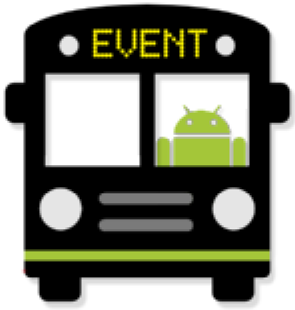
Event-bus archiectural pattern - used in Android app development and Push Notification Service

Event-bus architecture pattern is a distributed asynchronous architecture pattern to create highly scalable reactive applications. The pattern suits for every level application stack from small to complex ones. The main idea is delivering and processing events asynchronously.

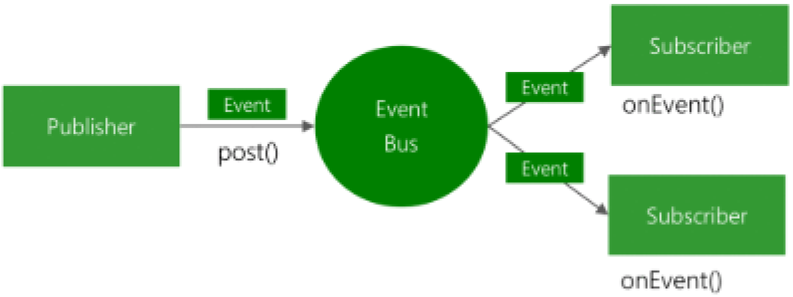
Gliffy Macro Error

You do not have permission to view this diagram.

There are a couple of ways to implement event-driven pattern to your stack. One of them is using “Broker Topology”. In broker topology, you send events to a central broker and all subscribers of these broker receive and process events asynchronously.



Android is also known as using Event-bus architecture. In Android, the main reason to use Event-Bus is loose coupling. Sometimes, you want to process specific events that are interested in multiple parts of your application, like the presentation layer, business layer, and data layer, so EventBus provides an easy solution for this.



The below is another case of building event-bus as backbone for decoupled microservice choreography.

Gliffy Macro Error

You do not have permission to view this diagram.