

The background of the entire image is a light gray gradient. It is decorated with numerous water droplets of various sizes and shapes. Some droplets are large and prominent, while others are small and subtle. They are scattered across the frame, with a higher concentration in the top-left and bottom-right corners. The droplets have a realistic, glossy appearance with highlights and shadows.

# FRAGMENTS

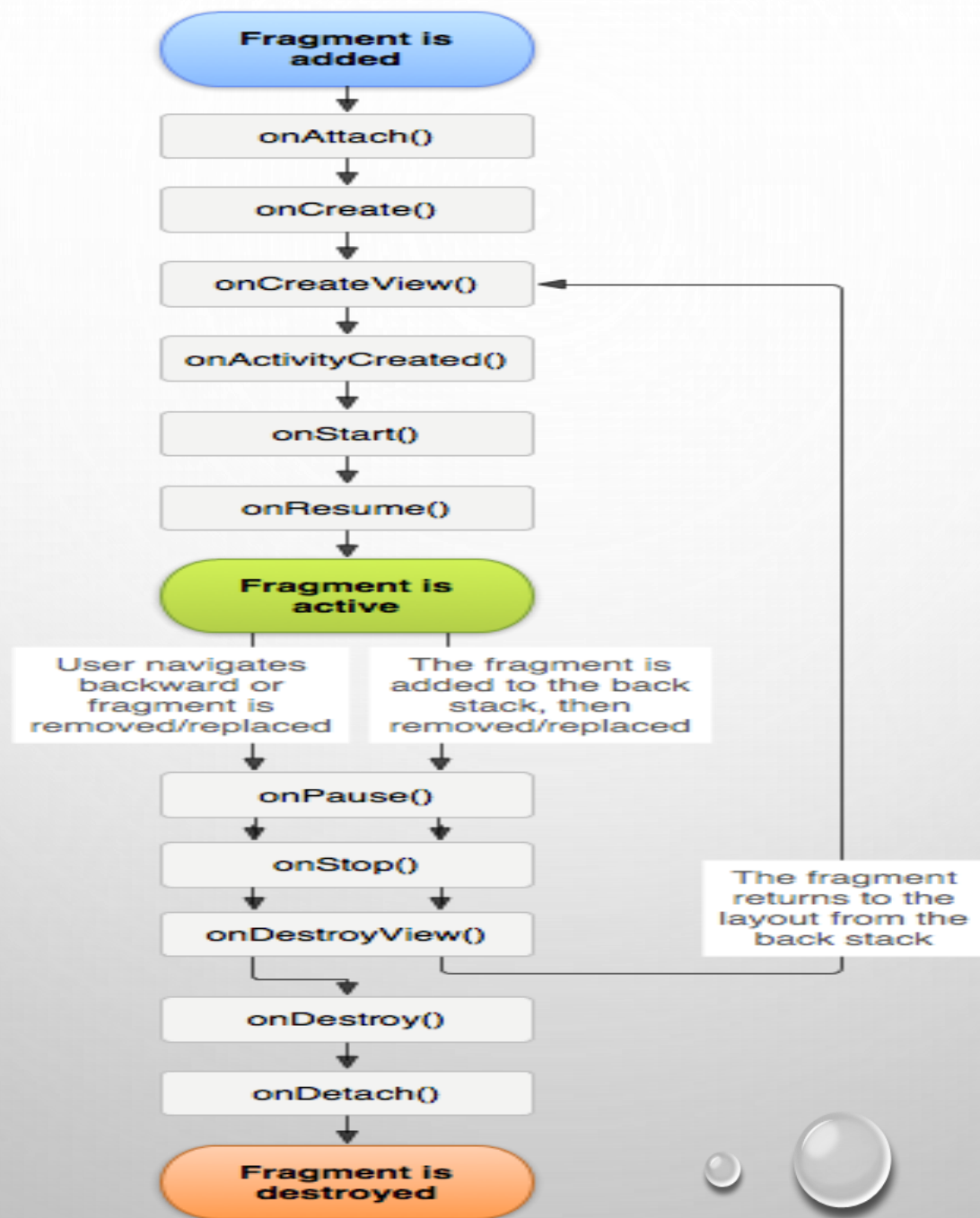
- A FRAGMENT REPRESENTS A BEHAVIOR OR A PORTION OF USER INTERFACE IN AN ACTIVITY
- YOU CAN COMBINE MULTIPLE FRAGMENTS IN A SINGLE ACTIVITY TO BUILD A MULTI-PANE UI AND REUSE A FRAGMENT IN MULTIPLE ACTIVITIES.
- YOU CAN THINK OF A FRAGMENT AS A MODULAR SECTION OF AN ACTIVITY, WHICH HAS ITS OWN LIFECYCLE, RECEIVES ITS OWN INPUT EVENTS, AND WHICH YOU CAN ADD OR REMOVE WHILE THE ACTIVITY IS RUNNING
- A FRAGMENT MUST ALWAYS BE EMBEDDED IN AN ACTIVITY AND THE FRAGMENT'S LIFECYCLE IS DIRECTLY AFFECTED BY THE HOST ACTIVITY'S LIFECYCLE.
- FOR EXAMPLE, WHEN THE ACTIVITY IS PAUSED, SO ARE ALL FRAGMENTS IN IT, AND WHEN THE ACTIVITY IS DESTROYED, SO ARE ALL FRAGMENTS.
- WHEN YOU ADD A FRAGMENT AS A PART OF YOUR ACTIVITY LAYOUT, IT LIVES IN A VIEWGROUP INSIDE THE ACTIVITY'S VIEW HIERARCHY AND THE FRAGMENT DEFINES ITS OWN VIEW LAYOUT.

- YOU CAN INSERT A FRAGMENT INTO YOUR ACTIVITY LAYOUT BY DECLARING THE FRAGMENT IN THE ACTIVITY'S LAYOUT FILE, AS A <FRAGMENT> ELEMENT, OR FROM YOUR APPLICATION CODE BY ADDING IT TO AN EXISTING VIEWGROUP.
- HOWEVER, A FRAGMENT IS NOT REQUIRED TO BE A PART OF THE ACTIVITY LAYOUT; YOU MAY ALSO USE A FRAGMENT WITHOUT ITS OWN UI AS AN INVISIBLE WORKER FOR THE ACTIVITY.

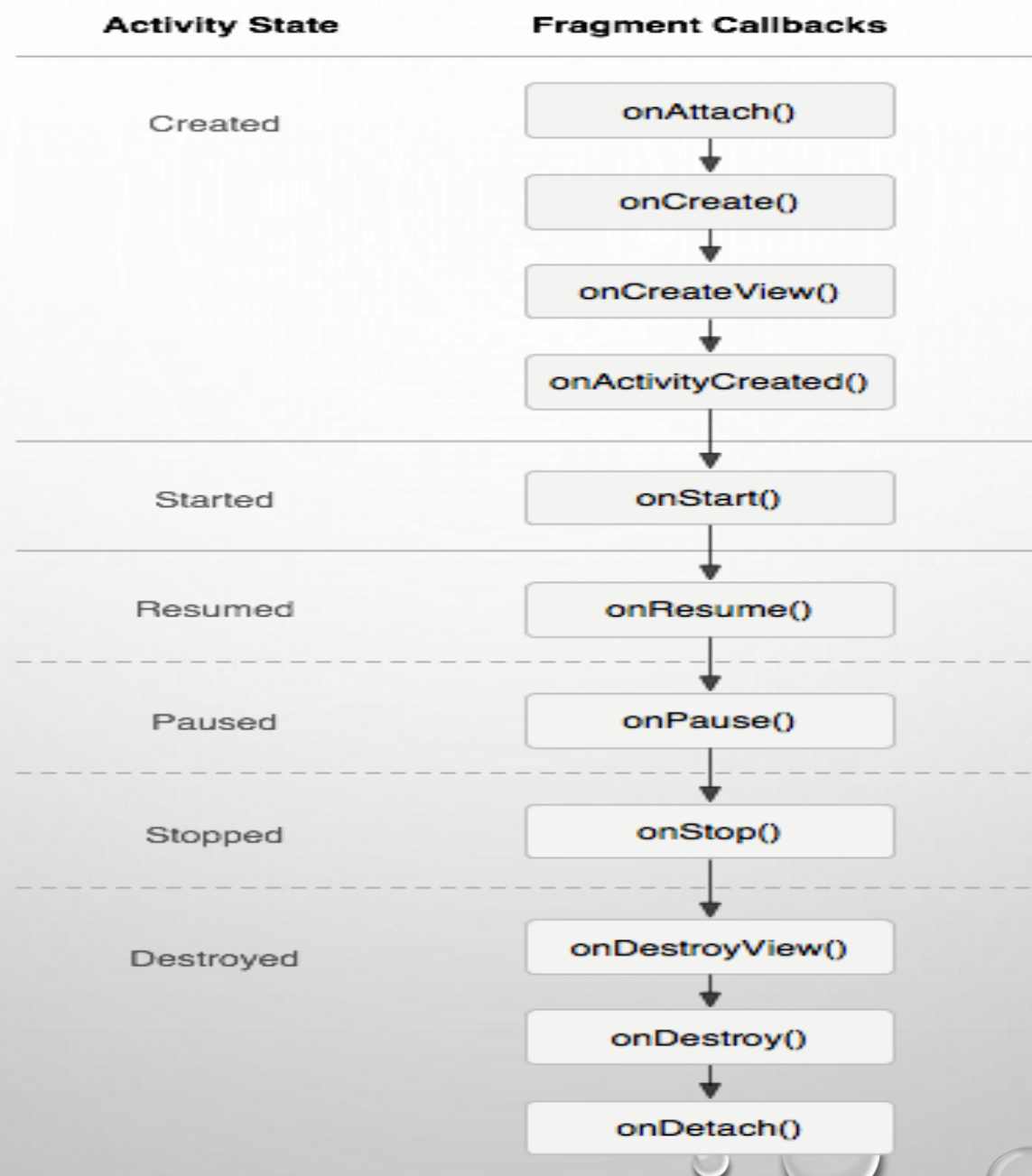
# CREATING A FRAGMENT

- TO CREATE A FRAGMENT, YOU MUST CREATE A SUBCLASS OF FRAGMENT (OR AN EXISTING SUBCLASS OF IT).
- THE FRAGMENT CLASS HAS CODE THAT LOOKS A LOT LIKE AN ACTIVITY.
- IT CONTAINS CALLBACK METHODS SIMILAR TO AN ACTIVITY, SUCH AS `ONCREATE()`, `ONSTART()`, `ONPAUSE()`, AND `ONSTOP()`.
- USUALLY, YOU SHOULD IMPLEMENT AT LEAST THE FOLLOWING LIFECYCLE METHODS:
  - `ONCREATE()`
    - THE SYSTEM CALLS THIS WHEN CREATING THE FRAGMENT.
    - WITHIN YOUR IMPLEMENTATION, YOU SHOULD INITIALIZE ESSENTIAL COMPONENTS OF THE FRAGMENT THAT YOU WANT TO RETAIN WHEN THE FRAGMENT IS PAUSED OR STOPPED, THEN RESUMED.
  - `ONCREATEVIEW()`
    - THE SYSTEM CALLS THIS WHEN IT'S TIME FOR THE FRAGMENT TO DRAW ITS USER INTERFACE FOR THE FIRST TIME. TO DRAW A UI FOR YOUR FRAGMENT, YOU MUST RETURN A VIEW FROM THIS METHOD THAT IS THE ROOT OF YOUR FRAGMENT'S LAYOUT. YOU CAN RETURN NULL IF THE FRAGMENT DOES NOT PROVIDE A UI.

- ONPAUSE()
  - THE SYSTEM CALLS THIS METHOD AS THE FIRST INDICATION THAT THE USER IS LEAVING THE FRAGMENT (THOUGH IT DOES NOT ALWAYS MEAN THE FRAGMENT IS BEING DESTROYED).
  - THIS IS USUALLY WHERE YOU SHOULD COMMIT ANY CHANGES THAT SHOULD BE PERSISTED BEYOND THE CURRENT USER SESSION (BECAUSE THE USER MIGHT NOT COME BACK).
- ONATTACH()
  - CALLED WHEN THE FRAGMENT HAS BEEN ASSOCIATED WITH THE ACTIVITY (THE ACTIVITY IS PASSED IN HERE).
- ONCREATEVIEW()
  - CALLED TO CREATE THE VIEW HIERARCHY ASSOCIATED WITH THE FRAGMENT.
- ONACTIVITYCREATED()
  - CALLED WHEN THE ACTIVITY'S ONCREATE() METHOD HAS RETURNED.
- ONDESTROYVIEW()
  - CALLED WHEN THE VIEW HIERARCHY ASSOCIATED WITH THE FRAGMENT IS BEING REMOVED.
- ONDETACH()
  - CALLED WHEN THE FRAGMENT IS BEING DISASSOCIATED FROM THE ACTIVITY.







# CONT ...

- THERE ARE ALSO A FEW SUBCLASSES THAT CAN BE EXTENDED INSTEAD OF THE BASE FRAGMENT CLASS:
  - DIALOGFRAGMENT
    - DISPLAYS A FLOATING DIALOG.
    - USING THIS CLASS TO CREATE A DIALOG IS A GOOD ALTERNATIVE TO USING THE DIALOG HELPER METHODS IN THE ACTIVITY CLASS, BECAUSE YOU CAN INCORPORATE A FRAGMENT DIALOG INTO THE BACK STACK OF FRAGMENTS MANAGED BY THE ACTIVITY, ALLOWING THE USER TO RETURN TO A DISMISSED FRAGMENT.
  - [LISTFRAGMENT](#)
    - DISPLAYS A LIST OF ITEMS THAT ARE MANAGED BY AN ADAPTER (SUCH AS A SIMPLECURSORADAPTER), SIMILAR TO LISTACTIVITY.
    - IT PROVIDES SEVERAL METHODS FOR MANAGING A LIST VIEW, SUCH AS THE ONLISTITEMCLICK() CALLBACK TO HANDLE CLICK EVENTS



# CONT ...

- [PREFERENCEFRAGMENT](#)
  - DISPLAYS A HIERARCHY OF PREFERENCE OBJECTS AS A LIST, SIMILAR TO PREFERENCEACTIVITY. THIS IS USEFUL WHEN CREATING A "SETTINGS" ACTIVITY FOR YOUR APPLICATION