Hubitat Community 2.2.9 singleThreaded Option for Apps & Drivers

(personal notes on this thread)

Notes

- **singleThreaded** - Boolean option
  - FALSE – Normal behavior
  - TRUE – Hub runs all methods for **an App|Driver instance** sequentially
    1. Load instance data (including state)
    2. Run a method
    3. Save the data (including state)
    4. Proceed to next method call.
  - This mode has lower overhead than using **atomicState**. The **App|Driver** behaves as though it is running in a single transaction **and** is always committed at the end, even if an exception is thrown.
  - **IMPORTANT CAVEAT:** The above applies to **top level methods only** – i.e., not to calls made by **App|Driver** methods.

```
definition ( // apps
  singleThreaded: true
)

metadata { // drivers
  definition (
    singleThreaded: true
  )
}
```

Questions & Answers

What happens if multiple threads attempt to interact simultaneously with a singleThreaded app or driver?

Hubitat's core software queues methods (FIFO) running one at a time. Other, non-singleThreaded callers bypass this queue behavior.

What about calls to other App|Driver methods from the same origin?

Hub-initiated methods – e.g., `runIn()`, `parse()` – run in a single-threaded context. When they invoke utility methods its within that single-threaded context.