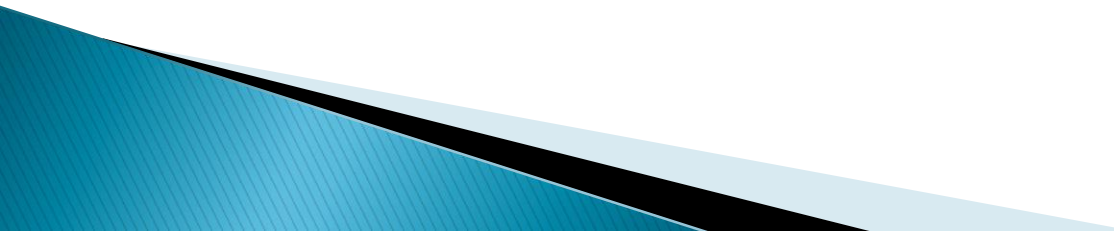


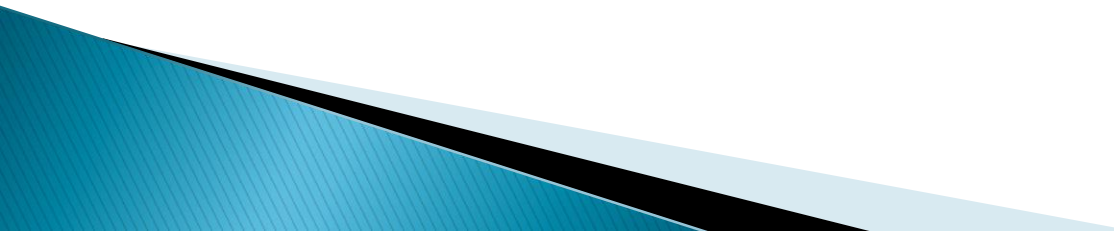
# Chapter 1

## Getting Started


# What is Flutter?

- Flutter is a software development toolkit from Google for building cross-platform apps.
  - Flutter apps consist of a series of packages, plugins and widgets — but that's not all.
  - Flutter is not is a language.
  - Flutter uses Dart as its programming language. If you know Kotlin, Swift, Java or Typescript, you'll find Dart familiar, since it's an object-oriented C-style language.
- 

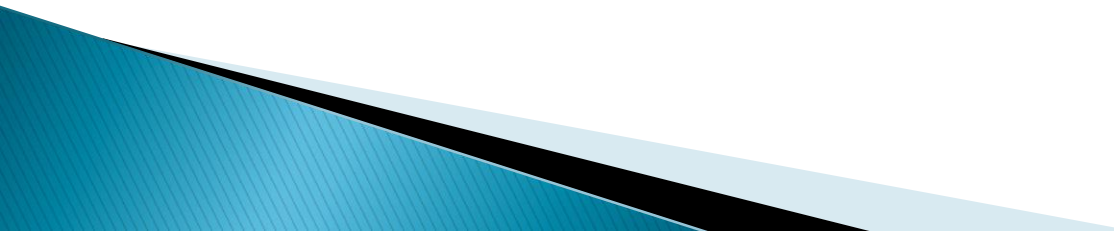
# What is Flutter? **cont**

- You can compile Dart to native code, which makes it fast.
  - It also uses a virtual machine (VM) with a special feature: **hot reload**.
  - This lets you update your code and see the changes live without redeploying it.
- 

# Who's Flutter for?

- Flutter is for both the new or experienced developer who wants to start a mobile app with minimal overhead.
  - Flutter is for someone looking to make an app that runs on multiple devices, either right away or in the future.
  - Flutter is for developers with experience on one platform who want to develop an app that works across many.
  - Flutter is a great way to develop something quickly to validate an idea or to build a full, multi-platform production app.
- 

# Great things about Flutter

- Flutter is **open-source**. That means you can watch its evolution and know what's coming — and even try out new features in development.
  - Flutter uses the **Dart** programming language.
  - It supports all the great language features people have come to like and expect, such as **async/await** for concurrency management and **type inference** for clean, type-safe code.
  - Hot restart takes a little longer than hot reload because it loads the changes, restarts the app and resets the state.
- 

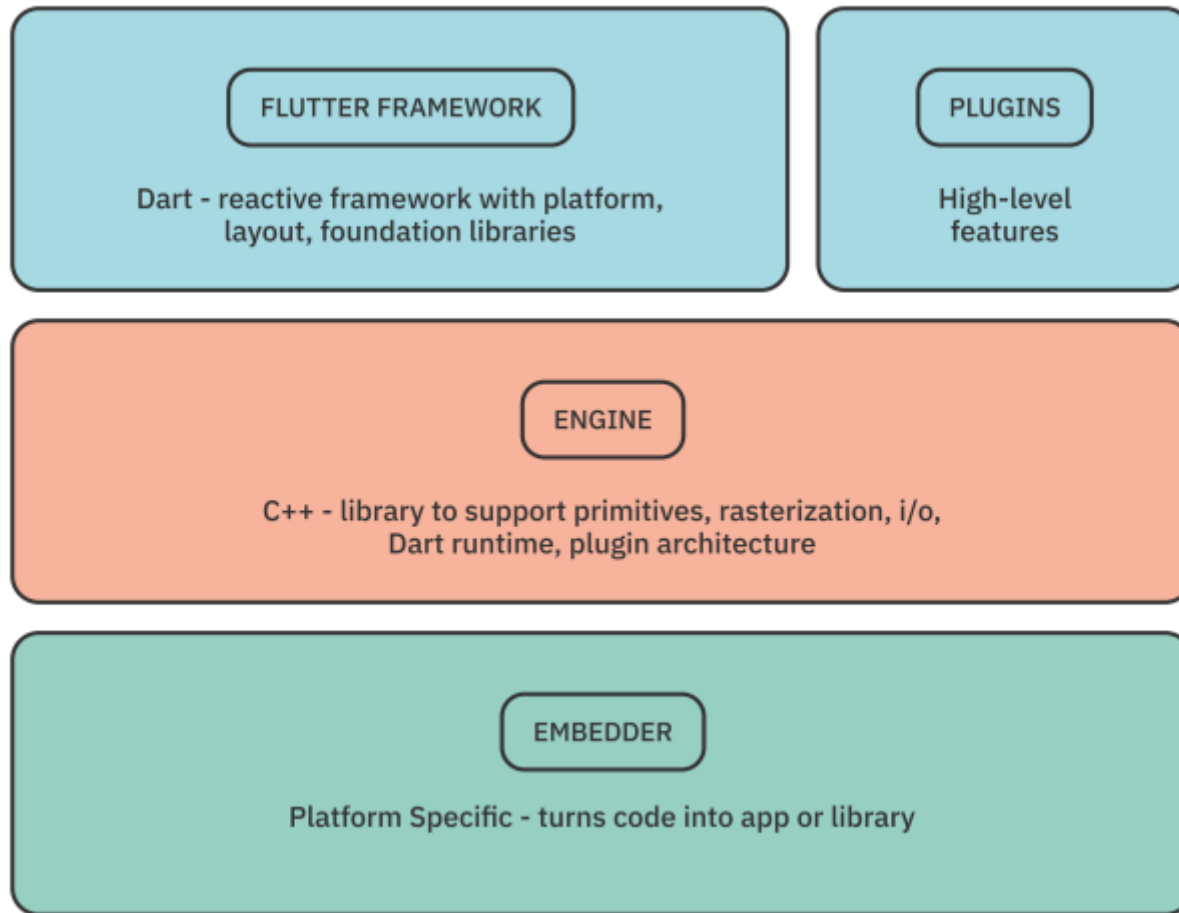
# Great things about Flutter **cont**

- Flutter comes with great animations and transitions, and you can build custom widgets as well.
- If you've used **SwiftUI** or **Jetpack Compose** recently, you're already familiar with many of Flutter's concepts.

# When not to use Flutter

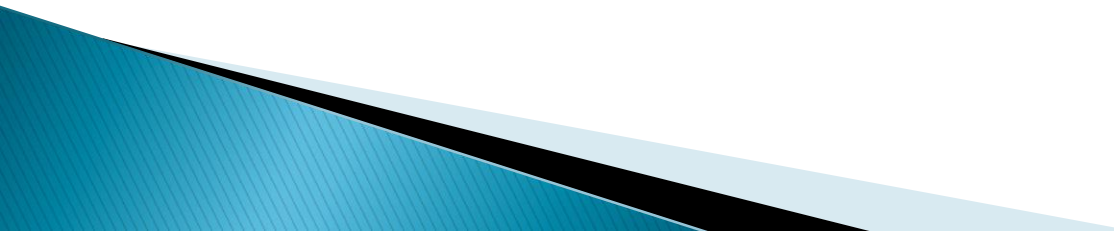
- Games and audio.
  - Apps with specific native SDK needs.
  - Certain platforms.
- 

# The Flutter architecture





# The Flutter architecture

- The **Framework** layer is written in Dart and contains the high-level libraries that you'll use directly to build apps.
  - The **Engine** layer contains the core C++ libraries that make up the primitives that support Flutter apps.
  - The **Embedder** is different for each target platform and handles packaging the code as a stand-alone app or embedded module.
- 

# What you need

- A **computer**. You can develop Flutter apps on Windows, macOS, Linux or ChromeOS. However, Xcode only runs on macOS, making a Mac necessary to build and deploy apps for iOS.
- The **Flutter SDK**.
- An **editor**, such as Android Studio or Visual Studio Code.
- At least one **device**. You can run in an iOS Simulator or Android emulator, but running Flutter apps on a physical device will give you the true user experience.
- **Developer accounts** (optional). To deploy to the Apple App Store or Google Play Store, you'll need a valid account on each

# What you need

- A **computer**. You can develop Flutter apps on Windows, macOS, Linux or ChromeOS. However, Xcode only runs on macOS, making a Mac necessary to build and deploy apps for iOS.
- The **Flutter SDK**.
- An **editor**, such as Android Studio or Visual Studio Code.
- At least one **device**. You can run in an iOS Simulator or Android emulator, but running Flutter apps on a physical device will give you the true user experience.
- **Developer accounts** (optional). To deploy to the Apple App Store or Google Play Store, you'll need a valid account on each