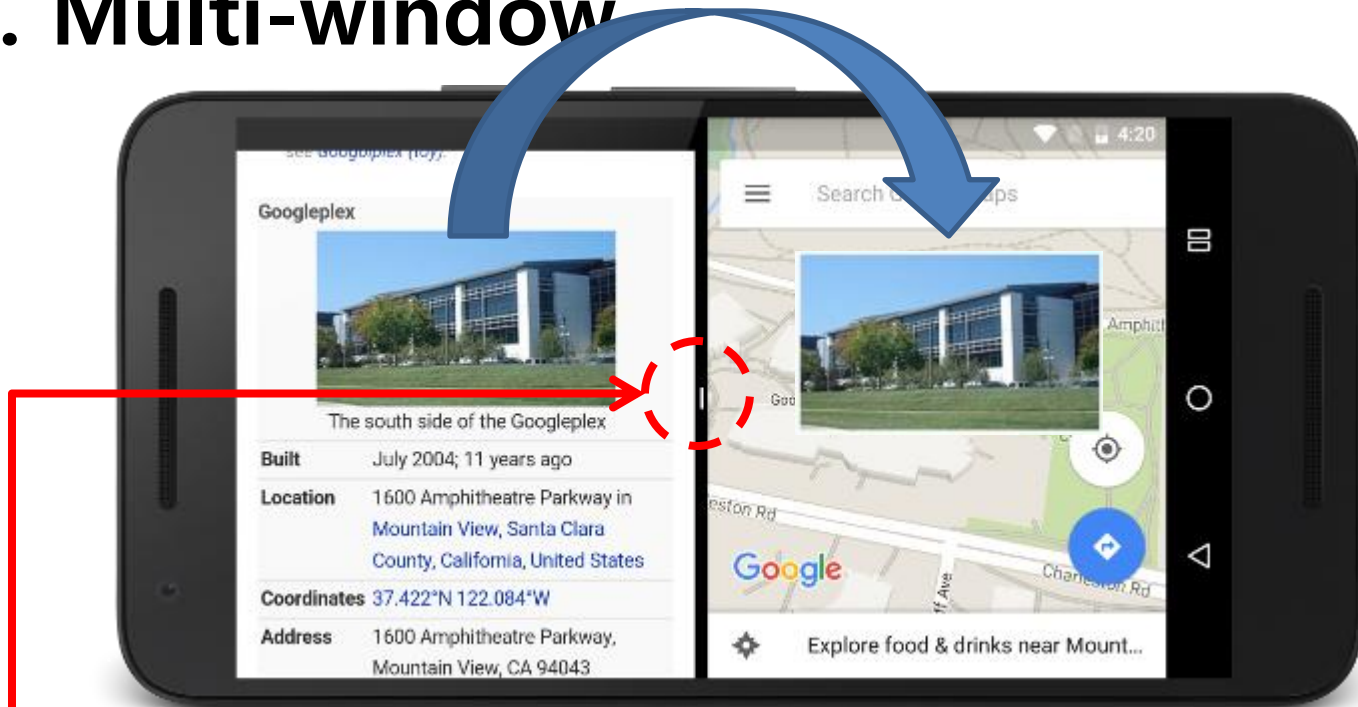


Android 7.x Nougat

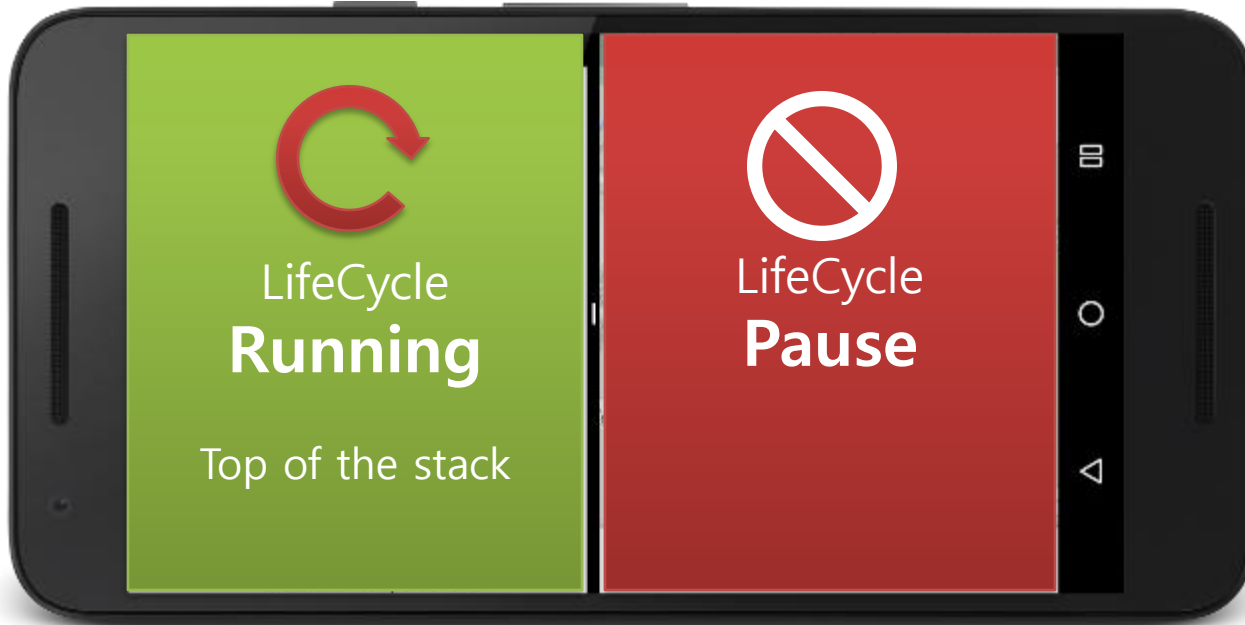
1.1. Multi-window



1. 화면 분할 모드에서 두 개의 앱을 나란히 또는 상하로 실행
2. 분할선 : 앱 사이에 있는 분할선을 끌어서 앱의 크기를 조정
3. 드래그 앤 드랍으로 앱과 앱 사이에서 콘텐츠 교환

1.2. Life Cycle of Multi-window

1. 사용자가 가장 최근에 상호작용했던 액티비티만 지정된 시간에 활성화
(액티비티 스택상에 최상위에 있음)
2. 다른 모든 액티비티는 표시되어 있더라도 일시 정지 상태
시스템은 **하지만 이 액티비티에 더 높은 우선 순위를 부여한다**



* 동영상 플레이어의 경우 onPause 가 아닌 onStop에서 일시정지해야 한다

1.3. Configuring for Multi-window Mode

1. 멀티윈도우 App을 구성하기 위해 다음과 같은 Manifest 옵션이 제공된다

- **android:resizeableActivity=["true" | "false"] :**

true로 설정하면, 액티비티를 Multi-window 모드로 구성할 수 있다.

false로 설정하면, 액티비티가 full screen으로 동작한다.

Nougat의 경우 default 가 true 이기 때문에 따로 설정하지 않으면 앱이 기본적으로 멀티윈도우 환경에서 사용가능하다

2. 다중창 변경알림

- **Activity.isInMultiWindowMode()** : 액티비티의 Multi-Window 상태확인

- **Activity.isInPictureInPictureMode()** : 액티비티의 PIP 모드 확인

- **Activity.onMultiWindowModeChanged()** :

액티비티가 다중 창 모드로 들어가거나 이 모드에서 나올 때 호출

- **Activity.onPictureInPictureModeChanged()**

액티비티가 PIP 모드로 들어가거나 이 모드에서 나올 때 호출

(Fragment 도 동일한 Api 들이 제공된다)

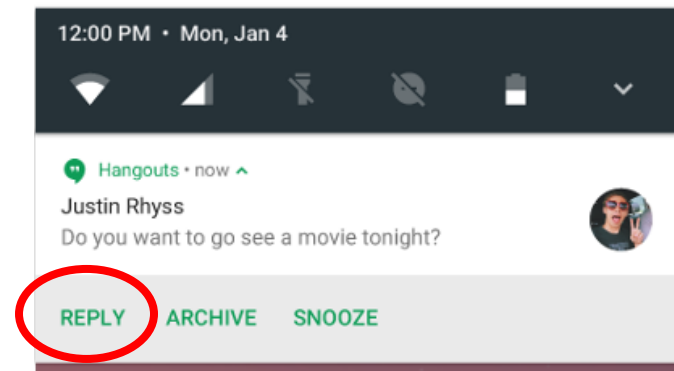
1.4. Drag and Drop on Multi-window

1. 멀티윈도우 상에서 Drag & Drop 기능을 지원하기 위해서 다음과 같은 API가 제공된다

- **android.view.DropPermissions** : 수신하는 앱에 부여되는 권한을 지정하기 위한 토큰 객체.
- **View.startDragAndDrop()** : View.startDrag()의 새로운 이름. 액티비티 간의 드래그 앤 드롭을 활성화하려면, 새 플래그 View.DRAG_FLAG_GLOBAL을 전달한다. 수신자 액티비티에게 URI 권한을 부여해야 하는 경우, 새 플래그 View.DRAG_FLAG_GLOBAL_URI_READ 또는 View.DRAG_FLAG_GLOBAL_URI_WRITE를 적절하게 전달한다.
- **View.cancelDragAndDrop()** : 현재 진행 중인 드래그 작업을 취소. 드래그 작업을 발생시킨 앱에 의해서만 호출가능.
- **View.updateDragShadow()** 현재 진행 중인 드래그 작업의 드래그 그림자를 대체. 드래그 작업을 발생시킨 앱에 의해서만 호출 가능.
- **Activity.requestDropPermissions()** : DragEvent에 포함된 ClipData로 전달되는 콘텐츠 URI의 권한을 요청합니다

2. Notification Enhancements

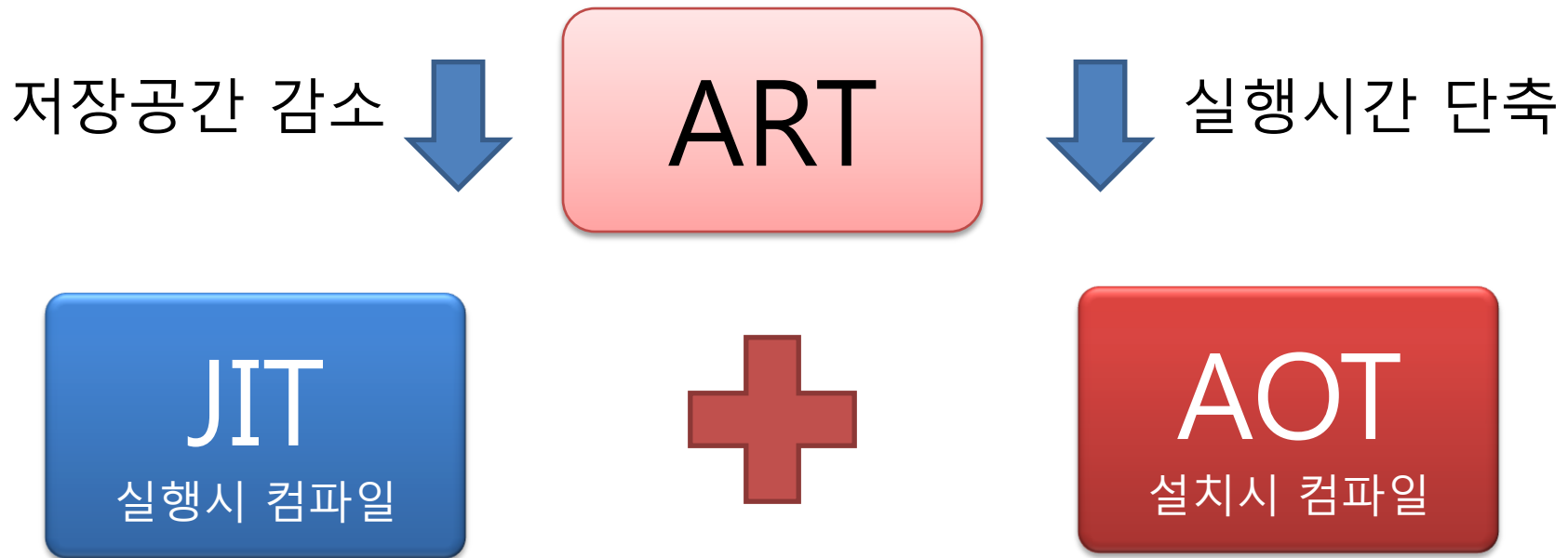
1. **Template updates** : 최소한의 코드 조정만으로 새로운 템플릿을 활용할 수 있다.
2. **Messaging style customization** : MessageStyle 클래스를 사용하여 알림과 관련된 사용자 인터페이스 레이블을 커스터마이징 할 수 있다.
3. **Bundled notification**: 시스템에서 메시지를 함께 그룹화하고 하나의 묶음으로 표시할 수 있다.
4. **Direct reply** : 인라인 회신을 지원하므로, 실시간 통신 앱의 경우 사용자가 알림바에서 바로 SMS 또는 텍스트 메시지에 직접 응답할 수 있다.



5. **Custom views**: 두 개의 새로운 API를 사용하면 알림에 사용자 지정 뷰를 사용할 때 header와 action 같은 시스템 decoration을 활용할 수 있다.

3. Profile-guided JIT/AOT Compilation

1. Android N에서 코드 프로파일링이 있는 JIT(Just in Time) 컴파일러를 런타임 엔진에 추가했습니다. JIT는 현재 ART가 가지고 있는 AOT(Ahead of Time) 컴파일러를 보완하여, 런타임 성능을 향상시키고, 저장 공간을 절약하며, 앱 업데이트와 시스템 업데이트 시간을 단축해 준다.



4. Quick Path to App Install

1. ART의 JIT 컴파일러의 도입으로 얻어지는 최대 강점은 시스템 업데이트 속도가 향상이다

Android 6.0에서 최적화하고 설치하는 데 몇 분이 걸리는 대용량 앱도 N에서는 단 몇 초 만에 설치할 수 있습니다. (항상 그렇지는 않다)
최적화 단계가 없으므로 시스템 업데이트도 더 빠르게 이루어집니다.



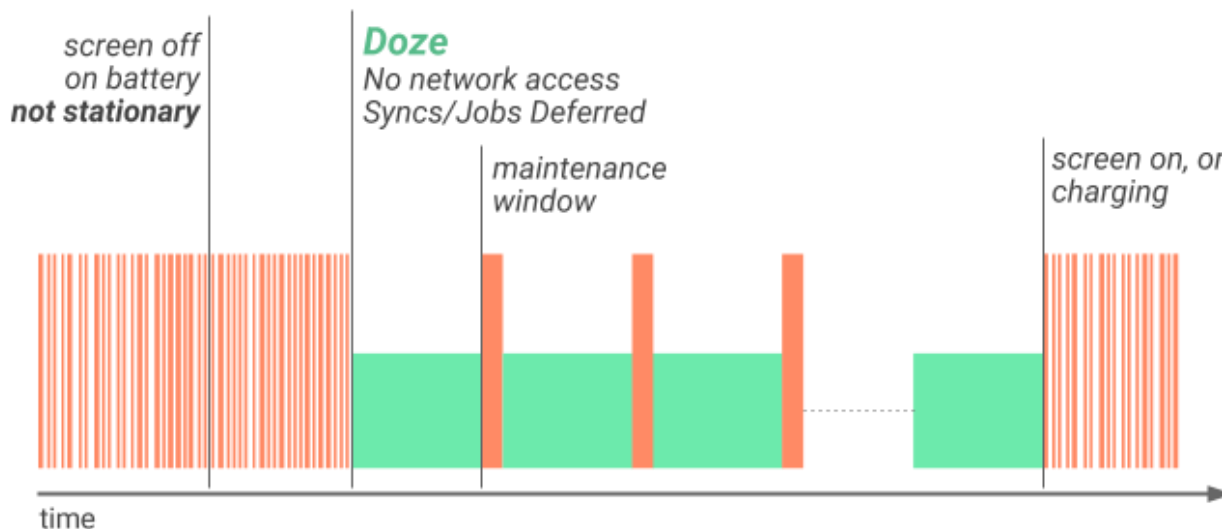
설치시간
및 공간



5. Doze on the Go

- 잠자기 모드 비교 Marshmallow vs Nougat

1. M : 스마트폰이 움직이지 않고(이동하지 않고) 유휴상태일 때 앱이 CPU와 네트워크 작업을 지연시켜서 배터리를 절약하는 방식
2. N : 화면이 일정 시간 동안 꺼져 있고 기기의 플러그가 뽑혀 있으면 이동중에도 잠자기 모드를 통해 일부 CPU 및 네트워크 제한을 앱에 적용한다. 즉, 사용자가 기기를 주머니에 휴대하고 다닐 때도 배터리를 절약할 수 있게 된다.



6. Background Optimizations

1. Restrictions on CONNECTIVITY_ACTION

모바일네트워크에서 Wifi로의 연결 상태 변화들을 알려주는데 **문제는** 네트워크 연결 상태가 변할 때 마다 CONNECTIVITY_ACTION을 받기 위해 **별도로 하는 일이 없음에도** 등록된 앱들이 실행되면서 메모리, 배터리 등의 자원을 낭비하게 된다

Android 6.0까지는 manifest에 CONNECTIVITY_ACTION 브로드캐스트를 등록할 수 있었지만 N 부터는 허용하지 않는다

따라서, N에서는

- JobScheduler 를 이용해서 특정조건에서 호출되도록 하고
- 앱이 실행시만 Network Connectivity 를 모니터링 한다

2. Restrictions on NEW_PICTURE and NEW_VIDEO

역시 위와 동일하게 manifest에 등록시 새로운 이미지나 비디오 등록시 해당 브로드캐스트를 등록한 모든앱에 영향을 미치게 되므로 사용제한

- N에서 JobInfo , JobParameters 를 확장한 Api를 제공한다

7. Surface View instead of TextureView

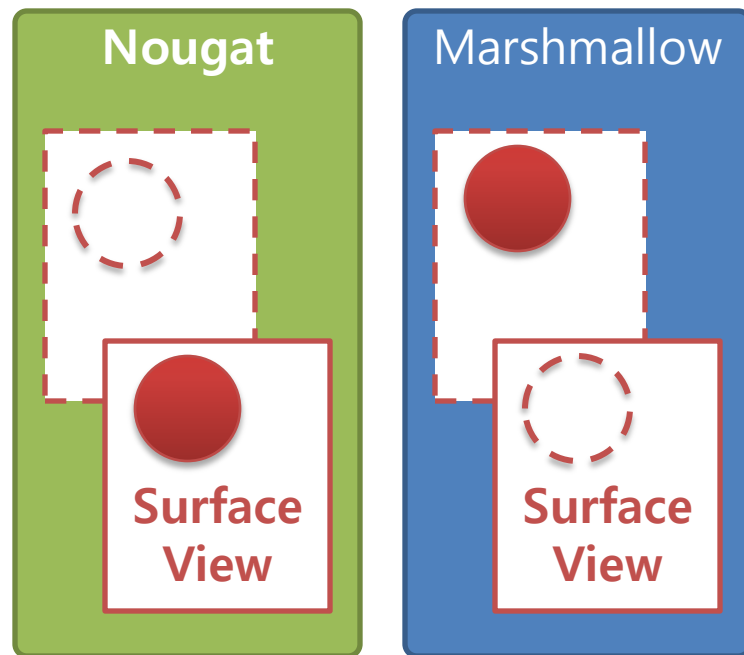
1. TextureView

SurfaceView 와 유사한 기능을 지원하고 있으나 View 계층구조상에서 다른 View 들과 같이 동작하고, 다른 View를 오버랩하거나 또는 오버랩 될 수 있다. View 합성이 발생할 경우 항상 *GL ES로 수행되면서 다른 View들을 다시 그려주기 때문에 의도하지 않은 **자원(CPU+GPU=Battery)낭비**가 발생할 수 있다

**(GL ES : 임베디드 단말을 위한 3D 그래픽 라이브러리)*

2. Surface View

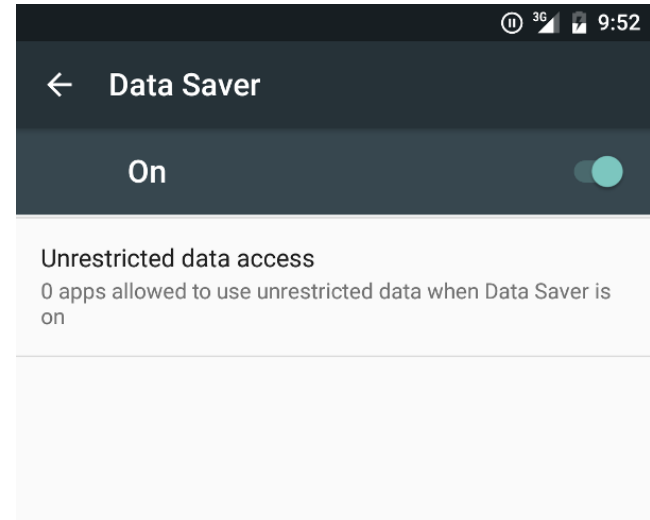
Surface 뷰는 안드로이드 View의 계층구조 밖에 존재한다. 기존 SurfaceView 는 애니메이션 등으로 View의 Position 변경시 동기화 되지 않는 문제점이 있었는데 N 부터는 View 계층구조간의 동기화를 default 로 제공한다



8. Data Saver

1. Data Saver 모드 지원

Android N에서는 새로운 시스템 서비스인 Data Saver 모드를 도입하였으며, 앱에 의해 사용되는 모바일 데이터 사용량을 줄여준다. Data Saver를 통해 사용자는 앱이 모바일 데이터를 사용하는 방식을 제어할 수 있으며, Data Saver가 켜져 있을 때 개발자가 더 효율적인 서비스를 제공할 수 있다.



2. 사용자는 Data Saver가 켜진 경우에도 백그라운드 데이터 통신 연결을 허용하기 위해 특정 앱을 허용 목록에 추가할 수 있다.

3. ConnectivityManager를 확장하여 사용자의 Data Saver 기본 설정을 검색하고 기본 설정 변경을 모니터링하는 방법을 앱에 제공한다. 모든 앱은 사용자가 Data Saver를 활성화했는지 여부를 확인해야 하며 포그라운드 및 백그라운드 데이터 사용을 제한하도록 해야한다.

9. Vulkan API

1. 새로운 3D Rendering API 지원

- Android N은 새로운 3D 렌더링 API인 Vulkan™을 플랫폼에 통합한다.

OpenGL™ ES와 마찬가지로, Vulkan은 Khronos Group에서 제공하는 3D 그래픽 및 렌더링을 위한 표준입니다.

Vulkan은 드라이버에서 CPU 오버헤드를 최소화하도록 설계되었고, 애플리케이션이 GPU 작업을 더욱 직접적으로 제어할 수 있게 해준다. 또한, Vulkan은 멀티스레드가 명령 버퍼 생성과 같은 작업을 한 번에 수행하도록 허용하여 병렬 처리 성능을 높입니다.

- 아래와 같은 Vulkan 개발 도구와 라이브러리가 Android NDK에 배포된다
 - 가. 헤더
 - 나. 유효성 검사 계층(디버그 라이브러리)
 - 다. SPIR-V 셰이더 컴파일러
 - 라. SPIR-V 런타임 셰이더 컴파일 라이브러리

10. Quick Settings Tile API

1. Expanded the scope of Quick Settings

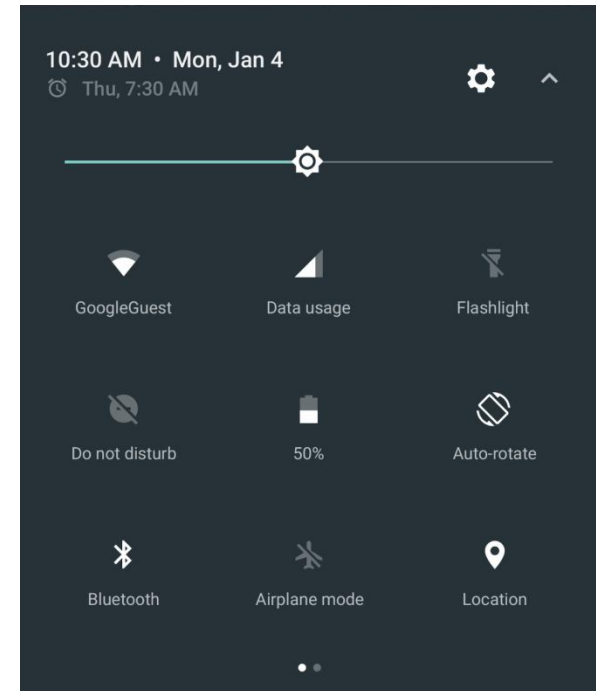
- Quick Settings의 범위를 더욱 확장했다.

Google에서는 Quick Settings 타일에 필요한 공간을 더 추가했으며, 페이지가 지정된 디스플레이 영역에서 왼쪽이나 오른쪽으로 Swipe하여 각 타일에 액세스할 수 있다.

또한, Quick Settings 타일의 모양과 표시 위치를 제어할 수 있다. 따라서 Drag & Drop으로 간단하게 타일을 추가하거나 이동시킬 수 있다.

- 개발자를 위해 새로운 API가 추가됐으며 이를 통해 Custom Quick Settings 타일을 정의할 수 있고, 사용자가 앱에서 쉽게 접근할 수 있다.

- Quick Settings Tile은 자주 사용되는 컨트롤 또는 작업을 위해 예약되어 있으며, 앱을 시작하기 위한 바로가기로 사용되어서는 안된다.



11. Number Blocking

1. 플랫폼에서 번호 차단을 지원하며, 서비스 공급자가 차단된 번호 목록을 유지 관리할 수 있도록 프레임워크 API를 제공한다. 기본 SMS 앱, 기본 전화 앱 및 이동통신사 앱은 차단된 번호 목록에서 읽기와 쓰기가 가능하다. 다른 앱은 이 목록에 액세스할 수 없다.

- 통화에서 차단된 번호가 텍스트에서도 차단된다.
- 차단된 번호는 재설정 시에도 백업을 통해 재사용 할 수 있다.
- 여러 앱이 동일한 차단된 번호 목록을 사용할 수 있다.

12. Call Screening

1. 기본 전화 앱이 수신 통화를 스크리닝하도록 허용한다. 전화 앱에는 CallScreeningService가 구현되며, 이를 통해 수신 통화의 Call.Details에 따라 다음과 같은 다양한 작업을 수행할 수 있다.

- 수신 통화 거절
- 통화를 통화 기록에 허용하지 않음
- 통화 알림을 사용자에게 표시하지 않음

13. 그외 변경사항

- 13. Multi-locale Support, More Languages - Settings 에서 다중 로케일 선택가능
- 14. New Emojis - 새로운 이모티콘
- 15. ICU4J APIs in Android - Java를 대체하기 위한 Open Source API
- 16. WebView - Chrome + WebView 지원
- 17. OpenGL™ ES 3.2 API - 일괄처리 및 스트리밍 성능을 향상, WebGL 오버헤드 감소
- 18. Android TV Recording - TV녹화
- 19. Android for Work - 앱에 대해 별개의 보안 인증 질문 지정, 상시 접속 VPN 등 추가
- 20. Accessibility Enhancements - 접근성 향상
- 21. Direct Boot - 시동시간 개선, 재시작 후에도 접근성 서비스 사용가능
- 22. Key Attestation - 하드웨어 기반 암호화 키 저장소
- 23. Network Security Config - 보안 프로토콜(HTTPS, TLS)을 수정없이 안전하게 사용가능
- 24. Default Trusted Certificate Authority - 시스템에서 제공되는 인증서만 신뢰
- 25. APK Signature Scheme v2 - 앱서명 Scheme 업그레이드, 앱설치시간 단축
- 26. Scoped Directory Access - 특정 외부저장소에 대한 접근요청 추가
- 27. Keyboard Shortcuts Helper - Alt + / 키를 눌러 Keyboard Shortcuts 화면 트리거
- 28. Custom Pointer API - 마우스나 키패드 사용자를 위한 커스텀 포인터 사용가능
- 29. Sustained Performance API - 장시간 실행되는 앱의 성능변화 정보 제공
- 30. VR Support - 고품질 모바일 VR 플랫폼 지원
- 31. Print Service Enhancements - 개별프린터에 대한 추가정보 제공
- 32. Frame Metrics API - UI 렌더링 성능 모니터링
- 33. Virtual Files - 사용자 파일에 대한 대체형식 제공

0. Nougat 변경사항 요약

Accessibility

Multi-window

Notification

Quick Settings Tile

Performance

JIT + AOT

SurfaceView

Vulkan

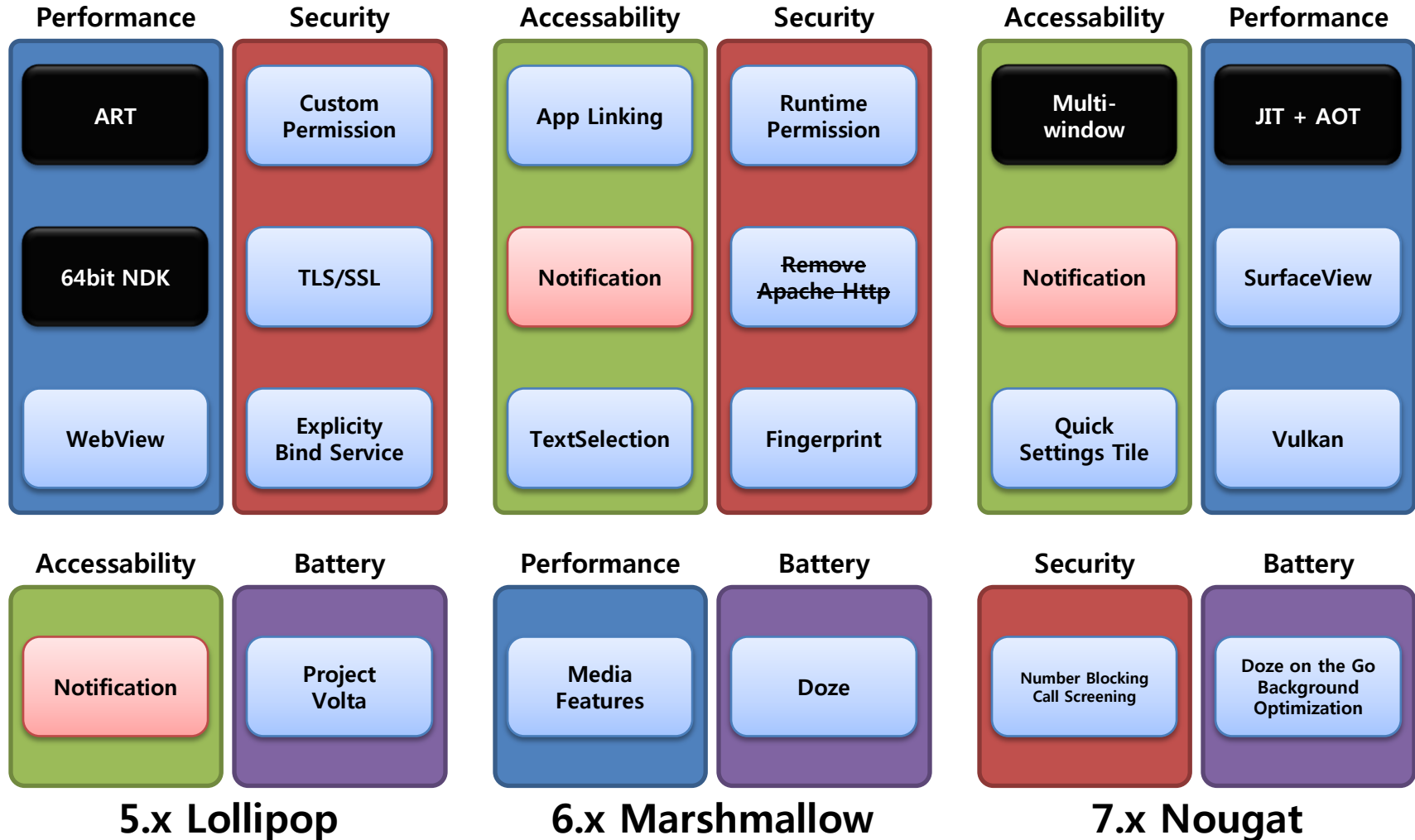
Security

Number Blocking
Call Screening

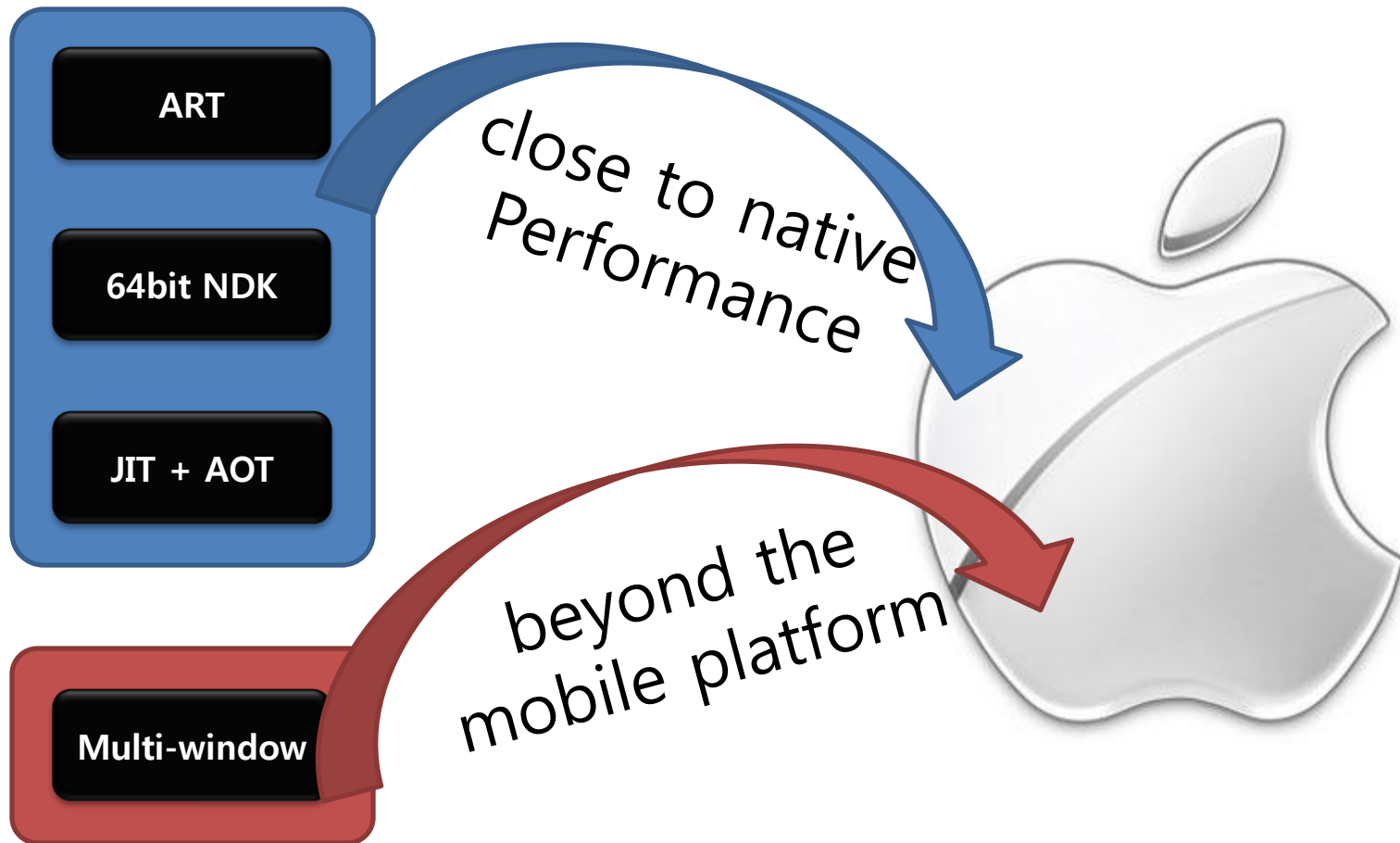
Battery

Doze on the Go
Background Optimization

A. Lollipop, Marshmallow and Nougat



B. what they follow

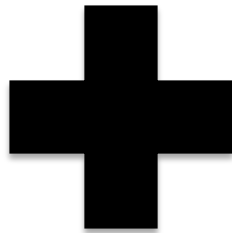


C. what android developers should do

Algorithm

NDK

Backend



ubuntu