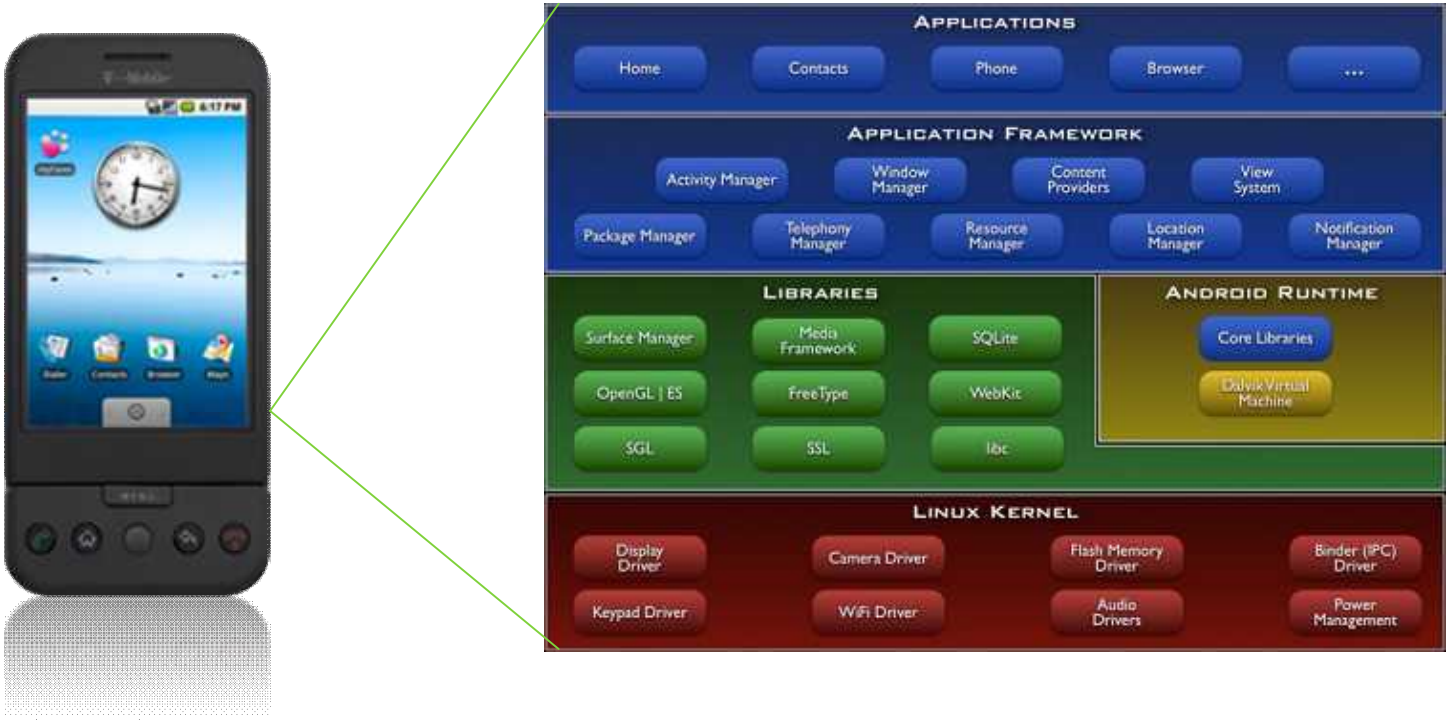
Androidアーキテクチャの特徴

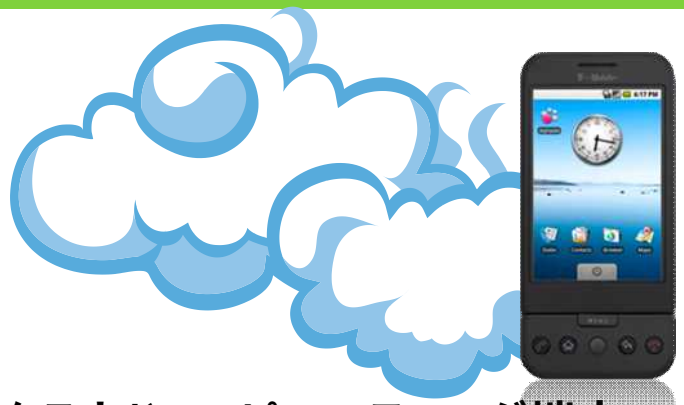
日本Androidの会
木南英夫
江川崇

Androidとは

- **携帯端末向けソフトウェアプラットフォーム**



Androidのポイント



クラウドコンピューティング端末

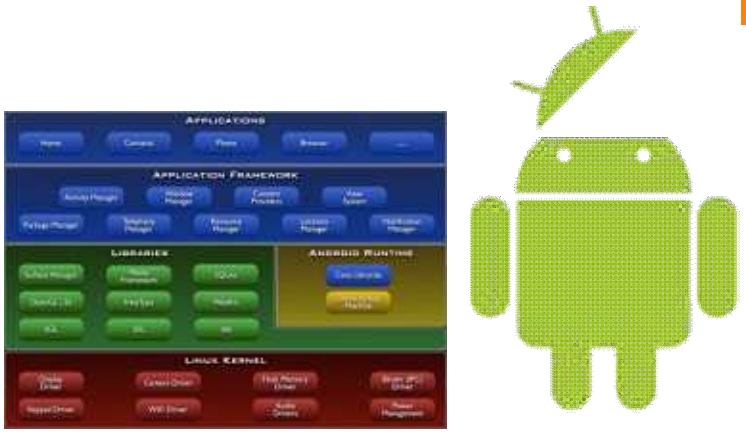
携帯端末向け

フルセット

open source

三つの視点

ユーザー



プラットフォーム提供企業

アプリケーション開発者

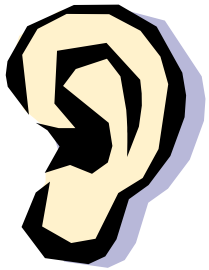
モバイルインターネット普及の理由

- データ定額制のプラン
- フルブラウザの採用
- タッチインタフェース

新たなユーザインタフェース

センサー

カメラ



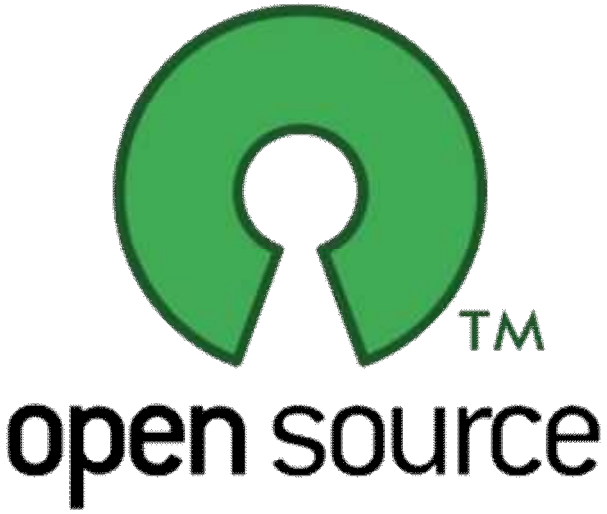
マイク

タッチパネル
バイブレータ

スピーカー

プラットフォーム提供者 (企業) の視点

- プラットフォーム開発者
  - 携帯端末メーカー
  - ミドルウェア提供メーカー
  - 商用化サポートメーカー
- オープンソースの可能性
  - 共通部品採用による開発コストの削減
  - 開発者確保の容易性



開発者：マーケット

- Android Market
  - **世界に向けたアプリケーションの発信**

バトンタッチする前に…

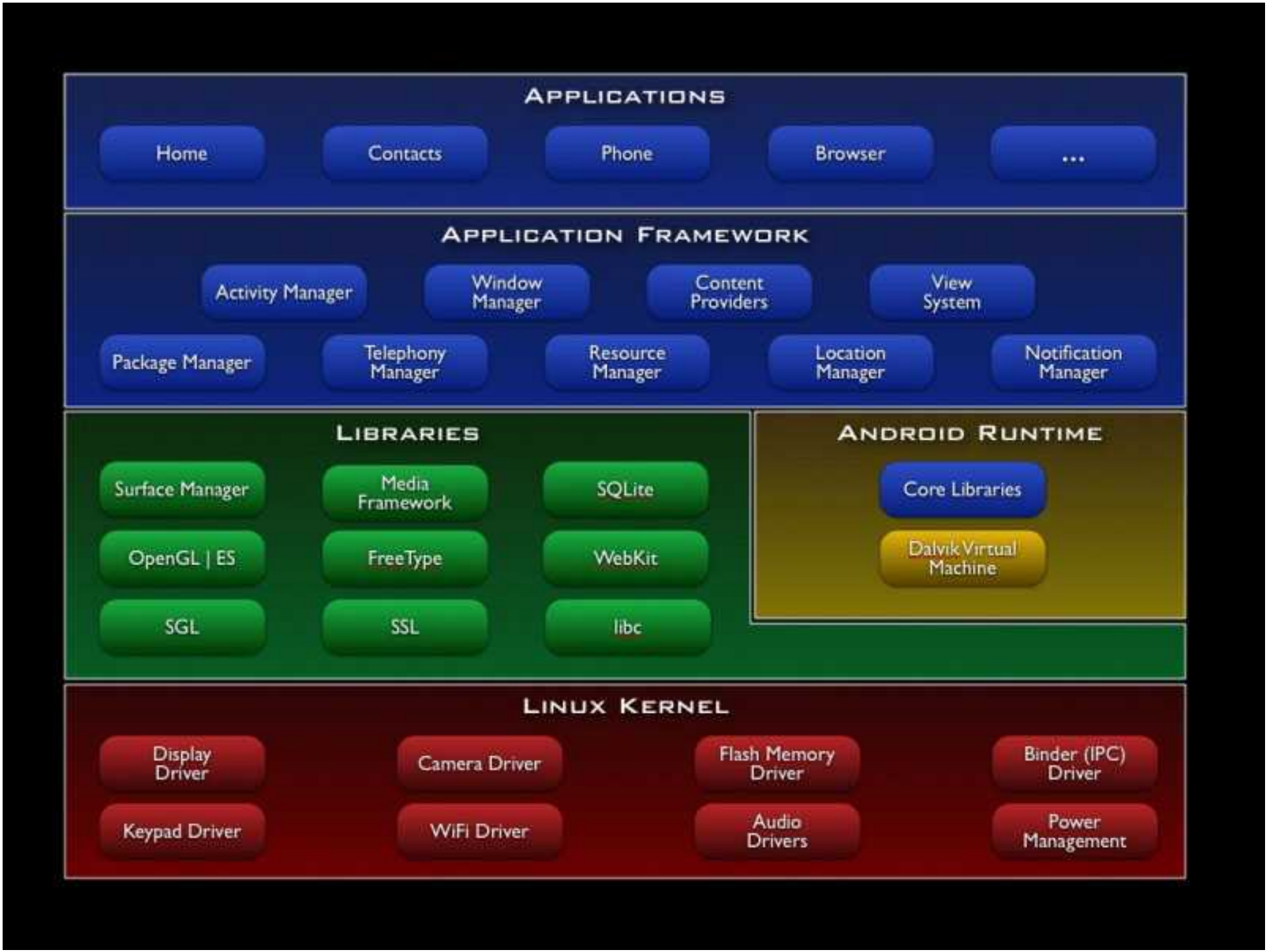
- コンピューターはコミュニケーションの道具
  - The Invisible Computer
  - 見えなくする方法は二つある。
    - 見えなくなるまで小さくすること
    - 見えなくなるまで大きくすること
- コミュニティーへ参加しよう！
  - ソフトウェアのユーザーコミュニティ
  - 日本Androidの会

日本Andoridの会

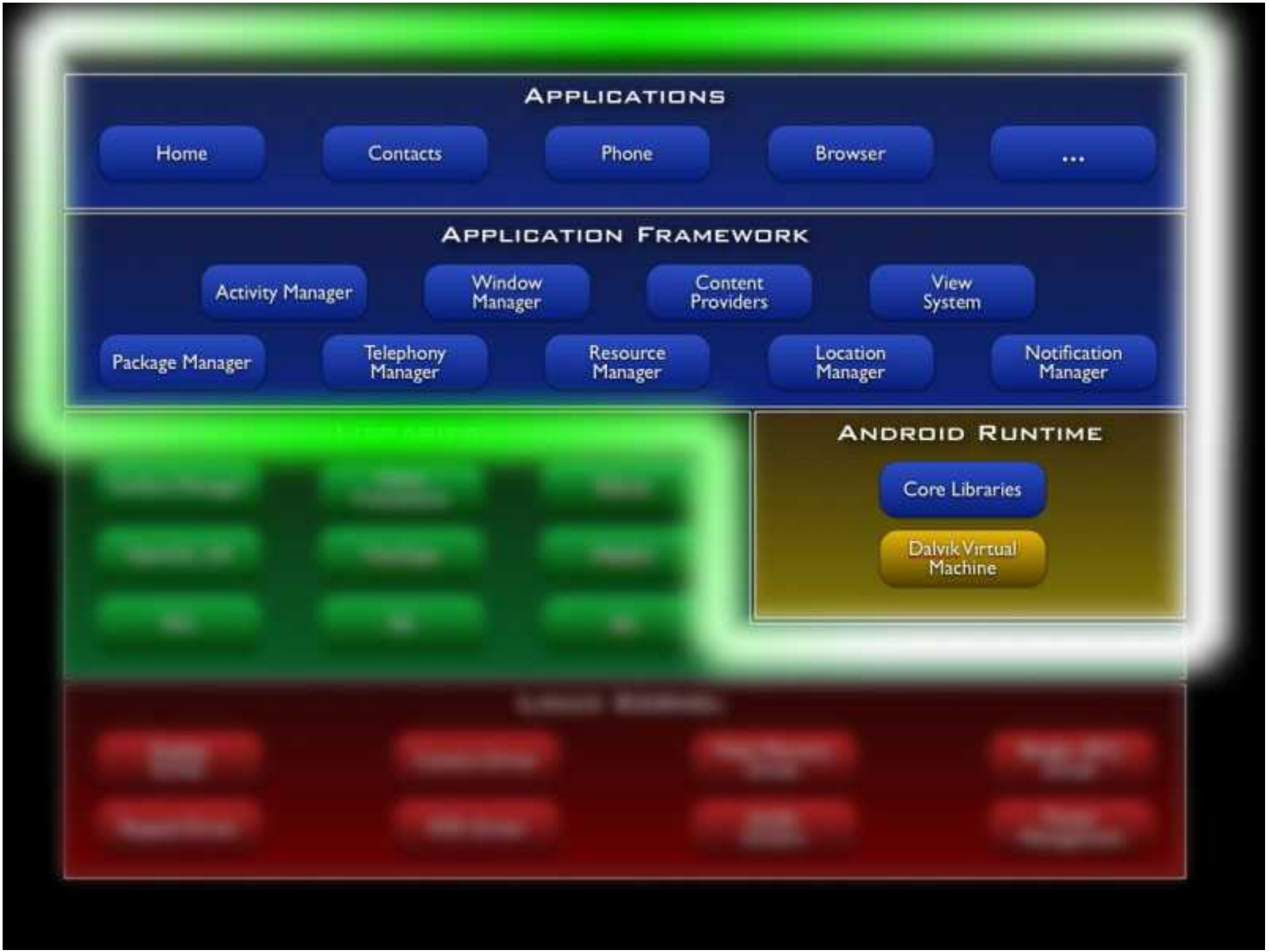
- 日本で、Androidの発展を促進させるユーザーグループ
- 会員募集中
  - http://www.android-group.jp/
  - MLに登録するだけ！

Androidの全体像

- モバイルデバイス向けのソフトウェアプラットフォーム（ソフトウェアの集合体）
  - アプリケーション
  - アプリケーションフレームワーク
  - ランタイム
  - ライブラリ
  - OS


Applications
Home
Contacts
Phone
Browser
...
Application Framework
Activity Manager
Window Manager
Content Providers
View System
Package Manager
Telephony Manager
Resource Manager
Location Manager
Notification Manager
Libraries
Surface Manager
Media Framework
SQLite
OpenGL | ES
FreeType
WebKit
SGL
SSL
libc
Android Runtime
Core Libraries
Dalvik Virtual Machine
Linux Kernel
Display Driver
Camera Driver
Flash Memory Driver
Binder (IPC) Driver
Keypad Driver
WiFi Driver
Audio Drivers
Power Management


LIBRARIES
Surface Manager
Media Framework
SQLite
OpenGL | ES
FreeType
WebKit
SGL
SSL
libc
LINUX KERNEL
Display Driver
Camera Driver
Flash Memory Driver
Binder (IPC) Driver
Keypad Driver
WiFi Driver
Audio Drivers
Power Management


Applications
Home
Contacts
Phone
Browser
...
Application Framework
Activity Manager
Window Manager
Content Providers
View System
Package Manager
Telephony Manager
Resource Manager
Location Manager
Notification Manager
Android Runtime
Core Libraries
Dalvik Virtual Machine

開発領域の違い

- プラットフォームに関する領域
  - アーキテクチャー図の下部に対応
  - ある端末の上で動作するAndroidを作る（ポーティング）
  - Linuxカーネルやデバイスドライバの知識が必要

- アプリケーション開発に関する領域
  - アーキテクチャー図の上部に対応
  - Androidの上で動作するアプリケーションを作る
  - JavaやWEB関連技術の知識が必要

開発領域の違い

- プラットフォームに関する領域
  - アーキテクチャー図の下部に対応
  - ある端末の上で動作するAndroidを作る（ポーティング）
  - Linuxカーネルやデバイスドライバの知識が必要

- アプリケーション開発に関する領域
  - アーキテクチャー図の上部に対応
  - Androidの上で動作するアプリケーションを作る
  - JavaやWEB関連技術の知識が必要

開発現場におけるAndroidの可能性

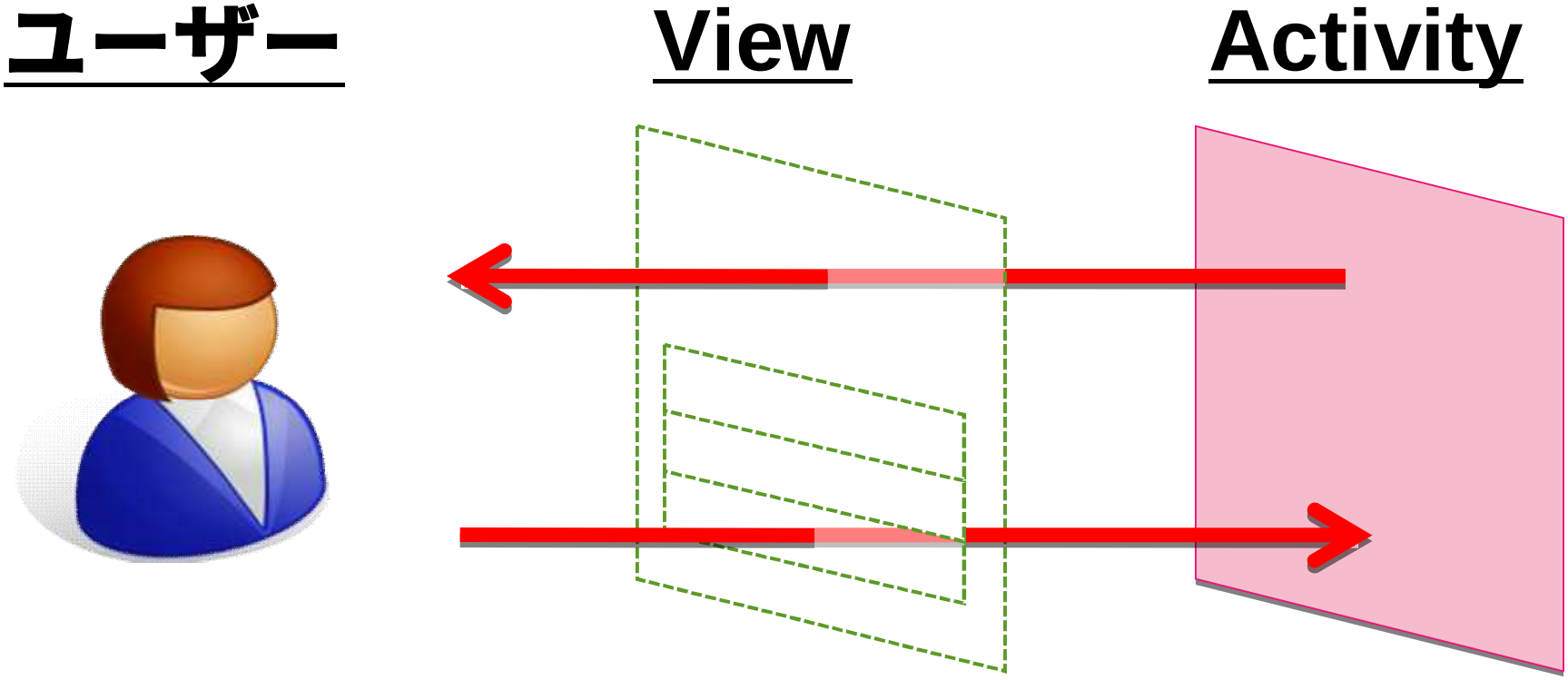
- 並行作業･分業が可能
  - ハードの完成を待たずにソフトが開発、テストできる
  - ハード、ミドルを知らなくてもソフトを開発できる
- アプリ開発者の調達や育成が容易
  - 製品に依存しない共通のAPI
  - 豊富な機能を持った開発環境（Eclipse）での開発
- ソフトウェア規模の抑制が可能
  - 基本的なアプリ、ライブラリ、ミドルウェアは標準搭載
  - オープンソース戦略

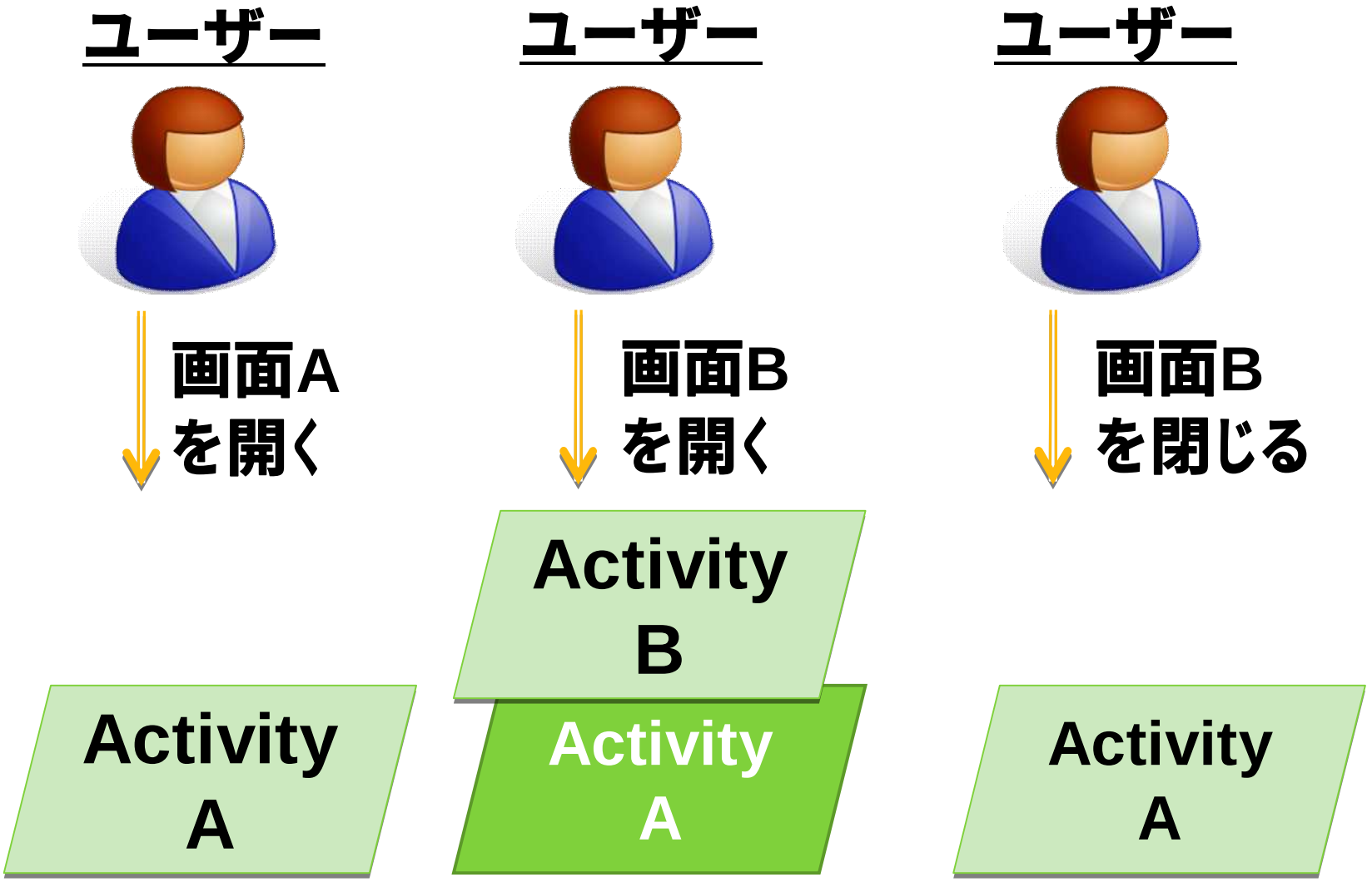
Androidアプリの主要な構成要素

- Activity
- Service
- Content Provider
- Intent

Activity

- ユーザーと相互作用する画面の単位
  - View (UI部品) と組み合わせて使う
- アプリケーションのエントリポイント



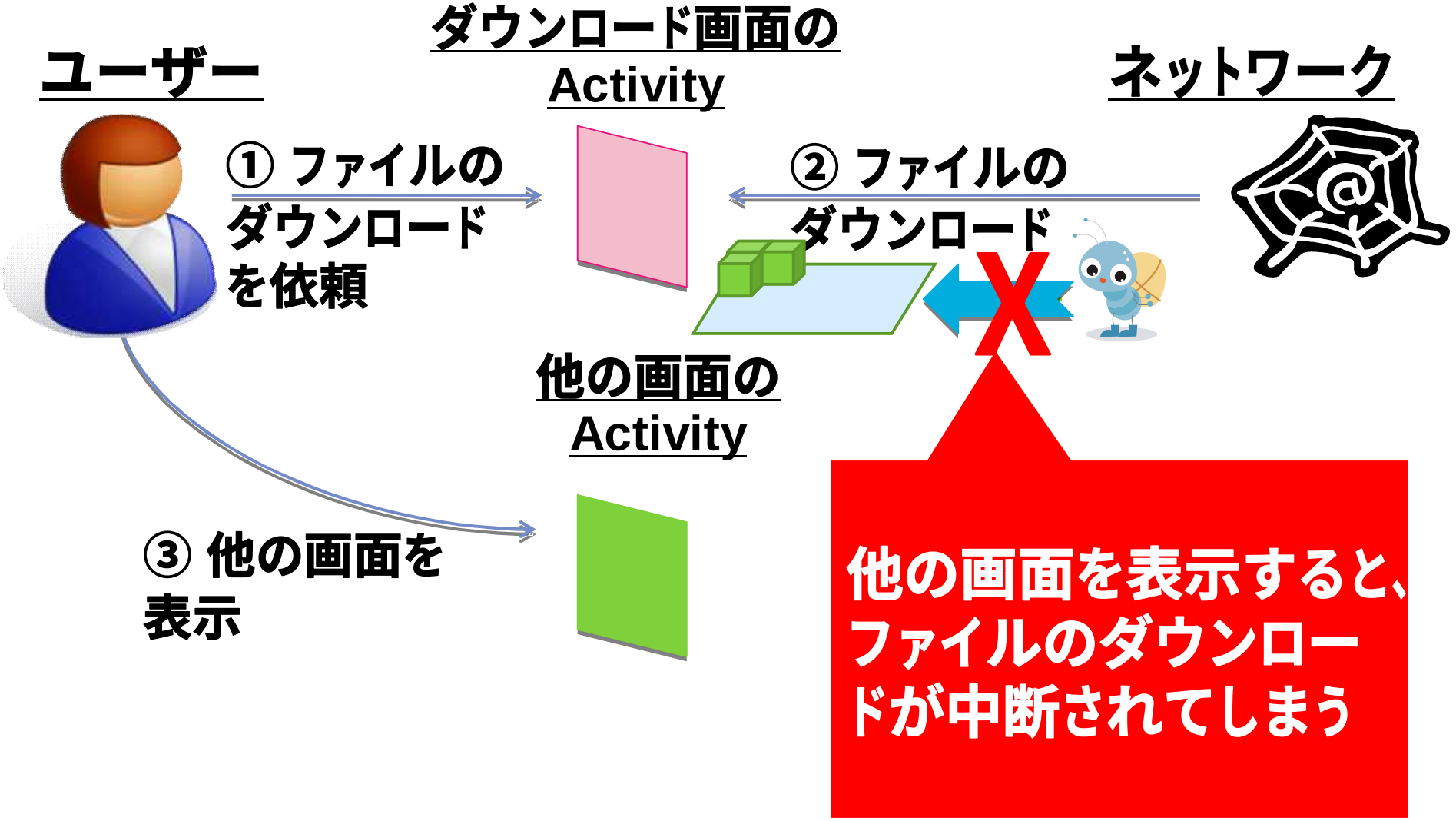

ユーザー
ユーザー
ユーザー
画面A
を開く
画面B
を開く
画面B
を閉じる
Activity
A
Activity
B
Activity
A
Activity
A

Service

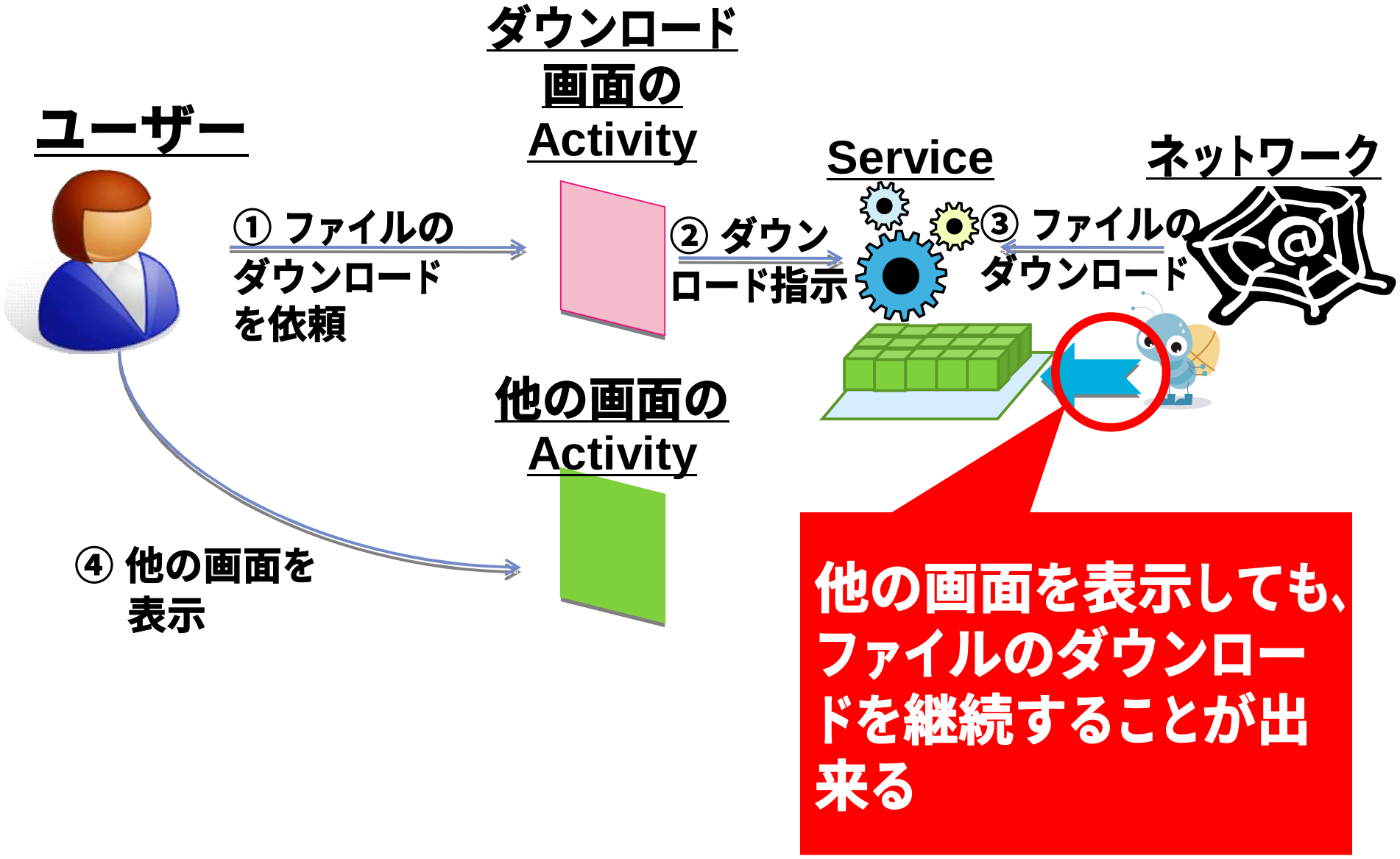
- 画面の状態と独立して処理する
- 別プロセスとして起動可能

- 利用する局面
  - ライフサイクルの長い処理
  - 常時動き続けている必要がある処理

ファイルのダウンロード処理をActivityで実装したイメージ

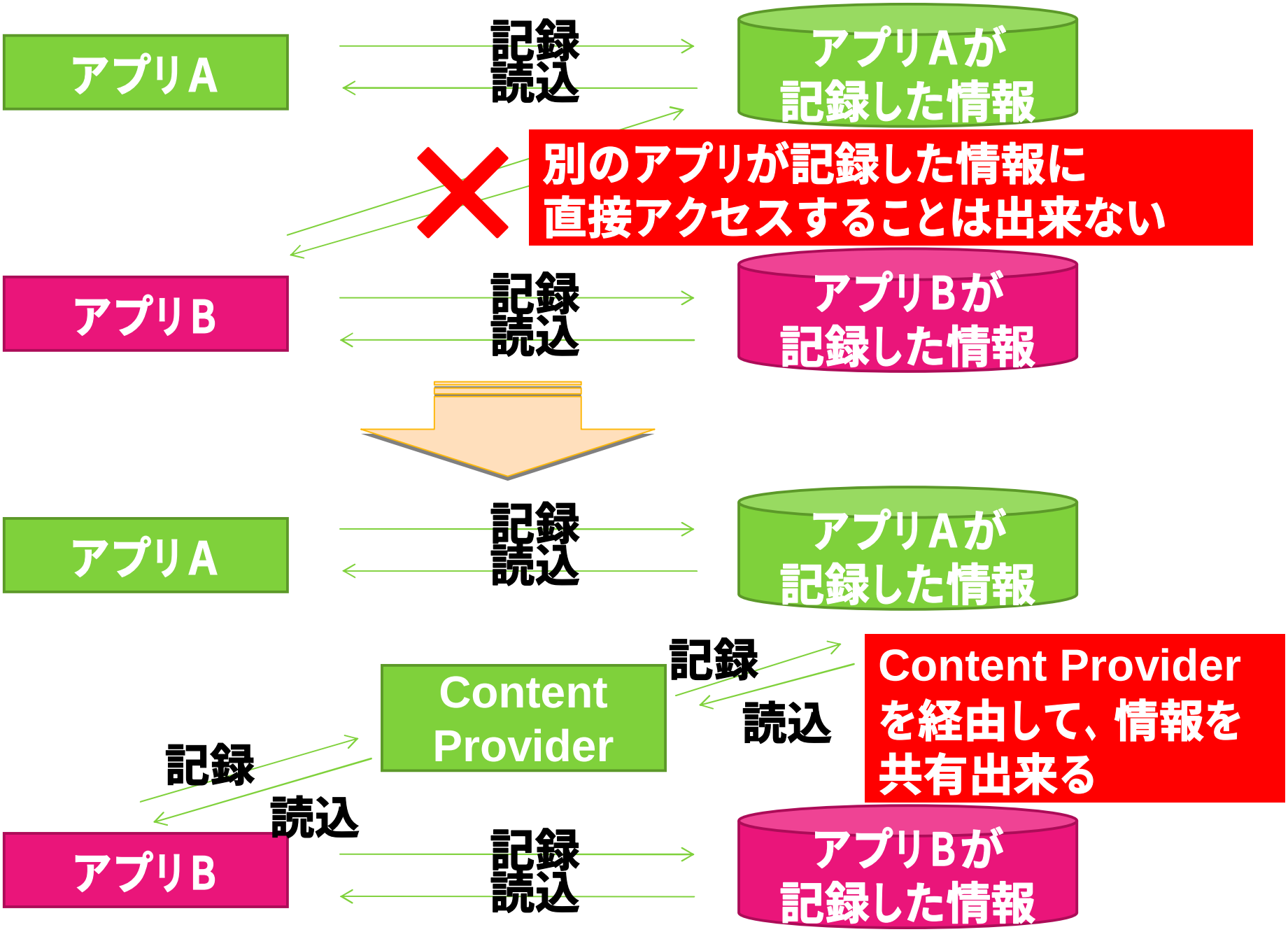


ファイルのダウンロード処理をServiceで実装したイメージ



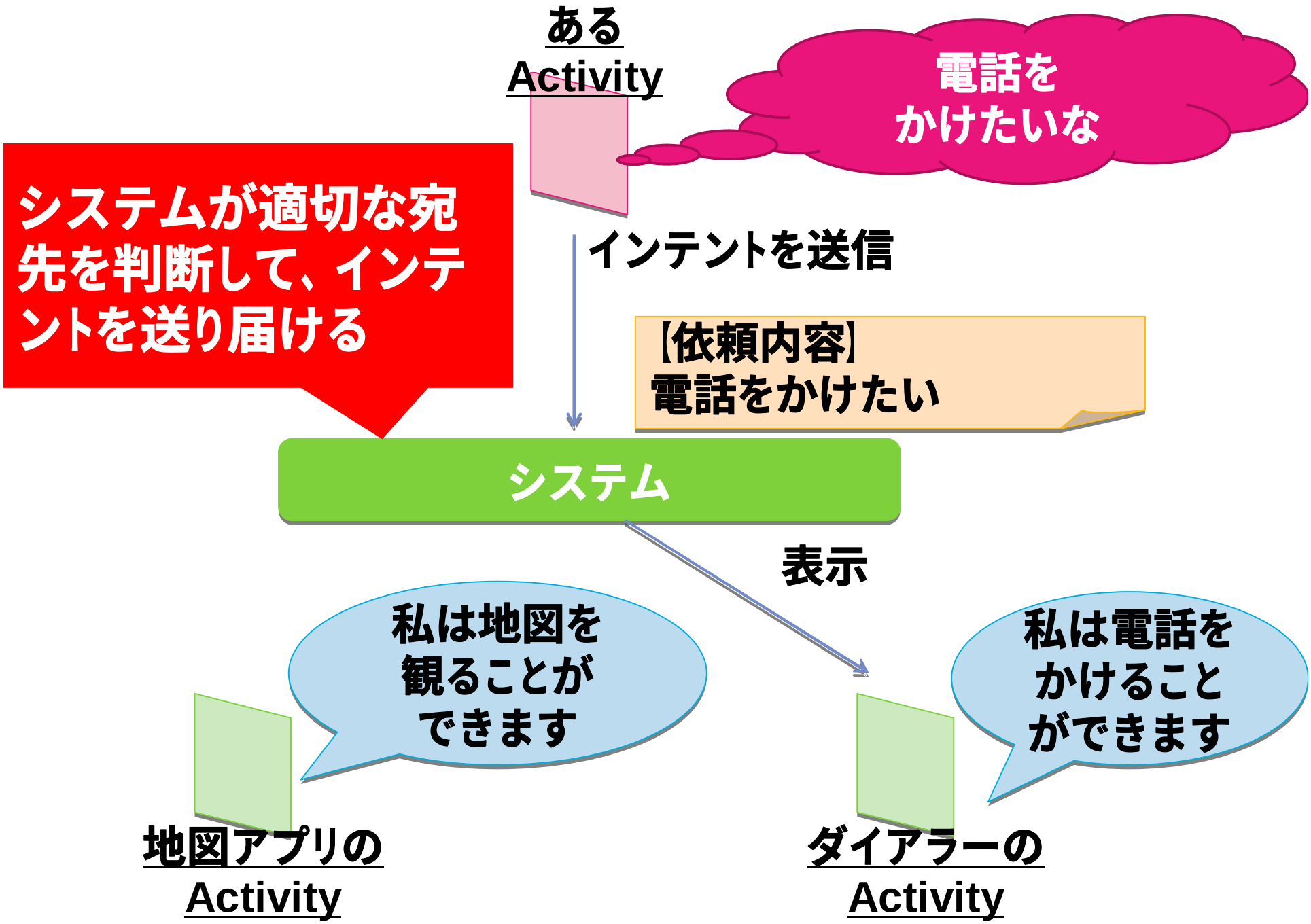
Content Provider

- アプリ間で情報を共有するもの
  - 通常はアプリ毎のパーミッション
- 情報への論理的なアクセス手段を提供
  - query (情報の読込)
  - insert (情報の新規追加)
  - update (情報の変更)
  - delete (情報の削除)


アプリA
記録
読込
アプリAが
記録した情報
別のアプリが記録した情報に
直接アクセスすることは出来ない
アプリB
記録
読込
アプリBが
記録した情報
アプリA
記録
読込
アプリAが
記録した情報
記録
読込
Content
Provider
Content Provider
を経由して、情報を
共有出来る
記録
読込
アプリB
記録
読込
アプリBが
記録した情報

Intent

- 各要素間を実行時に紐付けるための糊（のり）
- 「何かをする」という“intention（意図）”を示す


ある
Activity
電話を
かけたいな
システムが適切な宛先を判断して、インテントを送り届ける
インテントを送信
【依頼内容】
電話をかけたい
システム
表示
私は地図を
観ることが
できます
私は電話を
かけること
ができます
地図アプリの
Activity
ダイアラーの
Activity

組込製品開発分野への非組込系企業の参入

モバイルプラットフォームの登場により、

- 高スペックな携帯デバイスが
- 常にネットワークにつながる環境で
- 高度に抽象化されたアプリケーションフレームワーク越しに触れる

このことは、特に非組込系企業に対して、以下の2つの可能性を示す

- エンタープライズ系のエンジニアリング技術が活用可能に
- 組込製品が非ブラックボックスに

エンタープライズ系のエンジニアリング技術の活用

- 動的言語の利用
  - ex) JavaとJavaScriptのブリッジ
- 永続化の仕組 (RDB, SQL)
  - 慣れ親しんだ方法と同じ
- PoEAAやデザパタ、アーキテクチャーパターン
  - ホットスポットのインタフェースや責務分割など
- テスト駆動やCI
  - JUnitで統合し、サーバー側と同じ開発スタイルで

組込製品の非ブラックボックス化

- 組込製品の領域に立ち入ることが出来る
  - 今までのビジネスにプラスアルファの付加価値を与える

    ex)
    - 企業内のビジネスインフラのテーラリング
    - 物流や製造の在庫管理や購買管理と連携

止められない時代の流れがある

- 「組み込みの世界はわからないから」という時代は終わる
- 組込系とPC系の溝は埋まりつつある
- チャンスと考えるべきではないか

- 技術者としての夢
  - 長期
    - 自分が幸せになりたい　ので　自分の周囲を幸せにしたい
  - 短期
    - エンタープライズと組込の世界をつなげたい
- メッセージ
  - コミュニティで活動しましょう
  - Androidをやりましょう