A Service-based Architecture for Automated Guidance, Navigation, & Control Flight Software

Brendan O’Connor, Allen Brown, Kevin Gordon, Jason Schmidt, and Rafael de la Torre
PI Dr. Sun Hur-Diaz
Emergent Space Technologies
brendan.oconnor@emergentspace.com

Approved for Public Release, Distribution Unlimited
Notional System F6 On-Orbit Demo

F6 Tech Package

Spacecraft Bus Processor

Wireless Inter-Module Transceiver

Wireless Data

SB-SAT Terminal

F6 Tech Package

High-Powered Computer

High-Speed Ground Link Terminal

Inmarsat I-4 BGAN

Artist’s Concept

Courtesy DARPA

Approved for Public Release, Distribution Unlimited
Key Program Artifacts

- **On-Orbit Demo**—demonstrate universal attributes of fractionated architectures
  - Launch in 2015, 6-month duration, low-earth orbit (LEO)
  - Mission agnostic beyond demonstration of key fractionation capabilities
  - F6 payloads, spacecraft bus are being procured separately from other mission elements

- **F6 Developer’s Kit (FDK)**—everything needed for an independent 3rd party to develop a module that can fully participate in a fractionated cluster
  - Interface standards, network protocols, software, behaviors/rules
  - Freely distributed under an open source license, freely exportable
  - Solicited previously under FDK BAA (DARPA BAA-11-01)

- **F6 Technology Package (F6TP)**—low cost, commercialized physical instantiation of the FDK that enables a spacecraft bus to become a fractionated cluster module
  - Executes protocol stack, middleware, cluster flight software
  - Interfaces to wireless inter-module transceiver, F6 payloads, and spacecraft bus
  - Multiple sources, capable of supporting multiple spacecraft bus types
  - Goal is for a fully productized commercial off-the-shelf item

*Courtesy DARPA*
Cluster Flight Application (CFA)

Flight Software to Provide Autonomous Satellite Cluster Services

- Services
  - Autonomy
  - FDIR
  - Enhanced Navigation
  - Coordinated, optimized maneuver planning
  - Mission Management

- Development
  - Developed under a DARPA prime contract
  - Development cycle 30 month period ending 6/14
  - Primarily Use Case Driven
  - CMMI Level 3 Compliant
  - Scrum-based, agile software development process using small, distributed teams
  - Tested using NASA’s Trick Simulation environment
Architectural Characteristics

- Built to flight software standards
- Multiple Target Hardware Platforms
- Service-based, but message bus agnostic (Pub/Sub is the default assumption)
- “Reusable” components (Focused purpose, well documented, high-quality)
- Non-real-time
- Built around ARINC 653 partitions
- Models are the primary source artifacts for the algorithms
  - No modifications are made to the generated code
Code Development Flow

Matlab Model

Algorithm Code Generic C

UML Model

Mostly Hand generated Platform C++

Flight Code Platform C/C++

Blended code
Emergent’s Holistic Cluster Flight Solution

- **Station Keeping**
- **Ingress/Egress/Reconfiguration**

**Service-Based Architecture**
- Provides cluster flight algorithms as services
- Reside on any node including the ground

**Universal Maneuver Planning**
- Single maneuver planner for all scenarios
- Finds near optimal solution
- Capable of pre-calculating maneuvers for rapid response

**Cluster Flight System**
- **Cluster Flight Manager**
- **Fault Detection Isolation Recovery**
- **Orbit Maintenance**
- **Maneuver Planning**
- **Navigation**

**Flexible Navigation**
- Modules sharing sensor data
- Processes any measurements that are range, range rate, or angles
- Dynamically reallocate filters based on computational resources

**Scatter / Regather**

- **FDK**
Service-Based CFS Architecture Takes Advantage of Middleware-Based IA

• **Pros**
  - **Evolvable** – New services, clients, transactions can be added
  - **Adaptable** – Services can be moved from node to node, even to the ground
  - **Robust** – Can survive network disruptions and node failures

• **Cons**
  - Higher resource overhead related to messaging
Cluster Flight
Management
Maneuver
Planning
Orbit
Maintenance

Spacecraft 1 Message Bus

C&DH 1 Interface
Navigation
C&DH 2 Interface
Navigation

Maneuver Planning
Cluster Flight Management

Orbit Maintenance

Spacecraft 2 Message Bus

C&DH 4 Interface
Navigation
C&DH 3 Interface
Navigation

Ground Software Interface

Ground Node Message Bus

Spacecraft 4 Message Bus

Spacecraft 3 Message Bus

Approved for Public Release, Distribution Unlimited
Cluster Flight Manager (CFM)

- Software Agent orchestrating activities of the CFA
- Initiates periodic Cluster activities – Reconfiguring, Station-Keeping, Probability of Collision Monitoring
- Manages the Cluster Inventory
- Manages Maneuver Plans – Saves, distributes, and expires them updates Maneuver states as they are executed
- Synchronizes Cluster Environment Data (EOD Data) with NAV
Orbital Monitoring Service (OMS)

- Performs closed-loop cluster activities – Reconfiguring, Station-Keeping, Probability of Collision Monitoring
  - Determines whether predicted trajectories violate key target orbit constraints
  - Requests new maneuver plans ($\Delta V$ commands) from Maneuver Planning Service (MPS) as necessary
  - Validates candidate maneuver plans from MPS before distributing to CFM
  - Performs probability of collision determination upon request from CFM and during maneuver plan validation
Maneuver Planning Service (MPS)

- Designed to be deployed as a “job processing” Service
  - Message-based (service based architecture)
    - Clients submit maneuver requests on an as-needed basis
    - Application must remain responsive while computing
  - Maneuver planning returns, but does not store, results (“stateless”)
- A “standardized” interface
  - Supports all maneuver types (Ingress, SK, etc.)
  - With constraints, objectives, etc.
  - Single request = single result

\[\text{Request: Maneuver goals, constraints, design parameters}\]

\[\text{Maneuver Planning Service}\]

\[\text{Result: Optimal target and maneuver plans}\]

Requests originate anywhere: other CFA components, operators, etc.

Algorithm Tuning, Platform Tuning

Approved for Public Release, Distribution Unlimited
Simulated Annealing Search State
Description & Burn Solver Relationships

10 km keep-out ranges
On propagated orig traj's at T=5min

SA Scatter States (or each module):
Module destination relative orbit data
Earliest time of 1st burn (fixed for immediate scatter)
Latest time of final burn Scatter Target Azimuth
Scatter Target Elevation Scatter Target Range

LP solves for:
Number of burns, Impulsive burn dVs, burn times, burn vectors,
Given: module states at 1st and final times, Scatter Target position and time

Selected Module Destination Relative Orbit:
The trajectory of the module at the end of the maneuver

Times of interest:
To = cmd received
Ti = 1st burn time
Tx = intermediate burn times
Tf = final burn time
Tend = scatter maneuver time limit

Burn, Ti
Burns to target point

Burn, Tf

Target Point for module at T=5m (outside keepouts)

V at target point Must match both trajectories

Cluster Mean Orbit

Approved for Public Release, Distribution Unlimited
Navigation Service (NAV)

- Identical Navigation Flight Software (FSW) Components are located on each module in the cluster.
- Each component runs a GPS pseudorange filter and a relative range filter in parallel and exchange navigation solutions and lists of tracked GPS satellites with other modules in the cluster.
Requesting a Maneuver Plan

alt Maneuver Plan Required Check
[If A Maneuver Plan is Required]

handleMsg(MPRequestRequired)

CFM should adjust its time-out timer for OMS to account for the fact that a maneuver plan is required, so the OMS request will take longer.

ref

GenerateAClusterManeuverPlan

alt New Maneuver Plan
[A Valid Maneuver Plan is Generated]

handleMsg(MPRequestResult)

The MPRequestResult contains the MP.

handleMsg(MPGenerationResult)

The MPGenerationResult Message contains information on the success of the MP Generation, but it doesn’t contain the MP itself. The MPRequestResult contains the MP.

ref

DistributeClusterManeuverPlan

Valid CMP produced

CheckClusterManeuverStatus - Exit - CMP Distribution Complete

[No Valid Maneuver Plan is Generated]
Architectural Factors that Make a Good Flight Software Services

- Relatively few “state” parameters
- Process atomic transactions
- Focused purpose
- Self-contained requests in a relatively narrow range
- Continue to provide services, if at all possible
- Be mission, platform, and message bus agnostic
- Service must be testable on its own
CFA Forward Plans

- Port to the cFE/CFS
- Make it open source
- Add Monte Carlo Capability
  - Python based Monte Carlo driver for managing cloud-distributed jobs
- GPS Model Upgrades
  - Conversion of high-fidelity GPS constellation model to C++ classes
  - Validation with Google unit tests
- HWIL interfaces for testing with GPS signal simulator
- Flight platform benchmark testing
- CFA mission testing and performance characterization
Emergent Space Technologies

- **Small Business**
  - Founded August, 2001
  - Solid growth each year since

- **Headquartered in Greenbelt, MD**
  - Houston, Austin, Denver, Colorado Springs, Albuquerque

- **Reliable Flight Software Development**
  - Proven CMMI Level 3 Certified Agile Software Development

- **Experienced in Multi-Satellite System Development and Operations**
  - Orion, Orbital Express, F6, ANGELS, MMS, Globalstar, Iridium, HST SM4

Contact: Brendan O’Connor, Chief Software Architect
brendan.oconnor@emergentspace.com
(512) 791-5902