Prototype Flight Dynamics Ground System
Leveraging a Service-Oriented Architecture and Reusable COTS Services

Sara Case¹
Wsam Gazi²

¹Aerospace Engineer, Flight Dynamics Systems Division, a.i. solutions, Inc.
²Systems Engineer, Flight Dynamics Systems Division, a.i. solutions, Inc.

© 2012 by a.i. solutions, Inc. Published by The Aerospace Corporation with permission.

www.ai-solutions.com | 301.306.1756
Service Oriented Architectures

- **Approach:** Each ground system component acts as a service, communicating with other services through a message bus
  - Each service has a single, common interface to the bus, rather than a separate interface for each of the other components
  - Flight dynamics, TT&C, scheduling, etc.
- **Result:** Modular system architecture
  - Flexibility to incorporate services from multiple vendors
  - Services can be added or removed without having to rebuild surrounding components
  - Rapid system development – components can be designed, built, and tested without knowing implementation details of other services
  - Reduced effort and cost when adding, removing, or upgrading components over time
  - Services can be shared and reused in other systems
Prototype Flight Dynamics Ground System

• Goal: Develop a prototype flight dynamics ground system to demonstrate a service-oriented architecture with real-world astrodynamics applications

• Framework: GMSEC (Goddard Mission Services Evolution Center)
  – Publish/subscribe capabilities
  – Standardized message formats
  – Middleware for several message bus platforms

• Development Environment: COTS flight dynamics software FreeFlyer®
  – High-fidelity astrodynamics engine
  – Flexibility through scripting language and Extensions SDK
  – Unrestricted integration with any ground system environment
Flight Dynamics Services

• Six services perform real-time flight dynamics analysis for a set of 60 GPS and geosynchronous satellites
  – Each service is a reusable component that communicates through the message bus using standardized data formats
  – Services run simultaneously and autonomously, each performing a different function

• A separate instance of FreeFlyer executes each service
  – Scheduler
  – Tracking Data Simulation
  – Real-Time Orbit Determination
  – Ephemeris Generation
  – Collision Avoidance Screening
  – Maneuver Planning
Scheduler
- Publishes Start and End directive messages
  - SC ID, GS ID

Real-Time Orbit Determination
- At end of each pass, publishes estimated state
  - Epoch, Position, Velocity

Collision Avoidance Screening
- Screens predictive ephemeris against sample debris catalog
- Publishes conjunction detection messages

Tracking Data Simulation
- Publishes Simulated Tracking Data
  - SC ID, GS ID, Epoch, Range, Azimuth, Elevation

Ephemeris Generation
- Propagates state for 1 day
- Publishes predictive ephemeris
**Scheduling Service**

- Keeps track of the simulation epoch
- Identifies the spacecraft and ground station ID to simulate
- Publishes start and end simulation directives through the GMSEC connection
Tracking Data Simulation Service

- Subscribes to start and end tracking directives for GPS and GEO satellites
- No a-priori knowledge of the tracking schedule or the number of satellites or ground stations → single or multi-mission support
- Simulates observations of each satellite from the commanded ground station
- Observations are published in the AFSCN SIS 508 1/Sec Status message format
Real-Time Orbit Determination Service

- Uses separate Kalman Filters to estimate state and covariance for each satellite
- Processes each AFSCN 508 observation as it is received
- Updates estimated state for the corresponding satellite in real time
- 2D and 3D visualizations for situational awareness of constellation and ground station State of Health
- At the end of each pass, a solution state is published in CCSDS format
Ephemeris Generation Service

- The estimated state is used to create a 1-day predictive ephemeris in CCSDS format
- Predictive products are generated, including visibility times and look-angles to ground stations
- Such products are used operationally for contact planning
Collision Avoidance Screening Service

• The predictive ephemeris is screened against a sample catalog of 500 objects
• Catalog objects are filtered by apogee/perigee
• Objects within a specified miss distance will trigger a conjunction detection message which includes:
  – Time of closest approach (TCA)
  – Range at TCA
  – Probability of Collision (Pc)
Data Flow Diagram Revisited

Scheduler

Real-Time Orbit Determination

Collision Avoidance Screening

Message Bus

Tracking Data Simulation

Ephemeris Generation

Maneuver Planning
- Performs collision avoidance maneuver for a GPS spacecraft
- Publishes post-maneuver state
**Maneuver Planning Service**

- Operator requests a screening
- If a conjunction is detected, a risk mitigation maneuver is planned
- Custom GUI displays details
- If operator executes the maneuver, the service publishes the post-maneuver state
  - Generates a new predictive ephemeris
  - Re-screens against the catalog
Observations and Conclusions (1)

- Because of the standardized GMSEC message format and the flexibility of FreeFlyer’s scripting environment, individual astrodynamics services were easy to reconfigure
  - Adding messages for conjunction detection near the end of development required minimal effort
    - Desired metrics were already calculated by the COTS tool
    - Message architecture was already in place
  - A single Collision Avoidance Screening and Maneuver Planning service was easily separated into two services
    - No hardcoded dependencies on the number of components in the system or which component publishes particular messages
  - Additional services or space/ground assets could be added in the future
Observations and Conclusions (2)

• This prototype has also been used to demonstrate ease of interoperability with other GMSEC-compliant systems
  – An IRAD effort to create a real-time GMSEC interface in the web browser
  – The SMC/SN Compatible Satellite C2 Prototype coordinated by Aerospace Corporation

• In both cases, the integration effort was minimal, as each system had been configured to communicate using the same GMSEC standards prior to integration
Questions?