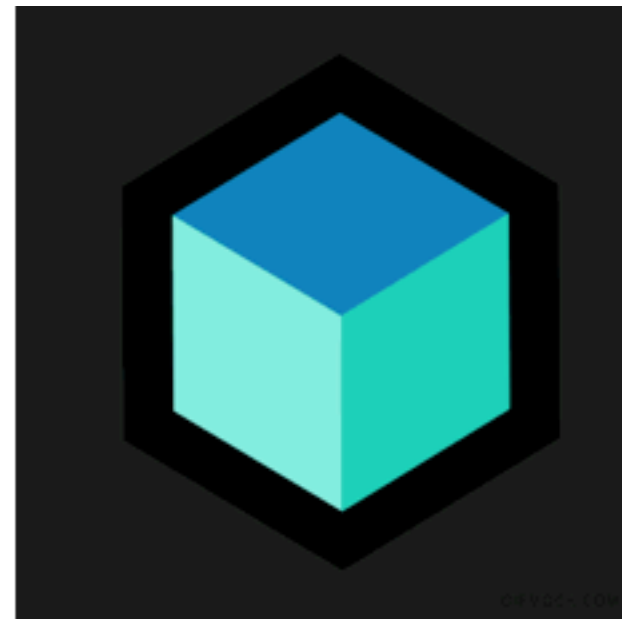


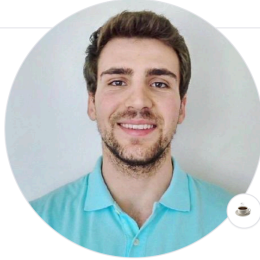
Bloc ❤️ Cubit

The new kid on the bloc



Felix Angelov @ Very Good Ventures

Very Good Ventures, Chicago 🕶️🦄



Felix Angelov
felangelov

software engineer by day, software engineer by night.

[Follow](#) [Sponsor](#) ...

👤 2.1k followers · 36 following · ⭐ 232

📁 @VG Ventures
📍 Chicago
✉ Sign in to view email
🐦 @felangelov

Highlights
🌟 Arctic Code Vault Contributor
⭐ PRO

Organizations
🏢

Overview Repositories 77 Projects

📄 felangelov / README.md

Hi there 🙌

I'm a software engineer at [Very Good Ventures](#) in Chicago, IL 🇺🇸

- 👩🏫 I'm currently working on the [bloc library](#)
- ☕ I'm learning how to make latte art
- 👤 I'm looking to collaborate on open source projects
- 💬 Ask me about [Flutter](#) and [Dart](#).
- 🗨 Pronouns: he/him/his
- 🐰 Fun fact: I have a pet bunny named Coco 🐰

Pinned

📄 [bloc](#)

A predictable state management library that helps implement the BLoC design pattern

🟢 Dart ⭐ 5.3k 🐙 1.2k

📄 [bloc.js](#)

A predictable state management library that helps implement the BLoC design pattern in JavaScript

🟢 TypeScript ⭐ 67 🐙 8

📄 [equatable](#)

Simplify Equality Comparisons | A Dart abstract class that helps to implement equality without needing to explicitly override == and hashCode.

🟢 Dart ⭐ 377 🐙 37

📄 [fresh](#)

🔥 A token refresh library for Dart.

🟢 Dart ⭐ 92 🐙 7

📄 [flutter](#)

Forked from flutter/flutter

Flutter makes it easy and fast to build beautiful mobile apps.

🟢 Dart ⭐ 1

📄 [sealed_flutter_bloc](#)

flutter_bloc state management extension that integrates sealed_unions.

🟢 Dart ⭐ 45 🐙 4



<https://verygood.ventures>

Agenda

- What's new in bloc v6.0.0
- Meet Cubit
- Cubit vs. Bloc
- Fluttersaurus (live coding)

What's New?

Initial State Changes

// OLD

```
class CounterBloc extends Bloc<CounterEvent, int> {  
  @override  
  int get initialState => 0;  
  
  ...  
}
```

// NEW

```
class CounterBloc extends Bloc<CounterEvent, int> {  
  CounterBloc() : super(0);  
  
  ...  
}
```

Bloc Test Simplifications

// OLD

```
blocTest<CounterBloc, CounterEvent, int>(
  'emits [1] when increment is called',
  build: () async => CounterBloc(),
  act: (bloc) async => bloc.add(CounterEvent.increment),
  expect: const <int>[1],
);
```

// NEW

```
blocTest<CounterBloc, int>(
  'emits [1] when increment is called',
  build: () => CounterBloc(),
  act: (bloc) => bloc.add(CounterEvent.increment),
  expect: const <int>[1],
);
```

BlocDelegate -> BlocObserver

// OLD

```
class MyBlocDelegate extends BlocDelegate {  
    ...  
}
```

// NEW

```
class MyBlocObserver extends BlocObserver {  
    ...  
}
```

BlocObserver onError accepts Cubit

// OLD

```
class MyBlocObserver extends BlocObserver {  
  @override  
  void onError(Bloc bloc, Object error, StackTrace stackTrace) {  
    super.onError(bloc, error, stackTrace);  
  }  
}
```

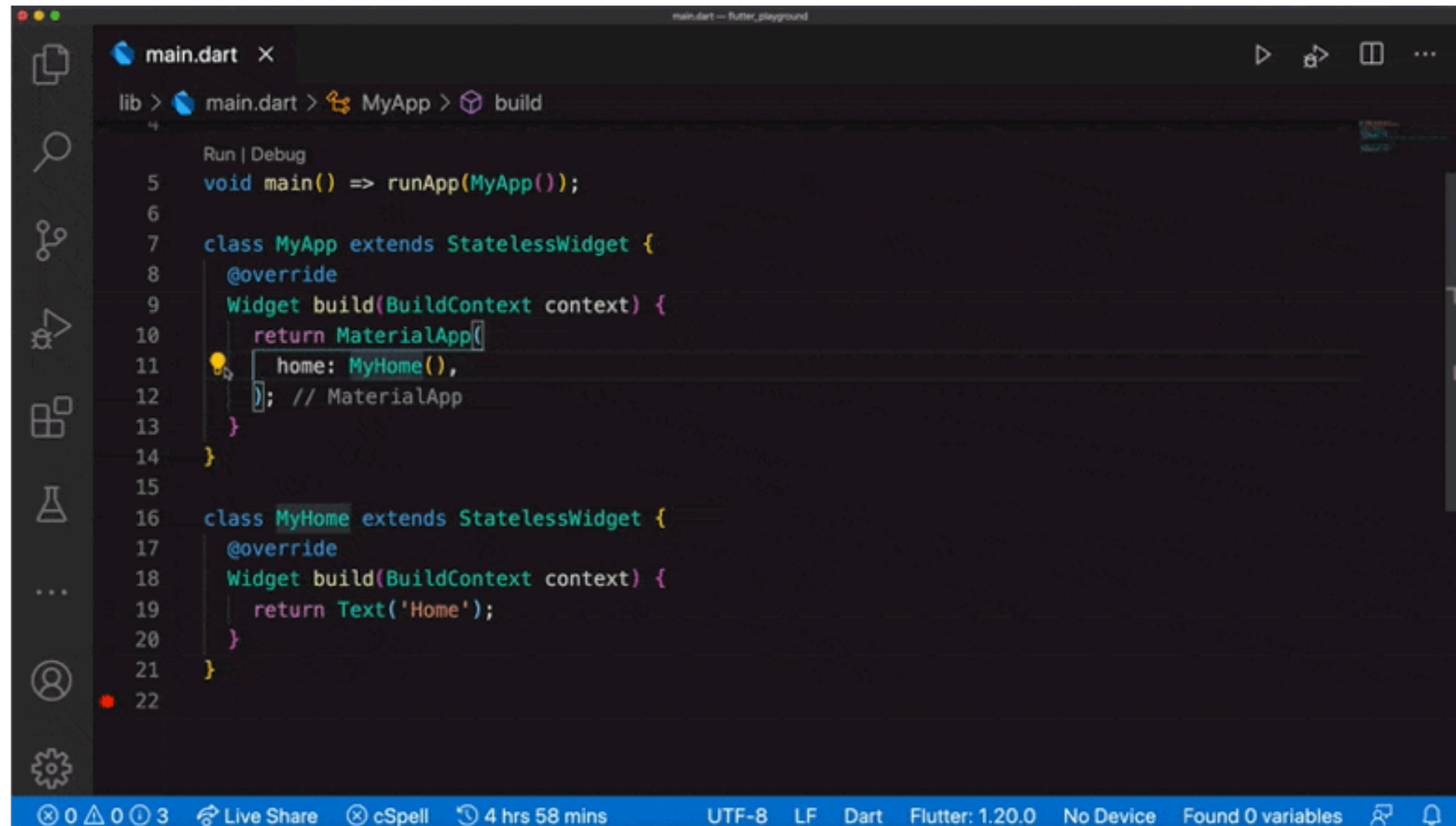
// NEW

```
class MyBlocObserver extends BlocObserver {  
  @override  
  void onError(Cubit cubit, Object error, StackTrace stackTrace) {  
    super.onError(cubit, error, stackTrace);  
  }  
}
```


BlocSupervisor Removed

```
void main() {  
  // OLD  
  BlocSupervisor.delegate = MyBlocDelegate();  
  
  // NEW  
  Bloc.observer = MyBlocObserver();  
}
```

Tooling Improvements



```
main.dart x
lib > main.dart > MyApp > build
Run | Debug
5 void main() => runApp(MyApp());
6
7 class MyApp extends StatelessWidget {
8   @override
9   Widget build(BuildContext context) {
10     return MaterialApp(
11       home: MyHome(),
12     ); // MaterialApp
13   }
14 }
15
16 class MyHome extends StatelessWidget {
17   @override
18   Widget build(BuildContext context) {
19     return Text('Home');
20   }
21 }
22
```

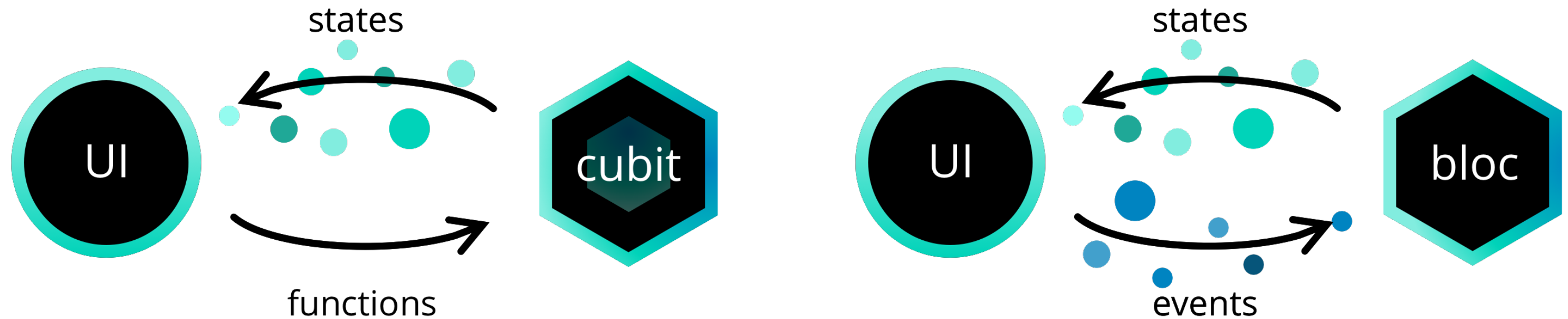
0 0 3 Live Share cSpell 4 hrs 58 mins UTF-8 LF Dart Flutter: 1.20.0 No Device Found 0 variables

Complete migration guide

<https://bloclibrary.dev/#/migration>

Meet Cubit

Cubit = Bloc - Events



CounterCubit

```
import 'package:bloc/bloc.dart';

class CounterCubit extends Cubit<int> {
  CounterCubit() : super(0);

  void increment() => emit(state + 1);
}
```

Cubit vs. Bloc

Cubit

```
class CounterCubit extends Cubit<int> {  
  CounterCubit() : super(0);  
  
  void increment() => emit(state + 1);  
}
```

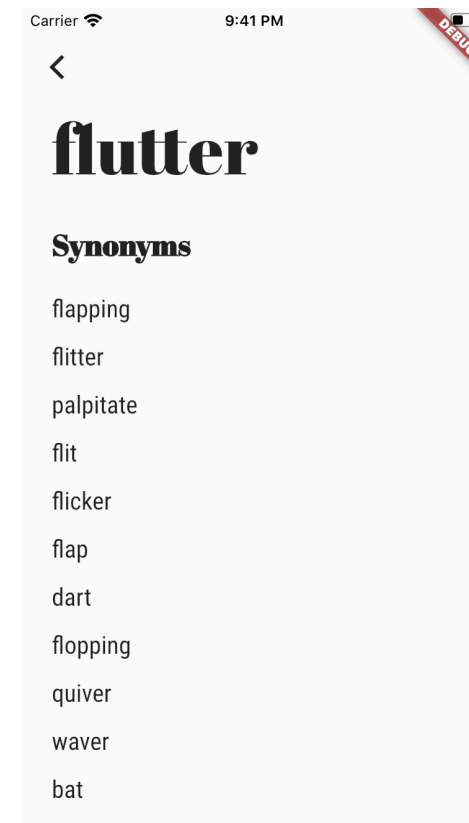
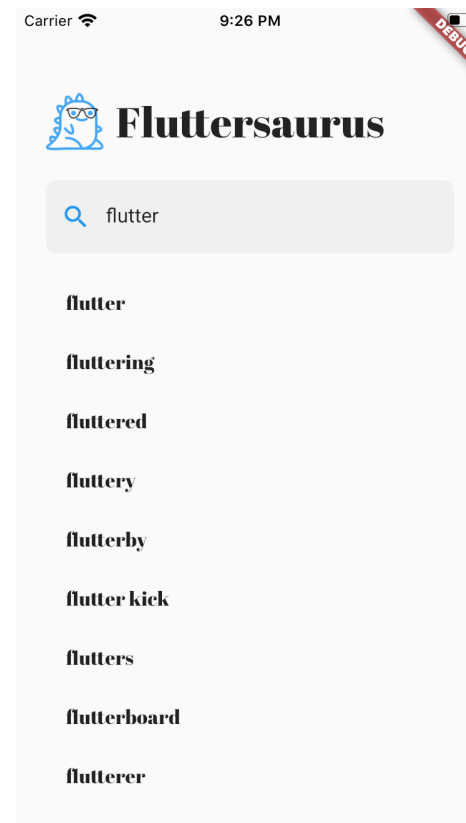
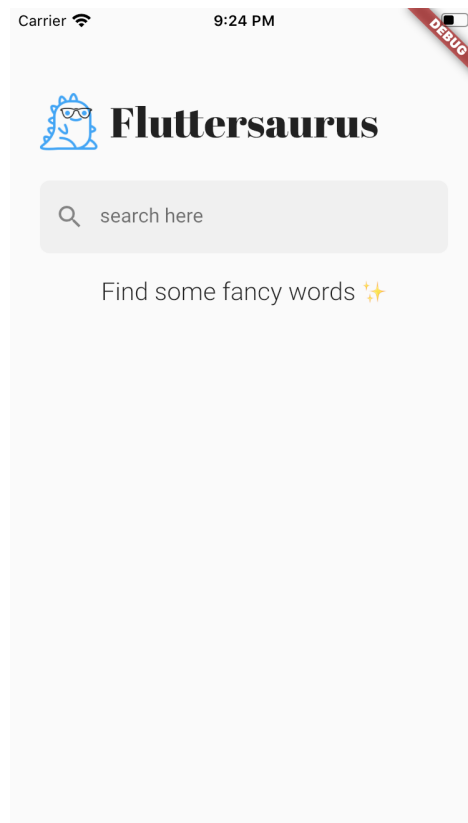
Bloc

```
enum CounterEvent { increment }  
  
class CounterBloc extends Bloc<CounterEvent, int> {  
  CounterBloc() : super(0);  
  
  @override  
  Stream<int> mapEventToState(CounterEvent event) async* {  
    switch (event) {  
      case CounterEvent.increment:  
        yield state + 1;  
        break;  
    }  
  }  
}
```

Cubit vs. Bloc

	Cubit	Bloc
Simple	✓	
Concise	✓	
Traceable		✓
ReactiveX Operators		✓
Testable	✓	✓
Scalable	✓	✓
Tooling Support	✓	✓

Fluttersaurus



https://github.com/felangel/flutter_saurus

Thanks! 🙏

Twitter @felangelov

Github @felangel