J. Astron. Space Sci. 26(2), 267–278 (2009) [기술논문]

Analysis and Design of the Automatic Flight Dynamics Operations For Geostationary Satellite Mission

Byoung-Sun Lee[1]†, Yoola Hwang[1], Sangwook Park[2], Young-Ran Lee[2], and Javier Santiago Noguero Galilea[3]
[1]Satellite Control and Navigation Research Team, ETRI, Daejeon 305-700, Korea
[2]Aerospace System Team, Satrec Initiative, Daejeon 305-811, Korea
[3]GMV Aerospace and Defence, S.A., E-28760 Madrid, Spain
email: lbs@etri.re.kr

(Received May 11, 2009; Accepted May 23, 2009)

Abstract

Automation of the key flight dynamics operations for the geostationary orbit satellite mission is analyzed and designed. The automation includes satellite orbit determination, orbit prediction, event prediction, and fuel accounting. An object-oriented analysis and design methodology is used for design of the automation system. Automation scenarios are investigated first and then the scenarios are allocated to use cases. Sequences of the use cases are diagramed. Then software components and graphical user interfaces are designed for automation. The automation will be applied to the Communication, Ocean, and Meteorology Satellite (COMS) flight dynamics system for daily routine operations.

Keywords: satellite orbit, flight dynamics, mission control, automation, COMS

1. Introduction

Automation of the satellite mission control is very important to modern satellite operation environment in two reasons. One is the saving of the man power in the satellite mission control and the other is to prevent satellite operator from mistakes in manual operation process. Normally, automation of the satellite mission operation starts from daily routine operations. The automation makes the satellite operator focus on more important and urgent mission operation. Nowadays autonomous systems should be applied not only in mission control centers on the ground, but also on spacecraft, on rovers, and other space assets (Truszkowski et al. 2006).

Satellite mission control consists in satellite tracking and ranging, telemetry reception and processing, command generation and transmission, satellite mission scheduling and planning, space flight dynamics operations, and satellite simulations. A successful satellite mission control operation can be achieved by the man power and the control system in harmony.

Flight dynamics operation for a geostationary satellite mission includes spacecraft Orbit Determination (OD), Orbit Prediction (OP), Event Prediction (EP), station-keeping maneuver planning, station-relocation maneuver planning, and fuel accounting (FA) (Lee et al. 2008b). Flight dynamics operations are normally carried out as an off-line processing using flight dynamics computer during

†corresponding author

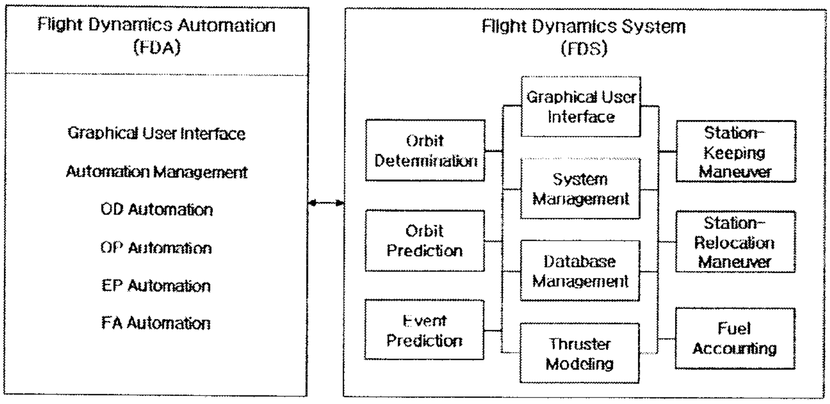

Figure 1. Functional architecture of the FDA and FDS.

the normal business hours. Input data to flight dynamics operations are satellite tracking data from ground station antenna and state-of-health data from real-time telemetry processing. Output data from flight dynamics operations are satellite antenna pointing data for tracking antenna, satellite orbit and event data for mission planning, and orbit determination and prediction data for payload processing.

Satellite orbit determination and prediction are the most important functions in normal flight dynamics operations. The orbital elements estimated from the orbit determination are used as basic input data for satellite orbit prediction and event prediction. The orbit prediction and event prediction data should be used for satellite mission planning.

Automation of the routine flight dynamics operations including orbit determination and orbit prediction has been successfully implemented and tested for the low Earth orbit satellite operations of the KOMPSAT series satellites (Lee et al. 2008a). It has a very similar functionality of NASA's Orbit Determination Automation System (ODAS) (Mardirossian et al. 1989, Chapman et al. 1994).

In this paper, automation of the key flight dynamics functions for satellite OD, OP, EP, and FA are analyzed and designed for the geostationary satellite mission operations. Object oriented analysis and design methodology is used for system automation. At first, automation scenario is investigated and then the scenario is allocated to use cases and sequences. Unified modeling language is used for the system analysis (Pooley & Stevens 1999). Software components are extracted and the graphical user interfaces are designed. Communication, Ocean, and Meteorology Satellite (COMS) Flight Dynamics System (FDS) is used for the target system for automation.

2. Analysis of the Flight Dynamics Automation

2.1 Flight Dynamics System and Its Automation

Flight Dynamics System (FDS) is a computer based software system. The FDS for the geostationary satellite mission includes satellite OD, OP, EP, station-keeping maneuver planning, station-relocation maneuver planning, and FA. The FDS is operated by the system management and database management. Thruster modeling is a hidden function of the FDS for thruster firing related calculations. The FDS is normally supported by graphical user interface for easy operation, regardless of

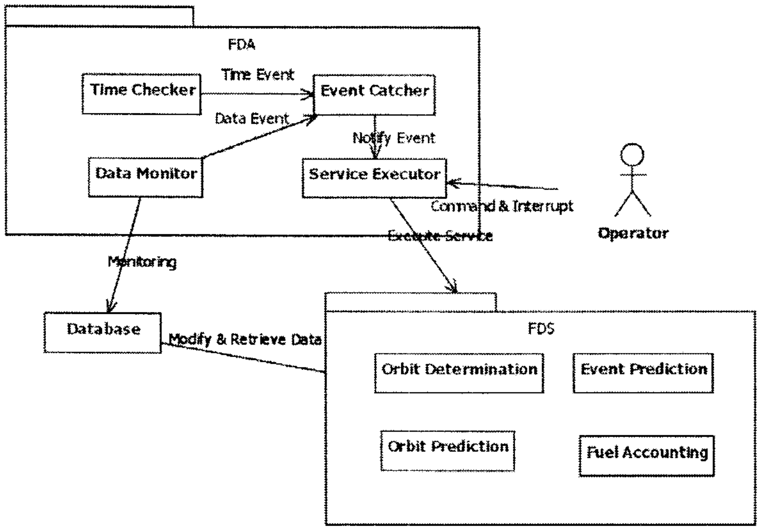

Figure 2. Domain modeling of FDA and FDS.

the automation.

Flight Dynamics Automation (FDA) focuses on automation of OD process, OP process, EP process, and FA process. OD estimates the orbital state of a satellite using tracking and ranging data. Normally, the satellite orbit will be determined once a day using two or three days tracking and ranging data. OP propagates the orbital state of a satellite using the orbit determination results. The OP results will be used for satellite mission planning and payload processing. EP finds various orbital events for satellite mission planning and operations. The EP results can be used in mission planning and payload data users. FA estimates the satellite fuel consumption during the thruster firing operations using telemetry data from satellite. All of the FDA will be processed once in a day. The FDA is the driver of the related functions in FDS. Figure 1 illustrates a high level S/W functional architecture of the FDA and FDS.

Figure 2 shows the domain modeling figure of FDA and FDS. FDA is invoked by the specified time and data. Time Checker is responsible to generate an event that is related with periodic job or specific time. Data Monitor is another event generator watching on the specific data. Event Catcher identifies an event that is triggered by Time Checker and Data Monitor. Then the Event Catcher generates event to execute the FDS services. The operator can command and interrupt the FDA.

2.2 Automation of Orbit Determination

OD provides the estimated position and velocity of the satellite based on the measurement data. A batch weighted least square estimation is normally performed everyday using the angle tracking and ranging data from ground station. A priori orbit state from orbit stack data file is used for starting of the orbit estimation. Automation of OD focuses on the input and output file handling and execution of the OD program. Automation scenario of OD is as follows.

- At predefined time, check if there is a new tracking data file in a designated directory (including data transmission failure check)
- Generate a new input data file for orbit determination program

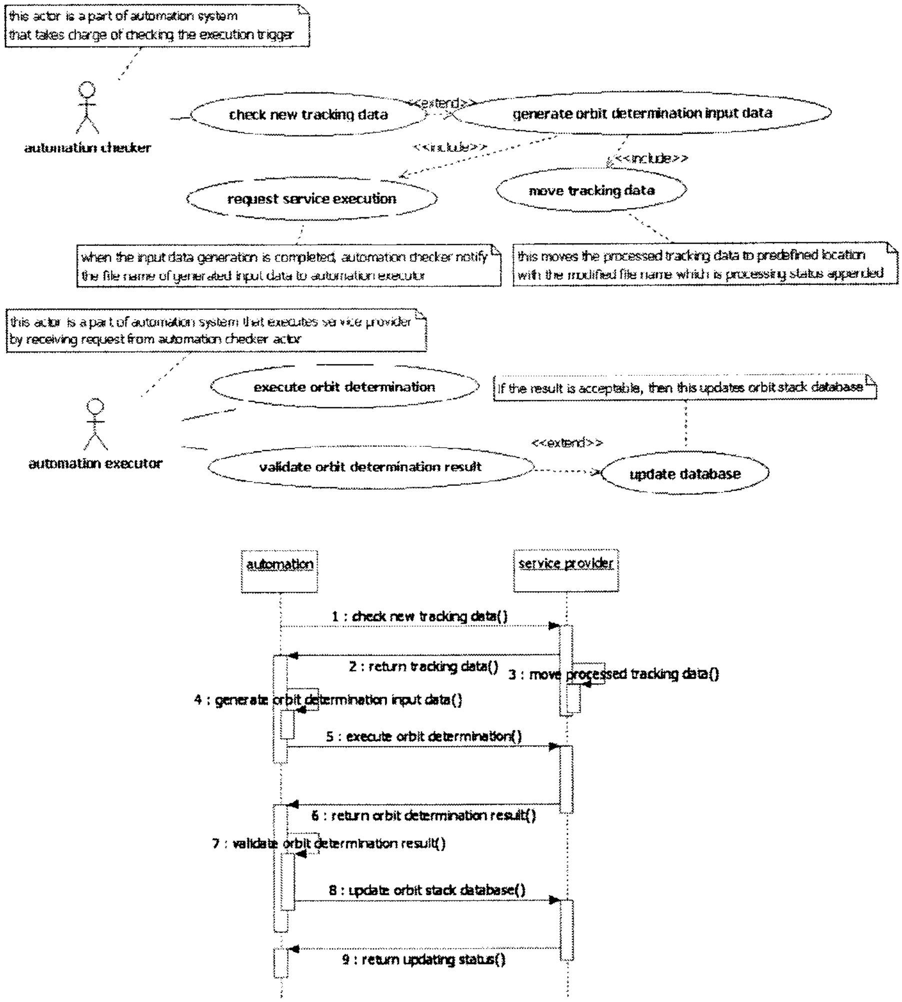

Figure 3. Use case diagram and sequence diagram for automatic orbit determination.

- Execute orbit determination program
- Evaluate the orbit determination results (including failure check)
- Generate a report file for orbit determination
- Update orbit stack data
- Move the old track data file to the specified directory according to the OD execution result (ex: Success/Fail/Update Success/ Update Fail)

Figure 3 shows the use case diagram and sequence diagram for automation of the orbit determination.

2.3 Automation of Orbit Prediction

OP provides the future position and velocity of the satellite based on the orbit determination results. Normally, the OP is performed using the numerical integration of the satellite orbit equation.

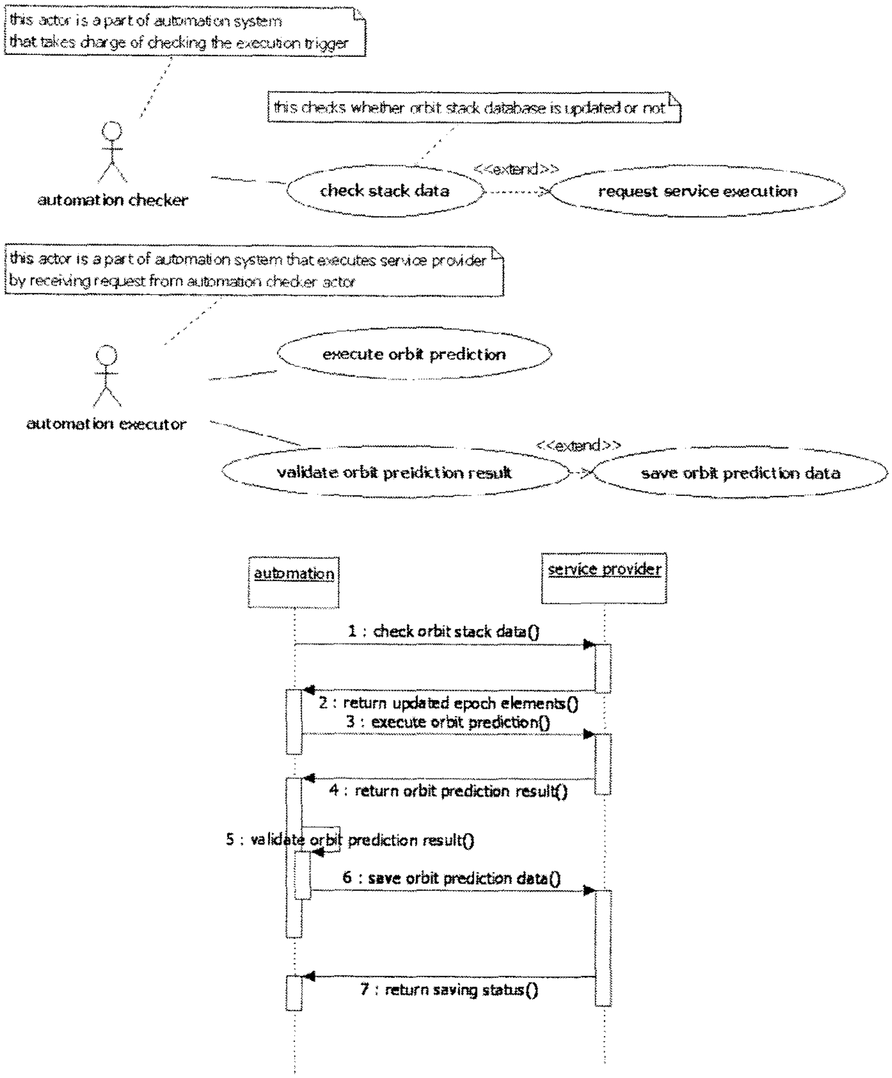

Figure 4. Use case diagram and sequence diagram for automatic orbit prediction.

Automation scenario of OP is as follows.

- At predefined time, check if there is a new orbit stack data in a designated directory
- Generate a new input data file for orbit prediction program
- Execute orbit prediction program
- Evaluate the orbit prediction results (including failure check)
- Generate a report file for orbit prediction

Figure 4 shows the use case diagram and sequence diagram for automation of the orbit prediction.

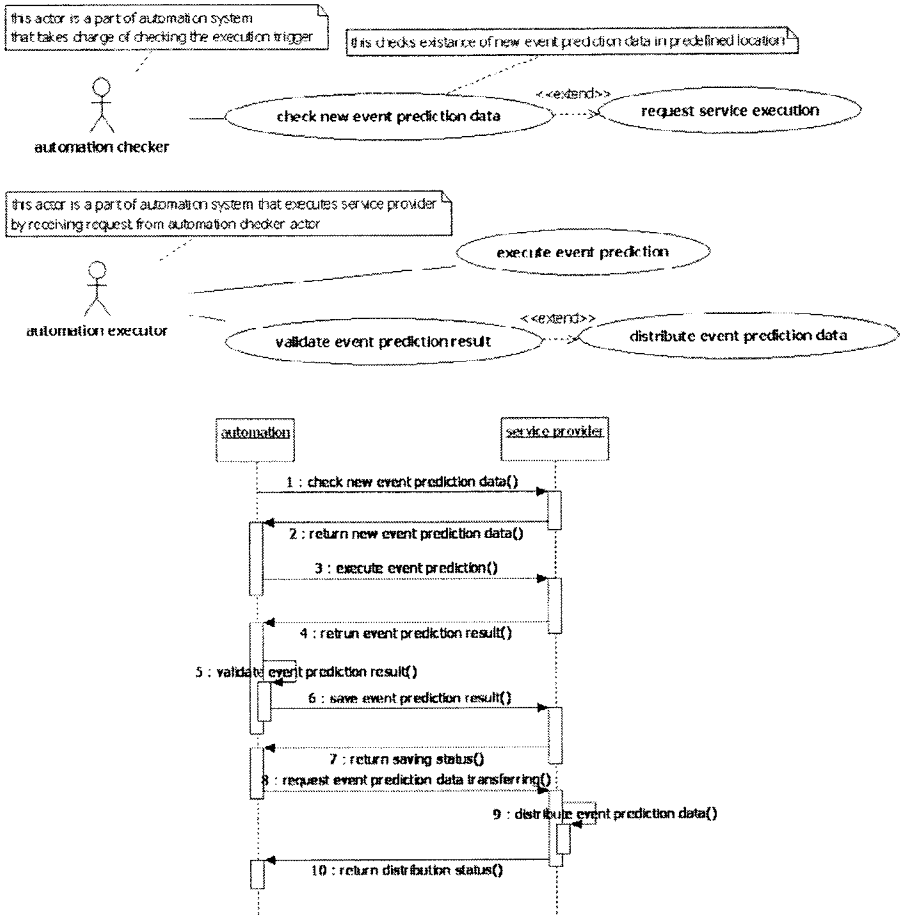

Figure 5. Use case diagram and sequence diagram for automatic event prediction.

2.4 Automation of Event Prediction

EP provides Sun eclipse time due to the Earth and the Moon, satellite sensor intrusion time, ground station sun-interference time, and station-keeping box boundary of the spacecraft. The event prediction also provides the various orbital events such as nodal crossing time and apsidal passing time. Automation scenario of EP is as follows.

- At predefined time, check periodically if there is a new event prediction data in a designated directory
- Generate a new input data file for event prediction
- Execute event prediction program if necessary
- Evaluate the event prediction results (including failure check)
- Generate a report file for event prediction
- File transfer to related subsystems for satellite operations

Figure 5 shows the use case diagram and sequence diagram for automation of the event prediction.

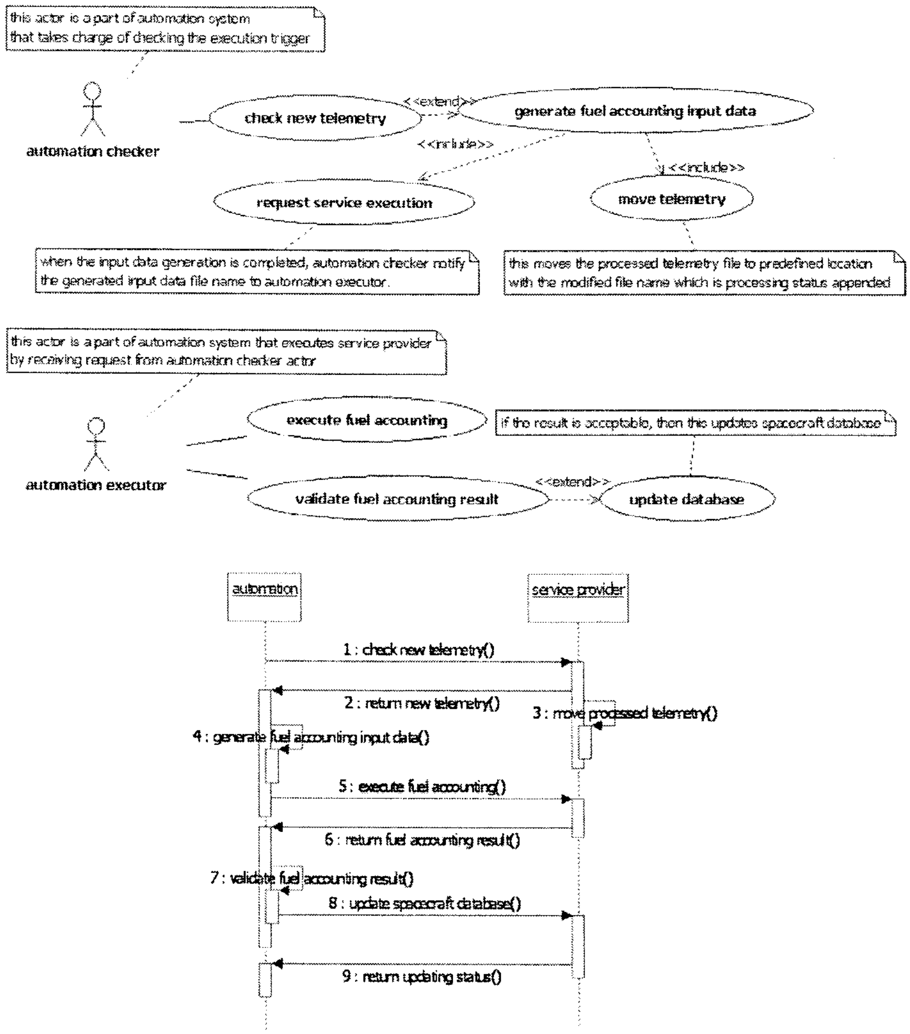

Figure 6. Use case diagram and sequence diagram for automatic fuel accounting.

2.5 Automation of Fuel Accounting (FA)

Fuel accounting provides the capabilities for analysis of remained fuel mass inside the spacecraft fuel tank by telemetry data. Fuel accounting consists of two programs. Pressure, Volume, and Temperature (PVT) method using temperature and pressure of the fuel tank transmitted from telemetry data and Thruster-On-Time (TOT) method that calculates remaining fuel accounts using pulse of the thruster firing from telemetry data. Automation scenario of FA is as follows.

- At predefined time, check if there is a new telemetry data in a designated directory (including failure check)
- Generate a new input data file for fuel accounting program
- Execute fuel accounting program

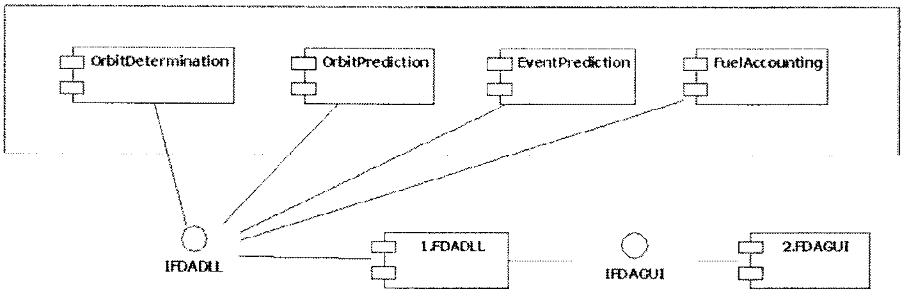

Figure 7. Package diagram of FDA and FDS.

- Evaluate the fuel accounting results (including failure check)
- Generate a report file for fuel accounting
- Update the spacecraft database

Figure 6 shows the use case diagram and sequence diagram for automation of the FA.

3. Design of the Flight Dynamics Automation

Design of the flight dynamics automation consists in package design, class design, component design, deployment design, and user interface design. It is a variation of the 4+1 model view of system architecture by Kruchten (1995). Figure 7 shows package diagram of the FDA and FDS. There are FDA Dynamic Link Library (DLL) and FDA Graphical User Interface (GUI) with their interfaces.

Figure 8 shows the component diagram of FDADLL and FDSGUI. In the FDADLL, CServiceExecuter plays an important role to manage the execution of the application program. By the request of CEventCatcher, CServiceExecuter executes the function of OD, OP, EP, and FA services. CTimeChecker defines the functions that manage the time elapsing sequence and the specific time conditions. CEventCatcher has the function to check the change of FDA related data and to request the FDS service execution by the time elapsing event handling. CDataManager has the function to generate and manage the FDA related input and output data. In the FDSGUI, FrmAutomationMain manages all of the parameter settings for the application program and related interfaces. FrmAutomationMain also displays FDA execution logs and results. The other classes in the FDSGUI provides the user interfaces for the setting of general parameter, external interface parameter, and operation parameters in each application program.

FDA will be collocated in the FDS computer system. And the FDS computer is linked via LAN with related system computers. FDS and the other systems are communicated via FTP protocol. Figure 9 shows the deployment diagram for FDA and other systems.

Figure 10 shows the design of automation main Graphical User Interface (GUI). The GUI shows the network status of Tracking, Telemetry and Command (TTC), Real-time Operations System (ROS), and Mission Planning System (MPS) for data exchange. The execution time and interval for the four FDS application programs can be set in the GUI. The status of the automation is displayed in the right hand side.

Figure 11 shows the design of FDA OD and OP parameters setting user interface. The parameters setting user interface is very similar to the COMS FDS GUI because COMS specific operations

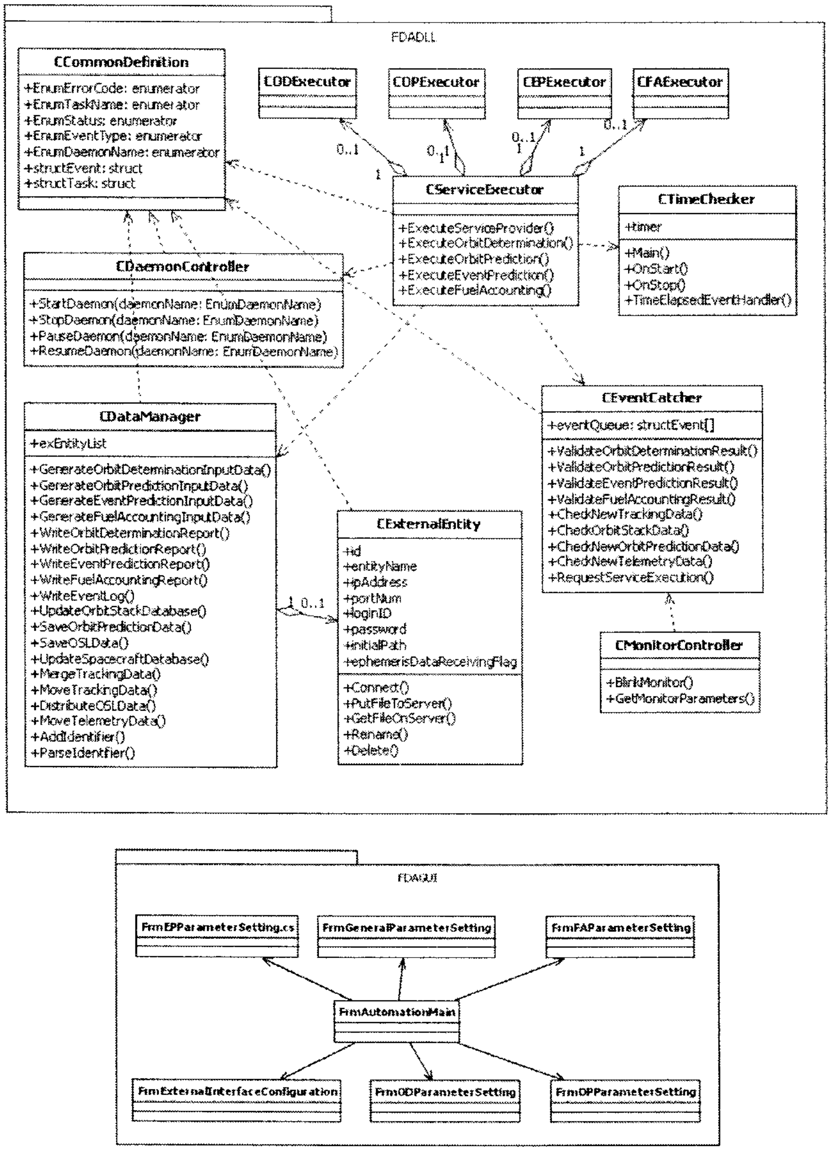

Figure 8. Component Diagram of FDADLL and FDAGUI.

such as wheel off-loading and station-keeping maneuver should be included.

Figure 12 shows the design of EP and FA parameters setting user interface. EP duration can be changed by operator for a day, a week, a month, or user defined days. Wheel off-loading set-point management is the COMS specific functions to calculate set-point parameters. Ground station should be selected in EP to calculate the time for the ground station antenna interference due to the sun. For the FA automation, either PVT or TOT can be selected to estimate the remaining fuel amount of the spacecraft.

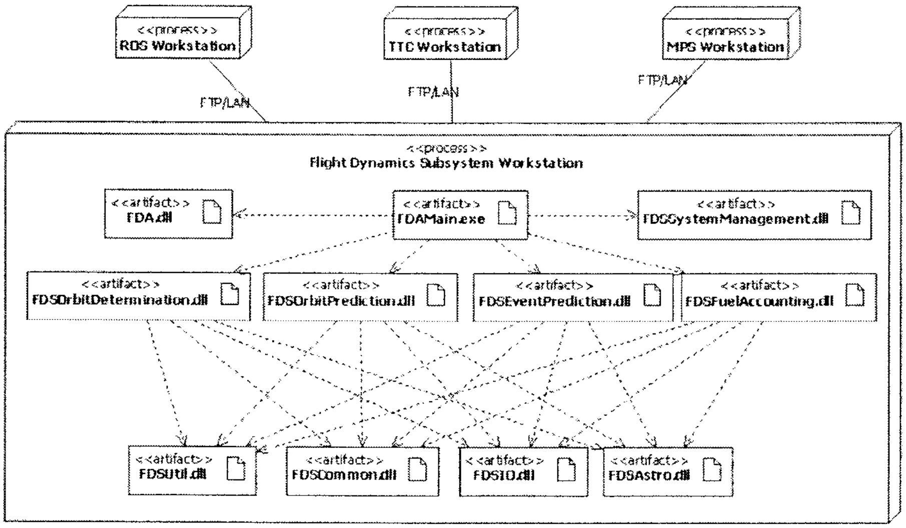

Figure 9. Deployment Diagram of FDA.

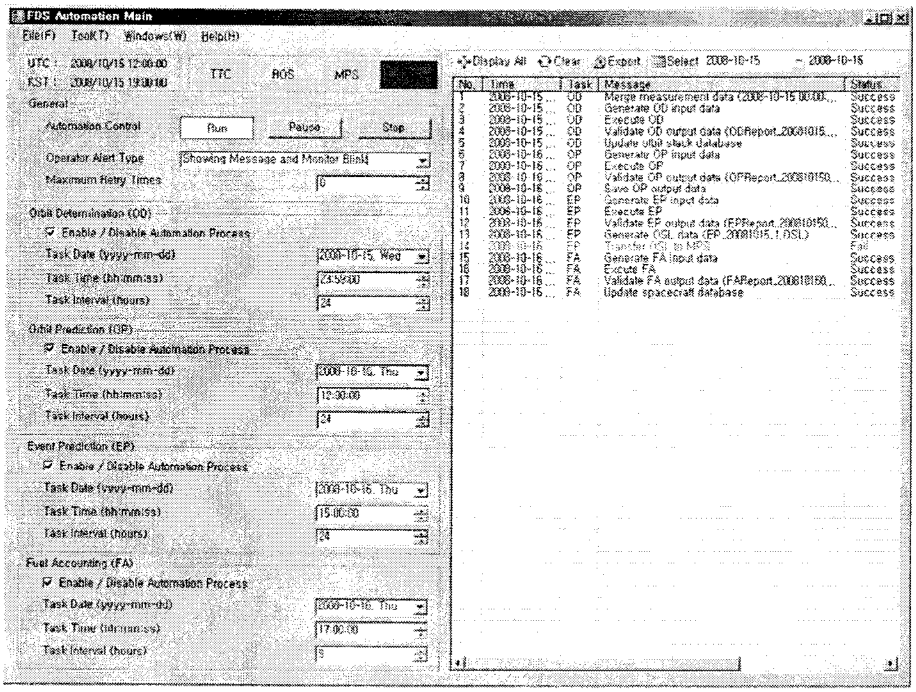

Figure 10. Main GUI design of FDS automation.

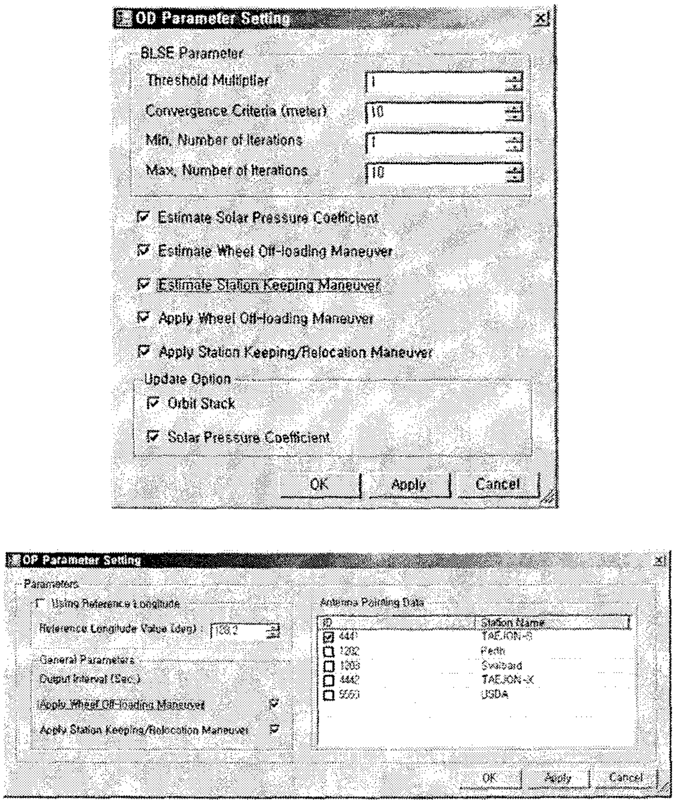

Figure 11. GUI design of OD and OP parameter setting.

4. Conclusions

Automation of the key flight dynamics operations for geostationary orbit satellite mission control have been analyzed and designed. Daily routine operations including satellite orbit determination, orbit prediction, event prediction, and fuel accounting have been successfully designed for implementing to COMS flight dynamics system. An object oriented analysis and design methodology have been applied to scenario investigation, use case modeling, domain modeling, and software architecture design. Software architecture design includes package design, class design, component design, deployment design, and user interface design. The flight dynamics automation will be implemented and tested in the near future for COMS mission operations.

Acknowledgements: This work was jointly carried out under the project of Korea & Spain Innovating (KSI). The work by the Electronics and Telecommunications Research Institute (ETRI) and Satrec Initiative was carried out under the contract with the Korea Evaluation Institute of Industrial Technology (KEIT) for 'Development of Core Technology for Automatic Satellite Control System'. The work by the GMV Aerospace and Defence was carried out under the contract with the Spain Center for the Development of Industrial Technology (CDTI).

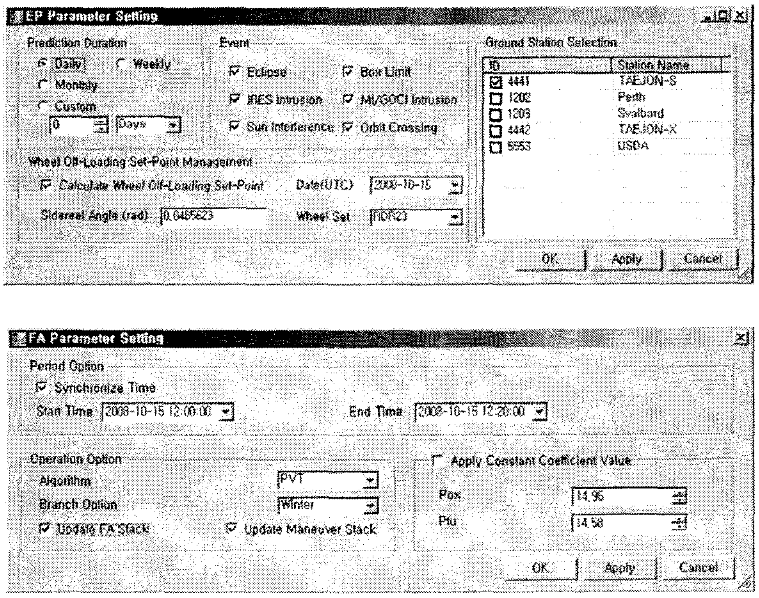

Figure 12. GUI design of EP and FA parameter setting.

References

Chapman, K. B., Cox, C. M., Thomas, C. W., Cuevas, O. O., & Beckman, R. M. 1994, Production and Quality Assurance Automation in the Goddard Space Flight Center Flight Dynamics Facility, Proceedings of the Third International Symposium on Space Mission Operations and Ground Data Systems, pp.509-516

Kruchten, P. B. 1995, The 4+1 View Model of Architecture, IEEE Software, pp.42-50

Lee, B.-S., Hwang, Y., & Kim, H.-Y. 2008a, Automation of the Flight Dynamics Operations for Low Earth Orbit Satellite Mission Control, Proceedings of International Conference on Control, Automation and Systems 2008, pp.2222-2225

Lee, B.-S., Kim, H.-Y., Hwang, Y., & Kim, J. 2008b, Flight Dynamics Operations for COMS Satellite Mission Control, Proceedings of the 26th International Communications Satellite Systems Conference (ICSSC), AIAA 2008-5500

Mardirossian, H., Heuerman, K., Beri, A., Samii, M., & Doll, C. E. 1989, Automation of Orbit Determination Functions for NASA Supported Satellite Missions, Proceedings of the Flight Mechanics/Estimation Theory Symposium, pp.349-366

Pooley, R. & Stevens, P. 1999, Using UML - Software Engineering with Objects and Components (Harlow: Addison-Wesley)

Truszkowski, W. F., Hinchey, M. G., Rash, J. L., & Rouff, C. A. 2006, IEEE Trans. Systems, Man, and Cybernetics-Part C: Applications and Reviews, 36, 279