(A5) GPS Service

GPS Service
extends
AbstractStatefulService

The intent that is passed to GPSService's onStartCommand(intent, flags, startId) is our action intent.

Wait for alarm to trigger.

Alarm triggers.

Android framework calls GPSService's onStartCommand(intent, flags, startId).

onStartCommand creates worker thread that calls AbstractStatefulService's onHandleIntent(intent)

AbstractStatefulService's onHandleIntent(intent) check for the intent's action code.

ACTION_START_SERVICE

ACTION_STOP_SERVICE

None of the other choices

ACTION_DEFERRED_TASK

Store information about the alarm (e.g. action code, action intent, pending intent, trigger time) in hashmaps within the service's serviceData.

Call AbstractStatefulService's handleStart(intent) In it, flush all flushable jobs in the job queue.

Call AbstractStatefulService's handleStop(intent)

Call AbstractStatefulService's handleIntent(intent)

Execute code in DEFERRED_TASK action.

PeriodicGetLocationState is the ONLY state for GPS Service!

Call current state's handleStop(intent, caller)

Call current state's handleIntent(intent, caller). The method is inherited from AbstractState.

intent action code does NOT have alarm prefix

Call current state's handleStart(intent, caller).

Call current state's leavingTo(newState, caller).

leavingTo is called when the service is going to stop. We don't want that to happen, so set service status to STARTING so that we can start the service again.

intent action code has an alarm prefix.

The intent is not an alarm intent. Call current state's processIntent(intent)

You can set the alarm trigger time in the alarmClockInfo.

Get AlarmManager and set an alarm by calling alarmManger.setAlarm(alarmClockInfo, pendingIntent).

Call current state's enteringFrom(oldState, caller)

Record GPS Service status as STARTING.

The intent is an alarm intent. Check if action code is in the service data.

In processIntent(intent), check if action code is UPLOAD_ONE_SERVICE

Set pull count number to 0.

Set current state to null.

Is In

NOT in

Is

Is NOT

Wrap a pending intent around the action intent by using PendingIntent class' getService(context, requestCode, intent, flags).

Pull count number is used in the formation of GPS alarm name.

Schedule GPS alarm.

Fetch alarm code from intent. Call current state's processAlarm(alarmCode).

Ignore delivery of alarm.

Upload GPS data to backend.

Log intent with unexpected action.

Record current GPS heartbeat time. Set service status to RUNNING.

In processAlarm(alarmCode), check alarm code to see what type of alarm it is.

We ignore because it was either scheduled by a different state (or by this state and we left it and returned).

Create new action intent(context, GPSService.class). Add action code to intent.

Schedule GPS heartbeat alarm.

GPS_KEEP_AWAKE_INTENT_ACTION (heartbeat)

Else

It's a GPS heartbeat alarm. Record current heartbeat time. Set service status to RUNNING.

It's a GPS alarm. Get current latitude and longitude. Check to see if we have last location.

Schedule a heartbeat alarm.

Have

Don't have.

Check the distance between last location and current location. If it meets a certain limit, save current GPS data and time.

Save current GPS data and time.

Schedule a GPS alarm.

Increment pull count.

alarmCallbackCompleted(actionCode) removes the data associated with the specific alarm from service data.

After processAlarm(alarmCode) is executed, ServiceData class' alarmCallbackCompleted(actionCode) is called.