텔레메트리 처리 유닛의 고장 방지 기법 모델링

이민규*, 권오현**
*플라스틱 소프트웨어 ㈜
** 동명정보대학교 컴퓨터공학과

Modeling of a Fault-Tolerance Approach For Telemetry Processing Unit

Min-kyu Lee*, Oh-Hyun Kwon**
* Plastic Software Inc.
**Department of Computer Engineering Tongmyung Univ. of IT

요 약

본 연구에서는 텔레메트리 처리장치의 고장 발생 원인과 복구 대책별로 장단점을 분석한 후 신뢰성 증대를 위하여 메시지 접근 방식을 제안하였으며 이를 객체지향적 방법론에 따라 UML 툴을 활용하여 모델링을 하였다.

1. 서 론

SOS(Satellite Operation System)는 크게 위성으로부터 전송되어 오는 텔레메트리(telemetry)의 값을 전송 받아 처리하는 텔레메트리 처리 유닛(telemetry processing unit)과 위성으로 원격명령(telecommand)을 전송하는 역할을 수행하는 원격명령 전송 유닛(telecommand transmission unit)으로 구성되어 있다.

KOMPSAT-2 위성은 저궤도 위성으로써 지상 관제소와의 contact time이 매우 짧으므로 그 시간 동안 이루어지는 telemetry값을 빠짐없이 전달 받아 처리하여야 한다.

Contact time동안 위성이 보내오는 telemetry는 255byte 크기의 프레임(frame)이며 매초 하나의 프레임이 전송되어 오며, SOS는 매초 전달되는 telemetry를 실시간으로 처리하여 수요자들에게 display하는 등의 기능을 수행한다. (그림1) 이 때 SOS의 telemetry 처리 기능에 장애가 발생하여 중단되는 경우 실시간으로 전송되어 오는 telemetry frame들을 모두 놓치게 되고 정상적으로 위성을 관

찰할 수 없게 된다.

따라서, telemetry processing unit의 장애에 대한 적절한 대책을 마련하고 있어야 한다.

2. 요구사항

SOS는 telemetry를 놓치지 않고 처리하기 위해서 실시간 처리 요구사항(real-time constraints)을 만족해야 하고, 또한 시스템에 문제가 발생하더라도 지속적으로 처리되어야 하며, telemetry를 요청하는 객체들의 개수에 영향을 받지 않아야 한다.

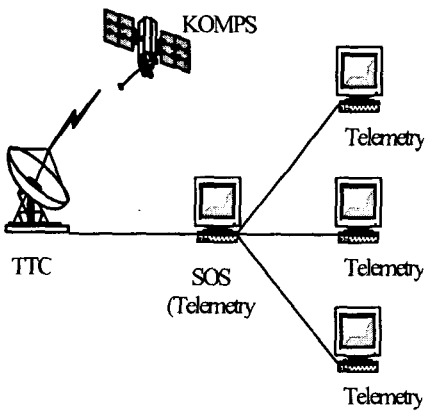
다음과 같은 Quality Attributes 들로 정리할 수 있다.

Fault-Tolerance : 위성이 Pass-Time에 있을 때 SOS는 실시간으로 telemetry frame을 전달받는데, 이 때 SOS 시스템이 문제를 발생시켜 중단되게 되면 telemetry frame을 놓치게 된다. 이러한 문제에 대처하기 위하여 SOS의 실시간 처리 부분은 fault-tolerance를 만족해야 한다.

Performance : 1초당 255byte의 크기를 갖는 telemetry frame이 SOS로 전송된다. 도착한 telemetry frame이 디스크와 같은 매체에 저장하고 telemetry의 값을 보기 원하는 사람들의 컴퓨터로 값이 전달되고 디스플레이 되어야 한다.

Scalability : TTC로부터 전달된 telemetry 값들은 여러가지 형태로 가공되어 보여지며 또한 여러 사람들이 보기를 원한다. 특히 LEOP기간에는 수십대의 컴퓨터가 연결되기 때문에 이러한 상황 때문에 성능에 대한 제약 조건을 해쳐서는 안된다.

KOMPSAT-2의 SOS는 객체지향 방법론을 이용하여 개발되며 시스템의 기본 미들웨어(middleware)로 CORBA[1]를 선택하였다. 텔레메트리 처리 유닛 역시 CORBA기반 위에서 개발될 예정이므로 장애에 대응하기 위한 접근법은 CORBA를 기반으로 할 것이다.



[그림 1] 텔레메트리 구성도

CORBA(Fault-Tolerant CORBA) 서비스[2][3] 혹은 그에 준하는 CORBA OGS(CORBA Object Group Service)[4][5] 등과 같은 일반적으로 설계된 CORBA 서비스를 이용하는 방법이고, 두번째는 그러한 서비스를 사용하지 않고 요구사항에 맞도록 직접 설계를 하는 접근법이다.

3. 아키텍쳐 제안

Telemetry frame은 TTC의 MODEM./BB라는 장비에 TCP 연결을 통해 전송받게 되고, 전송받은 telemetry frame은 필요로 하는 객체로 배포된다. 이 장에서는 이러한 특성을 고려하고 2장에서 제시한 3가지 속성(fault-tolerant, performance, scala-bility)을 만족하기 위한 아키텍쳐를 제안하고 그 내용에 대해

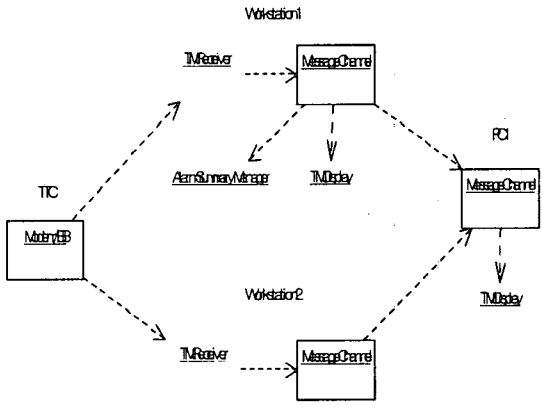
서 상세하게 기술한다.

3.1 메시지중심의 접근방법(Message-Oriented Approach)

SOS 시스템에서 telemetry processing unit은 두 개 이상의 시스템에 중복되어 있어야 한다. 설명을 간단하게 하기 위해 워크스테이션 두 개로 구성되어 있는 것으로 가정하면, 각 워크스테이션에 telemetry processing unit이 배치되어 있다. Telemetry processing unit은 TTC의 MODEM/BB 장비에 TCP 연결을 만들어 위성의 pass-time때 실시간 telemetry frame을 전송 받아 오는 TMReceiver 객체와 telemetry frame을 임시로 큐(queue)에 넣어두었다가 그것을 필요로 하는 객체에 배포해 주는 Message Channel 객체로 구성된다. 그리고 telemetry의 값을 display하기 위한 클라이언트가 설치되어 있는 PC(Personal Computer)에는 Message Channel 객체와 그것으로부터 telemetry frame을 받는 TMDisplay 객체가 배치되어 있다. [그림2]는 시스템의 아키텍쳐를 보여주고 있다.

TMReceiver는 위성의 pass-time이 되기 전에 TTC의 MODEM/BB와 TCP 연결을 건다. 위성이 TTC와 교신을 하기 시작하면서부터 telemetry frame이 MODEM/BB를 통해 TMReceiver로 전송되어 온다. TMReceiver는 전송받은 telemetry frame을 Message Channel로 전달하게 되고 Message Channel은 자신과 연결되어 있는 객체로 telemetry frame을 배포 한다. CORBA가 기반 미들웨어로 사용될 것이므로 지역 객체이든 원격 객체이든 전송을 받을 수 있게 된다.

PC1에 위치해 있는 Message Channel은 Workstation1과 Workstation2의 Message Channel들로부터 모든 telemetry frame을 두



[그림 2] Message-Oriented클래스 구조

번씩 전송받게 되는데 Message Channel 객체는 메시지의 시퀀스 번호(sequence number)를 통해 중복된 메시지를 한번만 처리하도록 구현하여야 한다.

3.2 클래스 구조

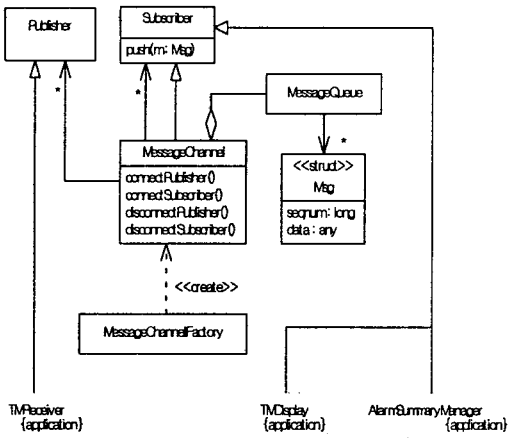
Message-Oriented Approach는 기본적으로 Publish/Subscribe 패턴을 차용하고 있다. 그리고 Publisher와 Subscriber사이에는 Message Channel이 존재한다. Message Channel은 Publisher가 생산한 메시지를 Subscriber가 받아가서 처리할 때까지 기다리지 않도록 하여 메시지 생산만을 할 수 있도록 하고, 그 이후는 Message Channel이 Subscriber에게 배포하도록 한다.

[그림3]에서 처럼 Message Channel은 여러 개의 Publisher와 Subscriber들과 연결될 수 있다. 그리고 자신 또한 Subscriber의 일종으로 다른 Message Channel의 Subsriber로써 연결될 수 있다.

이러한 구조로 MessageChannel은 다양한 형태로 구성될 수 있다.

Message Channel 내부에는 메시지를 임시로 보관할 Message Queue가 있으며 주고 받게 되는 Msg객체는 unique한 시퀀스 번호(seqnum)와 데이터(data)를 가진다.

TMReceiver는 telemetry frame을 TTC로부터 전송 받아 Message Channel로 전송하므로 Publisher로부터 서브타이핑을 하게 되고 TMDisplay나 Alarm Summary Manager 같이 telemetry frame 정보를 필요로 하는 객체는 Subscriber로부터 서브타이핑을 한다.

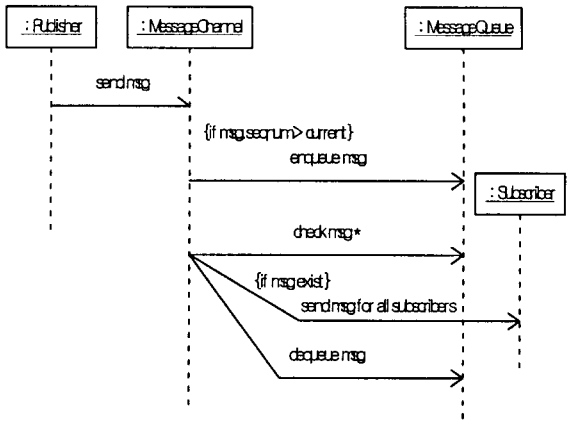


[그림3] 메시지 접근방식의 클래스 구조

3.3 객체들의 상호작용

[그림4]는 객체들이 어떻게 상호작용(interaction)하는가를 시퀀스 다이어그램(sequence diagram)으로 표현한 것이다. 우선 Publisher가 메시지를 생성하여 Message Channel로 전송하면 현재의 메시지 시퀀스 번호(current)와 메시지의 시퀀스 번호(msg.seqnum)를 비교하여 메시지의 시퀀스 번호가 같거나 작으면 메시지를 무시하고 크면 Message Queue에 넣는다.

Message Channel은 Message Queue에 메시지가 존재하는지를 항상 검사하고 존재한다면 모든 Subscriber에게 전송하고 해당 메시지를 Message Queue에서 제거한다.



[그림4] 객체들의 상호작용

4. 접근방법의 평가

제안된 메시지 중심의 접근방법이 어떻게 요구 사항들을 만족하는지를 생각해보자.

Fault-Tolerance : Message Channel은 여러 시스템에 중복되어 분산되어 있을 수 있으므로 하나가 고장 나더라도 다른 객체가 처리를 할 수 있다. 중복으로 인해 동일한 메시지가 중복적으로 수신될 수 있는데 이것 역시 Message Channel이 시퀀스 번호를 이용하여 해결하고 있다.

Performance : TMReceiver는 TTC로부터 전송 받아 온 telemetry frame을 Message Channel에 전달만 해주기 때문에 다음 frame을 처리하기 위한 상태로 즉시 돌아갈 수 있다. 그리고 중복된 시스템 중 하나가 고장이 나는 경우에 별도의 failure detection과 failure recovery의 과정이 필요 없으므로 telemetry frame을 놓치는 경우도 없다.

Scalability : Message Channel에는 복수개의 Subscriber가 연결될 수 있고 Message Channel 역시 Subscriber가 될 수 있으므로 여러 시스템에 각각 Message Channel을 두고

각 MessageChannel이 지역 객체들을 책임지는 형태로 구성할 수도 있다. 이렇게 유연한 구조로써 얼마든지 시스템의 스케일을 확장할 수 있다.

5. 결론

SOS시스템은 telemetry의 실시간 처리의 요구사항과 이중화를 통한 시스템의 reliability를 높이기 위한 요구 사항들이 있다. 현재 시스템을 이중화하기 위한 방법으로 FT-CORBA라는 접근법이 있으나, 실시간 처리라는 요구사항에 적절히 대처하기에는 부족한 점에 있는 구조라고 판단된다. 이러한 문제점을 극복하고 요구사항을 만족시키기 위하여 메시지 중심의 접근방법(Message-Oriented Approach)를 제안하였다. 이 접근방법은 메시지 큐(message queue)를 이용하고 Publisher/Subscriber 패턴을 적용하여 performance, fault-tolerant 그리고 scalability를 만족하도록 시스템을 구성할 수 있도록 도와준다.

[참고문헌]

[1] Object Management Group, "The Common Object Request Broker: Architecture and Specification," 2.3 edition, June 1999.

[2] Object Management Group, "Fault Tolerant CORBA Specification," OMG Document 99-12-08 edition, December 1999.

[3] B. Natarajan, A. Gokhale, S. Yajnik and D. C. Schmidt, "Applying Patterns to Improve the Performance of Fault Tolerant CORBA," 7th International Conference on High Performance Computing, ACM/IEEE, Bangalore, India, Dec. 17-20, 2000.

[4] P. Felber, "The CORBA Object Group Service: A Service Approach to Object Groups in CORBA," PhD thesis, École Polytechnique Fédérale de Lausanne, Switzerland, 1998.

[5] P. Felber, B. Garbinato, and R. Guerraoui., "The design of a CORBA group communication service," In Proceedings of the 15th Symposium on Reliable Distributed Systems (SRDS-15), pages 150-159, Niagara-on-the-Lake, Canada, October 1996.