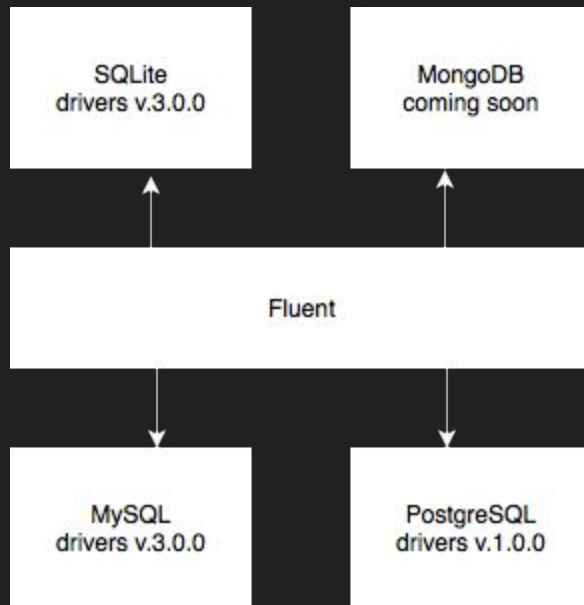
```
router.get("user", use: { (request) -> Future<[User]> in
    return User.query(on: request).all()
})

router.post("person", use: { (request) -> Future<People> in
    return try request.content.decode(People.self).flatMap { people in
        return people.save(on: request)
    }
})
```

Vapor

Sourcery

Sublime



Sourcery is a code generation tool for Swift based on SourceKit and Stencil

- SourceKit allows you to parse and explore the structure of Swift sources
- Stencil is a templating language (also used by Kitura and SwiftGen)

Sourcery is most used to quickly generate glue code and protocol conformance

for example

```
public struct Pet: AutoEquatable {  
    public let name: String  
    public let age: Int  
    public let gender: Gender  
  
    public init(name: String, age: Int, gender: Gender) {  
        self.name = name  
        self.age = age  
        self.gender = gender  
    }  
}
```


combined with

```
{% for type in types.implementing.AutoEquatable %}  
extension {{ type.name }}: Equatable {  
    public static func ==(lhs: {{ type.name }}, rhs: {{ type.name }}) -> Bool {  
        {% for var in type.variables %}  
            guard lhs.{{ var.name }} == rhs.{{ var.name }} else { return false }  
        {% endfor %}  
        return true  
    }  
}  
{% endfor %}
```

generates

```
extension Pet: Equatable {  
    public static func ==(lhs: Pet, rhs: Pet) -> Bool {  
        guard lhs.name == rhs.name else { return false }  
        guard lhs.age == rhs.age else { return false }  
        guard lhs.gender == rhs.gender else { return false }  
        return true  
    }  
}
```

Vapor

Sourcery

Sublimate

Why Sublimate

- experiment with Vapor & Sourcery
- bootstrap a backend for an MVP

Where to get it

- <https://github.com/gabrielepalma/sublimate>

Vapor

Sourcery

Sublimete

Features:

On server:

- Fluent models
- CRUD routes
- Authentication logic
 - access and refresh tokens (Oauth-like)
 - tokens are signed JWT

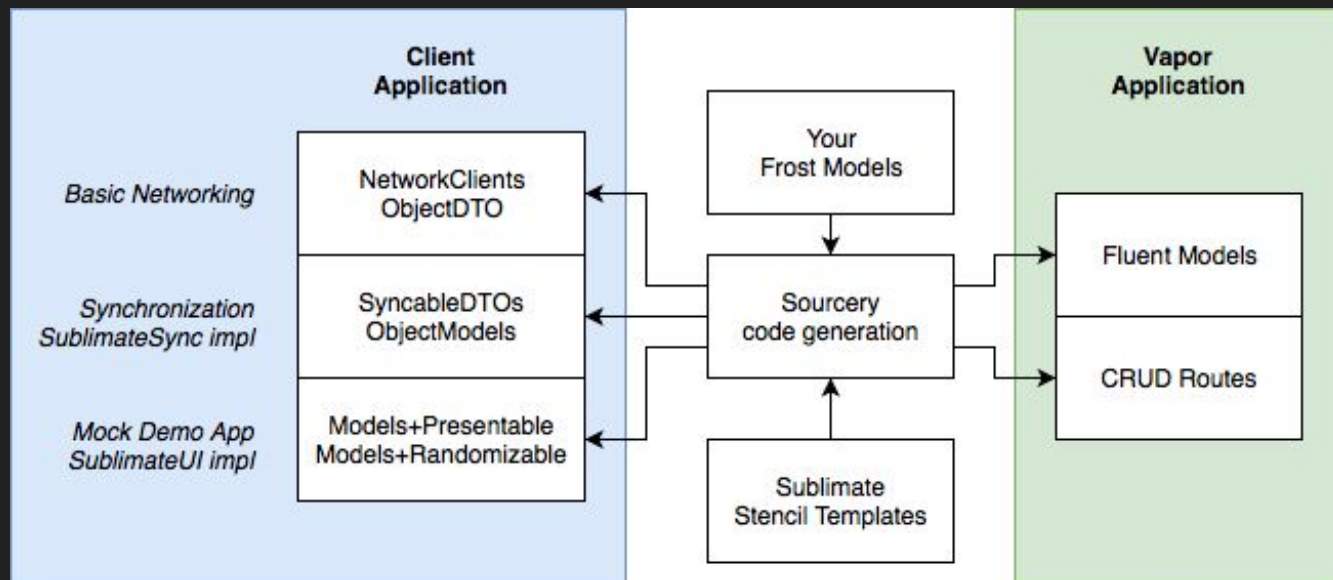
On client:

- Network clients
- Realm models
- Data synchronization
- Authentication logic
- Mock UI

Vapor

Sourcery

Sublime



Vapor

Sourcery

Sublimate

👷 coming next 👷

- Image handling serving arbitrary size, plus caching
- A profile table 👷 👷
- Putting backend developers out of business
- Profit 💰