



**dream**  
**IMPLEMENTATION**

<http://dream-implementation.com>

[info@dream-implementation.com](mailto:info@dream-implementation.com)



# Streams

---

- Standard part of dart
- StreamController
  - StreamSink
  - Stream
- StreamSubscription
- Single-subscription and Broadcast Streams
- StreamTransformer

# RxDart

---

- <https://pub.dartlang.org/packages/rxdart>
- Stream -> Observable
- StreamController -> Subject
- PublishSubject
- BehaviorSubject
- ReplaySubject

# StreamBuilder

---

- Flutter's way to use Streams in layout
- Rebuilds when data is passed to a listened Stream



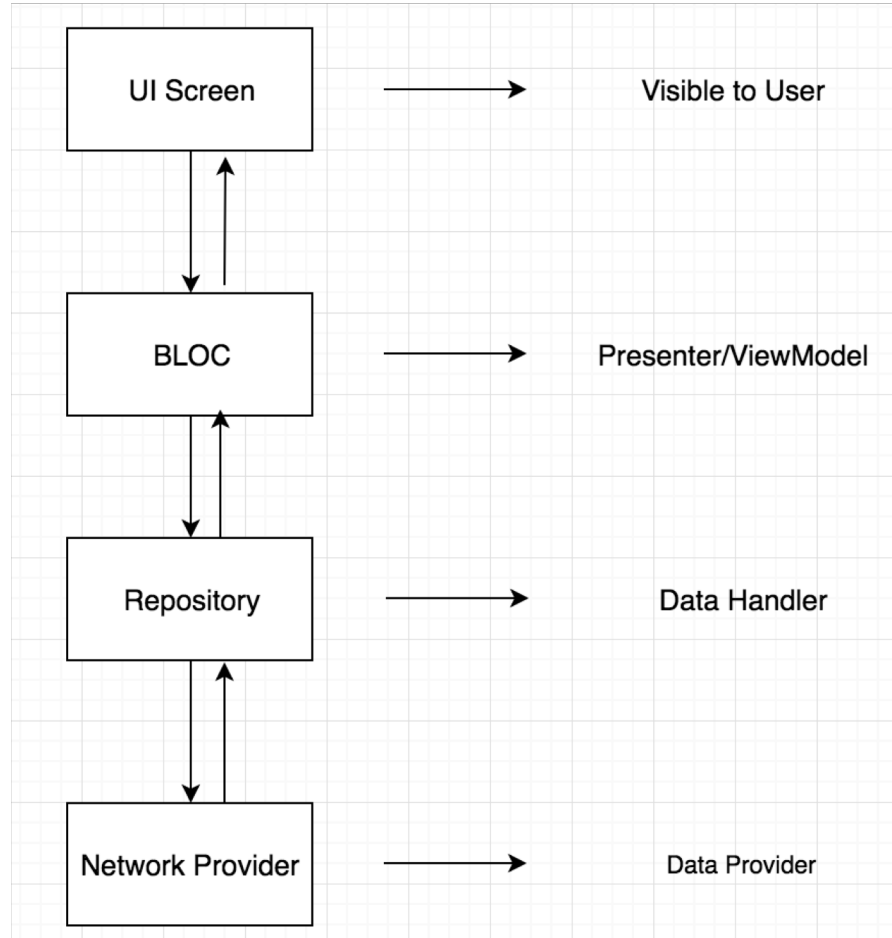
# BLoC

---

- Business Logic Component
- State management system
- Separates business logic from presentation layer
- Based on streams



# BLoC



# BLoC parts

---

- Bloc class
- Provider class
  - extends InheritedWidget
  - receives child widget and bloc object
  - static of method to get bloc object  
(`context.inheritFromWidgetOfExactType`)



# Useful resources

---

- <https://flutter.io/docs/development/data-and-backend/state-mgmt/intro>
- <https://www.didierboelens.com/2018/08/reactive-programming---streams---bloc/>
- <https://www.didierboelens.com/2018/12/reactive-programming---streams---bloc---practical-use-cases/>
- <https://medium.com/flutterpub/architecting-your-flutter-project-bd04e144a8f1>
- <https://medium.com/flutter-community/bloc-architecture-why-so-important-d9b29f06680e>
- <https://www.youtube.com/watch?v=RS36gBEp8OI>



# Thank you

Ivan Celija: [ivan@dream-implementation.com](mailto:ivan@dream-implementation.com)

Luka Knezić: [lukak@dream-implementation.com](mailto:lukak@dream-implementation.com)

