

MOBILE PROGRAM

PERTEMUAN 7

PLATFORM ARSITEKTUR

Android terdiri dari

- Linux Kernel
- Hardware Abstraction Layer (HAL)
- Android Runtime
- Native C/C++ Libraries
- Java API Framework
- System Apps

LINUX KERNEL

Landasan platform Android adalah kernel Linux. Sebagai contoh, Android Runtime (ART) bergantung pada kernel Linux untuk fungsi-fungsi seperti threading dan manajemen memori tingkat rendah.

- Menggunakan kernel Linux memungkinkan Android untuk mengambil keuntungan dari fitur keamanan utama dan memungkinkan produsen perangkat untuk mengembangkan driver perangkat keras untuk kernel yang terkenal.

Linux Kernel

Drivers

Audio

Binder (IPC)

Display

Keypad

Bluetooth

Camera

Shared Memory

USB

WIFI

Power Management



HARDWARE ABSTRACTION LAYER (HAL)

Hardware Abstraction Layer (HAL) menyediakan antarmuka standar yang mengekspos kemampuan perangkat keras perangkat ke kerangka Java API tingkat tinggi.

- HAL terdiri dari beberapa modul perpustakaan, yang masing-masing mengimplementasikan antarmuka untuk jenis komponen perangkat keras tertentu, seperti kamera atau modul bluetooth.
- Ketika API framework membuat panggilan untuk mengakses perangkat keras perangkat, sistem Android memuat modul perpustakaan untuk komponen perangkat keras itu.

Hardware Abstraction Layer (HAL)

Audio

Bluetooth

Camera

Sensors

...

ANDROID RUNTIME

Untuk perangkat yang menjalankan Android versi 5.0 (API level 21) atau lebih tinggi, setiap aplikasi berjalan dalam prosesnya sendiri dan dengan instance Android Runtime (ART) sendiri.

- ART ditulis untuk menjalankan beberapa mesin virtual pada perangkat dengan memori rendah dengan mengeksekusi file DEX, format bytecode yang dirancang khusus untuk Android yang dioptimalkan untuk jejak memori minimal.

Android Runtime

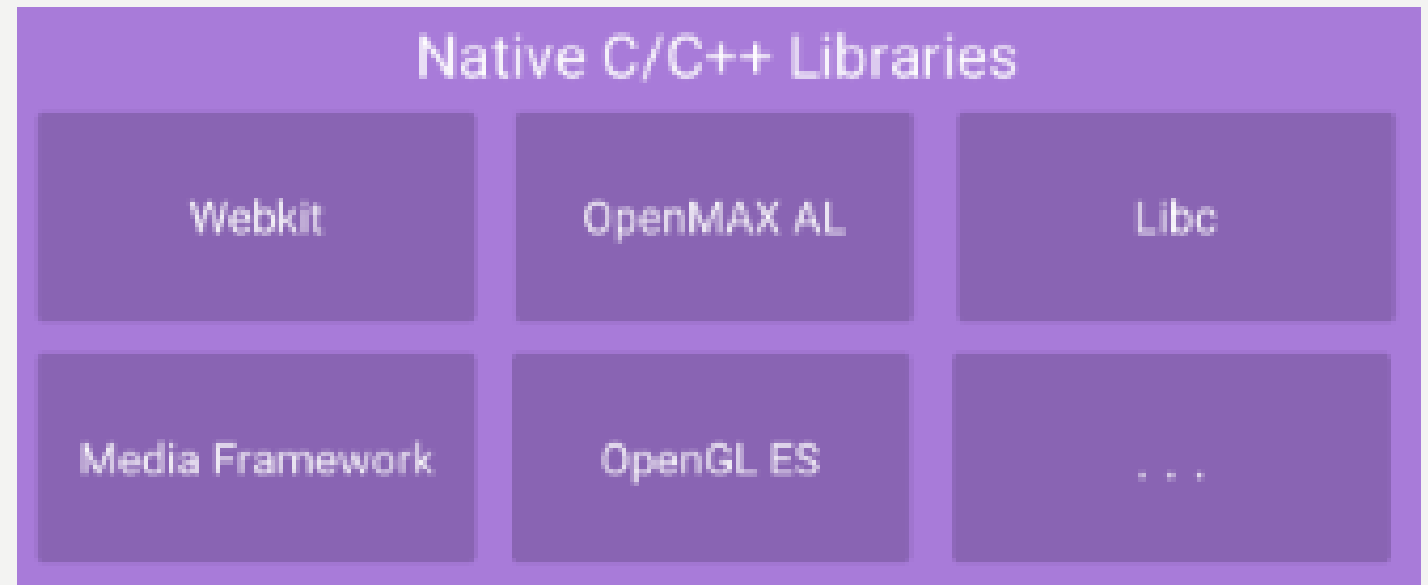
Android Runtime (ART)

Core Libraries

NATIVE C/C++ LIBRARIES

Banyak komponen dan layanan sistem inti Android, seperti ART dan HAL, dibangun dari kode asli yang memerlukan pustaka asli yang ditulis dalam C dan C++.

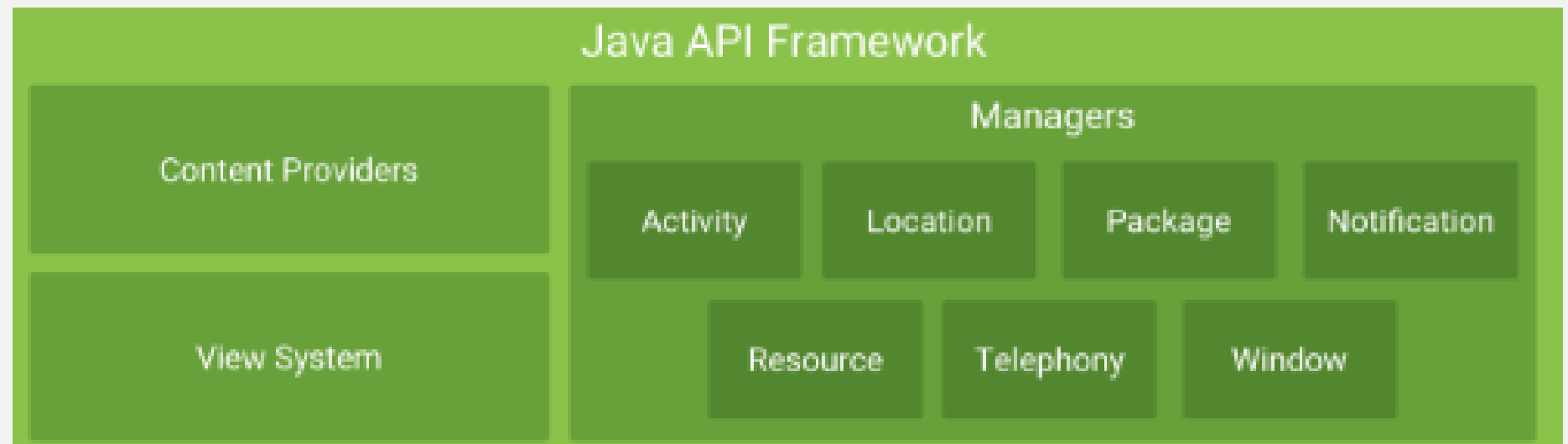
Platform Android menyediakan API framework Java untuk mengekspos fungsionalitas beberapa pustaka asli ini ke aplikasi



JAVA API FRAMEWORK

Seluruh rangkaian fitur OS Android tersedia untuk Anda melalui API yang ditulis dalam bahasa Java.

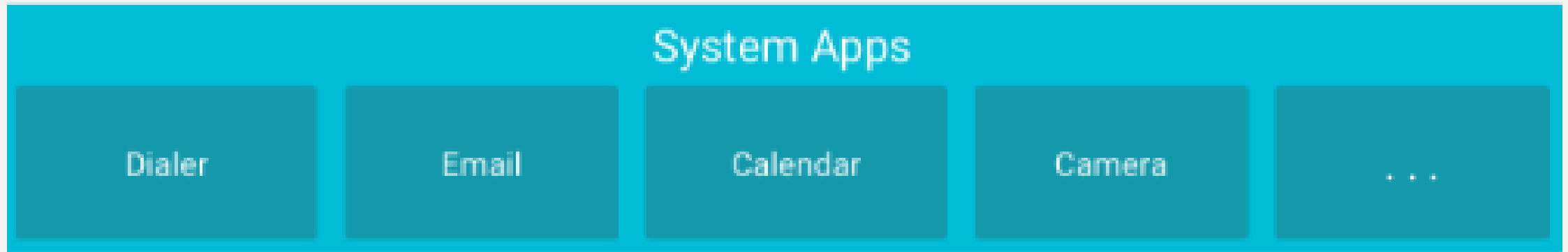
API ini membentuk blok bangunan yang Anda butuhkan untuk membuat aplikasi Android dengan menyederhanakan penggunaan kembali inti, komponen dan layanan sistem modular



SYSTEM APPS

Android hadir dengan serangkaian aplikasi inti untuk email, pengiriman pesan SMS, kalender, penelusuran internet, kontak, dan banyak lagi.

Aplikasi yang disertakan dengan platform tidak memiliki status khusus di antara aplikasi yang dipilih pengguna untuk diinstal.



KOTLIN PROGRAMMING

Fitur bahasa modern Kotlin memungkinkan Anda untuk fokus pada mengekspresikan ide-ide Anda dan menulis kode yang lebih sederhana. Lebih sedikit kode yang ditulis juga berarti lebih sedikit kode untuk diuji dan dirawat.

Tingkatkan kualitas aplikasi Anda dengan Kotlin. Tipe `@Nullable` dan `@NonNull` dimasukkan ke dalam sistem tipe Kotlin untuk membantu Anda menghindari `NullPointerExceptions`.

Panggil kode berbasis Java dari Kotlin, atau panggil Kotlin dari kode berbasis Java. Kotlin 100% dapat dioperasikan dengan bahasa pemrograman Java,

KOTLIN

```
class MainActivity : AppCompatActivity() {
```

```
    override fun onCreate(savedInstanceState: Bundle?) {
```

```
        ...
```

```
        fab.setOnClickListener { view ->
```

```
            Snackbar.make(view, "Hello $name", Snackbar.LENGTH_LONG).show()
```

```
        }
```

```
    }
```

```
}
```

Nullable and NonNull
types help reduce
NullPointerExceptions

Use lambdas for concise
event handling code

Use template expressions
in strings to avoid concatenation

Semicolons are optional

FITUR KOTLIN

Keringkasan

- Sifat ringkas Kotlin menyederhanakan pekerjaan pengembang dan mengurangi risiko kesalahan, tetapi Kotlin tidak mempraktikkan pengambilan keputusan hanya karena pengambilan keputusan.
- Interoperabilitas
- Pengembang cukup menulis modul di Kotlin yang bekerja dengan sempurna di dalam kode Java yang ada.

CONT'D

Keamanan Null

- `NullPointerException` bertanggung jawab atas kesalahan pengembangan Android. Kotlin memecahkan masalah ini dengan memasukkan keamanan null yang melekat. Tambahan ini menyelamatkan pengembang dari menulis kode tambahan untuk mengatasi masalah ini.
- Tidak Ada Raw Type
- Tipe RAW memungkinkan untuk kompatibilitas ke belakang, tetapi tipe RAW dapat melempar `CastClassException` dan kesalahan akan terjadi selama eksekusi dan bukan pada tahap kompilasi. Kotlin tidak mengizinkan tipe mentah, dan sebagai hasilnya, menghasilkan kode yang lebih aman.

CONTOH

```
override fun onCreate(savedInstanceState: Bundle?) {  
    super.onCreate(savedInstanceState)  
    setContentView(R.layout.activity_main)  
  
    // Inisialisasi DB  
    usersDBHelper = UsersDBHelper(this)  
  
    var resultText = findViewById<EditText>(R.id.result)
```

XAMARIN ANDROID

Framework yang disediakan oleh Microsoft dengan menggunakan C# atau F#

- Xamarin adalah platform open-source untuk membangun aplikasi modern dan berkinerja untuk iOS, Android, dan Windows dengan .NET.
- Xamarin adalah lapisan abstraksi yang mengelola komunikasi kode bersama dengan kode platform yang mendasarinya. Xamarin beroperasi di lingkungan yang dikelola yang memberikan kemudahan seperti alokasi memori dan pengumpulan sampah (Memori).

CARA KERJA XAMARIN



Android Native UI



iOS Native UI

Your Shared Application Code (C#/XAML)

.NET
APIs

Android
Bindings

MCW

Android.*

Java.*

Mono

ACW

Android Runtime
(ART)

Linux Kernel

.NET APIs

Bindings

iOS APIs

Mono (Full AOT)

Objective-C

Unix-Like Kernel

CONT'D

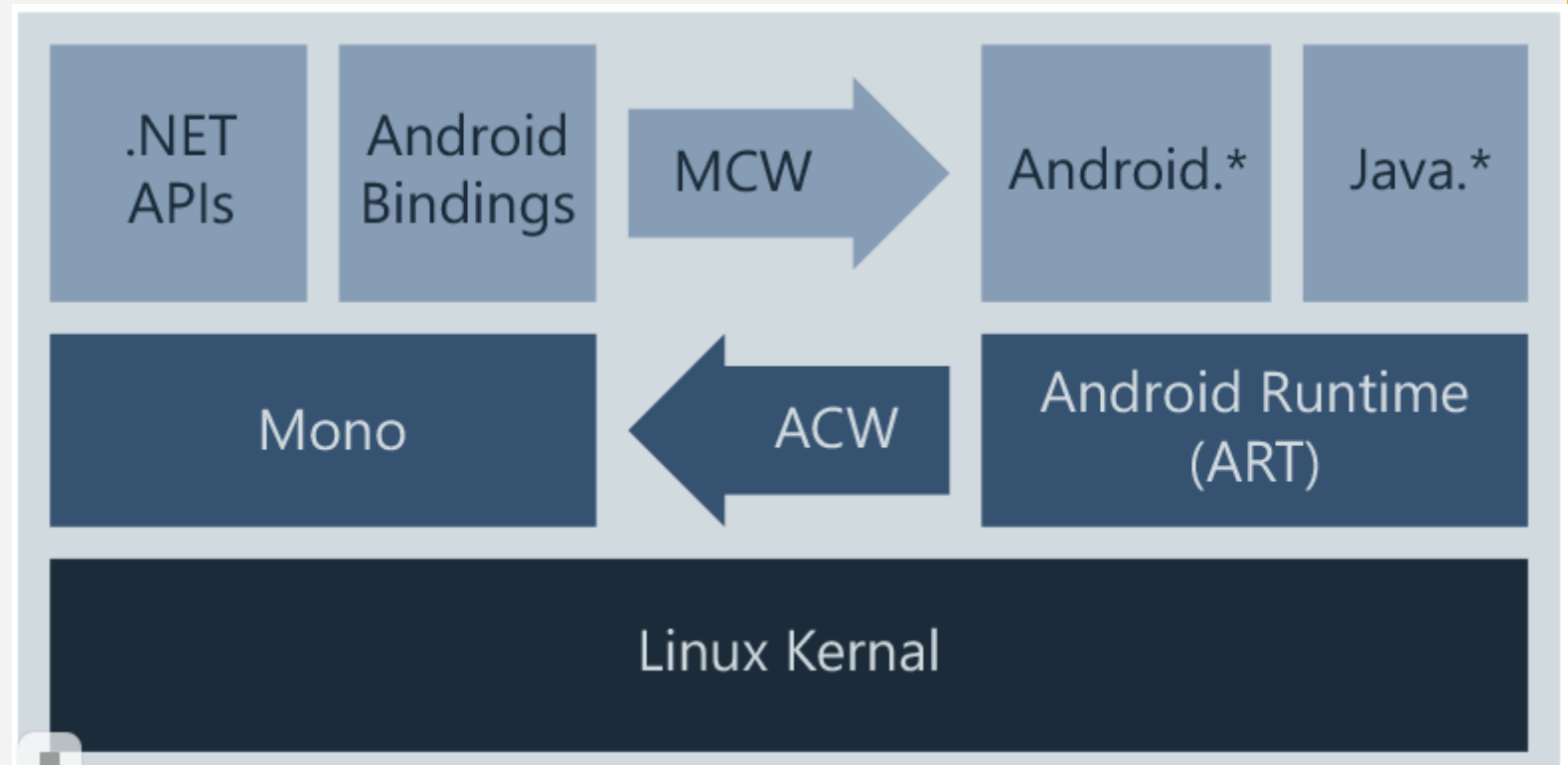
Xamarin dibangun di atas Mono, versi open-source dari .NET Framework berdasarkan standar .NET ECMA.

Mono telah ada hampir selama .NET Framework itu sendiri, dan berjalan pada sebagian besar platform termasuk Linux, Unix, FreeBSD, dan macOS.

Lingkungan eksekusi Mono secara otomatis menangani tugas-tugas seperti alokasi memori, pengumpulan sampah, dan interoperabilitas dengan platform yang mendasarinya.

CONT'D

- Aplikasi Xamarin.Android mengkompilasi dari C# ke Bahasa Intermediate (IL) yang kemudian Just-in-Time (JIT) dikompilasi ke Native Assembly ketika aplikasi diluncurkan.
- Aplikasi Xamarin.Android berjalan dalam lingkungan eksekusi Mono, berdampingan dengan mesin virtual Android Runtime (ART).





TERIMA KASIH

