

Uma Ponte entre seu Flutter App e Funções Nativas

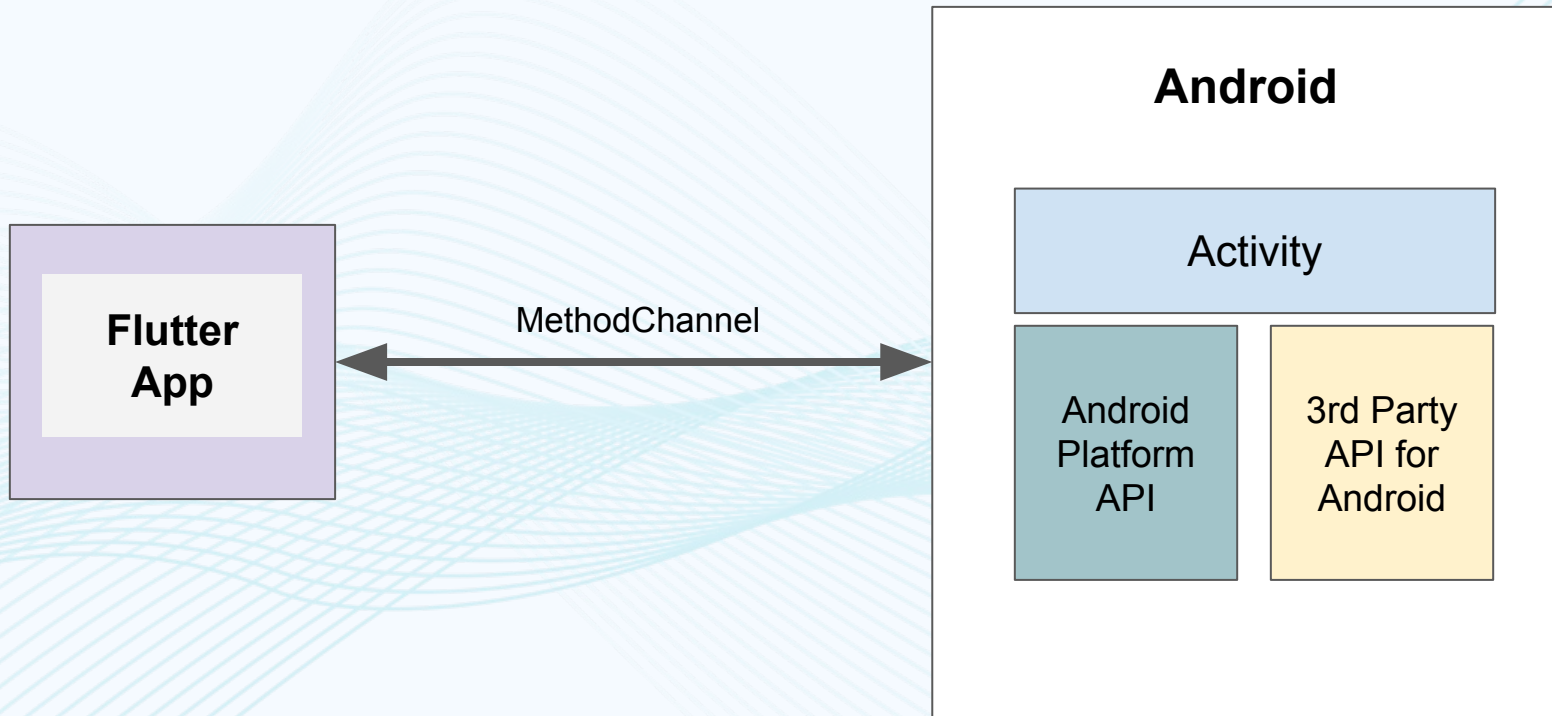
Entendendo o Method Channel

por Breno Araújo



Flutter Platform Channel

- Permite invocar código específico de plataforma;
- Comunicação bidirecional e assíncrona;
- Serialização e desserialização automáticos;
- Suporte à várias linguagens (Kotlin, Java, Swift, Objective-C, C++, ...)



Dart	Java
null	null
bool	java.lang.Boolean
int	java.lang.Integer
double	java.lang.Double
String	java.lang.String
List	java.util.ArrayList
Map	java.util.HashMap
Uint8List	byte[]
Int32List	int[]
Int64List	long[]



```
static const platformMethodChannel = const MethodChannel('com.demo/nerdfriday');
```


Lado Flutter



```
Future<Null> criarEstudante() async {  
  String _msg;  
  try {  
    final String result =  
      await platformMethodChannel.invokeMethod(  
        "criarEstudante",  
        {  
          "nome": "Breno Araujo",  
          "matricula": 123  
        }  
      );  
    _msg = result;  
    // print(result);  
  } on PlatformException catch(exp) {  
    _msg = "Deu muito errado óh ! ${exp.message}";  
  }  
}
```

Lado Java

```
public class MainActivity extends FlutterActivity {
    private static final String CHANNEL = "com.demo/nerdfriday";
    @Override
    public void configureFlutterEngine(@NonNull FlutterEngine flutterEngine) {
        GeneratedPluginRegistrant.registerWith(flutterEngine);
        new MethodChannel(flutterEngine.getDartExecutor().getBinaryMessenger(), CHANNEL)
            .setMethodCallHandler(
                (call, result) → {
                    switch(call.method) {
                        case "criarEstudante":
                        case "metodo2":
                        case "metodo3":
                    }
                }
            );
    }
}
```

Código Fonte: <https://github.com/brenoaraujolina/method-channel-demo>

Obrigado !