DroneDB

Much like the adoption of the cell phone, non-military personal and business drones will achieve the same level of ubiquity. The first practical cell phones emerged in the 1970's, and at first were mostly expensive curiosities. However, by this year, 2015, active cell phone subscriptions will outnumber the earth's human population (6.8 billion).

I predict that Drones will follow a similar, although probably accelerated, adoption rate. This year, inexpensive non-military drones are predicted to exceed one million units. As their usefulness grows, and the number of drone-related accidents diminishes, they will be incorporated seamlessly into daily life.

Unlike military drones, which usually have a remote pilot (and thus appears to be the primary reason for so many crashes), and excluding the toy drones that work with only with a remote control, non-military home and business drones will have autonomous missions (e.g. video surveillance of a stationary or moving object, package delivery, location monitoring). They will be self-guided within a set of mission parameters and will not rely on human interaction. Early in their evolution, drone missions will be relatively simple (e.g. fly from point A to point B at this elevation), and will likely not require much onboard intelligence. But, as the complexity of drone missions grows, as well as the societal and community requirements and restrictions, so will the need for a real-time onboard drone database and the server databases that control them.

In this position paper I (briefly) explore the technical requirements for the future onboard Drone Database System (DroneDB) and the coordinating drone server cluster (DroneSwarmDB).

The DroneDB will operate in real-time, constantly assessing its location relative to multiple overlapping geographic zones that will likely require different actions. Some zones will represent drone charge or repair stations (for extended missions). Some zones will represent residential or business tracking areas, which require drone registration and authentication, and will have requirements and restrictions on drone flight paths and behavior. Some zones represent no-fly areas (e.g. anything military, such as Area 51), which must be avoided altogether. Drone courses will be plotted at the beginning of a mission, but the inevitable autonomous course correction will need updated map and zone information.

As is the case for any mobile database, the DroneDB must tolerate network disconnection from its coordinating swarm network servers. Thus, it must operate autonomously in isolation for periods of time, and then when network connection is restored, it must synchronize with its coordinating server in order to get mission updates (e.g. changes in zones or zone rules, changes in mission participants, changes in destination). In addition to coordinating server communication, DroneDB will synchronize with other drones using proximity communication to exchange alerts, zone information and mutual backup information.

The database running on the server cluster that controls the swarm of drones, DroneSwarmDB, tracks the active state of all drones in service and updates them with changes in their environment and mission. It is time-sensitive, geo-spatial aware and multi-media oriented. It manages all of the active drone missions, as well as any interactions between them. It also tracks the pool of hot-stand-by drones and deploys either a stand-by drone or a nearby active drone that can absorb additional criteria in its mission. The drone air traffic controller application employs DroneSwarmDB to manage the drone video feeds, predict drone paths for possible air collisions (drone zone intersection) and overall drone status.

The database system requirements for DroneDB are merely an evolution of existing database technologies: high performance real-time storage of operational and video data, geo-aware queries and data, high-speed backup and data synchronization capability, and the ability to avoid motion sickness. However, the larger technological challenge rests with the cluster of coordinating servers, the DroneSwarmDB. For small numbers of drones, on the order of a hundred, the multi-media data will not swamp a modern high-performance database cluster, even if all drones are submitting multiple multi-media transactions every second. However, for larger enterprises that will be engaging tens or hundreds of thousands of drones, the HD video data (25mb/sec per onboard drone camera) alone will present a network and database storage challenge.