

IEC61850 기반의 변전소 자동화 시스템을 위한 객체지향 데이터 모델 연구

김정수¹, 김상식¹, 장혁수¹, 정태선¹, 장병태², 이재욱², 조성훈³
 명지대학교 컴퓨터 소프트웨어학과¹, 전력연구원², 한국전력공사³

Object Oriented Data Model for IEC 61850 based Substation Automation System

Jeong Soo Kim¹, Sang Sig Kim¹, Hyuk Soo Jang¹, Tae Sun Chung¹, Byung Tae Jang², Jae Wook Lee², Seong Hoon Joe³
 Dept. of Computer Software, Myong-Ji University¹, KEPRI², KEPCO³

Abstract - 데이터 모델링의 객체 지향적인 접근은 변전소 자동화 시스템의 설계 및 구현에 객체의 재활용과 객체 정보의 숨김과 같은 효율성을 제공한다. 변전소 자동화 시스템을 위한 IEC 61850 표준은 이러한 객체 지향 개념에 기반한 데이터 모델링을 IEC 61850 표준문서의 파트 7에서 제시하였다. 표준은 데이터를 교환하는 가장 작은 기능으로써 Logical Node(LN)를 모델링 하였고 본 논문에서는 LN을 비롯한 여러 객체 지향 데이터 모델들을 설명하고 분석하였다.

클래스들은 계층적인 구조를 가지는데, 계층적 클래스 구조에서 하위 클래스는 상위 클래스의 모든 속성을 그대로 유지하면서 하위 클래스만의 속성을 추가하거나, 기존 속성을 재 정의하여 정의된다. 사람이라는 클래스를 상속받아 학생 클래스나 교수 클래스가 정의될 수 있다. 학생과 교수는 모두 사람인 동시에 학생은 학생만의 상태와 행위를 가지고 있고, 교수는 교수만의 상태와 행위를 가지고 있기 때문에 가능한 것이다.

IEC 61850 표준의 데이터 모델은 이러한 클래스 정의의 집합이다. 그림 1은 Logical Node와 관련된 여러 클래스들의 상속관계를 나타내며, 각각은 다음 장에서 구체적으로 설명한다.

1. 소 개

최근 변전소 자동화 시스템에 관한 높은 관심이 집중되면서, IEC Technical Committee 57의 Working Group 10에서는 변전소 자동화 시스템에 관한 표준인 IEC 61850 표준[1]을 제정하였다. 현재 파트 10을 제외한 IEC 61850 표준 문서의 모든 파트는 국제표준(IS) 상태이다. IEC 61850 표준은 변전소 장비들 사이의 통신 규격뿐만 아니라 변전소 자동화 시스템과 관련된 요구사항들을 정의하였다.

IEC 61850 표준은 서로 다른 제조업체의 제품들 사이의 상호 운용성을 용이하게 하기위한 데이터 모델을 표준화 하였다. 데이터 모델의 객체 지향적 개념을 적용하였다. 이러한 객체 지향적인 개념은 프로그래밍을 비롯한 여러 분야에서 적용되었으며, IT 기반의 전력 시스템에서도 적용되고 있다. 예로써, IEC 61970 표준에서 정의된 CIM(Common Information Model)[2]에도 객체 지향 개념이 적용되었다. 본 논문은 IEC 61850 표준의 데이터 모델링에 적용된 객체 지향적 개념을 분석하고, 이러한 객체 지향적 접근으로 모델링된 데이터 모델에 대해 설명한다.

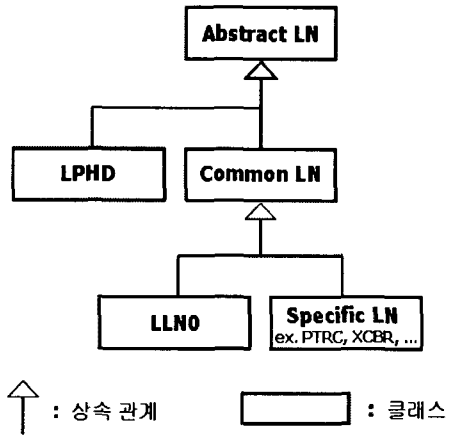


그림 1 클래스의 계층구조

2. 객체 지향 개념

객체 지향 개념은 크게 객체를 제공하는 클래스와 계층적인 클래스의 구조에서 발생하는 상속의 개념으로 정의할 수 있다. 객체는 실제계를 반영하는 것으로, 식별성(Identity), 상태(State), 행위(Behavior)를 가진다. 식별성은 단순히 이름과 같이 객체를 식별할 수 있는 것을 의미한다. 상태는 객체 상태를 나타내는 변수들을 의미하고, 행위는 객체의 행동을 나타내는, 객체 간 주고받는 메시지를 의미한다. 이러한 메시지의 구현 부분을 메소드(method)라고 정의하였다.

객체들은 클래스라는 틀로써 정의된다. 즉, 객체는 클래스를 인스턴스화 한 것이다. 사람을 클래스로써 정의하였다면, 실제 각각의 사람은 클래스를 인스턴스화 한 객체로써 표현된다. 생성된 각각의 객체는 자신만의 식별성, 즉 이름을 가지게 된다.

3. 객체 지향 데이터 모델

3.1 Logical Node

변전소 자동화 시스템에 필요한 기능에 근거하여, 표준에서는 정보를 교환하는 가장 기본적인 객체로써 Logical Node를 정의하고, 이를 객체 지향 개념의 클래스로써 설명하고 있다. 표준은 90개가 넘는 구체적인 LN 클래스를 정의하고 있다. 마이크로프로세서 기반의 IED를 구현하기 위해 필요한 LN 클래스들은 최상위 클래스로써 정의된 추상적인 LN 클래스를 상속받아 정의된다. 그림 1에서와 같이 LPHD 클래스와 Common LN 클래스는 추상적인 LN 클래스를 상속받는다. LPHD 클래스는 물리적인 장비에 대한 정보를 가지고 있는 클래스이고, Common LN 클래스는 변전소 자동화 시스템에 필요한 기능에 근거하여 모델링된 구체적인 LN 클래스를 정의하기 위한 상위 클래스이다. 즉, Common LN 클

표 1 추상적인 LN 클래스 정의

| Logical-Node class | |
|--------------------------------------|-----------------|
| Attribute name | Attribute type |
| LNName | ObjectName |
| LNRef | ObjectReference |
| Data[1 to n] | DATA |
| DataSet[0 to n] | DATA-SET |
| BufferedReportControlBlock[0 to n] | BRCB |
| UnBufferedReportControlBlock[0 to n] | URCB |
| LogControlBlock[0 to n] | LCB |
| Service | |
| GetLogicalNodeDirectory | |
| GetAllDataValues | |

래스를 상속받아 보호, 제어, 정보 수집 등의 기능을 위한 구체적인 LN 클래스들이 정의된다. 예를 들어, PTRC 클래스는 보호 기능과 관련된 LN 클래스로써 트립 신호를 발생시키는 기능으로 정의되어 있다. Common LN 클래스를 상속받아 정의되는 또 하나의 클래스는 LLNO 클래스이다.

표 2 Common LN 클래스 정의

| Common LN class | | | |
|---------------------------------|----------------|-------------------------------|-----|
| Attribute name | Attribute type | Explanation | M/O |
| LNName | - | 추상적인 LN 클래스로부터 상속받는 속성 | |
| LNRef | - | 상속받는 속성 | |
| Data | | | |
| LPHD를 제외한 모든 LN에서 필수적으로 상속받는 정보 | | | |
| Mod | INC | Mode | M |
| Beh | INS | Behavior | M |
| Health | INS | Health | M |
| NamePlt | LPL | Name plate | M |
| 선택적인 추가 정보 | | | |
| Loc | SPS | Local operation | O |
| EEHealth | INS | External equipment health | O |
| EEName | DPL | External equipment name plate | O |
| OpCntRs | INC | Operation counter reset | O |
| OpCnt | INS | Operation counter | O |
| OpTmh | INS | Operation time | O |
| Data Set | | | |
| 추상적인 LN 클래스로부터 상속받아 구체화 시킨다. | | | |
| Control Block | | | |
| 추상적인 LN 클래스로부터 상속받아 구체화 시킨다. | | | |
| Services | | | |
| 추상적인 LN 클래스로부터 상속받아 구체화 시킨다. | | | |

최 상위 클래스인 추상적인 LN 클래스는 표 1과 같이 정의된다. 객체 지향 개념의 상태에 대응되는 속성과 행위에 대응되는 서비스로 정의된다. LNName 속성과 LNRef 속성은 이름과 자기 자신의 식별을 위한 참조를 나타낸다. 나머지 속성은 추상적으로써 정의되어 하위 클래스에서 구체화(specialized)된다. 서비스는 추상적인 LN 클래스 레벨에서 제공되는 정보 교환 서비스이다.

Common LN 클래스는 표 2와 같이 정의된다. 이 클래스는 PTRC 클래스와 같은 기능 관련된 구체적인 클래스들의 공통적인 속성들을 정의하였다. 표 2는 추상적인 LN 클래스로부터 상속받는 속성과 구체화되는 Data, DataSet, 제어 블록, 그리고 서비스를 보여준다. Common LN 클래스의 Data 속성들은 구현에 따라 꼭 필요한 속성과 추가적으로 필요한 속성을 정의하였다. M은 필수적으로 구성되어야 할 속성이며, O는 선택적인

로 구성될 수 있는 속성으로 구현에 따라 결정된다.

표 3 LLNO 클래스 정의

| LLNO class | | | |
|---|----------------|----------------------------|-----|
| Attribute name | Attribute type | Explanation | M/O |
| LNName | - | 추상적인 LN 클래스로부터 상속받는 속성 | |
| LNRef | - | 상속받는 속성 | |
| Data | | | |
| Common LN 클래스의 속성 | | | |
| | | Common LN 클래스의 필수적인 속성을 상속 | M |
| Loc | SPS | Local operation | O |
| OpTmh | INS | Operation time | O |
| Controls | | | |
| Diag | SPC | Run Diagnostics | O |
| LEDs | SPC | LED reset | O |
| 서비스가 필요할 경우, 해당 제어 블록을 정의함 | | | |
| Attribute name | | Attribute type | |
| SettingGroupControlBlock[0 to 1] | | SGCB | |
| Log[0 to 1] | | LOG | |
| GOOSEControlBlock[0 to n] | | GoCB | |
| GSSEControlBlock[0 to n] | | GsCB | |
| MulticastSampledValueControlBlock[0 to n] | | MSVCB | |
| UnicastSampledValueControlBlock[0 to n] | | USVCB | |

LLNO 클래스는 Common LN 클래스를 상속하여 정의하고, 특별한 서비스를 제공하기 위해서 필요한 제어 블록도 아울러 정의한다. 셋팅을 위해 데이터를 그룹화하여 정의된 SettingGroupControlBlock, 로그의 저장과 관리를 책임지는 Log, 트립과 같은 일반적인 변전소의 이벤트 제어를 위한 GOOSEControlBlock과 GSSEControlBlock, 프로세스 레벨의 샘플 값의 송수신을 제어를 위한 MulticastSampledValueControlBlock과 UnicastSampledValueControlBlock이 정의된다. 이러한 제어 블록은 필요에 의해 정의된 것으로 표 3은 LLNO 클래스의 구성을 보여준다.

표 4 LN 그룹의 목록

| Group Indicator | LN groups | # of LNs |
|-----------------|---|----------|
| A | Automatic Control | 4 |
| C | Supervisory Control | 5 |
| G | Generic Function References | 3 |
| I | Interfacing and Archiving | 4 |
| M | Metering and Measurement | 8 |
| P | Protection Functions | 28 |
| R | Protection Related Functions | 12 |
| S | Sensors, Monitoring | 4 |
| T | Instrument Transformer | 2 |
| X | Switchgear | 2 |
| Y | Power Transformer and Related Functions | 4 |
| Z | Further (Power System) Equipment | 15 |

표준은 변전소 자동화 시스템을 위해 필요한 변전소의 모든 기능에 대응되는 데이터 모델을 정의하였다. 이러한 데이터 모델은 기능과 관련된 91개의 LN 클래스이다. 기능성의 분류에 의해 몇몇 LN를 그룹화 할 수 있다. 이러한 그룹화된 LN의 목록은 표 4에서 묘사된다. 기능관련 LN 클래스들은 Common LN 클래스를 상속하여, 자신의 기능에 근거하여 데이터를 구체화 하여 정의한다. 표 5는 Common LN 클래스를 상속하여 정의된 PTRC 클래스의 Data 속성을 보여준다.

IED는 물리적인 장비의 정보를 포함하는 LPHD 클래스, LLNO, 그리고 기능과 관련된 구체적인 LN들로 구현된다. 표준은 IED 구현을 위한 LN들의 계층적인 구조를

표 5 PTRC 클래스 정의

| PTRC class | | | |
|---------------------------|----------------|----------------------------|-----|
| Attribute name | Attribute type | Explanation | M/O |
| LNName | - | 추상적인 LN 클래스로부터 | |
| LNRef | - | 상속받는 속성 | |
| Data | | | |
| Common LN 클래스의 속성 | | | |
| | | Common LN 클래스의 필수적인 속성을 상속 | M |
| OpCntRs | INC | Operation counter reset | O |
| Status Information | | | |
| Tr | ACT | Trip | O |
| Op | ACT | Operate | O |
| Str | ACD | Directional protection | |
| Settings | | | |
| TrMod | INS | Trip Mode | O |
| TrPlsTmms | INS | Trip Pulse Time | O |

정의하였으며, 그림 2는 이를 보여준다. 그림 2에 따르면, 하나의 Logical Device는 여러 개의 기능관련 LN과 하나의 LLNO와 하나의 LPHD로 이뤄진다. 즉, Logical Device(LD)는 최소 3개의 LN이 필요하다. IED는 결국 LN들로 구성된 하나 이상의 LD로써 구현된다. LN 클래스들은 변전소 자동화 시스템에 필요한 모든 기능으로 모델링 되었기 때문에, 적당한 LN들의 집합으로 구현되는 IED는 기존 장비들의 통합(integration)을 가능하게 하고, LN들의 자유로운 할당 및 합성을 통해 구성될 수 있는 장점을 가진다. 그림 3은 LN의 자유로운 할당과 배치를 통한 IED 구성의 융통성을 보여준다.

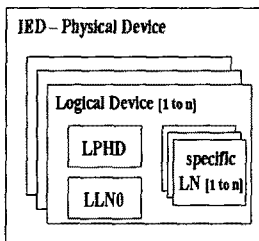


그림 2 IED의 구성도

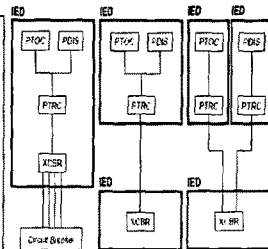


그림 3 IED의 동적인 구성

3.2 Common Data Class

표 6 DATA 클래스 정의

| DATA class | |
|----------------------|-------------------|
| Attribute name | Attribute type |
| DataName | ObjectName |
| DataRef | ObjectReference |
| Presence | BOOLEAN |
| DataAttribute[0..n] | |
| DataAttributeType | DAType |
| FunctionalConstraint | FC |
| TrgOp[0..n] | TriggerConditions |
| Service | |
| GetDataValues | |
| SetDataValues | |
| GetDataDirectory | |
| GetDataDirectory | |

LN 클래스는 여러 데이터들로 정의된다. LN 클래스를 구성하고 있는 데이터 역시 클래스로 정의되었다. 추상적인 LN 클래스와 비슷하게, DATA 클래스는 구체적인 Data 클래스를 정의하기 위한 공통적인 속성들로 정의된다. Common Data Class(CDC)는 Data 클래스를 상속받아 구체적인 Data를 정의하기 위한 클래스이다. 표

준은 29개의 CDC를 정의하고 있다. DATA 클래스는 표 6과 같이 정의된다.

표 7 ACT common data 클래스 정의

| ACT class | | | |
|----------------------|----------------|-----|-------|
| Attribute name | Attribute type | FC | TrgOp |
| DataAttribute | | | |
| general | BOOLEAN | ST | dchg |
| phsA | BOOLEAN | ST | dchg |
| phsB | BOOLEAN | ST | dchg |
| phsC | BOOLEAN | ST | dchg |
| ... | ... | ... | ... |
| Service | | | |
| DATA클래스와 관련된 서비스 | | | |

표 7은 표 5의 PTRC에 정의된 tr, op 데이터 속성에 대한 타입인 ACT(activation information) 클래스의 정의를 일부 보여준다. Data 클래스를 상속받아, DataAttribute 속성을 구체화하여 정의된다. Functional Constraint(FC)는 DataAttribute를 구체적인 사용에 근거하여 그룹화된 제약사항이다. TrgOp는 레포트나 로그 서비스에 관련된 트리거 옵션이며, 언제 해당 DataAttribute를 레포트/로그 서비스에 통해 보낼 것인가를 결정한다. 표준에서 정의한 FC와 TrgOp의 일부는 표 8과 같다.

표 8 Functional Constraint 와 Trigger Option의 의미

| FC | Semantic | TrgOp | Semantic |
|----|---------------|-------|-------------------|
| CO | Control | dchg | data-change |
| CF | Configuration | qchg | quality-change |
| ST | Status Info. | dupd | data value update |
| MX | measurands | | |

4. 결 론

전력 자동화 시스템과 관련된 여러 응용에 사용되고 있는 객체 지향적 개념은 변전소 자동화 시스템을 위한 IEC 61850 표준의 데이터 모델에 사용되었다. 데이터 모델을 표준은 제조업체가 다른 IED들 사이의 상호운용성을 용이하게 한다. 이러한 상호운용성은 IEC 61850 표준이 추구하는 목적이기도 하다. 본 논문은 이러한 데이터 모델에 적용된 객체 지향 개념에 대해 설명하였다.

[참고 문헌]

- [1] IEC, "INTERNATIONAL STANDARD IEC 61850"
- [2] A. deVos, S.E. Widgren, J. Zhu: "XML FOR COM MODEL EXCHANGE", 22nd IEEE Power Engineering Society International conference on 20-24 May 2001, 31-37p