



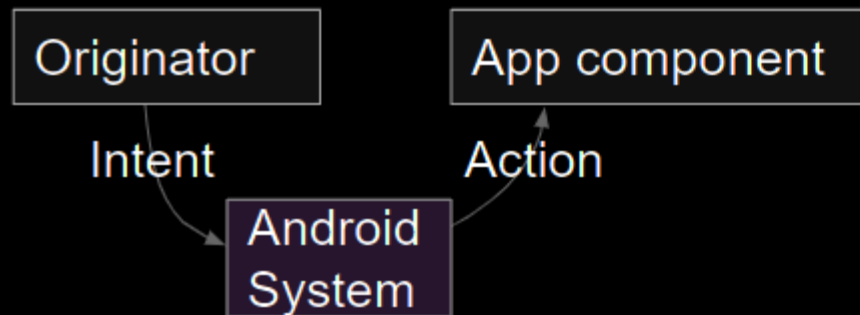
PROGRAMOWANIE URZĄDZEŃ MOBILNYCH

WYKŁAD 2

- ACTIVITY
- INTENT
- BUNDLE

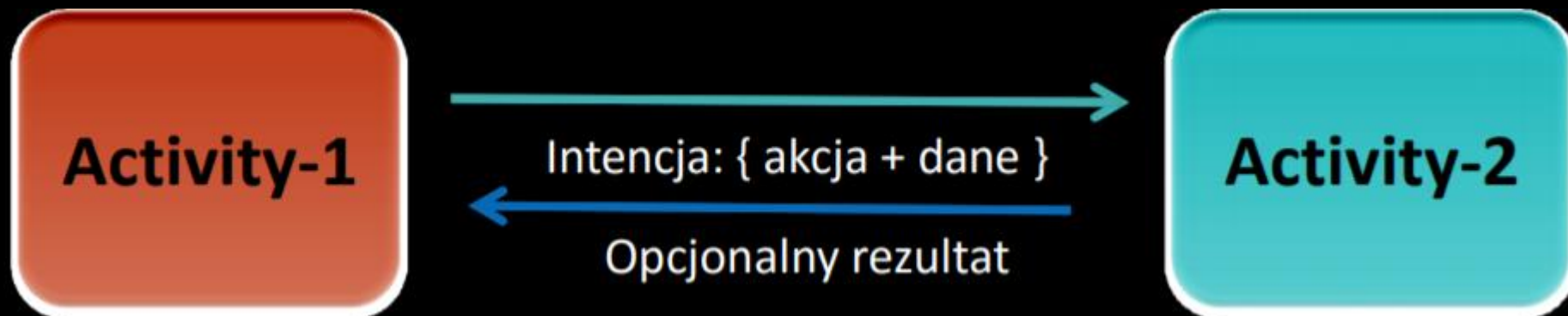
An Intent is a description of an operation to be performed.

An Intent is an object used to request an action from another app component via the Android system.



- Wywoływanie intencji

<code>startActivity (intent)</code>	Uruchamia (pokazuje) nową aktywność
<code>sendBroadcast (intent)</code>	Wysyła intencję do komponentu typu BroadcastReceiver
<code>startService (intent)</code> lub <code>bindService (intent, ...)</code>	Używane są do komunikacji z usługą działającą w tle (osobnym wątku).



Explicit Intent

- Starts a specific Activity
 - Request tea with milk delivered by Nikita
 - Main activity starts the ViewShoppingCart Activity

Implicit Intent

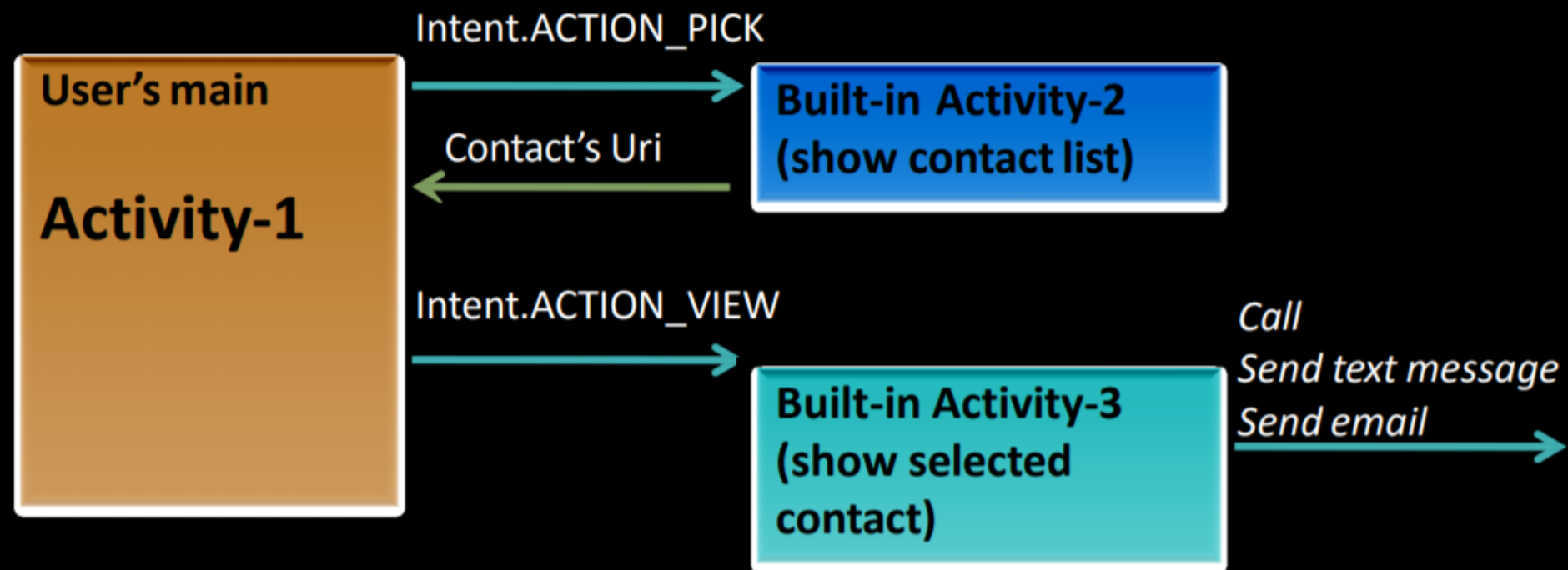
- Asks system to find an Activity that can handle this request
 - Find an open store that sells green tea
 - Clicking Share opens a chooser with a list of apps


```
Intent myActivity = new Intent (action, data);  
startActivity (myActivity);
```

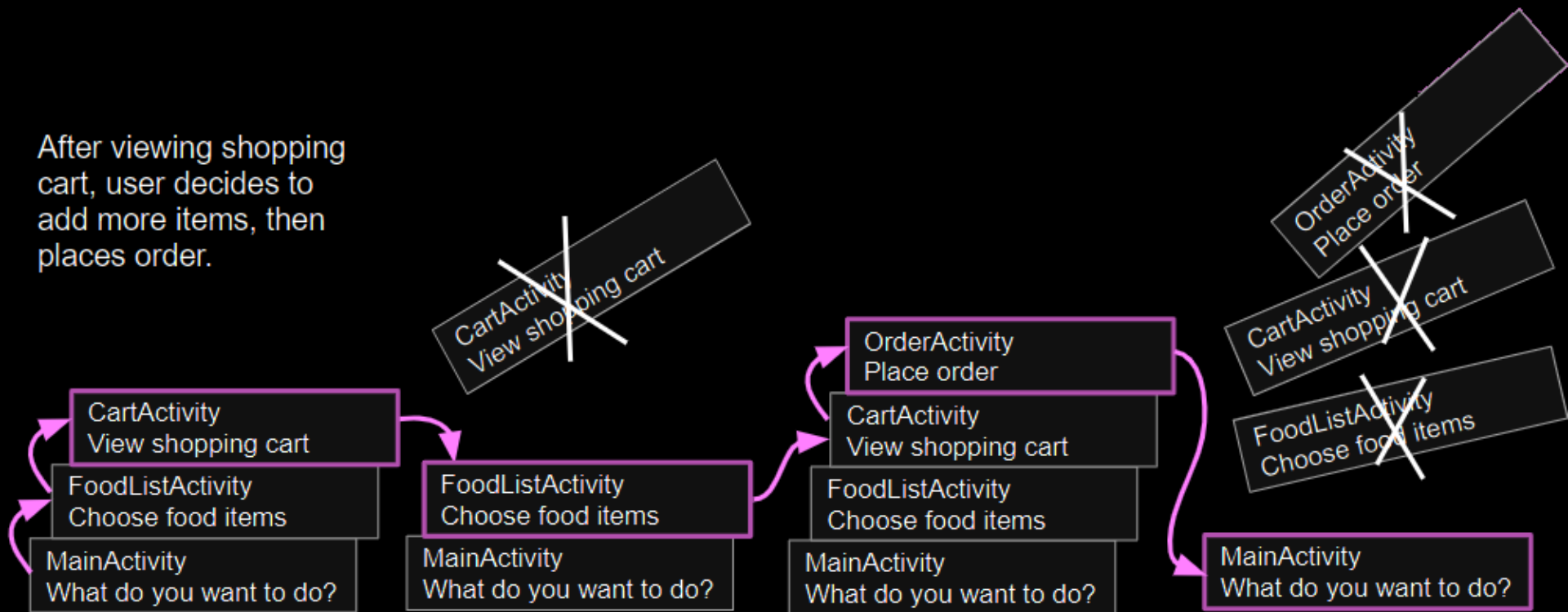
ACTION_DIAL	<i>tel:5551234</i> Wyświetla (uzupełnioną) aktywność do wybierania numerów.
ACTION_VIEW	<i><u>http://www.google.com</u></i> Pokazuje stronę Google w przeglądarce.
ACTION_EDIT	<i>content://contacts/people/2</i> Umożliwia edycję danych kontaktu o identyfikatorze "2".
ACTION_VIEW	<i>content://contacts/people/2</i> Uruchamia aktywność wyświetlającą dane kontaktowe użytkownika o identyfikatorze "2".
ACTION_VIEW	<i>content://contacts/people/</i> Wyświetla listę kontaktów użytkownika. Wybór poszczególnej osoby skutkuje wywołaniem kolejnej intencji.

Mechanizm Intencji

- Aplikacje działające pod kontrolą systemu Android składają się z wielu aktywności które są w stosunku do siebie niezależne. Każda aktywność działa w ramach swojego wątku
- Android wykorzystuje metodę `startActivity(Intent)` uruchamiając aktywność,
- Obiekt wywołujący znajduje się w obrębie swojego wątku.



After viewing shopping cart, user decides to add more items, then places order.



- Dwie formy nawigacji
 - Czasowa (Back Button)
 - Przycisk Wstecz na urządzeniu
 - Kontrolowana przez stos OS
 - Dziedziczna
 - Przycisk w aplikacji
 - Kontrolowana przez relacje dziecko-rodzic pomiędzy aktywnościami

- System saves only:
 - State of views with unique ID (`android:id`) such as text entered into `EditText`
 - Intent that started activity and data in its extras
- You are responsible for saving other activity and user progress data

Implement `onSaveInstanceState()` in your Activity

- Called by Android runtime when there is a possibility the Activity may be destroyed
- Saves data only for this instance of the Activity during current session

@Override

```
public void onSaveInstanceState(Bundle outState) {  
    super.onSaveInstanceState(outState);  
  
    // Add information for saving HelloToast counter  
    // to the to the outState bundle  
    outState.putString("count",  
        String.valueOf(mShowCount.getText()));  
}
```

Two ways to retrieve the saved Bundle

- in onCreate(Bundle mySavedState)
Preferred, to ensure that your user interface, including any saved state, is back up and running as quickly as possible
- Implement callback (called after onStart())
[onRestoreInstanceState\(Bundle mySavedState\)](#)

Bundle Class

- Można zapisać nawet cały obiekt
- Klasa musi implementować interfejs `Parcelable`
 - Konstruktor przyjmuje obiekt typu `Parcel`
 - `writeToParcel(Parcel dest, int flags)`
 - `Parcelable.Creator`
- To podejście działa dobrze gdy OS zabija aplikacje
- Nie zadziała gdy użytkownik zabija aplikacje

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    mShowCount = findViewById(R.id.show_count);

    if (savedInstanceState != null) {
        String count = savedInstanceState.getString("count");
        if (mShowCount != null)
            mShowCount.setText(count);
    }
}
```

```
@Override
public void onRestoreInstanceState (Bundle mySavedState) {
    super.onRestoreInstanceState(mySavedState);

    if (mySavedState != null) {
        String count = mySavedState.getString("count");
        if (count != null)
            mShowCount.setText(count);
    }
}
```