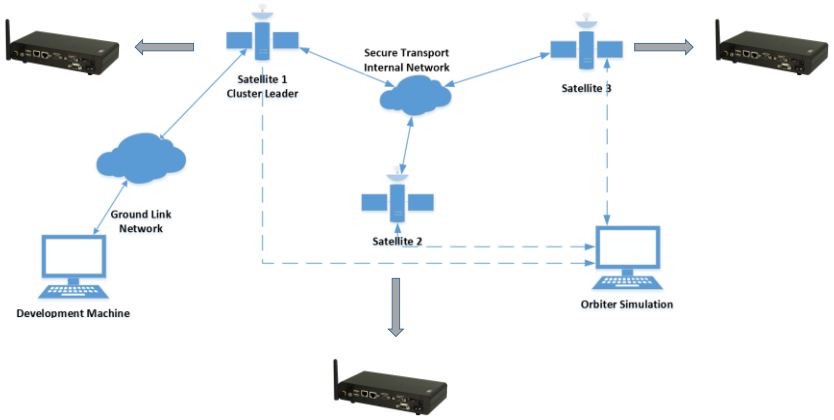
RTSS@Work Demo

3 Satellite Cluster Running Image Processing and Cluster Flight Software



To demonstrate the utility of DREMS for resource constrained systems running mixed-criticality applications, we configured a cluster of three embedded computing nodes connected to Orbiter Space Flight Simulator[1] which simulates the orbit physics and satellite hardware. Each satellite runs two applications: (1) the system-critical Cluster Flight Application (CFA), and (2) an Image Processing Application (IPA). The CFA consists of four actors which perform orbit maintenance and respond to emergency commands from a ground station. The three CFA actors involved with emergency response run as critical tasks, while the periodic orbit maintenance CFA actor runs in a temporal partition. Orbit maintenance periodically polls the satellite sensors and disseminates the new state to the rest of the cluster. Responding to emergency commands from the ground station is critical, i.e. when a *scatter* command is received by the cluster leader, all satellites should activate their thrusters simultaneously and maneuver away from each other. The IPA consists of four actors, each running image processing algorithms to consume as much of the CPU as possible. The four IPA actors are split evenly between two temporal partitions. These two applications were developed with the platform modeling tools, which required only 500 lines of code to be written between the two applications, comprising less than 0.1% of the applications' code.

The cluster communicates over a private gigabit subnet, to which is connected a traffic shaping node (TSN). All application network traffic must go through the TSN, which runs dummynet[2] for full control over the bandwidth, delay, and packet loss to simulate the cluster's satellite network. The nodes also support NTP with synchronization on the order of 10 µs. This synchronization allows accurate measurement of emergency response latency.

These demo applications show the performance of mixed-criticality real-time tasks on the system. The figure below shows the latency from the cluster leader's reception of the scatter command to each satellite's thruster activation. The scenarios demonstrate the invariance in the latency in response to changing application load and partition scheduling.

(1) http://orbit.medphys.ucl.ac.uk/

(2) http://info.iet.unipi.it/~luigi/dummynet/