11

Compose로 구현하는 반응형 앱

App 개발팀 이영우

TechTalk 2023

이영우

11번가 App개발팀

11번가 주식회사 App 개발팀

CONTENTS

11 TechTalk 2023

1. 시작에 앞서

11 TechTalk 2023

이 시간은 다음과 같은 분들을 대상으로 안내하는 시간입니다.

1. 앱을 신규로 개발 또는 기존 앱을 새롭게 리뉴얼을 고려 중인 경우

2. Compose를 새롭게 배워보고 싶지만 기존 개발 방식에 비한 장단점을 알고 싶음

3. 동료들과 Compose로 개발하고 싶은데 주위 사람을 설득하기 힘들때

2. Compose의 차이점

명령형 UI에서의 화면 그리기

2. Compose의 차이점

명령형 UI를 코드로 구현 할 때의 모습

1. XML Layout에 각각의 View를 미리 구성하여 배치

2. Extra Layer에서는 각각의 View를 읽어와서 (findViewById or binding)

3. 해당 View에 데이터에서 가져온 값을 셋팅하거나 상황에 맞게 노출

```
TextView sampleTitle = rootView.findViewById(R.id.sample_title);
TextView sampleMessage = rootView.findViewById(R.id.sample
ImageView sampleImage = rootView.findViewById(R.id.sampel_
```

```
sampleTitle.setText(model.titleText);
sampleMessage.setText(model.messageText);
if (model.imageVisibility) {
    sampleImage.setVisibility(View.VISIBLE);
} else {
    sampleImage.setVisibility(View.GONE);
}
```

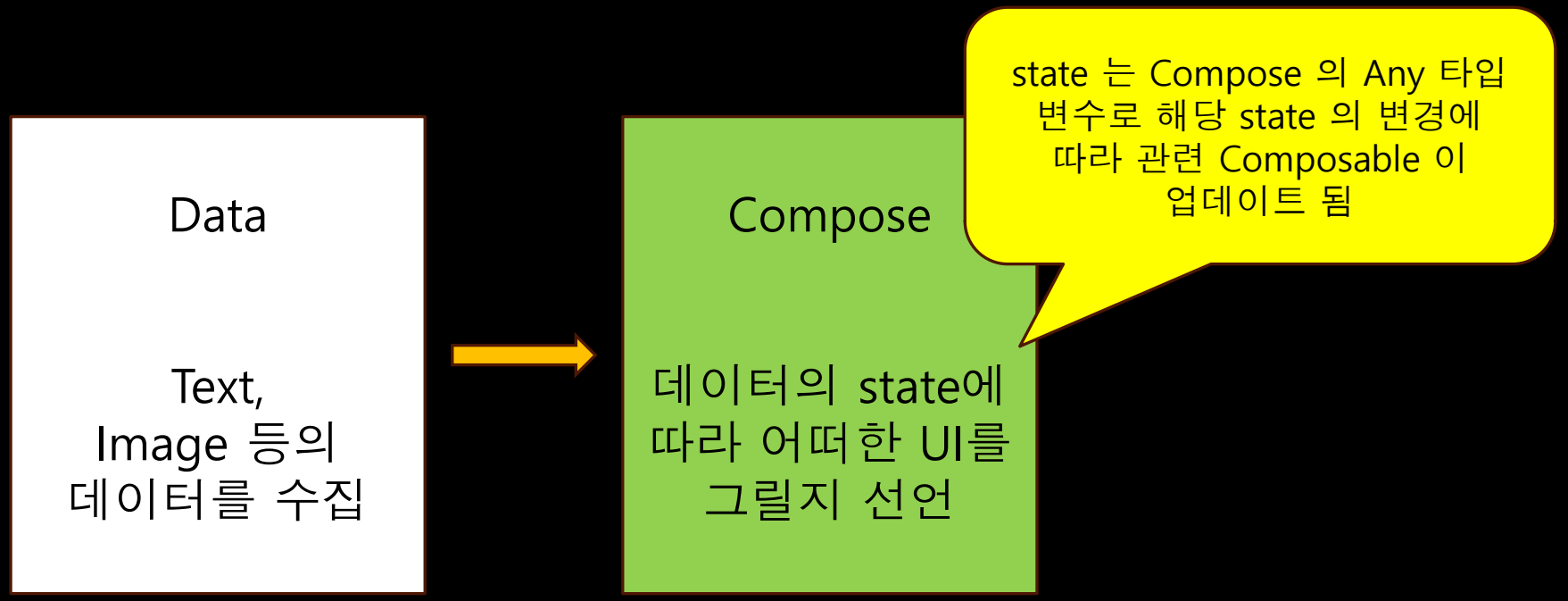
```
<TextView
    android:id="@+id/sample_title"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textSize="20sp"
    android:visibility="visible"/>

<TextView
    android:id="@+id/sample_message"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textSize="15sp"
    android:visibility="visible"/>

<ImageView
    android:id="@+id/sample_image"
    android:layout_width="50dp"
    android:layout_height="30dp"
    android:src="@drawable/logo" />
```

2. Compose의 차이점

선언형 UI에서의 화면 그리기



2. Compose의 차이점

선언형 UI를 코드로 구현 할 때의 모습

1. 데이터의 State에 따라 바로 해당 컴포넌트를 선언하고 데이터 셋팅

```
@Composable
fun SampleView(model: SampleModel) {
    Column {
        Text(model.sampleTitle)
        Text(model.sampleMessage)
        if (model.imageVisibility) {
            Image(
                painter = painterResource(id = R.drawable.logo),
                contentDescription = stringResource(id = R.string.sample_description)
            )
        }
    }
}
```

3. App에서 반응형 UI의 필요성

11 TechTalk 2023

다양한 기기 폼 팩터의 증가

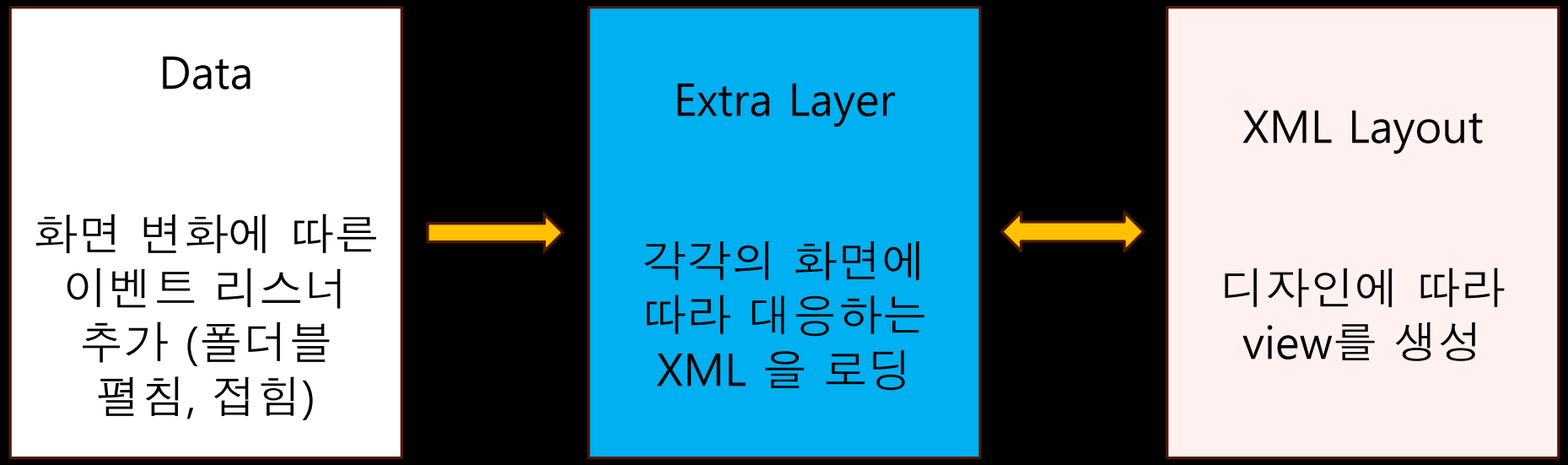
다 지원하기는 힘들지만 그렇다고 지원을 안할수는 없는 상황

Compose가 완벽한 대안은 아니지만 괜찮은 대안 중 하나

4. Compose로 만드는 반응형 UI

11번가 TechTalk 2023

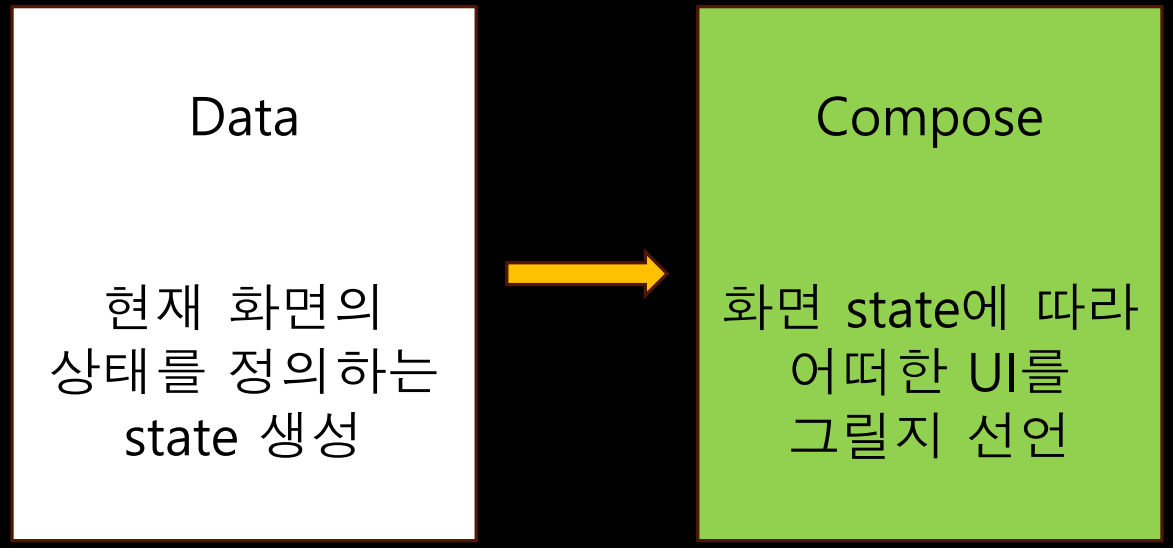
명령형 UI 의 앱에서 화면 변화에 따른 개발



4. Compose로 만드는 반응형 UI

11 TechTalk 2023

선언형 UI 의 앱에서 화면 변화에 따른 개발



4. Compose로 만드는 반응형 UI

11TechTalk 2023

Compose 에서는 다음과 같이 화면에 따라 대응하도록 개발이 가능합니다 (AS-IS)

```
MaterialTheme {
    NavHost(navController = navController, startDestination = "home") {
        val modifier : Modifier = Modifier.fillMaxSize()
        composable( route: "home") {
            DrawTabs(modifier, pages, onSelectionChange, pagerState, navController, onClicked)
        }
        composable( route: "detail") {
            DrawDetails(modifier, itemCount.value)
        }
    }
}
```

4. Compose로 만드는 반응형 UI

11 TechTalk 2023

Compose 에서는 다음과 같이 화면에 따라 대응하도록 개발이 가능합니다 (TO-BE)

```
when (rememberWidthSizeClass()) {
    WidthSizeClass.COMPACT -> {
        NavHost(navController = navController, startDestination = "home") {
            val modifier : Modifier = Modifier.fillMaxSize()
            composable( route: "home") {
                DrawTabs(modifier, pages, onSelectionChange, pagerState, navController, onClicked)
            }
            composable( route: "detail") {
                DrawDetails(modifier, itemCount.value)
            }
        }
    }
    WidthSizeClass.WIDE -> {
        Row(modifier = Modifier.fillMaxSize()) {
            val modifier : Modifier = Modifier.fillMaxSize().weight(1f)
            DrawTabs(modifier, pages, onSelectionChange, pagerState, navHostController: null, onClicked)
            if (itemCount.value >= 0) {
                DrawDetails(modifier, itemCount.value)
            }
        }
    }
}
```

5. 마무리

1. Compose는 현재 화면의 state 관리 만으로 다양한 환경의 UI에 대응하여 개발 가능

2. 익숙해진다면 개발이 빠르고 코드 양이 줄어 개발 및 유지보수가 좋아짐
   (단 복잡한 UI 가 많은 경우 개발이 힘들어 질 수 있으니 주의)

3. 상황에 따라 화면 개발의 복잡함을 많이 줄여줄 수 있음

Q&A

E-mail) lee.youngwoo@sk.com

Thank you