Benefits of Implementing Control Authority in Substation Automation Systems

Romulo Corna and Wellington Oliveira
Schweitzer Engineering Laboratories, Inc.

Presented at the
Protection, Automation, and Control World Americas Conference
Raleigh, North Carolina
August 11–14, 2025

Benefits of Implementing Control Authority in Substation Automation Systems

Romulo Corna and Wellington Oliveira, *Schweitzer Engineering Laboratories, Inc.*

***Abstract*—The IEC 61850 standard plays a vital role in modern power systems by standardizing communications and control across different operational levels. This paper explores control authority (CA), which manages hierarchical access to control commands within an SAS. Through practical implementations, it demonstrates how CA enforced via data objects (DOs), like Loc, LocSta, MltLev and LocKey, enhances security, reliability, and structured access. The standard's modeling approach supports interoperability, while service tracking ensures visibility into command origin and validity, contributing to safer and more efficient substation operations.**

I. Introduction

The substation automation system (SAS) is critical for ensuring the reliable, efficient, and secure operation of modern electrical substations. These systems integrate control, monitoring, and protection functionalities to manage complex power system operations. The Manufacturing Message Specification (MMS) protocol is used by remote operators to open and close breakers by sending control commands from various locations to an intelligent electronic device (IED), which interfaces with the equipment in the substation yard. An IED can receive commands from an operator located either inside the substation control room or in a different city, communicating via telecommunications. Given the possibility of multiple clients issuing commands, it is essential to differentiate the origin of each command. The IEC 61850 standard provides mechanisms such as control authority (CA), local or remote functionality, originator categories (orCat), and service tracking to manage these different clients through a hierarchical structure.

This paper explores IEC 61850 solutions for implementing CA within an SAS. It introduces the concept of a hierarchical command structure, detailing how the origin of control commands is managed across the levels. The paper thoroughly explains the components involved in CA, including logical nodes (LNs), data objects (DOs), and their interactions. In addition to the theoretical framework, it presents practical test scenarios that include configuration steps, command execution, and result analysis. Finally, the paper discusses the benefits of adopting CA in SAS projects, such as improved command traceability, enhanced system integrity, and better coordination between local and remote operators.

II. Operational Level

An SAS performs essential control, monitoring, and protection functions within a substation. To organize these functions, the IEC 61850 standard defines four logical levels: process, bay, station, and remote [1].

At the process level, sensors and actuators, like breakers and transformers, interact directly with power systems. Commands at this level are executed manually by field personnel and do not use the MMS protocol. The bay level involves local operators and automated controllers interacting with IEDs to perform protection and control functions. These IEDs interface directly with the sensors and actuators at the process level through digital outputs [1].

The station level includes operators using a human-machine interface (HMI) and automated systems that manage data and control across multiple bays or the entire substation. The remote level consists of control centers that oversee multiple substations via telecontrol interface (TCI) [1].

This hierarchical structure enables efficient, scalable, and standardized substation automation.

III. Control Authority

System operators can enforce access restrictions and enhance security by executing MMS control commands through various DOs defined in the IEC 61850 standard, which are distributed across multiple LNs and different hierarchical levels [2].

The CA concept within an IED determines which level is permitted to execute a command. For example, if the CA is configured to authorize only the bay level, a command issued from the remote level will be denied, while one from the bay level will be accepted. This ensures a structured hierarchy for substation control commands [2]. Fig. 1 shows all the levels at which a command can be executed.

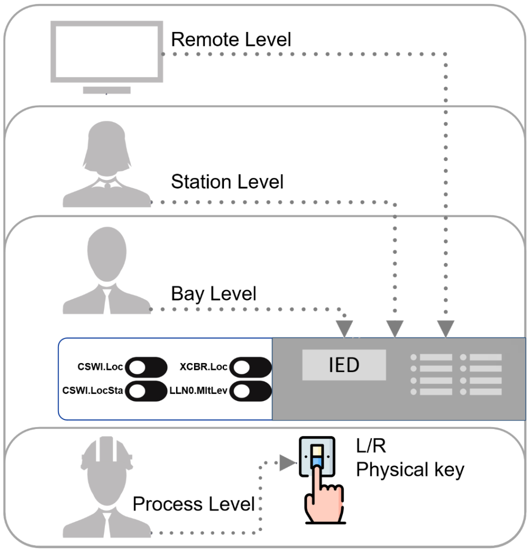


Fig. 1. CA levels [2].

IV. Local or Remote Functionality

Local or remote functionality is crucial for ensuring proper control and coordination within an SAS. It helps in defining and managing CA across different levels. The commands are based on their origin and are determined by the Loc, LocSta, and MltLev. This ensures a structured and hierarchical approach to control within the system [3]:

- Loc: This DO is located within LLN0 and in LNs such as CSWI, XCBR, and XSWI. When CSWI.Loc is asserted, it indicates that CA has been granted to the bay level for the associated switchgear, meaning that only commands issued from this level will be executed. Conversely, when XCBR.Loc or XSWI.Loc is asserted, it signifies that remote commands are blocked and only process-level commands, typically issued manually by field personnel, will be accepted.
- LocKey: This DO is located within LLN0 and in LNs such as CSWI, XCBR, and XSWI. CSWI.LocKey can be wired to a binary input that receives signals from a physical key or toggle switch located at the bay level. When CSWI.LocKey is asserted, it sets CSWI.Loc, meaning only commands issued from the bay level will be executed. Similarly, XCBR.LocKey or XSWI.LocKey can be wired to a binary input that receives signals from a physical key or toggle switch located at the process level. When either XCBR.LocKey or XSWI.LocKey is asserted, it sets XCBR.Loc or XSWI.Loc, respectively. This indicates that remote commands are blocked, and only process-level commands, typically issued manually by field personnel, will be accepted.
- LocSta: This DO is located within LNs such as CSWI and LLN0, but not in LNs linked to the process level, such as XCBR and XSWI. LocSta is used to set the CA at the station level and can be configured either by IED logic or via an MMS command. When LocSta is asserted, it signifies that the CA has been assigned to the station level. Consequently, only commands originating from the station level will be accepted.
- MltLev: Modeled exclusively in the LLN0, this DO indicates whether multiple sources of control commands are accepted simultaneously at a certain level and can be configured either by IED logic or via an MMS command. When LLN0.MltLev is asserted, it allows for control commands from various levels to be accepted concurrently. This means that commands from the bay, station, and remote levels can all be accepted.

A. Single-Level Control

Single-level control allows MMS commands from only one level at a time. The behavior of the CA depends on the specific combination of DOs, as shown in Table I. This table presents the DOs that can be configured in the IED, with each row representing a distinct configuration scenario, we can observe the LN DOs configured in the IED. Each row represents a scenario in which the IED can be configured. Therefore, when the IED receives commands from different levels, it must respond as specified illustrated in Table I [3].

Fig. 2 illustrates some examples regarding the relationship in the context of single-level control [3]. In the first scenario, the IED receives an input from the field that asserts XCBR.Loc, causing it to block MMS commands from all control levels. In the second scenario, the IED sets CSWI.Loc, allowing it to accept commands only from the bay level. In the third scenario, the IED sets CSWI.LocSta, restricting command acceptance to the station level. In the fourth scenario, all DOs are deasserted, enabling the IED to accept commands only from the remote level. Manual commands from the process level are never blocked by the CA.

B. Multiple-Level Control

Multiple-level control allows MMS commands from more than one level simultaneously. The behavior of the CA depends on the specific combination of DOs, as shown in Table II. This table presents the DOs that can be configured in the IED, with each row representing a distinct configuration scenario. We can observe the LN DOs configured in the IED. Each row represents a scenario in which the IED can be configured. Therefore, when the IED receives commands from different levels, it must respond as specified, illustrated in Table II [3].

Fig. 3 illustrates four examples of the relationship in the context of multiple-level control, with LLN0.MltLev asserted in all scenarios. In the first scenario, the IED receives an input from the field that asserts XCBR.Loc, causing it to block MMS commands from all control levels. In the second scenario, the IED sets CSWI.Loc, allowing it to accept commands only from the bay level. In the third scenario, the IED sets CSWI.LocSta, restricting command acceptance to the station and bay levels. In the fourth scenario, all DOs are deasserted except LLN0.MltLev, enabling the IED to accept commands from all levels.

TABLE I
DO AND CA RELATIONSHIP [3]

LN DOs				Commands From			
XCBR.Loc	**LLN0.MltLev**	**CSWI.Loc**	**CSWI.LocSta**	**Process Level**	**Bay Level**	**Station Level**	**Remote Level**
XSWI.Loc				**(Manual)**	**(orCat 1 and 4)**	**(orCat 2 and 5)**	**(orCat 3 and 6)**
TRUE	FALSE	TRUE OR FALSE	TRUE OR FALSE	Allowed	Not Allowed	Not Allowed	Not Allowed
FALSE	FALSE	TRUE	TRUE OR FALSE	Allowed	Allowed	Not Allowed	Not Allowed
FALSE	FALSE	FALSE	TRUE	Allowed	Not Allowed	Allowed	Not Allowed
FALSE	FALSE	FALSE	FALSE	Allowed	Not Allowed	Not Allowed	Allowed

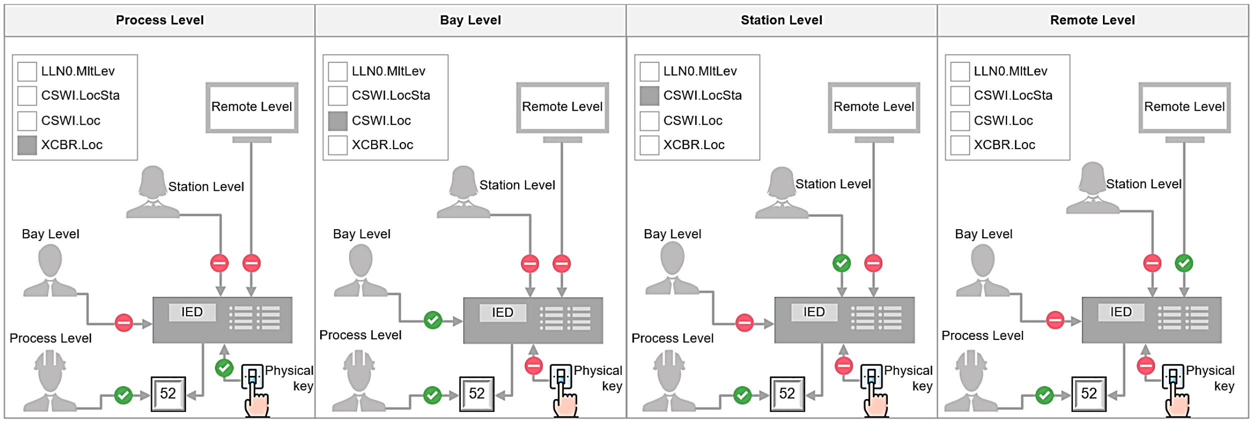


Fig. 2. Single-level control.

TABLE II
DO AND CA RELATIONSHIP [3]

LN DOs				Commands From			
XCBR.Loc	**LLN0.MltLev**	**CSWI.Loc**	**CSWI.LocSta**	**Process Level**	**Bay Level**	**Station Level**	**Remote Level**
XSWI.Loc				**Manual**	**orCat 1 and 4**	**orCat 2 and 5**	**orCat 3 and 6**
TRUE	TRUE	TRUE OR FALSE	TRUE OR FALSE	Allowed	Not Allowed	Not Allowed	Not Allowed
FALSE	TRUE	TRUE	TRUE OR FALSE	Allowed	Allowed	Not Allowed	Not Allowed
FALSE	TRUE	FALSE	TRUE	Allowed	Allowed	Allowed	Not Allowed
FALSE	TRUE	FALSE	FALSE	Allowed	Allowed	Allowed	Allowed

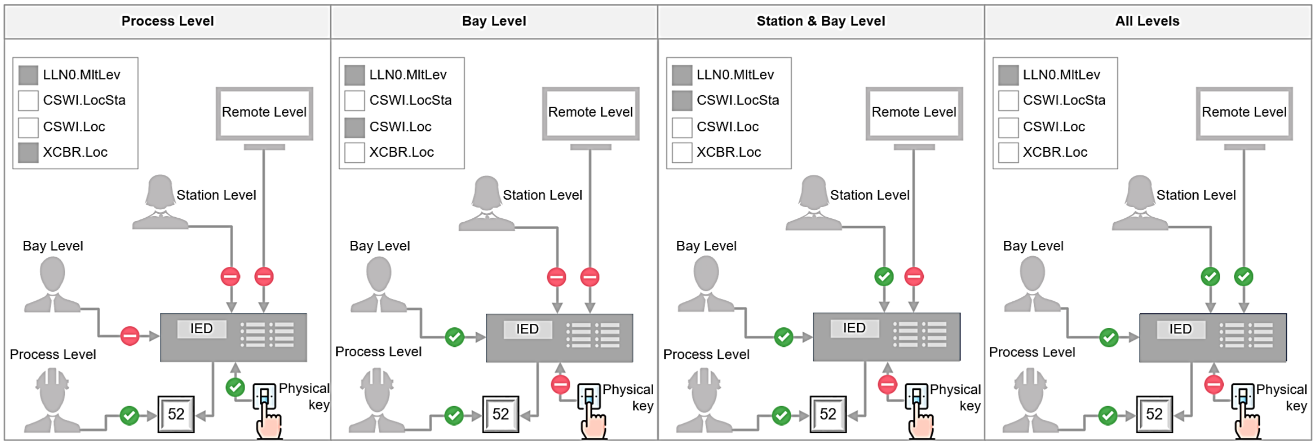


Fig. 3. Multiple-level control.

V. Relationship Between LNs and Logical Devices (LDs)

A. Relationship Between LNs Inside the Same LD

LLN0 establishes a connection with the LNs that reside within the same LD. This relationship ensures coordinated local or remote functionality among the components within that LD. When a DO is asserted at LLN0, the same DO is propagated throughout the entire LD. For example, asserting LLN0.Loc results in the Loc DO being set across the entire LD, thereby causing corresponding DOs in other LNs, such as CSWI.Loc, to also be asserted. However, XCBR.Loc and XSWI.Loc are exceptions since they indicate local or remote status of the physical switchgear and are usually asserted through a physical key. Fig. 4 illustrates this relationship. The same principle applies to other DOs, such as LocSta, LocKey, and MltLev.

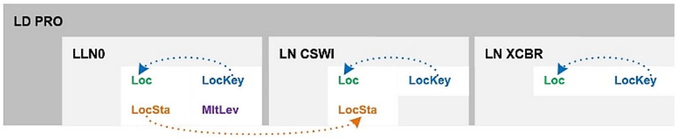


Fig. 4. Relation between LLN0 and LN.

B. Relationship Between LNs of Different LDs

LNs function independently within the same LD, following the definitions set by LLN0. This independence also applies across different LDs, each of which maintains its own unique local or remote functionality. However, an LD can be configured to follow another LD that holds a higher position in the hierarchy. This hierarchical relationship is defined by a group reference (GrRef) DO in LLN0. The GrRef DO serves as a reference to a higher-level LD. If both LDs have their GrRef DOs set to LD CFG, they will adhere to the decisions made within the LD CFG's DOs [2].

Fig. 5 illustrates LD PRO1 and LD PRO2, with the GrRef DOs defined in LLN0 set to LD CFG. This ensures that they follow LLN0's DOs regarding the local or remote functionality. When CFG.LLN0.Loc is set to True, both PRO1.LLN0.Loc and PRO2.LLN0.Loc follow to reflect the status of the higher-level LD.

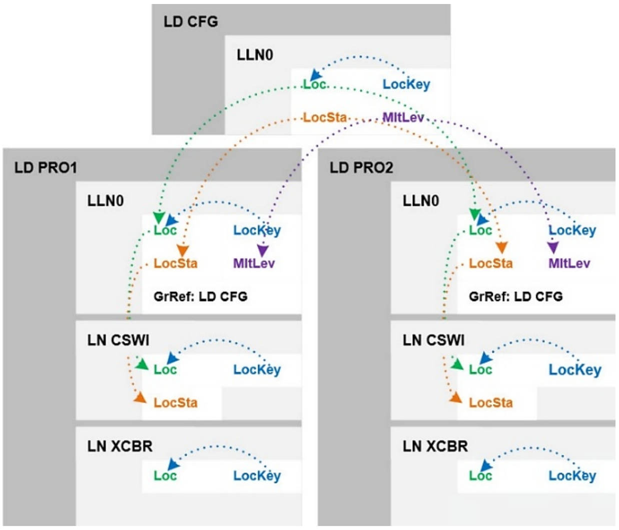


Fig. 5. Hierarchical relationship between LNs and LDs.

VI. Originator Categories

MMS clients issue control commands to DOs with controllable common data class (CDC) types, such as single point controllable (SPC) and double point controllable (DPC). These DOs have two attributes: orCat and orIdent.

Table III presents a complete list of all orCat values. These categories are essential as they identify the specific source or entity responsible for initiating the command operation. By examining the orCat values, it is possible to determine the level from which the command was issued and whether it was initiated by a human operator, an automatic tool, or software.

This orCat value identifies both the source level and the nature of a command, whether it was issued by a human operator or an automatic tool.

- At the bay level, the orCat value is set to 1 for operator commands and 4 for automated ones.
- At the station level, the orCat value is set to 2 for operator commands and 5 for automated ones.
- At the remote level, the orCat value is set to 3 for operator commands and 6 for automated ones.

Table III
orCat

Item	orCat Value	Description
Not supported	0	That value shall not be used.
Bay control	1	Operator command at bay level.
Station control	2	Operator command at station level.
Remote control	3	Operator command at remote level.
Automatic bay	4	Automatic command at bay level.
Automatic station	5	Automatic command at station level.
Automatic remote	6	Automatic command at remote level.
Maintenance	7	Command at a maintenance or service tool.
Process	8	Originator of a command is unidentified.

This classification helps IEDs determine how to handle incoming MMS commands based on their origin. There are three more orCat values:

- The orCat value 0 indicates that the IED does not support local or remote functionality.
- The orCat value 7 is specifically used for commands issued by a maintenance operator using a software application.
- The orCat value 8 is utilized when the originator of a command is unknown [4].

VII. Service Tracking

Service tracking is used to communicate information and values for analysis after a command is issued, which is especially useful for identifying and understanding negative responses. By capturing detailed data, it becomes easier to diagnose issues, determine their root causes, and implement corrective measures. Service tracking can be used by any client once the tracking DO is part of the data set associated with a log control block (LCB) or a buffered or unbuffered report control block (BRCB/URCB).

The LN LTRK is used to communicate service tracking, with each DO dedicated to a specific type of service. The LN LTRK includes several DOs focused on control and managing service tracking. DO DpcTrk, which services tracking for DPC data, was selected to illustrate how the mechanism works [3].

The DpcTrk DO contains a group of attributes used for service tracking. The attributes highlighted in this paper are service type, error code, and response AddCause. These attributes provide values that are enumerated in tables defined in IEC 61850-7-2.

A. Service Tracking—Success Command

Fig. 6 demonstrates a successful MMS command issued from the bay level, with CSWI.Loc asserted, allowing only bay-level commands. The example includes DpcTrk attributes, which log service tracking details for DPC operations.

B. Service Tracking—Blocked Command

Fig. 7 demonstrates a blocked MMS command issued from the remote level, with CSWI.Loc asserted, allowing only bay-level commands. The example includes DpcTrk attributes, which log service tracking details for DPC operations.

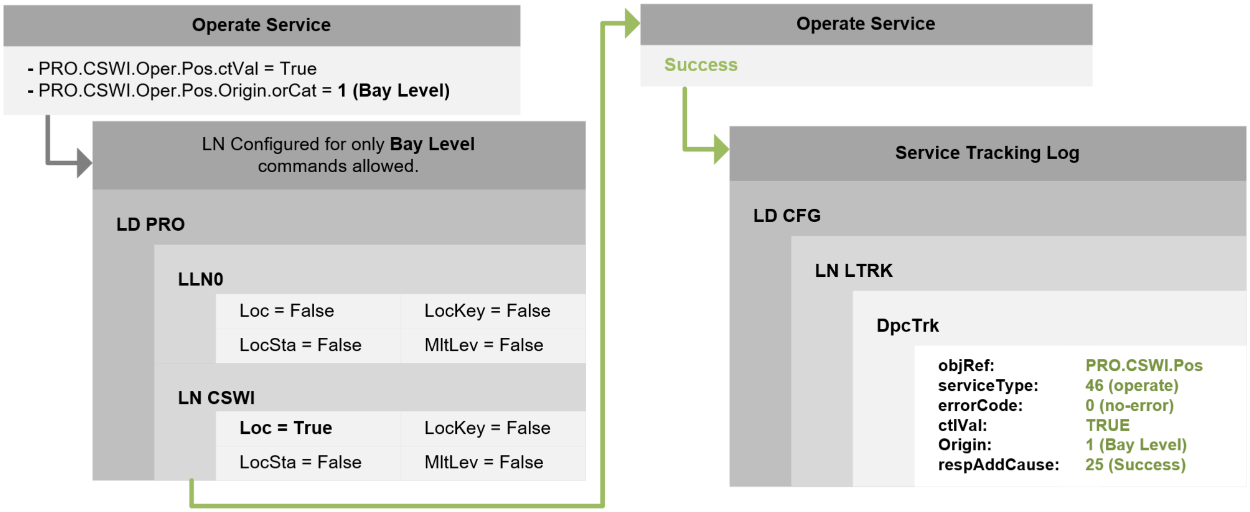


Fig. 6. Operate service—success.

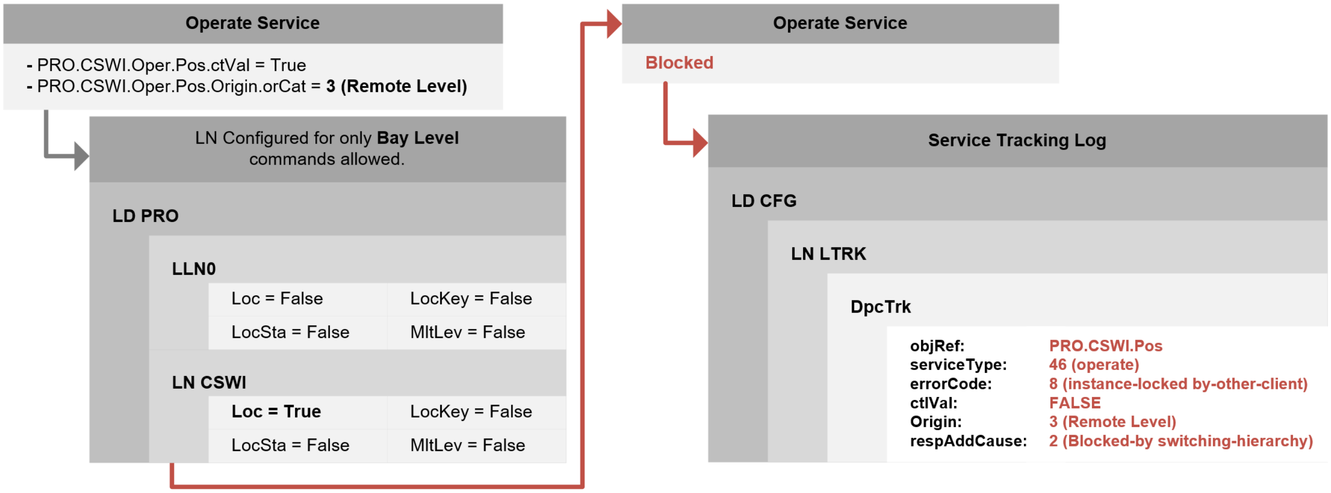


Fig. 7. Operate service—blocked.

VIII. Hands-On Scenarios

A. Hierarchical Equipment Architecture

Fig. 8 illustrates the communications architecture, showing the positioning of each device across different levels. The devices are positioned at different levels, and each one uses a specific orCat in its MMS command.

- At the bay level, there is an IED with orCat 1 and a local controller with orCat 4.
- At the station level, there is an HMI with orCat 2 and an automated controller with orCat 5.
- At the remote level, there is a supervisory control and data acquisition (SCADA) system with orCat 3 and an automated controller with orCat 6.

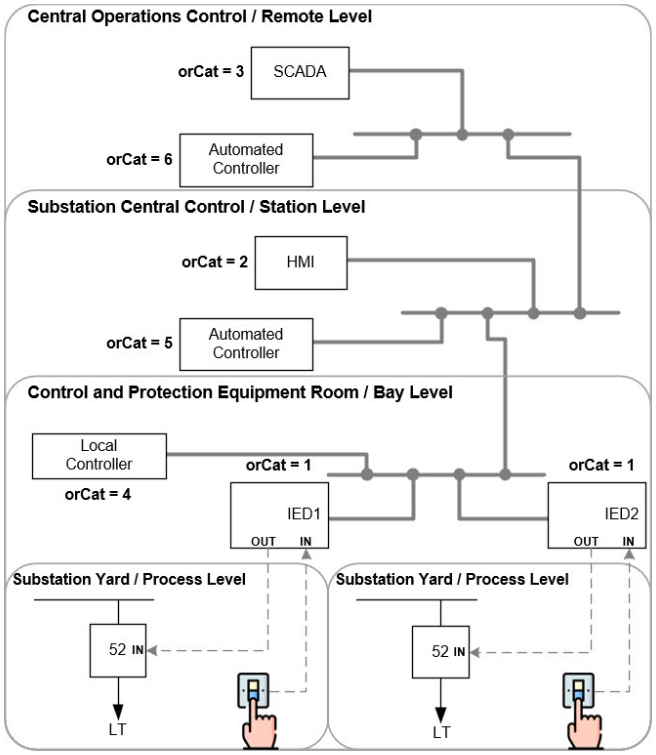

Central Operations Control / Remote Level
orCat = 3
SCADA
orCat = 6
Automated Controller
Substation Central Control / Station Level
orCat = 2
HMI
orCat = 5
Automated Controller
Control and Protection Equipment Room / Bay Level
Local Controller
orCat = 4
orCat = 1
IED1
OUT
IN
orCat = 1
IED2
OUT
IN
Substation Yard / Process Level
52
IN
LT
Substation Yard / Process Level
52
IN
LT

Fig. 8. Communications architecture.

B. Configuration

1) IED Configuration

Pushbutton_01, Pushbutton_02, Pushbutton_03, and Digital_input_01 are internal IED tags used by the IED logic to set the DOs. Operators configure these internal tags to establish the CA of an individual switchgear, as defined in Table I and Table II. For example, to set the CA at the bay level, the operator must assert Pushbutton_01. To set the CA at the process level, a physical key in the field must be activated, which asserts Digital_input_01 and, consequently, XCBR.Loc, thereby disabling remote commands. Fig. 9 illustrates a logic diagram within an IED, which includes internal variables used to determine the CA.

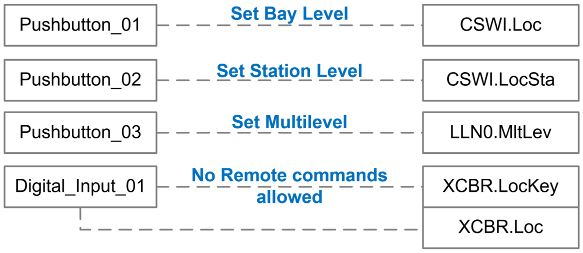


Fig. 9. IED logic.

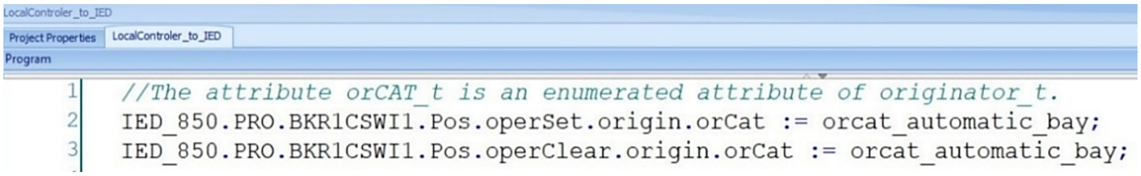
2) Bay-Level Configuration

a) IED (orCat 1)

The operator can execute commands using the IED display at the bay level while being physically present in front of the IED. These commands are executed with orCat 1.

b) Local Controller (orCat 4)

The local controller executes logic operations and autonomously controls breakers when enabled. It communicates status updates to HMI or SCADA via MMS, keeping operators informed. Automated functions like load shedding and capacitor bank control can be implemented. For example, for capacitor bank control, the local controller sends an MMS command with orCat 4 to an IED to open or close a breaker. The configuration is set in the local controller, as illustrated in Fig. 10.



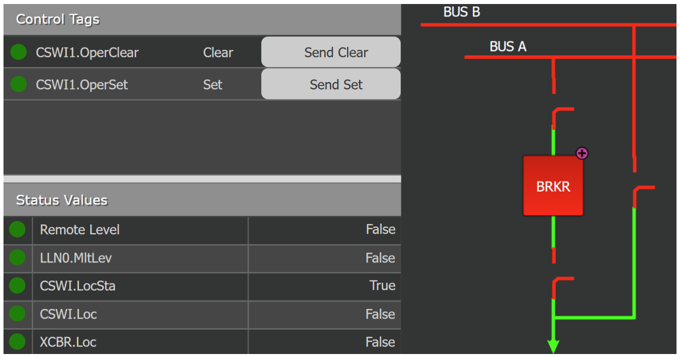
```
//The attribute orCAT_t is an enumerated attribute of originator_t.
IED_850.PRO.BKR1CSWI1.Pos.operSet.origin.orCat := orcat_automatic_bay;
IED_850.PRO.BKR1CSWI1.Pos.operClear.origin.orCat := orcat_automatic_bay;
```

Fig. 10. MMS command from local controller logic.

3) *Station-Level Configuration*

a) HMI (orCat 2)

The HMI provides real-time power system status to the substation operator and allows MMS commands to be sent to IEDs. In this project, the HMI displays breaker status and alarms, including the assigned CA, as shown in Fig. 11. When the operator issues an MMS command via the HMI, it is transmitted with orCat 2, as configured in Fig. 12.



Control Tags

● CSWI1.OperClear	Clear	Send Clear
● CSWI1.OperSet	Set	Send Set

Status Values

● Remote Level	False
● LLN0.MltLev	False
● CSWI.LocSta	True
● CSWI.Loc	False
● XCBR.Loc	False

Fig. 11. Operator interface at the station level.

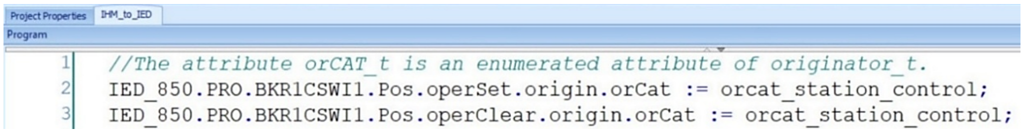
b) Automated Controller—Station Level (orCat 5)

The automated controller performs logic operations and autonomously controls breakers. It sends status updates to HMI or SCADA via MMS, keeping operators informed. It supports advanced automation functions like load shedding; black start; and fault location, isolation, and service restoration (FLISR). When sending MMS commands to IEDs (e.g., to shed load or restore service), the automated controller at the station level uses orCat 5. The IED decides which control level is authorized to operate the switchgear depending on the CA. The configuration is set in the automated controller, as illustrated in Fig. 13.

4) *Remote-Level Configuration*

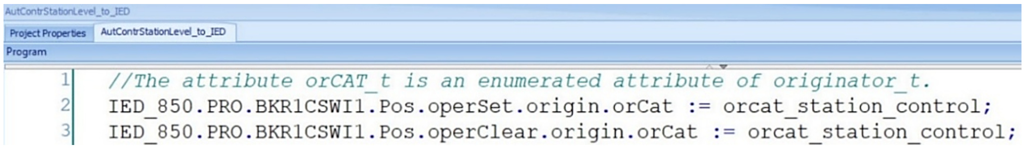
a) SCADA (orCat 3)

The SCADA system is responsible for providing the power system's status to the central control center. It can communicate with and manage multiple substations and may be located either within one of the substations or in a dedicated office for remote operations. SCADA can also send MMS commands to each IED in any substation. Fig. 14 shows the SCADA interface with a breaker, where the operator can send MMS commands to the IED. The operator can also view the DO's status to determine which CA is assigned to the breaker. When the operator sends a command from the SCADA, it is transmitted with orCat 3, which is configured as shown in Fig. 15.



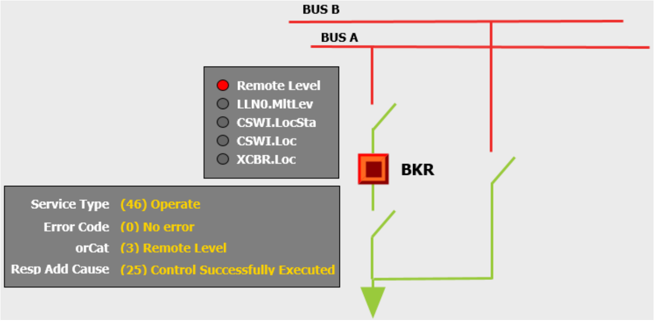
```
//The attribute orCAT_t is an enumerated attribute of originator_t.
IED_850.PRO.BKR1CSWI1.Pos.operSet.origin.orCat := orcat_station_control;
IED_850.PRO.BKR1CSWI1.Pos.operClear.origin.orCat := orcat_station_control;
```

Fig. 12. MMS command from station HMI.



```
//The attribute orCAT_t is an enumerated attribute of originator_t.
IED_850.PRO.BKR1CSWI1.Pos.operSet.origin.orCat := orcat_station_control;
IED_850.PRO.BKR1CSWI1.Pos.operClear.origin.orCat := orcat_station_control;
```

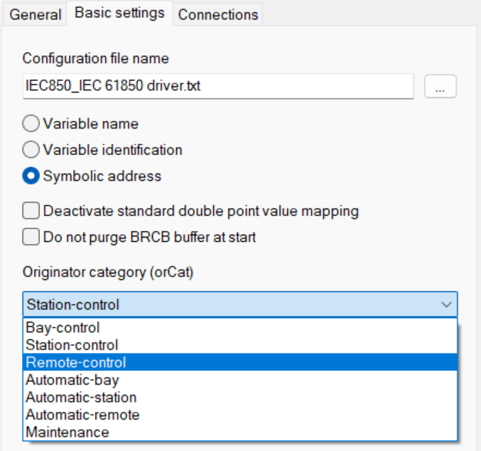
Fig. 13. MMS commands from automated controller logic.



● Remote Level
○ LLN0.MltLev
○ CSWI.LocSta
○ CSWI.Loc
○ XCBR.Loc

Service Type (46) Operate
Error Code (0) No error
orCat (3) Remote Level
Resp Add Cause (25) Control Successfully Executed

Fig. 14. SCADA interface.



Configuration file name
IEC850_IEC 61850 driver.txt

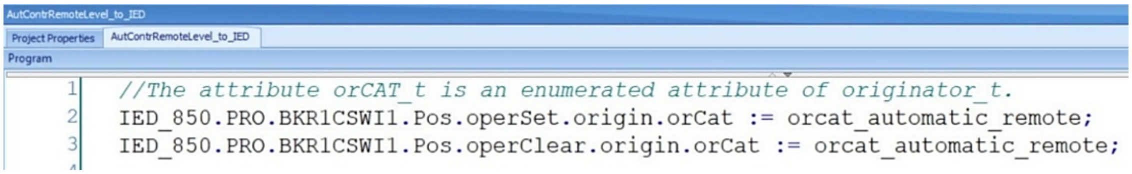
○ Variable name
○ Variable identification
● Symbolic address

☐ Deactivate standard double point value mapping
☐ Do not purge BRCB buffer at start

Originator category (orCat)
Station-control

Bay-control
Station-control
Remote-control
Automatic-bay
Automatic-station
Automatic-remote
Maintenance

Fig. 15. SCADA—remote-level configuration.



```
//The attribute orCAT_t is an enumerated attribute of originator_t.
IED_850.PRO.BKR1CSWI1.Pos.operSet.origin.orCat := orcat_automatic_remote;
IED_850.PRO.BKR1CSWI1.Pos.operClear.origin.orCat := orcat_automatic_remote;
```

Fig. 16. MMS command from automated controller logic (remote level).

b) Automated Controller—Remote Level (orCat 6)

The automated controller (remote level) is responsible for performing logic operations and executing actions. After completing certain actions, it provides status updates to SCADA via MMS, allowing the operator to understand what occurred. When the automated controller is enabled and ready, it can automatically open or close breakers without operator intervention, depending on the configured logic.

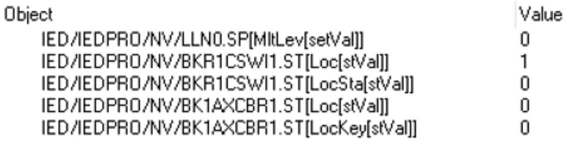
Several automated logic routines can be executed from central operations. One example is the automated service restoration of breakers, which saves operators time by avoiding the need to close them one by one. When the automated controller sends an MMS command to an IED to restore a breaker after a TRIP, the command is sent using orCat 6. The configuration is carried out as illustrated in Fig. 16.

C. Test Scenario

In this section, two scenarios are proposed, each with a different configuration. Scenario 1 configures the IED to accept commands only from the bay level and maintenance tool, while Scenario 2 is configured to block all remote commands. The objective is to send commands from the process level, bay level, station level, remote level, maintenance tool, unidentified origin, and nonexistent origin to determine whether the IED's local and remote functionalities allow the command to be executed or block it, and to verify the service tracking for each command.

1) Scenario 1: Only MMS Commands From Bay Level

In this scenario, only MMS commands from the bay level are allowed. The IED was configured to assert the CSWI.Loc. The XCBR.LockKey, CSWI.LocSta, and LLN0.MltLev remain deasserted. Fig. 17 shows the variable statuses, confirming that only CSWI.Loc is asserted.

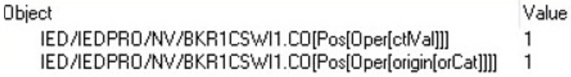


```
Object                                          Value
    IED/IEDPRO/NV/LLN0.SP[MltLev[setVal]]       0
    IED/IEDPRO/NV/BKR1CSWI1.ST[Loc[stVal]]      1
    IED/IEDPRO/NV/BKR1CSWI1.ST[LocSta[stVal]]   0
    IED/IEDPRO/NV/BK1AXCBR1.ST[Loc[stVal]]      0
    IED/IEDPRO/NV/BK1AXCBR1.ST[LocKey[stVal]]   0
```

Fig. 17. Bay level selected by IED.

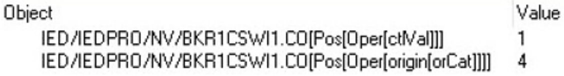
a) MMS Command From Bay Level

Fig. 18 shows the MMS command issued from the IED display (orCat 1). Fig. 19 shows the command issued from the local controller (orCat 4) to the IED. All these MMS commands are allowed and executed by the IED, closing the digital output.



```
Object                                                  Value
    IED/IEDPRO/NV/BKR1CSWI1.CO[Pos[Oper[ctlVal]]]       1
    IED/IEDPRO/NV/BKR1CSWI1.CO[Pos[Oper[origin[orCat]]]] 1
```

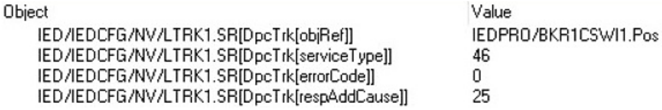
Fig. 18. MMS command with orCat 1.



```
Object                                                  Value
    IED/IEDPRO/NV/BKR1CSWI1.CO[Pos[Oper[ctlVal]]]       1
    IED/IEDPRO/NV/BKR1CSWI1.CO[Pos[Oper[origin[orCat]]]] 4
```

Fig. 19. MMS command with orCat 4.

The service tracking report from the IED is presented in Fig. 20. The service type is identified as 46, which corresponds to an operating command. The error code is 0, indicating that the command was executed successfully without any errors. Finally, the response AddCause is 25, confirming that the control action was successfully executed.

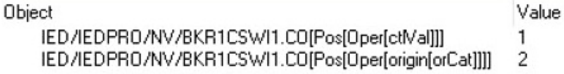


```
Object                                              Value
    IED/IEDCFG/NV/LTRK1.SR[DpcTrk[objRef]]          IEDPRO/BKR1CSWI1.Pos
    IED/IEDCFG/NV/LTRK1.SR[DpcTrk[serviceType]]     46
    IED/IEDCFG/NV/LTRK1.SR[DpcTrk[errorCode]]       0
    IED/IEDCFG/NV/LTRK1.SR[DpcTrk[respAddCause]]    25
```

Fig. 20. Service tracking from IED.

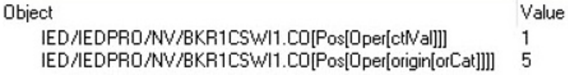
b) MMS Commands From Station and Remote Levels

Fig. 21 shows the MMS command issued from the HMI (orCat 2), Fig. 22 shows the command issued from the automated controller at the station level (orCat 5) to the IED, Fig. 23 shows the command issued from the SCADA (orCat 3) to the IED, and Fig. 24 shows the command issued from the automated controller at the remote level (orCat 6) to the IED. All these commands are blocked, and the digital output is not executed by the IED.



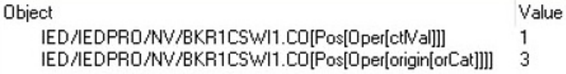
```
Object                                                  Value
    IED/IEDPRO/NV/BKR1CSWI1.CO[Pos[Oper[ctlVal]]]       1
    IED/IEDPRO/NV/BKR1CSWI1.CO[Pos[Oper[origin[orCat]]]] 2
```

Fig. 21. MMS command with orCat 2.



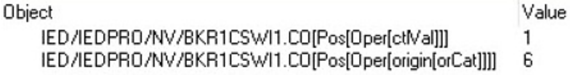
```
Object                                                  Value
    IED/IEDPRO/NV/BKR1CSWI1.CO[Pos[Oper[ctlVal]]]       1
    IED/IEDPRO/NV/BKR1CSWI1.CO[Pos[Oper[origin[orCat]]]] 5
```

Fig. 22. MMS command with orCat 5.



```
Object                                                  Value
    IED/IEDPRO/NV/BKR1CSWI1.CO[Pos[Oper[ctlVal]]]       1
    IED/IEDPRO/NV/BKR1CSWI1.CO[Pos[Oper[origin[orCat]]]] 3
```

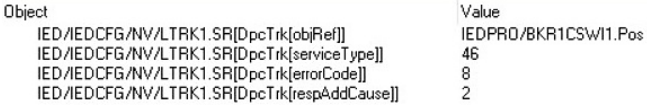
Fig. 23. MMS command with orCat 3.



```
Object                                                  Value
    IED/IEDPRO/NV/BKR1CSWI1.CO[Pos[Oper[ctlVal]]]       1
    IED/IEDPRO/NV/BKR1CSWI1.CO[Pos[Oper[origin[orCat]]]] 6
```

Fig. 24. MMS command with orCat 6.

The service tracking report from the IED is presented in Fig. 25. The service type is identified as 46, which corresponds to an operating command. The error code is 8, indicating that the command was blocked by a client. Finally, the response AddCause is 2, confirming that the control action was blocked by the switching hierarchy.

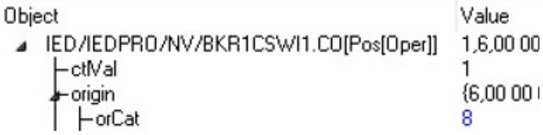


```
Object                                              Value
    IED/IEDCFG/NV/LTRK1.SR[DpcTrk[objRef]]          IEDPRO/BKR1CSWI1.Pos
    IED/IEDCFG/NV/LTRK1.SR[DpcTrk[serviceType]]     46
    IED/IEDCFG/NV/LTRK1.SR[DpcTrk[errorCode]]       8
    IED/IEDCFG/NV/LTRK1.SR[DpcTrk[respAddCause]]    2
```

Fig. 25. Service tracking from IED.

c) *Unidentified MMS Commands*

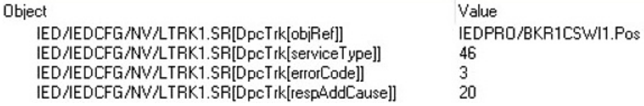
Fig. 26 shows the MMS command issued from the unidentified sender (orCat 8). The command is blocked, and the digital output is not executed by the IED.



```
Object                                          Value
 ◢ IED/IEDPRO/NV/BKR1CSWI1.CO[Pos[Oper]]        1,6,00 00
     ├ctlVal                                    1
     ├origin                                    {6,00 00 I
     │ ├orCat                                   8
```

Fig. 26. MMS command with orCat 8.

The service tracking report from the IED is presented in Fig. 27. The service type is identified as 46, which corresponds to an operating command. The error code is 3, indicating that the command was an access violation. Lastly, the response AddCause is 20, which confirms that the control action had no access authority.

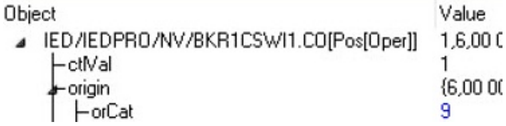


```
Object                                              Value
    IED/IEDCFG/NV/LTRK1.SR[DpcTrk[objRef]]          IEDPRO/BKR1CSWI1.Pos
    IED/IEDCFG/NV/LTRK1.SR[DpcTrk[serviceType]]     46
    IED/IEDCFG/NV/LTRK1.SR[DpcTrk[errorCode]]       3
    IED/IEDCFG/NV/LTRK1.SR[DpcTrk[respAddCause]]    20
```

Fig. 27. Service tracking from IED.

d) *Nonexistent Origin MMS Commands*

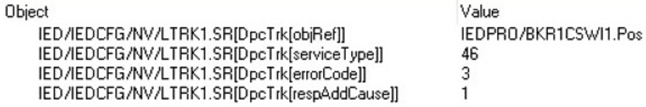
Fig. 28 shows the MMS command issued from the nonexistent orCat; in this example, it is shown by orCat 9. The command is blocked, and the digital output is not executed by the IED.



```
Object                                          Value
 ◢ IED/IEDPRO/NV/BKR1CSWI1.CO[Pos[Oper]]        1,6,00 (
     ├ctlVal                                    1
     ├origin                                    {6,00 0(
     │ ├orCat                                   9
```

Fig. 28. MMS command from nonexistent orCat.

The service tracking report from the IED is presented in Fig. 29. The service type is identified as 46, which corresponds to an operating command. The error code is 3, indicating that the command was an access violation. Lastly, the response AddCause is 1, which confirms that the control action was not supported.



```
Object                                              Value
    IED/IEDCFG/NV/LTRK1.SR[DpcTrk[objRef]]          IEDPRO/BKR1CSWI1.Pos
    IED/IEDCFG/NV/LTRK1.SR[DpcTrk[serviceType]]     46
    IED/IEDCFG/NV/LTRK1.SR[DpcTrk[errorCode]]       3
    IED/IEDCFG/NV/LTRK1.SR[DpcTrk[respAddCause]]    1
```

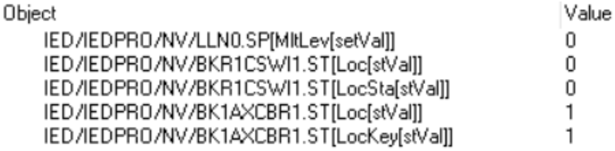
Fig. 29. Service tracking from IED.

e) *Resume of Scenario 1*

Table IV provides a summary of all allowed and blocked commands, along with the service tracking information from the IED.

2) *Scenario 2: Only MMS Commands From Process Level Allowed*

In this scenario, only MMS commands performed directly by the operator on the equipment in the substation yard are allowed. The IED input was asserted via a physical key located in the substation yard, which asserted XCBR.LocKey and XCBR.Loc. The CSWI.Loc, CSWI.LocSta, and LLN0.MltLev remain deasserted. Fig. 30 shows the variable statuses, confirming that XCBR.LocKey and XCBR.Loc are asserted.



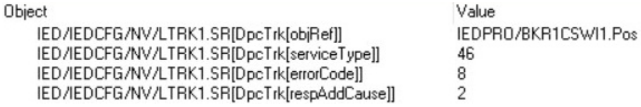
```
Object                                              Value
    IED/IEDPRO/NV/LLN0.SP[MltLev[setVal]]           0
    IED/IEDPRO/NV/BKR1CSWI1.ST[Loc[stVal]]          0
    IED/IEDPRO/NV/BKR1CSWI1.ST[LocSta[stVal]]       0
    IED/IEDPRO/NV/BK1AXCBR1.ST[Loc[stVal]]          1
    IED/IEDPRO/NV/BK1AXCBR1.ST[LocKey[stVal]]       1
```

Fig. 30. Process level selected by IED.

a) *MMS Command From All Levels*

All the MMS commands with orCat values from 1 to 6 were sent and the IED response was the same for all of them. All commands are blocked, and the digital output is not executed by the IED.

The service tracking report from the IED is presented in Fig. 31. The service type is identified as 46, which corresponds to an operating command. The error code is 8, indicating that the command was blocked by a client. Lastly, the response AddCause is 2, which confirms that the control action was blocked by the switching hierarchy.



```
Object                                              Value
    IED/IEDCFG/NV/LTRK1.SR[DpcTrk[objRef]]          IEDPRO/BKR1CSWI1.Pos
    IED/IEDCFG/NV/LTRK1.SR[DpcTrk[serviceType]]     46
    IED/IEDCFG/NV/LTRK1.SR[DpcTrk[errorCode]]       8
    IED/IEDCFG/NV/LTRK1.SR[DpcTrk[respAddCause]]    2
```

Fig. 31. Service tracking from IED.

b) *Unidentified and Nonexistent Origin MMS Commands*

The IED response was the same as observed in Scenario 1 for orCat Values 8 and 9, as shown in Sections 8.3.1.C and 8.3.1.D.

a) *Resume of Scenario 2*

Table V provides a summary of all allowed and blocked commands, along with the service tracking information from the IED.

3) *Conclusions From the Test Scenarios*

It was observed that the IEC 61850 solutions for CA, implemented through local and remote functionalities, enable the IED to respond appropriately to each MMS command based on its specific origin and to provide feedback via service tracking, which can be visualized at any level. These features contribute to safer and more predictable automation in electrical power substations. Table VI shows when each command was allowed or blocked.

TABLE IV
SCENARIO 1 SERVICE TRACKING

Service Tracking	Scenario 1			
orCat Value	1, 4	2, 3, 5, 6	8	9
Command	Allowed	Blocked	Blocked	Blocked
Service Type	46 (operate)	46 (operate)	46 (operate)	46 (operate)
Error Code	0 (no error)	8 (instance blocked)	3 (access violation)	3 (access violation)
RespAddCause	25 (successfully executed)	2 (blocked by hierarchy)	20 (no access authority)	1 (not supported)

TABLE V
SCENARIO 2 SERVICE TRACKING

Service Tracking	Scenario 2		
orCat Value	1, 2, 3, 4, 5, 6	8	9
Command	Blocked	Blocked	Blocked
Service Type	46 (operate)	46 (operate)	46 (operate)
Error Code	8 (instance blocked)	3 (access violation)	3 (access violation)
RespAddCause	2 (blocked by hierarchy)	20 (no access authority)	1 (not supported)

TABLE VI
LIST OF ALLOWED AND BLOCKED COMMANDS

Scenario	Process Level	Bay Level	Station Level	Remote Level	Unidentified	Nonexistent
		orCat 1 and 4	**orCat 2 and 5**	**orCat 3 and 6**	**orCat 8**	**orCat 9**
1	Allowed	Allowed	Blocked	Blocked	Blocked	Blocked
2	Allowed	Blocked	Blocked	Blocked	Blocked	Blocked

IX. COMPARING WITH OTHER PROTOCOLS

In an SAS, protocols such as Modbus, IEC 60870-5-104, and Distributed Network Protocol (DNP3) are used to communicate with IEDs. While the protocols support command functions, they lack mechanisms to trace the origin of commands:

- Modbus uses function code (e.g., 05, 15) but cannot identify the source of a command.
- IEC 60870-5-104 uses Application Service Data Units (ASDUs) for commands (e.g., C_SC_NA_1) and provides status responses but does not indicate the origin of the command.
- DNP3 employs object-based commands (e.g., Object 12), which provide status responses and internal indications (e.g., internal failures and buffer overflows), but it also lacks source identification.

None of these protocols support command traceability or source identification, which limit advanced features such as command blocking or service tracking, capabilities that are available in IEC 61850 [5].

X. BENEFITS

This paper outlines the CA features of IEC 61850, highlighting their practical benefits for an SAS:

- CA enables hierarchical control in an SAS, allowing users to assign authorization levels to MMS commands. This ensures that only designated control levels can operate specific breakers and provides visibility about the origin of each command.
- orCat attributes in CA specify the exact source of a command, enabling IEDs to respond with precise and appropriate actions.
- CA improves diagnostics by using service tracking to explain why unauthorized commands are rejected—a feature not available in protocols like Modbus, DNP3, or IEC 60870-5-104.
- CA standardizes local and remote control schemes, replacing custom IED logic with a unified approach. This simplifies substation operation and enhances team and system integration across projects.

- CA eliminates the need for custom logic by integrating local and remote control schemes directly into the MMS protocol. This reduces maintenance complexity and supports flexible, scalable substation operations.

XI. Conclusion

This paper provides an explanation of how to implement CA within an SAS, covering key concepts such as local and remote functionality, hierarchical command structure, interactions between LNs and LDs, the origin of control commands, and the group of attributes used for service tracking. In addition to the theoretical concepts of IEC 61850, practical test scenarios are presented, including configuration steps, command execution, and result analysis.

The test scenarios demonstrated the effectiveness of the IEC 61850 CA concept in managing MMS commands from different levels. In Scenario 1, the IED correctly accepted commands only from the bay level and the maintenance tool, while blocking all others—including those from the station level, remote sources, unidentified origins, and nonexistent sources. In Scenario 2, only process-level commands were permitted, with all remote commands blocked. The service tracking reports provided detailed feedback for each command, confirming whether the MMS command was executed or blocked and explaining the reason.

This behavior validates the IED’s ability to enforce hierarchical control and enhance operational security. Overall, the results confirm that the CA concept is a relevant solution for achieving safer, more reliable, and more predictable automation in modern substations, aligning with the goals of IEC 61850.

XII. References

[1] IEC 61850-5, *Communication Networks and Systems for Power Utility Automation* – Part 5: Communication Requirements for Functions and Device Models, 2022.

[2] IEC 61850-7-1, *Communication Networks and Systems for Power Utility Automation* – Part 7-1: Basic Communication Structure – Principles and Models, 2020.

[3] IEC 61850-7-4, *Communication Networks and Systems for Power Utility Automation* – Part 7-4: Basic Communication Structure – Compatible Logical Node Classes and Data Object Classes, 2020.

[4] IEC 61850-7-2, *Communication Networks and Systems for Power Utility Automation* – Part 7-2: Basic Information and Communication Structure – Abstract Communication Service Interface (ACSI), 2020.

[5] G. Clarke, D. Reynders, and E. Wright, *Practical Modern SCADA Protocols: DNP3, 60870.5 and Related Systems*, Newnes, Oxford, UK, 2004.

XIII. Biographies

Romulo Corna earned his BSc in Electrical Engineering from the Federal Technological University of Paraná in 2010. He specialized in network computers in 2015 and completed his professional master’s degree in electrical engineering at LACTEC of Paraná in 2018. He joined Arteche EDC in 2009. In 2014, he began working at Schweitzer Engineering Laboratories, Inc. (SEL), where he has held various roles including project engineer, application engineer, and SEL University instructor. In 2022, he relocated to Pullman, Washington, to join the Research and Development division, where he currently serves as a development lead engineer.

Wellington Oliveira holds a BSc in Electrical Engineering from Faculdade de Ciência e Tecnologia and brings nearly 25 years of experience in power system communication, automation, and protection. An IEEE Senior Member, he has collaborated extensively with utilities and manufacturers, specializing in testing, integration, and commissioning of critical infrastructure. Since 2012, Wellington has also been actively involved in a wide range of IEC 61850-related projects, deepening his expertise in substation communication and automation standards.

20250731 • TP7234