FACE™ Vehicle Control, Navigation, and Guidance Architecture Experiments

August 17, 2017
BLUF

• PEO Aviation funded a FACE reference implementation (BALSA*)
  – Started with Remote Car to learn FACE/BALSA

• Using BALSA, the Air Force Resilient EGI team (on IRaD funding) rapidly prototyped an end-to-end vehicle control, navigation, and guidance system (for air and ground)
  – Demonstrated using a Piper Cub scale model (The Piper Cub scale model is electromechanically the same as a "real airplane")

• Resulting software is portable to “real” aircraft with reduced integration times
  – Shadow, Scan Eagle, Raven, Hunter, Apache, A-10, etc.

• Also integrated the safety-certified FACE TSS from RTI with Green Hills INTEGRITY RTOS

* Basic Avionics Lightweight Source Archetype

Result is a technically advanced and operationally ready framework ready to make any aircraft, missile, or spacecraft operate with a FACE computing environment
Component Integration into FACE

- Components designed with FACE interfaces
  - Embedded GPS/INS interface and Alt-Nav interfaces: magnetometer and LIDAR (IOSS and PSSS apps, 2-3 days ea.)
    - Already had interfaces and created new FACE™ apps
  - Position, Navigation, and Timing (PNT) app that estimates aggregate PNT variables (PCS app, 3 man months)
    - New app, FACE™ integration took a few days
  - Control Surface Servomotor interfaces (aileron, elevator, flaps, rudder) (IOSS and PSSS apps, 2 days ea.)
    - Already had interfaces and created new FACE™ apps
  - Propulsion interfaces (IOSS and PSSS apps, 2 days ea.)
    - Already had an interface and created new FACE™ apps
  - Flight Control app that has a stability augmentation system and does auto-launch, auto-land and waypoint navigation (new PCS app, 3 man months)
- Partitioned into two OFPs
Modified BALSA Data Flow
Phase IVa R/C Airplane

N.B. Servos: 14a, 15a Elevator; 14b, 15b Rudder; 14c, 15c Right Aileron; 14d, 15d; Left Aileron; 14e, 15e Right Flap; 14f, 15f Left Flap; 14g, 15g Electronic Speed Control (Throttle)
Lessons Learned

• Integration time reduced by a factor of 4 in most cases
  – Tasks that would normally take a month took a few days
  – Integration testing was almost trivial, the FACE data model method reduced data mismatches and integration/interoperability problems

• FACE reference implementation (BALSA) is essential; Bonus that the source code was available

• Concise reference implementation of the FACE standard made it easy for developers to get to the productive work of EGI/Alt-Nav design

• Very easy to work with FACE Standard compared to other DoD standards
  – Joint Unmanned Autonomous Systems
  – High Level Architecture

• IDI experience and technology allowed the programmers to pick up on BALSA easily; Some BALSA training could further reduce development time
Conclusion

- Demonstrates that the FACE approach enables integration of avionics components with reduced integration times (and thus cost)
- FACE Technical Standard and its supporting ecosystem is sufficiently mature to warrant incorporation into government solicitations
- FACE Reference Implementation (i.e. BALSA) has proven a good investment as it greatly aids the process
  - Developers will always need a publicly available reference architecture implementation
  - Ideas for improvements going forward will be prototyped and tested
- Accordingly, BALSA will be advanced to incorporate the FACE Technical Standard, Edition 3.0
Questions?
Keys To Success

- BALSA, the existing FACE reference implementation
  - Essential, and a bonus that the source code was available
  - Commercial products like the FACE TSS from RTI exist to upgrade this

- Data Model Training
  - Probably need specific training in EA and GME as well

- Integration/Interoperability Framework Training
  - IDI experience and technology allowed the programmers to pick up on BALSA easily
  - Could be supplanted by BALSA training
Recommendations

- Create a data modeling tutorial for the community and teach it at the meetings in the evenings
  - Same for BALSA
- Advance BALSA to v3.0 of the FACE standard
  - Must always have a publicly available reference architecture implementation
  - Ideas for improvements going forward should be prototyped and tested