

Defining the Data Sources: Local and Remote



Kaushal Dhruw

APP DEVELOPER / AUTHOR

@drulabs

linkedin.com/in/kaushal-dhruw/

Overview



Room - AAC

Local data source with Room

Testing local data source

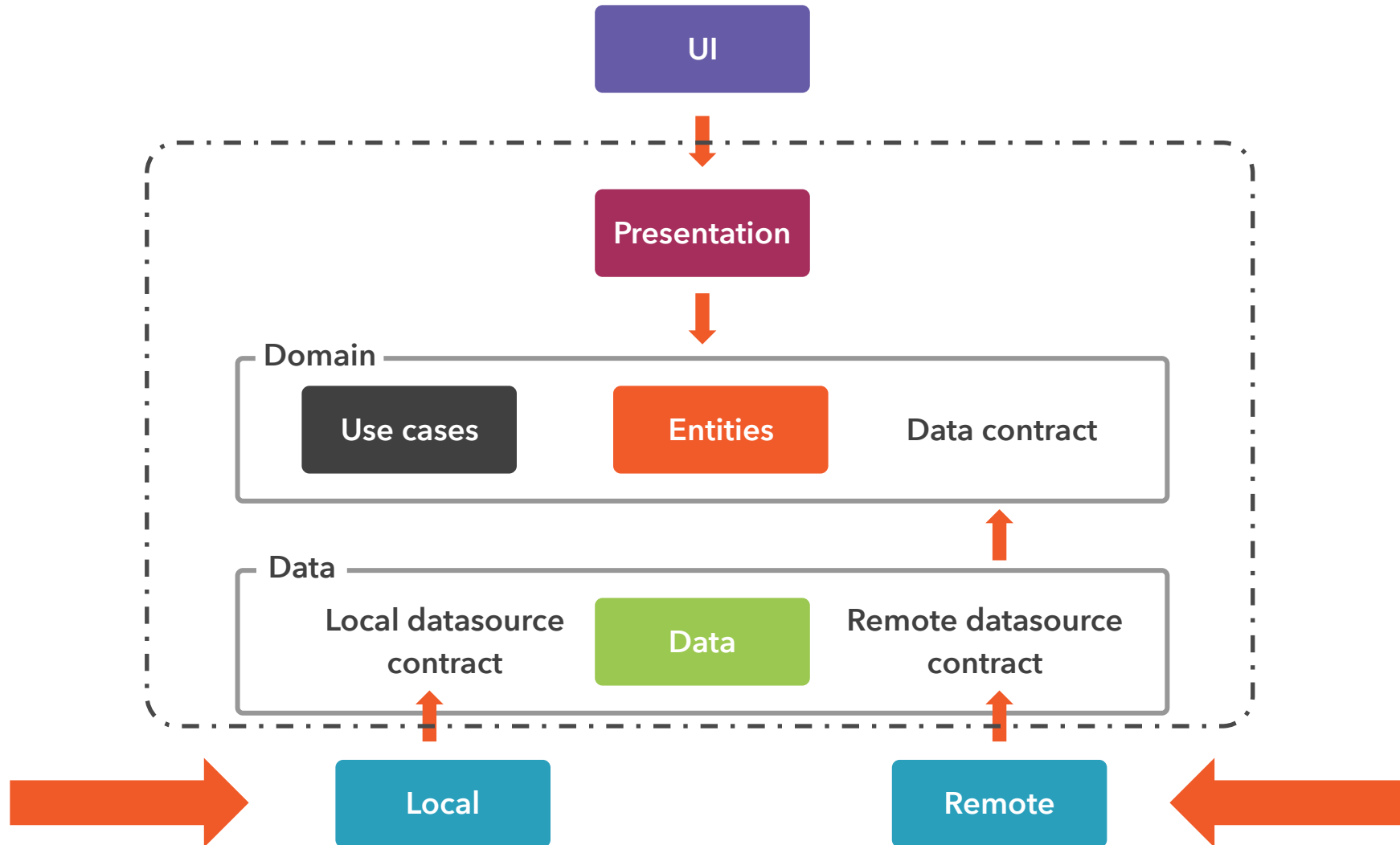
Remote data source with Retrofit

Testing remote data source

Why so many data classes?

Summary

Banking App Layers



Yes. Switching the network
and persistence library is
this easy.

Room Persistence Library

Major Components of Room

@Database

- Represents a Database
- Is an abstract class
- Provides DAO (@Dao)

@Dao

- Represents Data Access Objects
- Can be an interface or abstract class
- DB operations

@Entity

- Represents a database table
- A POJO or Kotlin data class
- Every object is a row in database

Why Room?

Supported by Google

No SQL for simple database operations

No boilerplate for converting data classes

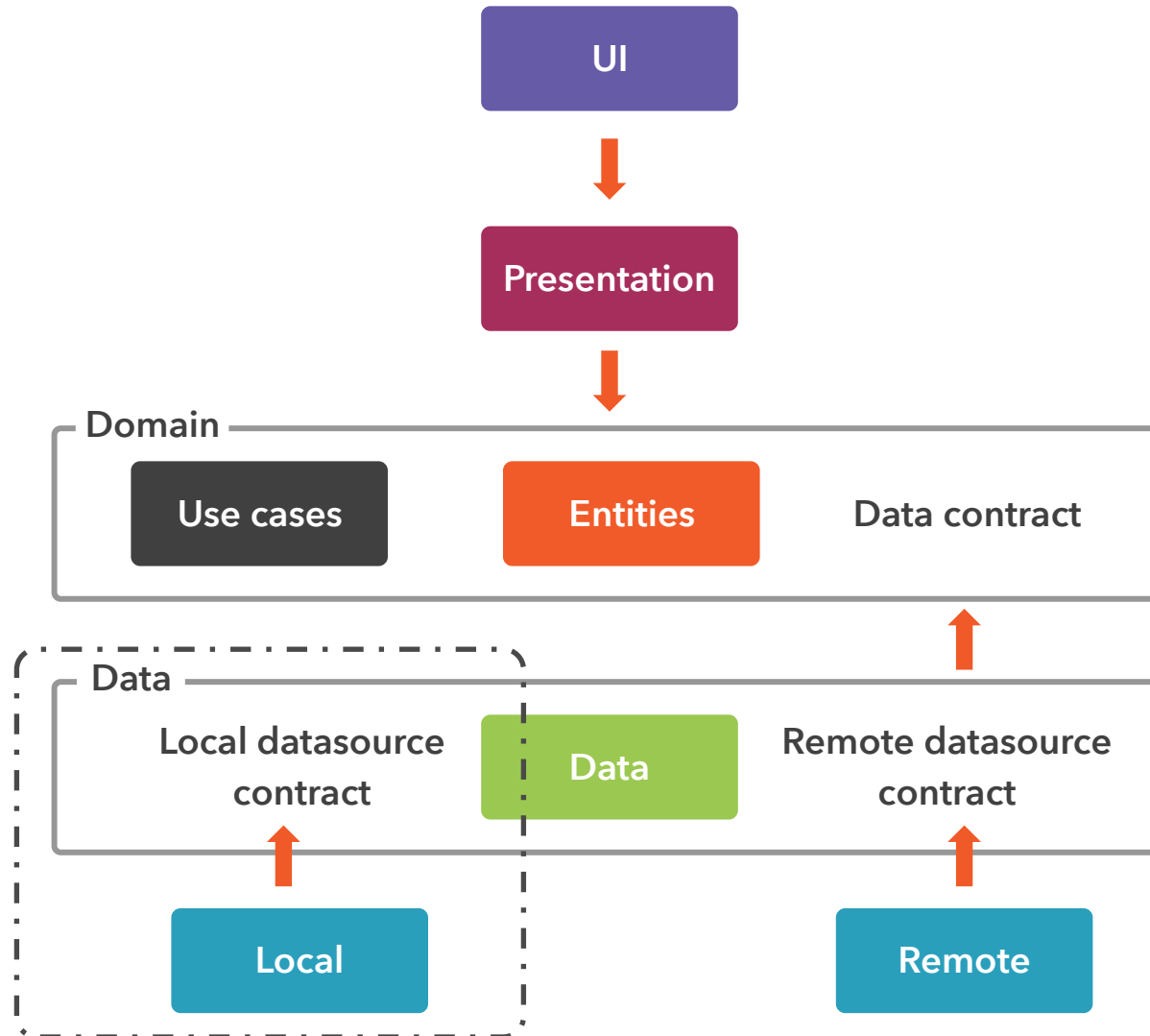
SQL Query validation in compile time

Live updates with LiveData<T>

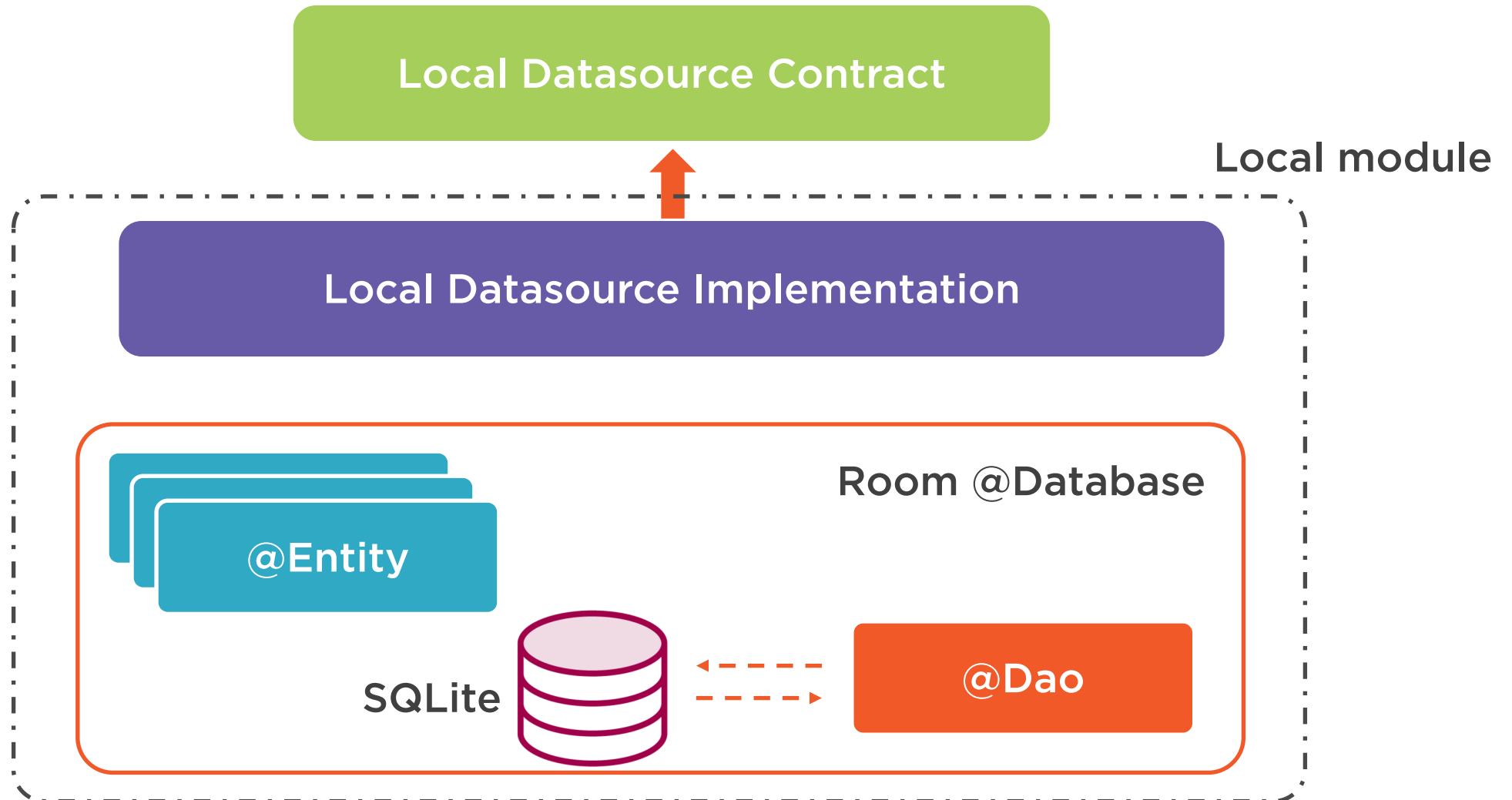
Supports RxJava Observables

Easy testing

Local Datasource Module

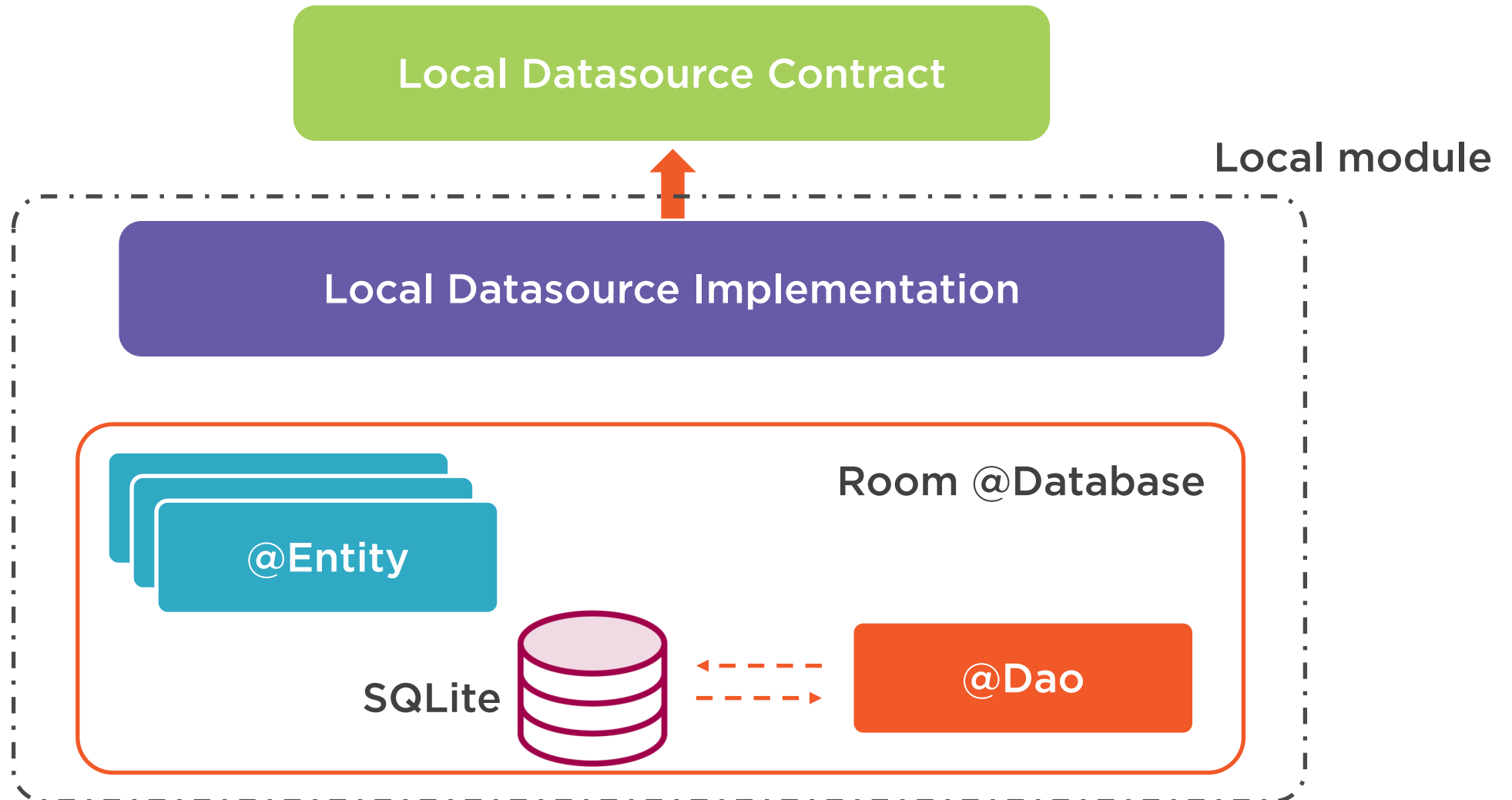


Local Datasource Module



Defining the Local Datasource

Local Datasource Module



Demo



Create relevant entities

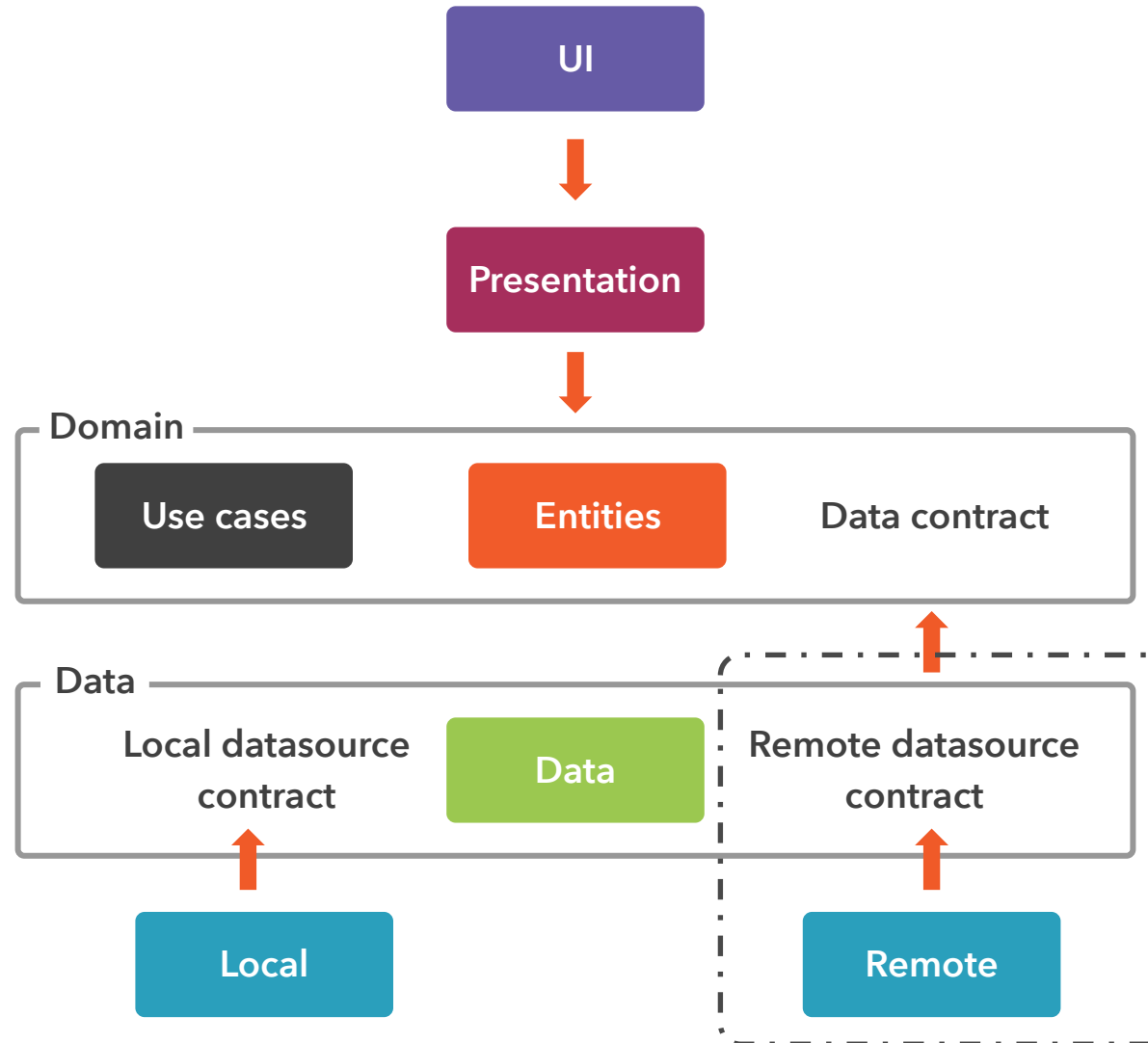
Create Room components

Implement LocalDataSource interface

Testing local datasource layer

Defining the Remote Datasource

Retrofit in the Banking App



Demo



Define the remote APIs using Retrofit

Implement RemoteDataSource

Test the remote data source layer

Why the Redundant Data Classes?

A Typical Data Class

```
data class User(  
    val name: String,  
    val age: Int,  
    ....  
)
```

A Typical Data Class

```
data class User(  
    @SerializedName("user-name")  
    val name: String  
)
```

A Typical Data Class

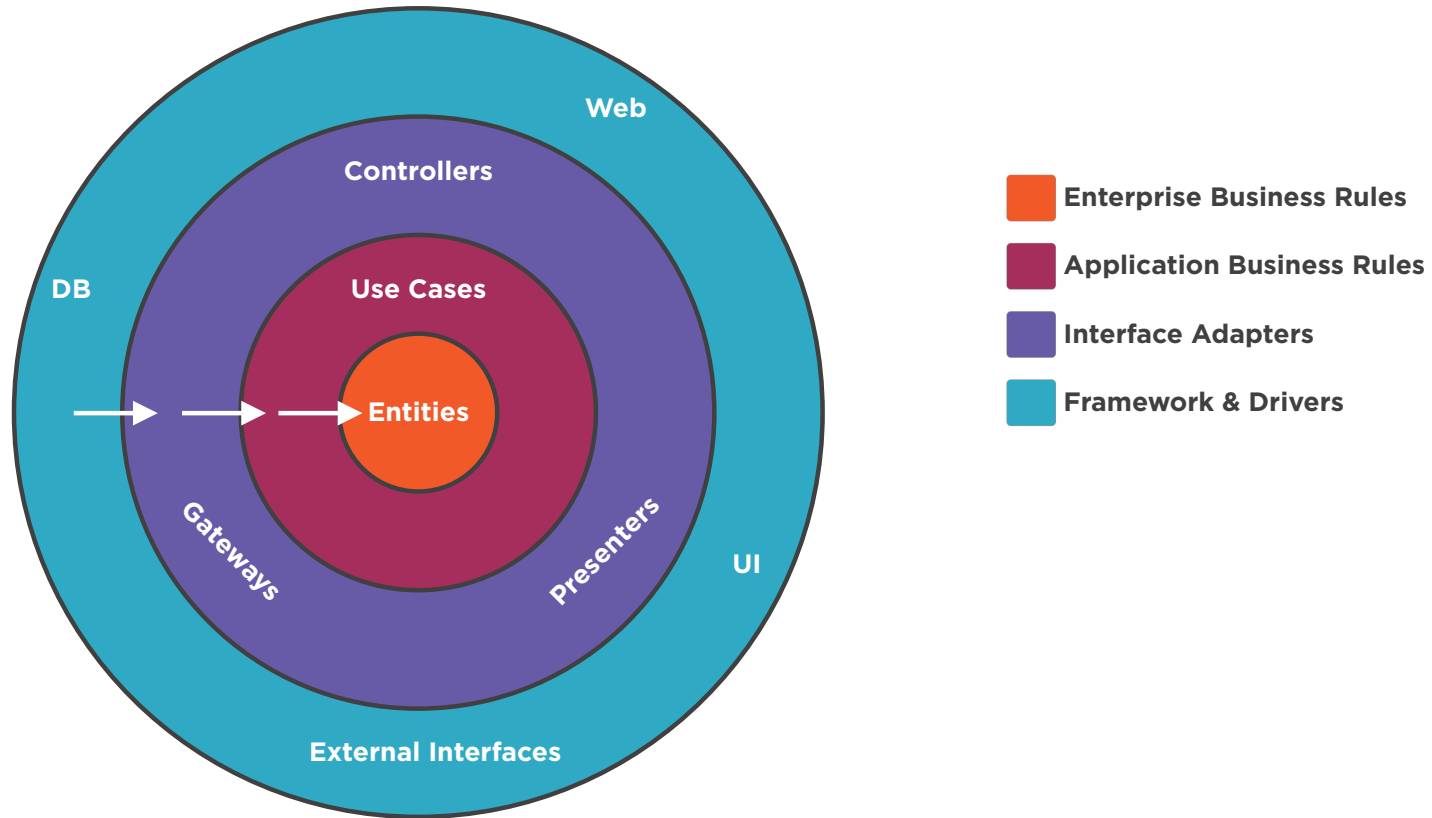
@Entity

```
data class User(  
    @SerializedName("user-name")  
    @PrimaryKey  
    @ColumnInfo(name = "column_user_name")  
    val name: String  
)
```

A Typical Data Class

```
@Entity(name)
data class User(
    @SerializedName("user-name")
    @PrimaryKey
    @ColumnInfo(name = "column_user_name")
    @Exclude
    val name: String
)
```

Layers of CLEAN Architecture



Summary



Room – AAC

Local data source layer with Room

Unit and instrumentation testing

Networking with Retrofit

Remote layer with Retrofit

Remote layer testing

Reason for duplicate data classes