PC37.115 Draft 9

December 2002

Draft Standard Test Method for Use in the Evaluation of Message Communications between Intelligent Electronic Devices in an Integrated Substation Protection, Control and Data Acquisition System

Sponsored by the

Power System Relaying Committee
of the
IEEE Power Engineering Society

IEEE Standards Activities Department
Standards Licensing and Contracts
445 Hoes Lane, P.O. Box 1331
Piscataway, NJ 08855-1331, USA

Abstract: A standard test method, including evaluation criteria and performance measures, test scenarios for communication between intelligent electronic devices (IEDs) that implement substation integrated protection, control and data acquisition is defined. Test scenarios are used to describe what data is exchanged between IEDs to perform a specified function that may be distributed between IEDs. All test scenarios use a standard Unified Modeling Language (UML) and a core reference model to build-out an implementation.

Keywords: automatic control, data acquisition, SCADA, supervisory control, communication, intelligent electronic devices, IED, distributed processing, test scenario, Unified Modeling Language, UML, communication performance measures, communication evaluation criteria.

Introduction

(This introduction is not part of IEEE C37.115, Standard for Substation Protection, Control and Data Acquisition Communications.)

This standard applies to systems used to communicate between intelligent electronic devices (IEDs) for substation integrated protection, control and data acquisition. The requirements of this standard are in addition to those contained in standards for individual devices (e.g., relays, switchgear)

This standard applies to a rapidly changing technology. It is anticipated that frequent revision may be desirable.

At the time this standard was completed, Working Group H4, Standards for Evaluation of Message Communication between IEDs of the Relaying Communications H Subcommittee had the following membership:

Dennis Holstein, *Chair*

Mark Adamiak	Jerry Hohn	Charles Sufana
Alex Apostolov	Kazik Kuras	John Tengdin
John Burger	Jeff McElray	Mike Thompson
Mason Clark	Gary Michel	Eric Udren
Douglas Dawson	Fredric Planchon	Murty Yalla
George Gresko	Jacques Prevost	
Ameen Hamdon	Tevfik Sezi	
Wayne Hartman	Dilip Shroff	

At the time this standard was completed, the Relaying Communications H Subcommittee had the following membership:

Mark Simon, *Chair*

Mark Adamiak
John Burger
Paul Drum
Ken Fodero
Jerry Hohn
Dennis Holstein
Jim Huddleston
Jim Ingleson
Lowe
Ken Martin
Jeff McElray
Gary Michel
Brad Nelson
Tim Phillippe
Roger Ray
Robert Ryan
Miriam Sanders
Chuck Shank
John Tengdin
Eric Udren
Murty Yalla
John Zip

The following persons were on the balloting committee: (To be provided by IEEE editor at time of publication.)

William Ackerman
Mark Adamiak
David Bassett
Stuart Borlase
Stuart Bouchey
Gustavo Brunello
John Burger
James Carlo
John Chadwick
Mason Clark
Kay Clinard
Guru Dutt Dhingra
Robert Dempsey
Paul Drum
Ernest Duckworth
Gary Engmann
Kenneth Fodero
James Gardner
Jeffrey Gilbert
Tony Giuliante
Daniel Gregory
Erich Gunther
Roger Hedding
Jerry Hohn
Dennis Holstein
James Hrabliuk
Viaplana John
Robert JeanJean
Hermann Koch
Joseph L. Koepfinger
Terry Krummrey
Robert Landman
Daniel Love
Greg Luri
John McDonald
Jeff McElray
Jesus Martinez
Gary Michel
Gary L. Nelson
Daniel Nordell
Arun Phadke
Paul Pillitteri
Barry Pokorney
Peter Rashcio
Paulo Ribeiro
James Ruggieri
Mohindar S. Sachdev
Thomas Schossig
Robert Sodergren
Richard Taylor
John Tengdin
Demetrios Tziouvaras
Eric Udren
Benton Vandiver
James Wardin
Philip Winston
Peter Wong
Murty Yalla

Contents

Table of Figures

Draft Standard Test Method for Use in the Evaluation of Message Communications between Intelligent Electronic Devices in an Integrated Substation Protection, Control and Data Acquisition System

1 Overview

1.1 Scope

This standard defines standard communication modeling, terminology, evaluation criteria and performance measures for communication test scenarios, which specify messages to be exchanged between electrical power substation intelligent electronic devices (IEDs). These scenarios define message transactions between applications within the substation, and between substation IEDs and remotely located applications. The scenarios do not specify the communication protocol required to implement the transactions.

1.2 Purpose

There are currently no coherent communication modeling, terminology and communication test scenarios for the evaluation of one or more implementation concepts for communication between substation IEDs within a substation or between a substation and remote IEDs. Utilities and vendors will use this standard to evaluate, on a common basis, one or more implementation solutions.

1.3 Document organization

In addition to the overview, described in this clause, normative references are described in Clause 2.

Communication network structures are described in Clause 4 for the substation LAN architecture and for the utility enterprise WAN architecture. The WAN architecture is further described in terms of the requirements for bridging communication networks, switching between communication networks, and routing between communication networks.

Network management and application development tools are described in Clause5. These tools include the network analyzer, network management workstations, monitoring software, server-based databases, and test methods and measurements.

Communication test method is formally defined in Clause6. This method is specified in terms of functional and performance requirements for conformance testing, interoperability testing, and operability testing. Test concepts are specified in terms of

requirements for performance measures, evaluation requirements, procurement specification and vendor requirements.

Annexes listed below are included to add both informative and normative specifications.

Annex	Description
A (Informative)	Bibliography describing references that are commonly used in power system engineering and communications between intelligent electronic devices.
B (Informative)	Synch check to close breaker.
C (Informative)	Load tap changer control with remote voltage measurements.
D (Informative)	Distributed generation on utility feeders.

2 References

This standard shall be used in conjunction with the following publications. When the following standards are superseded by an approval revision, the revision shall apply unless noted otherwise.

ISO/IEC 9646, Information Technology – Open Systems Interconnection – Conformance testing methodology and framework.

3 Definitions and acronyms

3.1 Definitions

A list of some of the terms used in technical descriptions in this document is described below. The Unified Modeling Language (UML) descriptions related to the Object Management Architecture (OMA) object model is identified by the designation [OMA].

When brackets enclose a multiword term (one or more words), it indicates that those words are optional when referring to the term. For example, use case [class] may be referred to as use case.

The following conventions are used.

Contrast: <term>. Refers to a term that has an opposed or substantively different meaning.

See: <term>. Refers to a related term that has a similar, but not synonymous meaning.

Synonym: <term>. Indicates that the term has the same meaning as another term, which is referenced.

Acronym: <term>. This indicates that the term is an acronym. The reader is usually referred to the spelled out term for the definition, unless the spelled-out term is rarely used.

3.1.1 Action A computational or algorithmic procedure.

3.1.2 Action state A state with an internal action and one or more outgoing transitions involving the completion of an internal action.

3.1.3 Active class A class whose instances are active objects. See: active object.

3.1.4 Active object An object that owns a thread and can initiate control activity. An instance of active class. See: active class.

3.1.5 Activity diagram A special case of a state diagram in which all or most of the states are action states and in which all or most of the transitions are triggered by completion of actions in the source states. Contrast: state diagram.

3.1.6 Actor [class] A predefined stereotype of type denoting an entity outside the system that interacts with use cases.

3.1.7 Addressing Means to identify the source and sink (recipients) of all information transfers.

3.1.8 Address Resolution Protocol (ARP) An address used alongside IP to discover IP-to-MAC address mappings. NOTE: ARP is described in RFC-826.

3.1.9 Adjacent Substation Protection Protection of power system equipment at one substation based on data measured at others. Examples are line differential protection and teleprotection schemes.

3.1.10 Agent Servers that are designated to work with compatible client stubs known as user agents, which share the same server protocol. Agents are responsible for picking up and delivering messages between senders and receivers.

3.1.11 Aggregate [class] A class that represents the 'whole' in an aggregation (whole-part) relationship. See: aggregation.

3.1.12 Aggregation A special form of association that specifies a whole-part relationship between the aggregate (whole) and a component part. Contrast: composition.

3.1.13 Alarm processing Alarm analysis procedures to improve presentation of alarm data. It ranges from updating alarm lists and producing group alarms up to more intelligent evaluations.

3.1.14 Artifact A piece of information that is used or produced by a software development process. An artifact can be a model, a description or software.

3.1.15 Association role The roles that type or class plays in an association.

3.1.16 Asynchronous message A message where the sending objects do not pause to wait for results. Synonym: asynchronous request [OMA]. Contrast: synchronous message.

3.1.17 Asynchronous transmission In data communications, a method of transmission in which sending and receiving of data is controlled by control characters rather than by a timing sequence. Contrast: synchronous transmission.

3.1.18 Attribute description

a) Text describing an object's attribute. It is a filled-in template of information about an attribute, including: description, legal values, unit of measure, required (yes/no) get/set constraints, rules for getting a default value, applicable states, and tractability codes.

b) For real-time systems, categorize each attribute as: state, state-dependent, or state independent.

3.1.19 Automatic switching sequences Automatic sequential operation of groups of power system devices to reduce operator workload and/or switching time and to avoid unsuccessful or unnecessary switching attempts.

3.1.20 Availability of data State in which data are where the user needs them, when the user needs them, and how the user needs them.(see also multiple - transparent data access).

3.1.21 Bandwidth (data) The rate at which a communications link is capable of carrying data, usually measured in bits per second (bps)~~The volume of~~

~~data that a communications link is capable of carrying, usually measured in bits/sec.~~ See also latency.

3.1.22 Binary association An association between two classes. A special case of an n-ary association

3.1.23 Binary Large Object Very large (may be several megabytes in size) binary representation of an image data type.

3.1.24 Boolean expression An expression that evaluates to a boolean value.

3.1.25 Breaker A device that connects and disconnects power circuits, with fault-interrupting capability. (Synonymous with circuit breaker).

3.1.26 Breaker (health) monitoring A function that measure breaker's parameters mainly for maintenance purpose.

3.1.27 Breaker failure protection Backup protection scheme to trip all connected breakers if a breaker fails to clear a detected fault.

3.1.28 Broadcast mode Concurrent transfer mode of information to all connected receivers with one message from the information source. Contrast: unicast and multicast modes.

3.1.29 Circuit breaker See breaker.

3.1.30 Class diagram A diagram that shows a collection of declarative (static) model elements such as classes, types, and their contents and relationships.

3.1.31 Collaboration diagram A diagram that shows object interactions organized around the objects and their links to each other. Unlike a sequence diagram, a collaboration diagram shows the relationships among the objects. Sequence diagrams and collaboration diagrams express similar information but show it in different ways. See: sequence diagram.

3.1.32 Cold load pickup Restoration of a circuit where all load diversity is lost.

3.1.33 Communication association In a deployment diagram, an association between two nodes that implies a communication. See: deployment diagram.

3.1.34 Communication interface Serial interface of a device that allows exchange of (physical and logical) information among devices of the same or different functional levels in a hierarchical system. An interface specifies the connection of a communication link, with regard to the mechanical connection as well as to the signal's physical and functional characteristics.

3.1.35 Communication safety Measures and controls to avoid any deterioration or losses of information (reliability).

3.1.36 Communication security Measures and controls taken to deny unauthorized persons access to and information derived from communication facilities; and, to ensure the authenticity of communication transactions.

3.1.37 Component An executable software module with identity and a well-defined interface. Contrast: component [OMA].

3.1.38 Component diagram A diagram that shows the organizations and dependencies among components.

3.1.39 Composite [class] A class that is related to one or more classes by a composition relationship. See: composition.

3.1.40 Composite aggregation Synonym: composition.

3.1.41 Composite state A state that consists of substates. Contrast: substate.

3.1.42 Composition A form of aggregation with strong ownership and coincident lifetime as part of the whole. Parts with non-fixed multiplicity may be created after the composite itself, but once created they live and die with it (i.e., they share lifetimes). Such parts can also be removed before the death of a composite. Composition may be recursive. Synonym: composite aggregation.

3.1.43 Concurrent substate A substate that can be held simultaneously with other concurrent substates contained in the same composite state. See: composite state. Contrast: disjoint substate.

3.1.44 Connect function Process of initiating communication between two applications.

3.1.45 Core object model Term used to describe components of the reference object model that are used and reused to build-out the object model for each virtual device.

3.1.46 Container

a) An object that exists to contain other objects and that provides operations to access or iterate over its contents. For example, arrays, sets, dictionaries.

b) A component that exists to contain other components.

3.1.47 Delegation

The ability of an object to issue a message to another object in response to a message. Delegation can be used as an alternative to inheritance. Contrast: inheritance.

3.1.48 Deployment diagram

A diagram that shows the configuration of run-time processing nodes and the components, processes, and objects that live on them. Components represent run-time manifestations of code units. See: component diagrams.

3.1.49 Device

Physical entity connected to the communication network composed of at least on communication element (the network element), which may have a control element, and/or a monitoring element (transducer, actuator, etc.).

3.1.50 Digital fault recorder (DFR)

Device that samples and stores analog and related binary data during power system transients. See: Transient fault recorder (TFR).

3.1.51 Directory services

a) Service for resolving user identifications of network components to network routing information.

b) Automated services permitting users to determine network address of data and processing resources via symbolic names.

3.1.52 Disconnect function

Process of terminating communication between two applications.

3.1.53 Disjoint substate

A substate that cannot be held simultaneously with other concurrent substates contained in the came composite state. See: composite state. Contrast: concurrent substate.

3.1.54 Distribution tree

A set of routers and subnetworks that allows a (set of) group members(s) to receive traffic from any source. Depending on the algorithm in use by the multicast routing protocol, the tree may be routed at the source or at some central point in the network.

3.1.55 Dynamic classification

A semantic variation of generalization in which an object may change type or role. Contrast: static classification.

3.1.56 Equipment clock synchronization Automated procedure to maintain consistent time data throughout the substation or power system; e.g., for time tagging or synchronized sampling.

3.1.57 Equipment load monitoring Automated procedures to detect equipment overload. The goal is to tolerate temporary overload, support for adaptive protection and maintenance on request.

3.1.58 Embedded system An embedded system includes a microprocessor, involves the immediate control of hardware, and is limited to performing a specific function.

3.1.59 Export In the context of packages, to make an element visible outside its enclosing namespace. See: visibility. Contrast: export [OMA], import.

3.1.60 Extends A relationship from one use case to another, specifying how the behavior defined for the first use case can be inserted into the behavior defined for the second use case.

3.1.61 Fire To cause a state transition. Example, to fire off a communication message. See: transition.

3.1.62 Firewall A device or combination of hardware and software used to enforce security.

3.1.63 Focus of control A symbol on a sequence diagram that shows the period of time during which an object is performing an action, either directly or through a subordinate procedure.

3.1.64 Function block An autonomous function, such as auto-reclosing, operational units, breaker failure protection, disturbance recording, etc., which might be implemented in separate hardware.

3.1.65 Gen-spec structure Generalization-specialization is a relationship between classes. A specialization class inherits the responsibilities of its generalization classes.

3.1.66 Guard condition A condition that must be satisfied in order to cause an associated transition to fire.

3.1.67 Hard real-time A system is hard real-time if failure to respond to an event within a specified time is considered a system failure. A hard real-time system must make all its deadlines all the time. This requirement is a result of the fact that the correctness of the system is tied to the timeliness of its response.

3.1.68 High-speed sampled data Raw sampled data provided by an analog-to-digital converter located close to the VTs and CTs. A process bus, or dedicated point-to-point links, transfers such data samples from the converter to protection IEDs. (Synonym: Streaming data).

3.1.69 High-speed yard data Raw sampled data provided by an analog-to-digital converter located in the substation yard close to the VTs and CTs. Such data samples are transferred from the converter to protection IEDs via dedicated point-to-point links, or over a process bus.

3.1.70 Implementation inheritance The inheritance of the implementation of a more specific element. Includes inheritance of the interface. Contrast: interface inheritance.

3.1.71 Import In the context of packages, a dependency that shows the packages whose classes may be referenced within a given package (including packages recursively embedded within it). Contrast: import [OMA] and export.

3.1.72 Instance connection An association in the problem domain that maps one object needs with other objects in order to fulfill its responsibilities. An instance object is shown as a line drawn between objects. The end-points of an instance connection are position between individual objects (rather than between classes). Each object has amounts or range markings on each of its instance connection, reflecting its constraints with other objects.

3.1.73 Integrity Immunity requirements to the network of data transfer errors due to accidental or intentional interference. Three levels are defined:

High: where a vanishing small probability of undetected error must be achieved.

Medium: where inherent data redundancy provides adequate error immunity

Low: where errors are merely a nuisance to the data recipient.

3.1.74 Interaction A behavioral specification that comprises a set of message exchanges among a set of objects within a particular context to accomplish a specific purpose. One or more scenarios may illustrate an interaction.

3.1.75 Interchangeability Two IEDs are interchangeable when one can replace the other without changing external functionality or performances.

3.1.76 Interface inheritance The inheritance of the interface of a more specific element. Does not include inheritance of the implementation. Contrast: implementation inheritance.

3.1.77 Internet Control Message Protocol (ICMP) An Internet Protocol used to report used to report errors in processing datagrams. NOTE: ICMP is specified in RFC-792. ICMPv6 is specified in RFC-1885. ICMPv6 incorporates many of the functions from ICMP for IPv4, and it also includes IGMP's functionality. There is no separate protocol known as IGMPv6.

3.1.78 Internet Engineering Task Force (IETF) Publishes standards[1] for Internet operation, including TCP/IP and SNMP.

3.1.79 Internet Group Management Protocol (IGMP) An Internet Protocol used to keep neighboring multicast routers informed of the host group memberships present on a particular local network. IP hosts report their host group memberships to any immediately neighboring multicast routers use the IGMP IGMP is an asymmetric protocol and is specified here from the point of view of a host, rather than a multicast router. (IGMP may also be used, symmetrically or asymmetrically, between multicast routers. Such use is not specified here.) [NOTE: IGMPv1 is specified in Appendix 1 of RFC-1112.]

3.1.80 Internet Protocol (IP) A communication protocol used to move datagrams through an interconnected set of networks. Passing the datagrams from one internet module to another until the destination is reached does this. The internet modules reside in hosts and gateways in the internet system. The datagrams are routed from one internet module to another through individual networks based on the interpretation of an internet address. [NOTE: IP is defined in RFC-791. IPv6 is specified in RFC-1883.]

3.1.81 Interoperability testing Check whether two or more communication network implementations operate together for some range of applications over a specific medium.

3.1.82 Journaling Means to maintain an audit trail of all control activities.

3.1.83 Latency (communications) The delay between the time the data is sent from its origin and received at its destination. Latency determines how responsive the link will be. See also bandwidth.

[1] The IETF is not an accredited standards development organization, but it writes standards anyway.

3.1.84 Legacy systems Information systems built on older technology that is not very adaptable or scalable to integrate new microprocessor-based intelligent electronic devices.

3.1.85 Link role An instance of an association role. See: association role.

3.1.86 Master/Slave Communication management scheme called polling in which one IED (the Master) requests one IED, or group of IEDs, (Slaves) to deliver specified information. Only Masters, not Slaves, may issue unsolicited data or commands. Used where data flows primarily between the Slaves and the Master.

3.1.87 Metaclass A class whose instances are classes. Metaclasses are typically used to construct metamodels.

3.1.88 Meta-metamodel A model that defines the language for expressing a metamodel. The relationship between a meta-metamodel and metamodel is analogous to the relationship between a metamodel and a model.

3.1.89 Metamodel A model that defines the language for expressing a model. An instance of a meta-metamodel.

3.1.90 Metaobject A generic term for all metaentities in a metamodeling language. For example, metatypes, metaclasses, metaatrributes, and meta associations.

3.1.91 Model aspect A dimension of modeling that emphasizes particular qualities of the metamodel. For example, the structural model aspect emphasizes the structural qualities of the metamodel.

3.1.92 Model element An element that is an abstraction drawn from the system being modeled.

3.1.93 Monitor function Process of repeatedly comparing the current value of one variable with a predefined value, and reporting detected changes in the comparison state.

3.1.94 Multicast forwarding For each multicast IP packet received, a forwarding decision must be made. Typically, each packet must arrive on a specific upstream interface (i.e., incoming interface), and then must be copied onto a (set of) downstream “outgoing interfaces.”

3.1.95 Multicast mode Concurrent transfer mode of information to a predefined subset of all connected receivers with one message from the information source. Contrast: unicast and broadcast modes.

3.1.96 Multicast routing Activities performed by IP routers in order to determine how to forward multicast IP packets, either from some particular source to a group, or from any source to a group.

3.1.97 Multiple classification A semantic variation of generalization in which an object may belong to more than one class. See: dynamic classification.

3.1.98 Multiple inheritance A semantic variation of generalization in which a type may have more than one supertype. Contrast: single inheritance.

3.1.99 Multiplicity A specification of the range of allowable cardinalities that a set may assume. Multiplicity specifications may be given for roles within associations, parts within composites, repetitions, and other purposes. Essentially a multiplicity is a (possible infinite) subset of the non-negative integers. Contrast: cardinality.

3.1.100 n-ary association An association among three or more classes. Each instance of the association in an n-tuple of values from the respective classes. Contrast: binary association.

3.1.101 Non-repudiation Method by which the sender of data is provided with proof of delivery and the recipient is assured of the sender's identity, so that neither can later deny having processed the data.

3.1.102 Object diagram A diagram that encompasses objects and their relationships at a point in time. An object diagram may be considered a special case of a class diagram or a collaboration diagram. See: class, diagram, collaboration diagram.

3.1.103 Object lifetime A line in a sequence diagram that represents the existence of an object over a period of time. See: sequence diagram.

3.1.104 Operability testing Checks functionality within a communication network implementation, using an application to test in a multi-layer environment.

3.1.105 Persistent object An object that exists after the process or threads that created it are terminated. For example, the data that persists beyond the lifetime of a single program execution is stored in a permanent data store.

3.1.106 Portability Ability to move, with minimal changes, application software between computers.

3.1.107 A predefined basic type, such as an integer or a string.

Primitive type

3.1.108 Privileged user Subject that is granted special discretionary access privileges.

3.1.109 Process bus The serial bus closest to the process providing raw data or data processed very near to the switchgear or instrument transformers. In the area of process automation, very often called the field bus.

3.1.110 Projection A mapping from a set to a subset of it.

3.1.111 Pseudo-state A vertex in a state machine that has the form of a state but doesn't behave as a state. Pseudo-states include initial, final and history connections.

3.1.112 Qualifier An association attribute or tuple of attributes whose values partition the set of objects related to an object across an association.

3.1.113 Receive [a message] The handling of a message passed from a sender object. See: sender, receiver.

3.1.114 Receiver [object] The object handling a message passed from a sender object. Contrast: sender.

3.1.115 Reference A denotation of a model element.

3.1.116 Reference object model Term used to describe an object model for substation integrated protection, control and data acquisition. The reference object model provides abstract views of state-of-the-art IEDs than can be integrated over a communication network. The model provides the basis for developing a conceptual design that maps object attributes to a data structure ant to communication protocol services. It also provides the basis for comparing conceptual designs against a standard reference.

3.1.117 Refinement A relationship that represents the fuller specification of something that has already been specified at a certain level of detail. For example, a design class is a refinement of an analysis class.

3.1.118 Request for Comments (RFC) A document that is treated as pseudo-standard but is subject to change before published as a standard.

3.1.119 The use of a pre-existing artifact.

Reuse

3.1.120 Security
Immunity of network resources to accidental or intentional unauthorized access. Three levels are defined:
High: where access is limited to predefined and validated clients.
Medium: where access is granted to any client meeting simple criteria.
Low: where access (usually read-only) is granted to any client.

3.1.121 Semantic variation
A particular interpretation choice for a semantic variation point. See: semantic variation point.

3.1.122 Semantic variation point
A point of variation in the semantics of a metamodel. It provides an intentional degree of freedom for the interpretation of the metamodel semantics. See: semantic variation.

3.1.123 Sequence diagram
A diagram that shows object interactions arranged in time sequence. In particular, it shows the objects participating in the interaction and the sequence of messages exchanged. Unlike a collaboration diagram, a sequence diagram includes time sequences but does not include object relationships. A sequence diagram can exist in a generic form (describes all possible scenarios) and in an instance form (describes one actual scenario). Sequence diagrams and collaboration diagrams express similar information, but shows it in different ways. See: collaboration diagram.

3.1.124 Sequence of events (SOE) recorder
Device that samples and stores events like contact status changes, trips, limit violations, etc., for later replay and analysis. The events are time tagged.

3.1.125 Single inheritance
A semantic variation of generalization in which a type may have only one supertype. Contrast: multiple inheritances.

3.1.126 Soft real-time
A system in which timeliness of response is important but not a matter of complete system failure. The acceptable frequency of missed deadlines is dictated by design.

3.1.127 Static classification
Semantic variations of generalization in which an object may not change type or may not change role. Contrast: dynamic classification.

3.1.128 Stereotype
A type of modeling element that extends the semantics of the metamodel. Stereotypes must be based on certain existing types or classes in the metamodel. Stereotypes may extend the semantics, but not the structure of pre-existing types and classes. Certain stereotypes are

predefined in UML; the user may define others. Stereotypes are one of three extendibility mechanisms in the UML.

3.1.129 Structural model aspect

A model aspect that emphasizes the structure of the objects in a system, including their types, classes, relationships, attributes, and operations.

3.1.130 Substate

A state that is part of a composite state. A substate can either be a concurrent or disjoint substate. See: concurrent state, disjoint state.

3.1.131 Supplier

Others can invoke a type, class, or component that provides services that. Synonym: server object [OMA]. Contrast: client.

3.1.132 Switch (in data network)

Network (communication) device that filters, forwards, and floods frames based on the destination address of each frame. The switch operates generally at the data link layer (layer 2) of the OSI model. Switches provide a unique network segment on each port, thereby separating collision domain. They are more and more used to replace hubs (concentrators) to increase network performance and bandwidth. Some switches perform network switching (layer 3) in order to alleviate the burden on centralized routers.

3.1.133 Synchronous message

A message where the sending-object pauses to wait for results. Synonym: synchronous request [OMA]. Contrast: asynchronous message.

3.1.134 System reconfiguration

a) Procedure to manage changes in power system connectivity.

b) Procedure to overcome failures in redundant secondary systems.

3.1.135 Tagged value

The explicit definition of a property as a named-value pair. In a tagged value, the name is referred as the tag. Certain tags are predefined in the UML; the user may define others. See: constraint, stereotype.

3.1.136 Time event

An event that occurs at a particular time. It may be expressed as a time expression. See: event.

3.1.137 Time expression

An expression that resolves to an absolute or relative value of time.

3.1.138 Time to Live (TTL)

A field in the IP packet that controls how far an IP packet may travel. Each router decrements a packet's TTL field by one when forwarding it. The largest possible TTL is 255. If a router ever receives a packet whose TTL equals

1, it cannot forward the packet any further.

3.1.139 Timing mark

A denotation for the time at which an event or message occurs. Timing marks may be used in constraints.

3.1.140 Transient object

An object that exists only during the execution of the process or threads that created it.

3.1.141 Transmission Control Protocol (TCP)

A reliable connection-oriented communication protocol. NOTE: TCP is defined by RFC 793.

3.1.142 Transparent data access

Access to data from any point of the system without knowing the details of the data link.

3.1.143 Type expression

An expression that evaluates to a reference to one or more types.

3.1.144 Unified Modeling Language (UML)

The modeling language is the (mainly graphical) notation that methods use to express designs. UML rigorously defines the semantics of the object metamodel and provides a notation for capturing and communication object structure and behavior.

3.1.145 Use case [class]

A class that defines a set of use-case instances.

3.1.146 Use case diagram

A diagram that shows the relationships among actors and used cases within a system.

3.1.147 Use case instance

A sequence of actions a system performs that yields an observable result of value to a particular actor. Usually scenarios illustrate prototypical use case instances. An instance of a use case class. See: use case class.

3.1.148 Use case model

A model that describes a system's functional requirements in terms of use cases.

3.1.149 User Datagram Protocol (UDP)

A connectionless, unreliable, transport layer network protocol for the exchange of requests and replies between connected stations. NOTE: UDP defers reliability issues to the next higher layers: session, presentation or application layers of the OSI Reference Model, application layer of the Internet Reference Model as defined in RFC 768.

3.1.150 Uses

A relationship from a concrete use case to an abstract use case in which the behavior defined for the concrete use case employs the behavior defined for the abstract use case.

3.1.151 View element A textual and/or graphical projection of a collection of model elements.

3.1.152 View projection A projection of model elements onto view elements. A projection provides a location and a style for each view element.

3.1.153 Virtual device Class of power system devices.

3.1.154 Virus Self-replicating, malicious program segment that attaches itself to an application program or other executable system component.

3.1.155 Vulnerability Weakness in an information system, or cryptographic system, or components (e.g., system security procedures, hardware design, internal controls) that could be exploited.

3.1.156 Whole-part structure A structure which portrays a mapping between objects that takes on three basic forms: assembly-parts, container-contents, and collection-members. The notation is directional; so the structure can be drawn at any angle. The end-points are position to reflect a mapping between the objects. Each end of the whole-part structure is marked with an amount or range, indicating the number of parts that a whole may have.

3.2 Acronyms and abbreviations

A list of acronyms and abbreviations used in the technical descriptions in this document is described below.

3.2.1	**ARP**	Address Resolution Protocol
3.2.2	**ASE**	Application Service Element
3.2.3	**BFI**	Breaker Failure Initiate
3.2.4	**BLOB**	Binary Large Object
3.2.5	**CAS**	Common Application Service
3.2.6	**CASM**	Common Application Service Model
3.2.7	**CID**	Connection Identifier
3.2.8	**CL**	Connectionless
3.2.9	**CLTP**	Connectionless-mode Transport Service
3.2.10	**CO**	Connection-oriented
3.2.11	**COMTRADE**	Common Format for Transient Data Exchange
3.2.12	**CSOM**	Client-Server Object Model
3.2.13	**DCR**	Directional Comparison Relaying
3.2.14	**DFR**	Digital Fault Recorder
3.2.15	**DNP**	Distributed Network Protocol
3.2.16	**DTT**	Direct Transfer Trip
3.2.17	**DUT**	Device Under Test
3.2.18	**ESP**	Encapsulating Security Protocol
3.2.19	**ETP**	Engineering Test Platform
3.2.20	**GMT**	Greenwich Mean Time
3.2.21	**GOMSFE**	Generic Object Models for Substation and Feeder Equipment
3.2.22	**GOOSE**	Generic Object-Oriented Substation Event
3.2.23	**GPS**	Global Positioning System

3.2.24 GSP	Generic Service Provider
3.2.25 ICMP	Internet Control Message Protocol
3.2.26 ID	Identifier (i.e., name)
3.2.27 IDL	Interface Definition Language
3.2.28 IETF	Internet Engineering Task Force
3.2.29 IGMP	Internet Group Management Protocol
3.2.30 IKE	Internet Key Exchange
3.2.31 IP	Internet Protocol
3.2.32 IRIG	Inter Range Instrument Group
3.2.33 ISAKMP	Internet Security Association and Key Management Protocol
3.2.34 MMS	Manufacturing Message Specification (ISO 9506)
3.2.35 MICS	Model Implementation Conformance Specification
3.2.36 ODBMS	Object-oriented Data Base Management System
3.2.37 OMA	Object Management Architecture
3.2.38 OOA	Object-oriented Analysis
3.2.39 PACE	Priority Access Control Enabled
3.2.40 PIXIT	Protocol Implementation Extra Information for Testing
3.2.41 PMU	Phase Measurement Unit
3.2.42 PSOM	Power System Object Model
3.2.43 RFC	Request For Comment
3.2.44 QoS	Quality Of Service
3.2.45 RBE	Report By Exception
3.2.46 RCB	Report Control Block
3.2.47 RCL	Relay Control Logic
3.2.48 RDBMS	Relational Data Base Management System

3.2.49 RLC Reactive Line Component

3.2.50 RPC Remote Procedure Call

3.2.51 SBO Select-Before-Operate

3.2.52 SNMP Simple Network Management Protocol

3.2.53 TAL Time Allowed to Live

3.2.54 TCP Transmission Control Protocol

3.2.55 TFR Transient Fault Recorder

3.2.56 TOE Thread Of Execution

3.2.57 TTL Time To Live

3.2.58 UCA® Utility Communication Architecture

3.2.59 UDP User Datagram Protocol

3.2.60 UML Unified Modeling Language

3.2.61 VMD Virtual Manufacturing Device

4 Communication network structures

Several communication network structures are used in this standard. Near real-time information transfer between IEDs within a substation, over the substation Local Area Network (LAN), is one communication network structure. Another structure for control and monitoring uses the Utility enterprise Wide Area Network (WAN) for communication between the control center or technical services center and the substation. And a third structure includes the transfer of information between two utility communication networks or between the utility WAN and the customer's WAN. All three structures are used in the scenarios described in the annexes.

4.1 Substation LAN architecture

Figure 4-1~~Figure 4-1~~ shows an example reference architecture, which is used in this standard for substation-distributed communication. A firewall is shown between the Substation router and the Substation LAN. A transceiver, connected to the Substation router, is included to transmit or receive data from either a satellite or radio. System interfaces to the Utility WAN are described in IEC 61968 and IEC 61970, which use a common interface reference model.

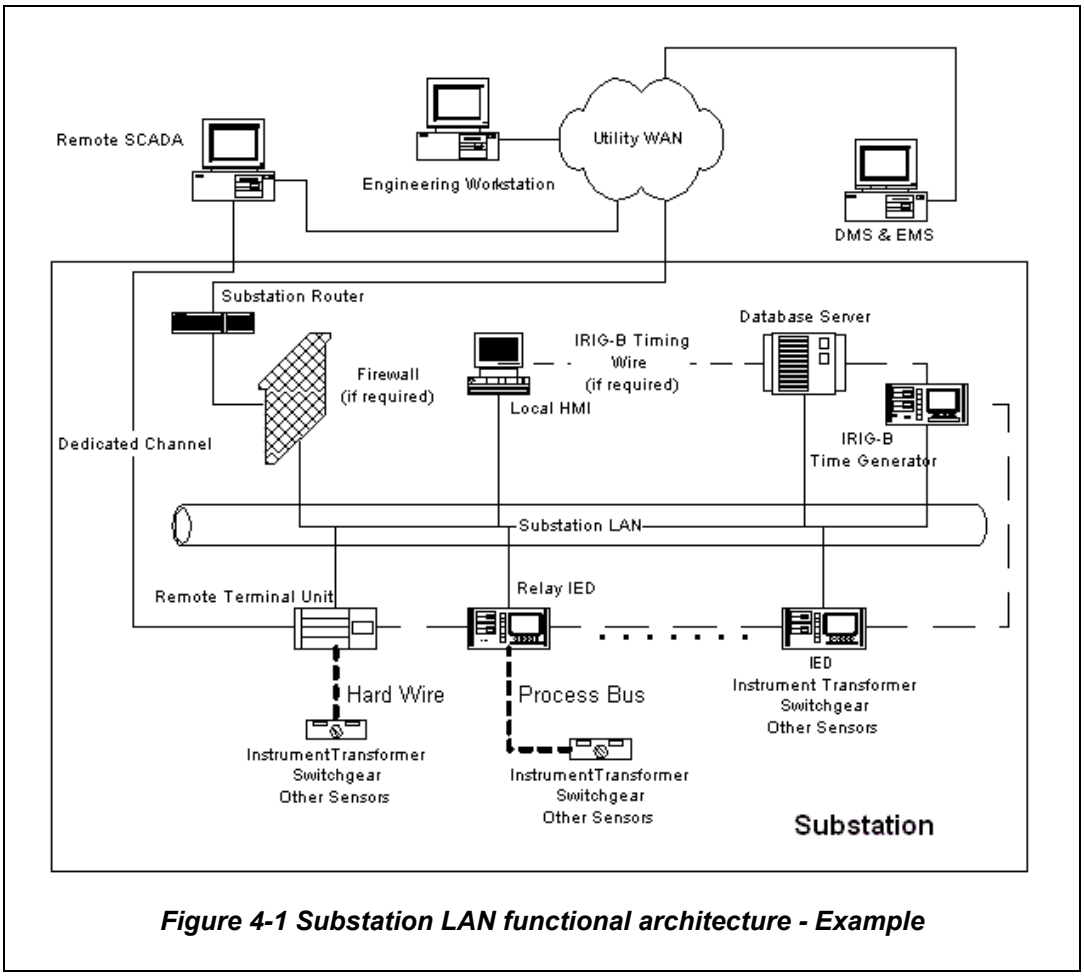

Figure 4-1 Substation LAN functional architecture - Example

If required for precise time synch, an IRIG-B time generator and timing wire may be used to synchronize the IED clocks so that data from multiple sources can be combined for post-fault analysis. If less time synch precision is needed, time synchronization may be achieved by sending time synch messages over the substation LAN. Time synch precision requirements are defined in IEEE P1525 [A45].

A high-speed database server connected to the LAN is used to maintain all substation configuration data, and to record all substation event data.

Instrument transformers, switchgear and other sensors are connected to the IEDs by either hardwire to a concentrator such as a Remote Terminal Unit (RTU), or through a process bus as described in IEC 61850. Some instrument transformers, switchgear and other sensors have IEDs that provide the capability to communicate over the LAN which is extended from the control house into the switchyard as described in IEEE P1525 [A45][2]. Figure 4-1~~Figure 4-1~~ shows "**…**" to indicate that there may be many IEDs on the substation LAN of the type describe above.

Information security requires the use of a firewall, or other security provisions, to provide access protection to the substation IEDs. The firewall shown in Figure 4-1~~Figure 4-1~~ provides a reasonable level of protection for all information passing through the substation router[3].

However, a potential security problem exists through dedicated channels that by-pass the substation router. The dedicated channel, which is typical in today's substations, between the remote SCADA and the substation RTU is one example of opening a back door that could result in a security problem. C37.115 recommends that future dedicated channels be protected by the firewall. Other firewalls, or security provisions, external to the substation are not considered in this standard.

4.2 Utility enterprise WAN architecture

The complex functionality of the modern telecommunication network can be understood by dividing the telecommunication network into logical layers. These represent different function of the network, and have often been implemented using separate equipment. However, the functional layers are not separate in a geographical sense, but equipment of several layers is typically co-located in the bigger network nodes[4].

4.2.1 Bridging communication networks

Bridges are used to extend the size of a network, both with regard to the number of workstations and IEDs and also geographical coverage. Bridges are also used to divide a network into subnets, or to connect networks in different areas as shown in Figure 4-2~~Figure 4-2~~. As opposed to a router, a bridge works on the data link and physical

[2] This assumes that data has been digitized and the IED is embedded in the switchgear, instrument transformer, or sensors and does not include a process bus, but outputs the data directly onto the extended substation LAN. Although possible, and under consideration by the development team for IEC 61850, this may not be practical. The concept is included in C37.115 for completeness.

[3] A firewall may be implemented in a separate physical device, or it can be implemented within the substation router. If security is only required in one or two of the substation IEDs, such as a substation computer (host) or database server, the firewall may be implemented on a board in the computer or database server, which provides security to that device only.

[4] A detailed description of this architectural concept is provided in the CIGRE 35.07 report (see Annex A).

layers of the Open System Interconnect (OSI) model, which is defined in ISO 7498. A bridge doesn't care what network protocols are in use – it only tries to transfer packets between networks. Since a bridge does not have the overhead associated with a router, a higher level of performance is possible.

Bridges also prevent local messages from leaving the subnet and restrict the flow of faulty Data Link layer-2-packets to a certain part of the network.

Figure 4-2~~Figure 4-2~~ also includes gateways for protocol conversion if the networks use dissimilar protocols. Gateways are used to exchange information at the transport, session, presentation, and application layers of the OSI reference model. The gateways know nothing about the network, data link and physical layers of the network.

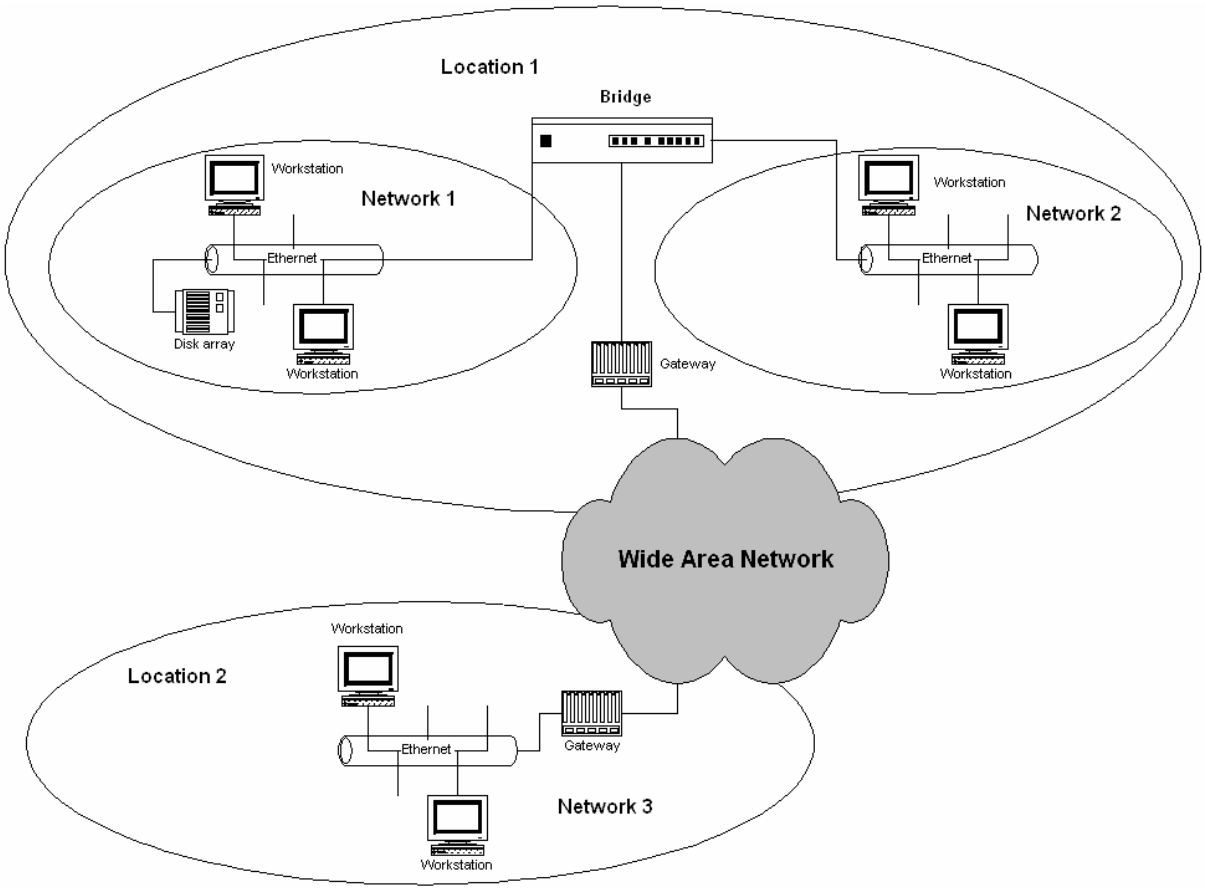

Figure 4-2 Extension of Local Area Networks via bridges and gateways

4.2.2 Switching between communication network segments

The idea behind switching is the attempt to solve the problems with system loading within a Local Area Network. If the aggregate demand of users on a network segment requires more than 50% of the available communication bandwidth, the network segment is too big and should be sub-divided. Communication switches should then connect sub-networks. The connection between clients and servers located on different sub-networks is done peer-to-peer by a LAN-Switch. LAN-Switches are products that use the functionality of components of Hub-systems and multi-port bridges.

4.2.3 Routing between communication networks

Routing refers to the transmission of an IP-datagram from one node to another on the same or a different network. Basically, routing can be described as a mechanism for forwarding of IP-datagrams on a “next-hop” strategy. The route refers to the paths that are chosen to transmit an IP datagram from its origin to its destination, based on the IP addresses contained in the datagram.

Routers are coupling elements, capable to connect different sub-networks on Layer 3 (Network Layer) according to the OSI Reference Model. Normally, IP-datagrams are forwarded with an IP-Network based on the destination address. The following decisions are possible:

1. Is it a local IP-address: If so, the packet has to be handled locally.
2. Is it a broadcast (or multicast) address of a direct connected network: If so the packet will be handled locally, and if necessary send as broadcast or multicast to the directly connected network.
3. Is it a destination address of a directly connected LAN: If so, the packet will be forwarded to a Router in the LAN (“next-hop”).

Within a network, a node sending an IP datagram can perform the following action directly:

- Query all the other nodes on the network for the physical address corresponding to an IP address.
- Encapsulate the IP datagram in a physical frame containing that physical address.
- Send the encapsulated IP datagram directly to the destination node’s physical address on the network.

When an IP datagram is sent to a node on other networks, the network portions of the originating IP address and the destination address are different. The sending node recognizes this difference and sends the packet to the router that connects the originating network with another. Two networks can be connected only if one router is attached to both networks and can pass data in a form that is compatible with both networks as shown in Figure 4-3~~Figure 4-3~~.

The sending node has a table of IP addresses for one or more workstations, computers, database servers, or IEDs on the network that serve as routers to other networks. It looks for the IP address of a router in its table and broadcasts an Address Resolution Protocol (ARP) request to the router from the physical address of the router. It then sends the packet containing the IP datagram to the router’s physical address. When the router receives the IP datagram, it uses the IP address in the datagram to send the packets to its final destination in a similar manner. If necessary, the router sends the packet to the address of another router that can route the packet to its final destination.

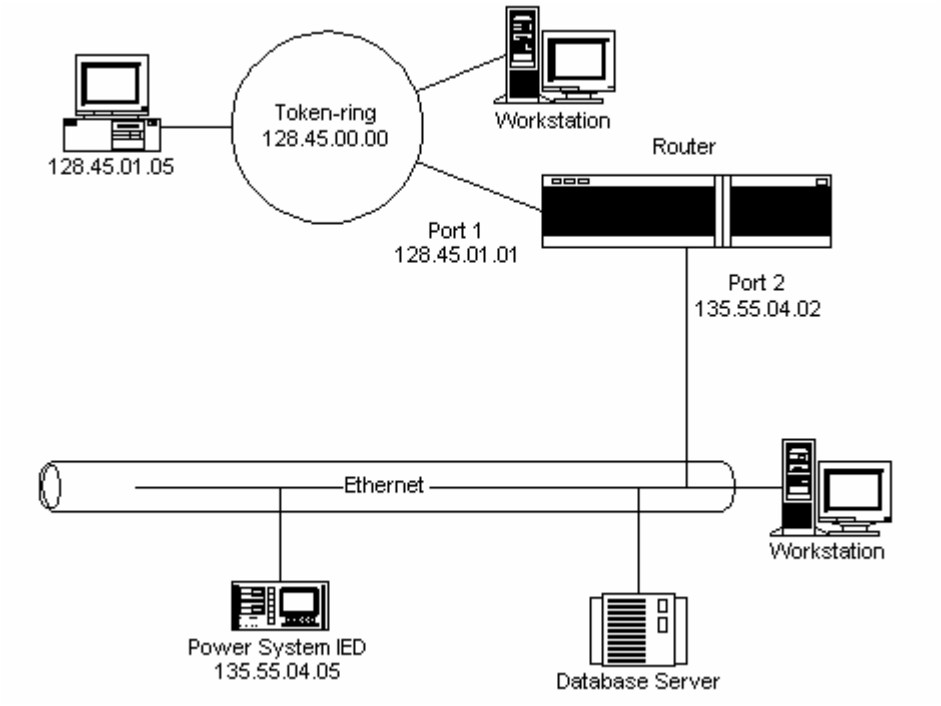

Figure 4-3 Data exchange via a router

5 Network management and application development tools

Figure 5-1~~Figure 5-1~~ shows the network management and application development tools needed to implement C37.115 test methods[5]. Network switches operate at the data link and network layers of the OSI model. The router performs no modification of network information. The router only monitors network addresses to know if inbound traffic is bound for some other LAN segment. A link layer (level 2) switch, or a network (level 3) switch (router), is used to provide a dedicated high-speed link between LAN segments, and to allow each network segment to function as a standalone network[6].

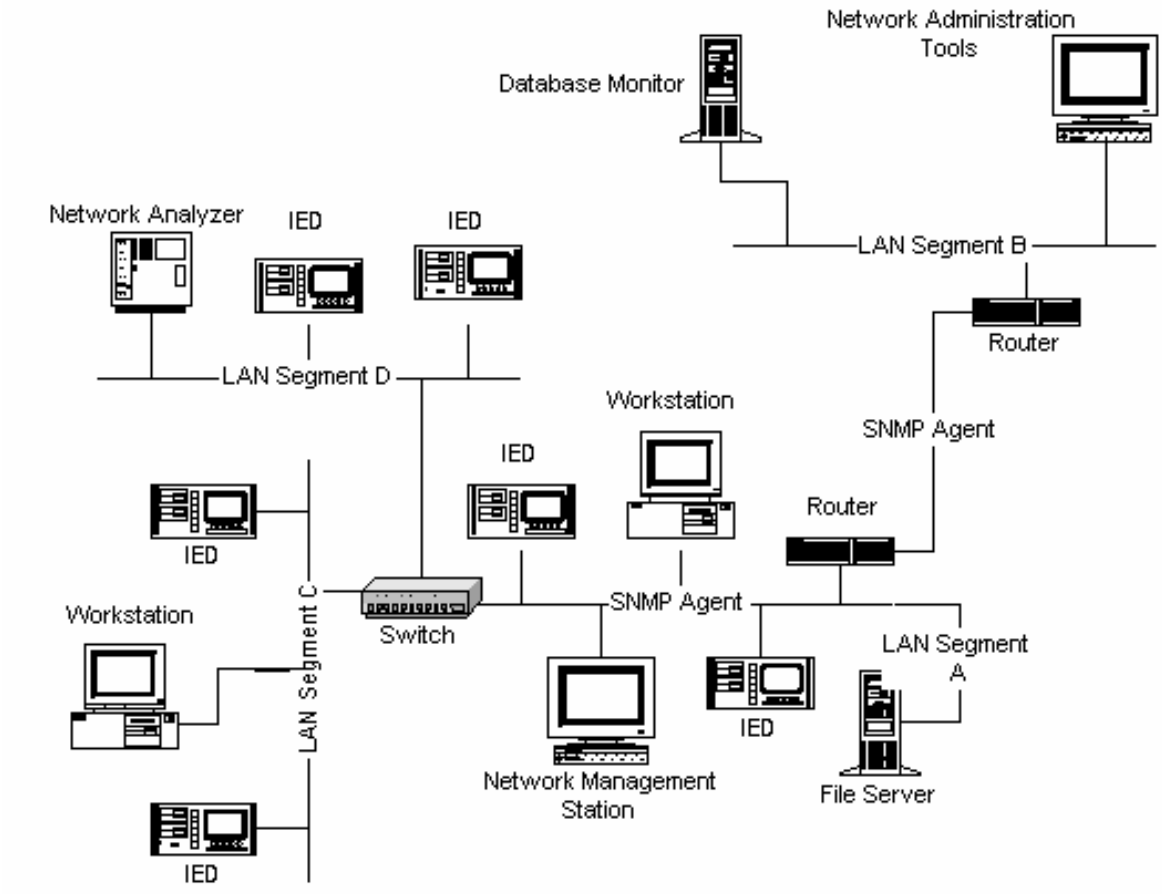

Figure 5-1 Network management and application development tools

[5] The location of routers is shown as an example. Each substation LAN architecture needs to be evaluated to determine whether a switch or a router should be used to connect LAN segments. If protocol conversion is required to gracefully migrate legacy systems into a peer-to-peer architecture, then they need to be used to connect LAN segments.

[6] Some level 3 switches will support multiple protocols, bridge between the WAN and LAN segments, provide hundreds of configurable parameters, and use sophisticated packet filtering. The switch capability required will depend on the communication protocol and capabilities designed into the IEDs on each LAN segment.

The routers shown in Figure 5-1~~Figure 5-1~~ must be “intelligent” routers in the sense that they are required to handle multicast messages, and which pass from one network segment to another. The information is exchanged using only logical address information, and functions only at the network layer of the OSI model.

Although four LAN segments are shown in Figure 5-1~~Figure 5-1~~, the Substation LAN is one “Logical” Substation LAN comprised of four segments.

5.1.1 Network analyzer

Network analyzers shall be used to capture packets or frames of information flowing through the network. A packet (or frame) typically includes three fundamental types of information: source and destination addresses, data, and control bits. Different network protocols have different packet formats that the network analyzer must be able to recognize. Analyzer features required by C37.115 shall include:

- Packet capture, including address filtering, which ensures that only packets with specific source or destination addresses are captured[7].
- Packet decode, which is specific to each protocol. The level of decoding varies from simple decoding of packet type and address to sophisticated decoding, which interprets the data portion of the packet for commands, such as file open or file read.
- Packet playback or generation, which transmits packets from the analyzer onto the network.
- Other functions such as graphical displays, current and trend statistics, and programmable operations, which assist the user in displaying and interpreting, captured data.

5.1.2 Network management stations

Network management stations shall be used to provide graphical representation of the network configuration, and to enable the network engineer to monitor the network and collect utilization statistics, alerts, and other pertinent information from all network nodes.

The network management stations shall be based on Standard Network Management Protocol (SNMP) standards, and SNMP agents running on the network nodes shall collect and summarize statistics that are sent to the management station’s management information base (MIB)[8].

Vendor supplied resource management and test configuration tools shall be used to reconfigure the test configuration of the network.

5.1.3 Monitoring software

Vendor supplied resource monitoring software running on the individual network node shall be used to collect intranodal information; e.g., cache hits, send/receive buffer utilization, memory utilization, and file open/reads/writes.

---

[7] Network analyzers capture and copy packet header information without degrading network performance. This is non-intrusive testing.

[8] This capability provides passive monitoring of the network to detect network failures or alerts. Alerts arise when network or nodal thresholds set on the network management station are exceeded. These thresholds, such as network bandwidth utilization or dropped packets per second by a router, measure characteristics of the network that may indicate a potential problem.

The monitoring software shall be used to confirm that a baseline test load is representative of the real-world operating load (it creates the same level of activity for key system parameters) on a comparably configured node. It shall also be used to measure the impact on that particular node as the test load is increased to reflect more IEDs or network activity.

5.1.4 Server-based databases

Server-based databases shall be used for monitoring the resource utilization of the database. Similar to monitoring software running on a server, database utilization, vendor supplied software shall be used to collect statistics pertinent to the database, such as cache utilization, file read/writes, connected IEDs, average response time for database queries, and average record size.

6 Communication test method

This standard defines communication modeling, terminology, evaluation criteria and performance measures for communication test scenarios, which specify messages to be exchanged between electrical power substation IEDs. These scenarios also define message transactions between substation IEDs and remotely located applications.

Clause 4 first described the network structures addressed in this standard. Clause 5 then described the network management and application development tools needed to implement C37.115 test methods. This clause defines how these structures shall be tested.

Implementing the communication between IEDs in an integrated substation protection, control and data acquisition system requires a specification of the communication interfaces and the performance requirements associated with each interface. IEEE C37.115 incorporates all specifications of IEEE P1525 [A45], and the scenarios described in the informative annexes use extensions of the models described in IEEE P1525 [A45].

This communication test method must be tailored by the utility or system integrator to include conformance testing, interoperability testing, operability testing and performance testing that implements the operating procedures of the using utility[9].

6.1 Functional and performance requirements

Functional requirements and performance requirements allocated to the communication IEDs specified in IEEE P1525 [A45] could be implemented on a single network using one communication protocol. This will not be true in the near term because of the need to transition from legacy systems[10]. It may not be true in the long term because of the performance requirements; an operational policy to retain separation of organizational responsibility between the control network, the protection network and the information network, or because of different IED interface costs.

For these reasons, the requirements will probably be implemented on sub-networks connected to form the utility enterprise network. A complex network may have multiple bridges, routers, and other components. The correct component for each internetworking location is determined by the amount of network traffic, distance of LAN segments, and network topologies involved. For example, when it is necessary to translate one network protocol into a different network protocol, a gateway is used. A router may be used to connect networks to take care of compatibility issues. A bridge can be used to filter data for security and traffic isolation.

6.1.1 Conformance testing

The formal testing methodology specified in IEC/ISO 9646 shall be used for all conformance testing. IEC 61850-10 also defines conformance-testing requirements.

---

[9] CIGRE 35.07 report provided a basic framework for developing the detailed specifications in this standard.

[10] Legacy systems are information systems built on older technology that is not very adaptable or scalable to integrate new microprocessor-based intelligent electronic devices.

Part of conformance testing shall check that the underlying transport mechanism is error-free. This shall be determined by testing the “pipe” through physical layer tests such as bit error rate (BER), power, signal level, and pulse shape, as well as timing tests to ensure they conform to specified values for fiber, copper or radio transmission links.

Both static conformance tests, which check the list of features and options claimed to have been implemented, and dynamic conformance tests, which check the communication system under load conditions, shall be performed.

Tests shall be designed to load the communication system with background traffic so as to test the communication system when it is moderately busy[11].

6.1.2 Interoperability testing framework

A suitable set of tests for checking the interoperability of communication systems shall be performed to verify the data link connectivity. These include:

- Address translation and mapping table functionality.
- Protocol translation verification.
- Protocol encapsulation and de-capsulation.
- Congestion indication translation.
- Connection status management.

6.1.3 IP testing

The essential element of IP networks is the provision of end-to-end Quality of Service (QoS) for user traffic over one or more networks. The key network quality parameters that affect IP QoS over the IP networks are packet delay, packet loss, and delay variation (jitter). Tests shall be performed to monitor the protocol exchange between terminal, gateway, and gatekeeper so as to address the following interoperability problems:

- Routing table errors.
- Failure of session connection negotiation due to terminal capabilities mismatch.
- Configuration errors.

6.1.4 Operability tests

Emulating message traffic and data streaming traffic to ensure that proper signal routing occurs without errors shall test functionality of the communication switches and hubs. ~~Functionality of the communication switches and hubs shall be tested by emulating message traffic and data streaming traffic to ensure that the proper signal routing occurs without errors.~~ Specific tests shall include virtual path identifier and virtual channel handling, switch or hub response to header error control, and verifying proper alarm generation and response.

~~When checking the performance of the communication system, special test equipment shall be used to emulate message traffic and data streaming traffic to verify proper routing through the communication switches and hubs. Special test equipment shall also be used to identify a~~All communication traffic at the switch or hub output shall be

[11] The worst-case scenario should be evaluated by analysis using simulation tools. It is not practical to design a test procedure for the worst-case scenario.

scanned to preview ~~by scanning the signal and previewing~~ all addresses that are active and storing them for use[12].

6.2 Test concepts and requirements

The interoperability test method defines the following:

- Test concepts and requirements, performance measures for testing the device(s)-under-test (DUT).
- Evaluation requirements for determining whether or not the DUT performed its communication functions correctly.
- Information that should be provided in the procurement specification.
- Information that should be provided by the device vendor.

Interoperability tests shall be conducted to evaluate product communication capability. Testing to evaluate the product for its performance, as a power system device, is not within the scope of this standard.

While a single set of communication interoperability tests applied to all products might seem an ideal approach to testing power system devices, the wide range of products required for substation automation precludes the application of a blanket test set. However, there are sets of services required for all products and those services will be tested in all cases. Beyond the required services, interoperability testing should be customized for each product based on the services identified in the Protocol Implementation Conformance Statement (PICS), Protocol Implementation Extra Information for Testing (PIXIT) and Model Implementation Conformance Specification (MICS) provided by the vendor.

True interoperability testing requires that each vendor's product be tested with complementary and/or competitive products from all other vendors. The problem is that complementary or competitive products will not always be available for testing. Therefore, the test concept must provide a test environment simulating the target environment using products from the vendors of the communications software. Given these initial conditions, products developed to operate in the role of a Client will be tested with simulated servers to verify that they perform the required messaging functions in that role (see Figure 6-1~~Figure 6-1~~).

Products developed to operate in the role of a Server will be tested with simulated Clients to verify that they perform the required messaging functions in that role.

Products developed to operate as a Server under specified conditions, and as a Client under other specified conditions would be tested with simulated Clients and simulated Servers that emulate the specified conditions of each.

6.2.1 Performance measures

All performance measures shall be defined in accordance with the procurement specification described in Clause 6.2.3. In general, the following communication times shall be measured:

- Message delivery time shall be measured in terms of the time required to send a message from the sending application interface to its receipt at the receiver application

[12] The scan identifies mapping problems up front, providing easy analysis.

interface (this is the sum of a + b + c described in Figure 6-2~~Figure 6-2~~ discussed in the next paragraph).

- Functional performance shall measure the time required to communicate, from the sender to the receiver, the information needed to execute the specified function, plus the time to execute the function.

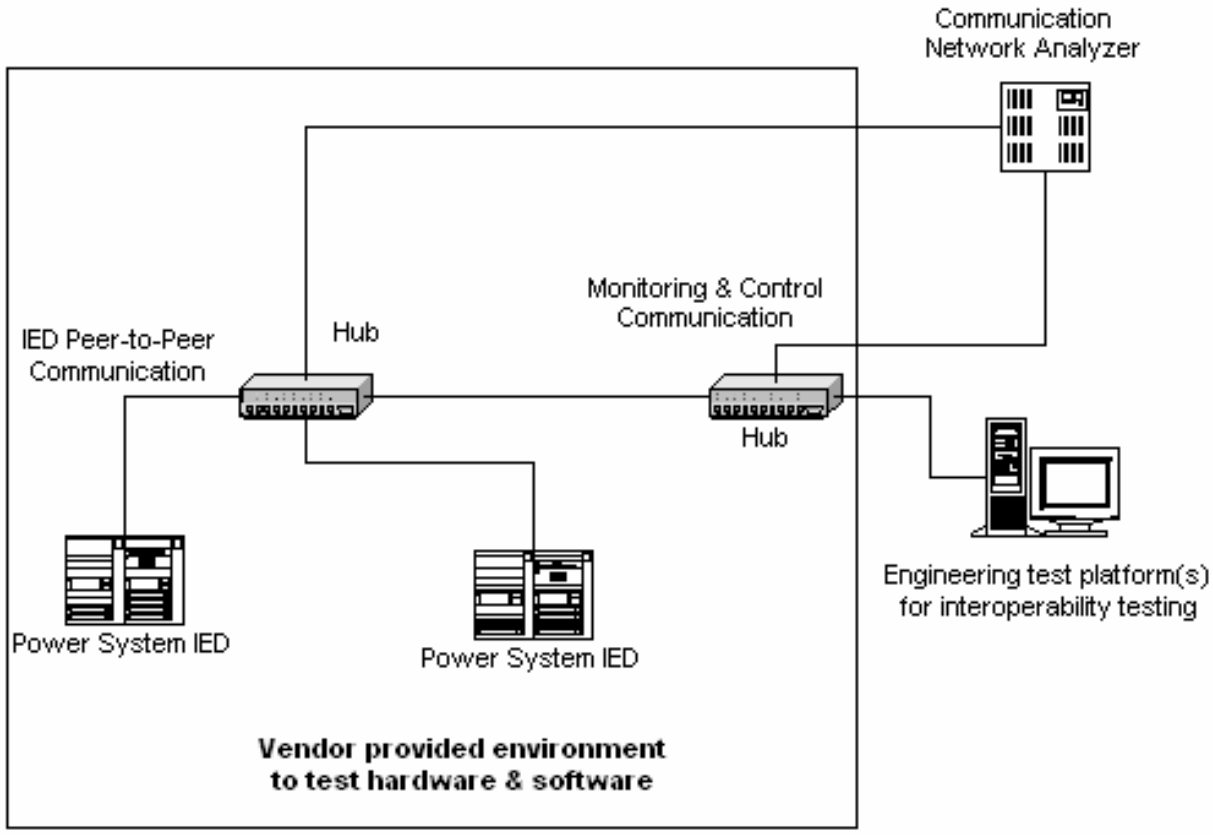

Figure 6-1 Test configuration

Response time is specified in terms of the time when the message leaves the sending IED application to the time when the receiving application gets the message. Figure 6-2~~Figure 6-2~~ shows time components that define the time requirement. Application-to-application time is defined as the sum of the times required for the sending IED communication processor to accept the data from the sending application and exit the output queue of the sender “a”, plus time over the communication network (including processor time required by routers, bridges, gateways, etc.) “b”, plus the time required for the receiving IED communication processor to extract the message content and present it to the receiving application “c”. Time requirements are specified in IEEE P1525 [A45].

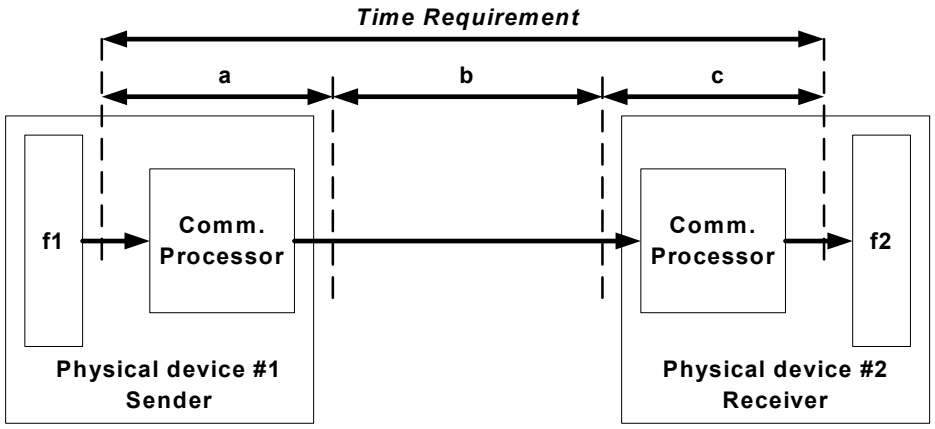

Figure 6-2 Application-to-application communication times

6.2.1.1 Application response time testing

Application response time testing shall be used to measure how long it takes an application to complete a series of tasks, and best represents the utility's perception of the network system (application network operating system and network components).

Tests shall measure how long it takes to switch between different applications tasks or to load new software overlays.

Tests shall be run at various loads, numbers of real or emulated IEDs, to create a load versus response time curve for each application tested.

Application tests shall use a series of commands that execute typical network activity, such as file opens, reads, writes, searches, and closes to provide an representative load model. The time it takes to complete commands shall be measured for each workstation or IED running under test.

Response time testing shall include monitoring the system for reliability. A reliability problem, such as a high number of dropped packets at a router or server, or a high number of bad packets because of a malfunctioning network component, can significantly impact response time measurements.

Network analyzers shall be used to monitor the system for errors during testing.

6.2.1.2 Application feature/functional testing

Feature testing shall be used to verify individual commands and capabilities of the application. Feature testing shall be performed with minimal to light loads to measure the IED's interface and application operations or transactions invoked by the client IED.

Functional testing shall be used to verify that the application's multi-IED characteristics and background functions work correctly under heavy loads. Functional testing shall be performed under loading that closely models the substation's real-world operating environment.

6.2.1.3 Regression testing

Regression testing is not one test, but a series of tests that measure critical aspects of the IEDs and communication network. A regression test plan identifies which basic test objectives should be run against each new product release.

Regression testing shall be used to compare the performance, reliability, and functionality of a new release of hardware or software to the current release to ensure that the product upgrades will not adversely impact the operational network.

6.2.1.4 Throughput testing

Throughput testing shall be used to measure data transfer rates (e.g., kilobits per second or packets per second) to evaluate performance, find bottlenecks, compare different products, and size individual components of the network.

6.2.1.5 Acceptance testing

~~Acceptance testing is a shake down of the system prior to commissioning the substation. It is an excellent method of guaranteeing that the new system will be stable and provide acceptable performance in its initial release.~~To ensure that the new system will be stable and provide acceptable performance in its initial release, acceptance testing is used to verify proper system operation prior to commissioning the substation.

Acceptance testing of the communication network is conducted using real-world test configuration and emulated load. Acceptance testing is not conducted on an IED, or group of IEDs, because this is not a final target configuration. Like regression testing, acceptance testing does not have a single objective, but a combination of one or more test objectives.

Acceptance testing (including response time, reliability, and feature/functionality tests) shall be used to ensure that the new system is stable and provides acceptable performance in its initial release. Acceptance testing shall comply with the following, adapted from IEEE Standard C37.1-1994 [A.46], Clauses 9.1.2, 9.1.3 and Table 13:

- Factory tests and inspections include tests conducted at a system integrator's facility on a multi-vendor system. Included are the inspection and approval of interface drawings prior to staging of the system, and all functional tests and inspections on the actual system to be supplied to the user prior to the shipment of the system from the supplier's facilities. The factory tests shall be a highly structured procedure designed to demonstrate as completely as possible that the system will perform correctly and reliably in its intended application. Factory tests and inspections include functional tests of I/O point checkout, communications, user interface, and special functions. Also included are performance tests and inspections of loading, data acquisition, control, and user interface. Computer & disc stability, maintainability, and expandability tests are optional.
- Field tests and inspections are performed on the system after it has been shipped from the supplier's facilities. These include pre-installation inspections and tests to ensure the equipment has not been damaged during shipment and post installation tests to verify the system performs its functions reliably and correctly. Field tests and inspections include I/O point checkout, communications, user interface, and special functions. Availability tests are optional.

6.2.1.6 Configuration sizing

Results from application response or throughput tests shall be used to size network components by evaluating load versus response time for alternative configurations. A target response time shall be specified by the Utility to select the configuration that

provides the best cost/performance margin for the maximum number of IEDs on the network.

6.2.1.7 Reliability testing

Reliability tests shall be run for an extended period of time (24 to 72 hours), under medium to heavy load to monitor the network for errors and failures. ~~Reliability testing forces the DUT or the communication network to handle in a compressed time period the activity it would normally experience over weeks, months, or years on a production network.~~ Reliability testing attempts to accelerate failure of the IED communication processors or other communication network devices caused by:

- Cumulative errors that result from repeating an operation multiple times in a fashion that results in an error.
- Timing errors caused by two time-dependent operations that occur out of sequence or without the proper delay.
- Statistical errors resulting from the probability that a highly unusual error condition will occur or a seldom-invoked sequence of events will occur[13].

Tests shall be performed to check the reliability of the communication network's ability to handle faulty traffic. Graceful recovery from a loss of signal, loss of synchronization, or discarded traffic shall be included in these tests to show that hardware and software failures do not corrupt persistent data.

~~Reliability tests shall be run for an extended period of time (24 to 72 hours), under medium to heavy load to monitor the network for errors and failures.~~

6.2.1.8 Bottleneck identification and problem isolation

Maximum sustainable throughput on ~~each~~ the system under test ~~component~~ shall be measured or calculated. Tests shall then be run on individual components of the system under test to determine their capacity limits. The difference between the maximum capacity of individual components and the maximum sustainable throughput of the system shall be used to determine where system bottlenecks and excess capacity exist.[A.7]

6.2.2 Evaluation requirements

All performance measures shall be defined in accordance with the procurement specification described in Clause 6.2.3. In general, the following communication performance shall be reported and evaluated:

- An evaluation of the communication time between sender application and receiver application.
- The repeatability of the DUT to perform its communication function.
- The types of failures that DUT contained.
- An evaluation of how the DUT handled each failure.

[13] In developing and testing a complex substation automation system, it is virtually impossible to test and verify every possible path through the software code. Reliability testing increases the probability that a statistical error will occur.

6.2.3 Procurement specification

The procurement specification issued by the buyer shall define the Substation Automation System communication capabilities required to operate in a defined environment. The procurement specification will define the following:

- Communication protocols required on the device for each specified communication network that will be connected to the device.
- Communication network resources available for configuring and operating the device.
- Operating constraints and procedures required configuring and operating the device over specified communication networks.
- Configuration diagrams showing the Substation Automation System communication networks and devices, including the operational, non-operational and remote access data paths from the substation to the utility enterprise.

6.2.4 Vendor requirements

Vendors are expected to build their Substation Automation System and/or device product in accordance with the specifications defined in the procurement specification.

When submitting device products for testing, vendors are required to supply the following items:

- A sample product for testing.
- A Protocol Implementation Conformance Statement (PICS) and Protocol Implementation Extra Information for Testing (PIXIT) statement detailing the standard protocol services supported by the product.
- A Model Implementation Conformance Specification (MICS) detailing the object model supported by the product.
- Instruction manuals detailing the operation of the product and installation specifications.
- Vendor test configuration hardware to support a simulated target-operating environment (example: power supplies, sensors).
- Vendor test configuration software to support a simulated target-operating environment.

When submitting Substation Automation System products for testing, vendors are required to supply the following, adapted from IEEE Standard C37.1-1994 [A.46], Clauses 10, 10.1, 10.2, 10.3, 10.4 and 10.5:

- Functional and design documentation, including system architecture overview, list of deliverables, system component (e.g., data concentrator, host processor, software applications, device interface modules) documentation, I/O requirements (e.g., historical data points list, SCADA points list, individual device points lists), technical requirements (e.g., contract table of compliance, vendor system technical requirements), and implementation (e.g., implementation plan, project overview, implementation team).
- Installation documentation including interface wiring procedures, equipment mounting methods, safety precautions or guards, grounding or bonding procedures, clearances for access and ventilation, testing and alignment methods, environmental procedures, and any other procedures needed to properly install the equipment.
- Operating instructions and records, including vendor-operating instructions, user-operating instructions, and records to support the availability and reliability of the system.
- Maintenance instructions and records, including performance information, preventive maintenance instructions, corrective maintenance instructions, and parts information.

- Test documentation, including a system test plan, test procedures, and certified test reports as required.

A Bibliography (informative)

A.1 Modeling tools

There are several commercial tool kits to model all aspects of communication networks and distributed processing. Below is a list of the tools that were used a basis for the technical descriptions used in this document.

[A.1] OSS ASN.1 Tools for Standards Editors: A family of tools for compiling ASN.1 data objects. Open Systems Solutions, Inc.

[A.2] Togethersoft: A family of object modeling tools used to develop the object models and transaction diagrams. Object International, Inc.

A.1 Text books

There is an extensive literature on all aspects of communication networks and distributed processing. Below is a list of some literature that was used a basis for the technical descriptions used in this document.

[A.3] Atchison, Lee: "Object-Oriented Test & Measurement Software Development in C++", 1997, Prentice-Hall, Inc.

[A.4] Black, Uyless: "OSI, A Model for Computer Communication Standards", 1991, Prentice-Hall, Inc.

[A.5] Berson, Alex: "Client/Server Architecture", 1992, McGraw-Hill, Inc.

[A.6] Breyer, Robert and Riley, Sean: "Switched and Fast Ethernet", 2nd Edition, 1996, Ziff-Davis Press.

[A.7] Buchanan, Jr., Robert W.: "The Art of Testing Network Systems", 1996, John Wiley & Sons, Inc.

[A.8] Coad, Peter: "Object Models: Strategies, Patterns, and Applications", 1995, Prentice-Hall, Inc.

[A.9] Comer, Douglas E. and Stevens, David L.: "Internetworking with TCP/IP, Volume II, Design Implementation and Internals", 1994, Prentice-Hall, Inc.

[A.10] Doraswamy, Naganand and Harkins, Dan: "IPSec, The new Security Standard for the Internet, Intranets, and Virtual Private Networks", 1999, Prentice-Hall, Inc.

[A.11] Douglass, Bruce Powell: "Real-Time UML, Developing Efficient Objects for Embedded Systems", 1997, Addison-Wesley Longman, Inc.

[A.12] Dubuisson, Olivier: "ASN.1 Communication Between Heterogeneous Systems", 2001 Academic Press.

[A.13] Edwards, Jeri: "3-Tier Client/Server at Work", 1997, John Wiley & Sons, Inc.

[A.14] Firesimith, Donald G.: "Object-Oriented Requirements Analysis and Logical Design", 1993, John Wiley & Sons, Inc.

[A.15] Forouzan, Behrouz: "TCP/I Protocol Suite", 2000, McGraw-Hill Higher Education.

[A.16] Fowler, Martin: "UML Distilled, Applying the Standard Object Modeling Language", 7th printing, 1998, Addison-Wesley Longman, Inc.

[A.17] Ganti, Narsim and Brayman, William: "The Transition of Legacy Systems to a Distributed Architecture", 1995, John Wiley & Sons, Inc.

[A.18] Graham, Buck: "TCP/IP Addressing – Designing and Optimizing your IP Addressing Scheme, 1997, Academic Press.

[A.19] Graham, Rick, Moote, Robert and Cyliax, Ingo: "Real-time Programming", 1998, Addison Wesley Publishing Company.

[A.20] Johnson, Howard W.: "Fast Ethernet, Dawn of a New Network", 1996, Prentice-Hall, Inc.

[A.21] Kee, Eddie: "Networking illustrated", 1994, Que Corporation.

[A.22] Loshin, Pete: "TCP/IP Clearly Explained, 2nd Edition, 1997, Academic Press.

[A.23] Maufer, Thomas: "Deploying IP Multicast in the Enterprise", 1998, Prentice Hall PTR.

[A.24] Northcutt, Stephen: "Network Intrusion Detection – An Analyst's Handbook", 1999, New Riders Publishing.

[A.25] Orfali, Robert & Harkey, Dan & Edwards, Jeri: "The Essential Distributed Object Survival Guide", 1996, John Wiley and Sons.

[A.26] Piscitello, David M. & Chapin, A. Lyman: "Open Systems Networking", 1993, Addison-Wesley Publishing Company.

[A.27] Roberts, David: Internet Protocols Handbook", 1996, The Coriolis Group.

[A.28] Stallings, William: "Handbook of Computer Communication Standards – The Open Systems (OSI) Model and OSI-Related Standards", Volume 1, 2nd Edition, 1990, Macmillan Publishing Company.

[A.29] Stallings, William: "Handbook of Computer Communication Standards – Local Area Network Standards", Volume 2, 2nd Edition, 1990, Macmillan Publishing Company.

[A.30] Tanenbaum, Andrew S.: "Computer Networks", 3rd Edition, 1996, Prentice-Hall, Inc.

[A.31] Theriault, Marlene & Heney, William: "Oracle Security", 1998, O'Reilly & Associates, Inc.

[A.32] Tsai, Thomas C.: "A Network of Objects", 1995, Van Nostrand Reinhold.

[A.33] Udupa, Divakara K.: "Network Management System Essentials", 1996, McGraw-Hill, Inc.

A.2 EPRI reports

[A.34] RP3599: "Substation Integrated Protection, Control and Data Acquisition, Phase 1, Task 2, Requirements Specification", Preliminary Report, Version 1.2, February 16, 1998.

[A.35] TR 109474: "Substation Integrated Protection, Control and Data Acquisition, UCA®, Interoperability Test Specification for the Utility Substation Demonstration Initiative", Draft Version 1.0, September 1, 1998.

[A.36] UCA® 2.0 – CASM: "Common Application Service Models (CASM) and Mapping to MMS, Editorial Draft 1.5, September 1, 1998.

[A.37] UCA®® 2.0 – Profiles: Draft Input for the Utility Communications Architecture Version 2.0, Editorial Draft 1.0R, undated.

[A.38] UCA® 2.0 – GOMSFE: "Generic Object Models for Substation & Feeder Equipment (GOMSFE), Version .91, May 2000.

A.3 CIGRE reports

[A.39] CIGRE WG 34.03: "Communication Requirements in Terms of Data Flow Within Substations", August 23,1996.

[A.40] CIGRE WG 34.07: “The automation of new and existing substations: why and how”, in work.

[A.41] CIGRE WG 35.07: Draft Technical Brochure – “The Use of IP Technology in the Power Utility Environment”, December 1999.

A.4 IEEE technical reports, papers, standards, and draft standards

[A.42] IEEE 100, The Authoritative Dictionary of IEEE Standard Terms, Seventh Edition.

[A.43] IEEE-SA TR 1550-1999: “Utility Communications Architecture (UCA®) Version 2.0”. [GOMSFE version .82 is included in this report; the version listed under EPRI reports is a later version.]

[A.44] IEEE Report: “Application of Peer-to-Peer Communications for Protective Relaying” December 22, 2000.

[A.45] IEEE STD P1525, 2000, Draft IEEE Standard for Substation Integrated Protection, Control, and Data Acquisition Communications.

[A.46] IEEE C37.1-1994, Standard for Definition, Specification, and Analysis of Systems Used for Supervisory Control, Data Acquisition, and Automatic Control.

A.5 IEC/ISO specifications

[A.47] IEC 61850: “Communication networks and systems in substations”, to be published.

[A.48] IEC 61968: “System Interfaces for Distribution Management”, to be published.

[A.49] IEC 61970: “Energy management system application program interface”, to be published.

B Synch check to close breaker (informative)

The application uses a high performance substation LAN for the transmission of synch check voltage information to the synch check relay IED, and the transmission of the “Synch Close” message to the breaker IED.

Absent a LAN, all possible sources of bus voltage must be wired into each synch check relay. Assuming redundant bus VTs, and that the center breaker in a breaker and a half substation is the one to be closed, four sets of bus voltages would have to be wired to the synch check relay. In a ring bus, the possibilities are even greater for multiple VT sources. In either case, without a substation host, it would be difficult for the synch check relay to determine which bus voltage source to use for synch check information. Further, a separate synch check relay would be required for each breaker to be synch check closed.

With the LAN approach, one or two (for redundancy) synch check relay IEDs can serve the entire station. The benefits are simplified wiring to synch check relays, and added flexibility in selecting bus side VTs.

B.1 Performance requirements

Time tag of positive going voltage waveform zero crossings to the nearest 100 microseconds and to a common time reference (100 microseconds corresponds to approximately 2° on a 60 hertz sine wave).

Note: This requires the clock in the VT IED to be set to ± 10 μs, which may require a separate timing wire from the substation master clock to the VT IEDs.

B.2 Evaluation criteria

Evaluation requires:

- The substation host determines the proper VTs to be used.
- The VT IEDs time tag the zero crossings of the two voltages (bus and line) to ± 0.1 ms, calculate the voltage and frequency information, and send the required messages.
- The synch check relay IED interprets the time tag messages and issues the required Synch Close message.
- The breaker control IED receives and properly responds to the required messages.

B.3 Functional configuration

An overview of the functional configuration is described first from a power system point-of-view. Next the communication configuration is described to identify both the WAN and LAN connectivity. Then the communication functions are allocated to specific IEDs.

B.3.1 Overview

The use of distributed voltage measurements for synch (synchronism) check over a substation LAN can only be achieved if a number of requirements are met. The hypothetical substation in question includes the following:

- An IED (Intelligent Electronic Device) hard wired[14] to each breaker for control (trip and close)
- Protection IEDs hard wired to their input CTs and VTs
- Measurement IEDs hard wired to bus VTs
- IED to IED communication is via a substation LAN (local area network)
- A LAN delivery time requirement, application to application, of 4 milliseconds or less
- A substation host with up-to-date topology information of the substation components

The synch check function is used to ensure that a breaker will not be closed if the power systems on both sides of the breaker are not already synchronized. The function measures the angle between single-phase voltages on each side (bus and line) of the breaker, and the slip rate, to determine if they are within the limits set. The function may be programmed to allow closing if there is either a dead bus or a dead line condition.

A utility's overall transmission system design and operating procedures define which breakers, when commanded to be closed, must first be subjected to synchronism check. This is pre-established and, in today's substations, the synch check relays are hard wired to the correct VTs and breakers.

B.3.2 Communication configuration

The high performance LAN in Figure B-1~~Figure B-1~~ is shown with only those IEDs necessary for the synch check of a particular breaker:

- A synch check relay IED.
- Bus and line VT IEDs (may actually be relaying IEDs).
- A breaker control IED.
- A substation host, with the current status of every breaker and disconnect switch, and a complete description of the substation configuration.
- An interface to a remote HMI (SCADA master).

Only the breaker IED and the two VTs needed for the following scenario are shown. In an actual substation, there will be many breaker, relay, and VT IEDs. Depending on which breaker is to be synch check closed and the substation topology, a unique pair of VT IEDs would be used for bus and line side voltages.

Now consider an operator initiated breaker close command. The breaker in question is one previously identified as requiring synch check before closing. The operator might be at the remote SCADA master or at a substation console. In either case, the command is sent to the substation host.

From its database, the substation host knows that the breaker requires synch check and the synch check parameters for that breaker. From its database of substation topology, it also knows which VTs should be used for the synch check function for the present topology. Assuming the operator's identity had already been validated, the substation host initiates the synch check closing by issuing an SBO "Select"[15] message to the breaker control IED.

---

[14] Direct metallic connection via individual wires

[15] Message names in quotation marks are intended to be generic.

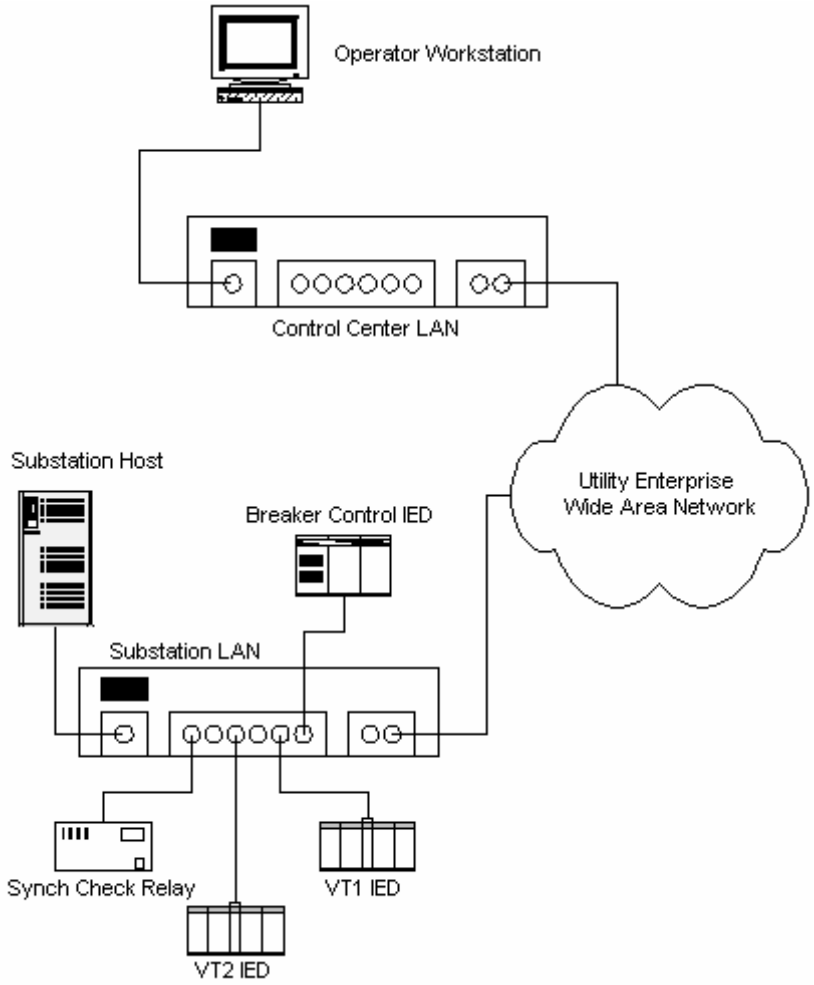

Figure B-1 Functional configuration

The breaker control IED receives the SBO “Select” message, confirms its validity, and sends a multicast message confirming its state change to “Selected”.

A state change multicast message is repeated for reliability, at geometrically increasing intervals out to a maximum of 1 minute. Thus, every IED in the substation will report its state at least every minute. Any state change will cause a new state change message to be multicast on the LAN. The substation host receives all these multicast messages to maintain its database.

As stated earlier, the multicast state change message is sent from the breaker IED indicating that it is in the “Selected” state. This is the first step in a Select Before Operate (SBO) sequence. In this state, the breaker is locked out from any other SBO operation, and will not respond to a close command from any other SBO source. If the “Select” operation had been initiated from a remote HMI, the substation host would forward the new state information to that HMI.

There is a timer associated with the “Select” state. If the next step in the SBO sequence is not received within the time set, the breaker IED deselects and sends a state change multicast message indicating it is in the deselected state.

The operator, in viewing his monitor, confirms that the selected breaker is the one to be closed and operates the “Close” button on the display. The substation host then issues a “Close Permissive”[16] command via the LAN to the breaker control IED. On receipt of the “Close Permissive” message, the breaker control IED issues a state change multicast message indicating it is in the Close Permissive” state.[17] The breaker control IED now awaits a “Synch Close” message from the synch check relay IED. The substation host also issues a multi-cast ”Initiate” message addressed to the synch check relay IED, and to the two VT IEDs.

The ‘Initiate” message contains:

- The ID[18] of the synch check relay IED and the synch check parameters to be used.
- The ID of the breaker control IED.
- The ID of the bus side VT IED.
- The ID of the line side VT IED.
- The refresh rate (number of times per second that the VT IEDs are to send their voltage information).

The synch check relay IED and the VT IEDs shown in Figure B-1~~Figure B-1~~ receive the “Initiate” message, confirm its validity, start their own functions, and send their own state change multicast messages confirming their state change. The VT IEDs begin periodic reporting of voltage data, and the synch check relay IED starts processing the voltage data.

A conventional synch check relay uses hard-wired voltage inputs from two sources. In a LAN environment, replication of the actual voltage waveforms would require very high data rates, and would flood the LAN. Thus, this scenario uses only selected data from the VT IEDs; i.e. data which only needs to be updated at longer intervals.

The periodic VT report includes the following digitized information:

- ID of the VT IED.
- Time tag of the measurement set.
- Voltage magnitude, RMS.
- Frequency.
- Rate of change of frequency.
- Positive going zero crossing time tag (to the nearest 0.1 millisecond).

The de facto standard for time tagging in most substation applications, such as sequence of events, is ± one millisecond. In a 60 Hz system, 1 ms = 21.6° so

---

[16] Closing is only permitted if the synch check conditions are satisfied.

[17] This is referred to as “end to end” SBO. The SBO may also be accomplished in two steps. The first step is entirely between the operator and his GUI. Only after the operator operates the “Close” button does the Select Before Operate sequence begin with the substation.

[18] ID is the substation communication address. Address naming techniques are not addressed in this paper.

measurement of the same event by two different IEDs could be off by 2 ms or ~ 43°. This is too great a variance for a synch check function that might be set for a permissible angular difference of only 10 or 15°. Thus, the requirement for zero crossing time tagging in this application is 0.1 ms.

The synch check relay IED receives the periodic data from the VT IEDs, and compares it to the downloaded parameters:

- From the voltage magnitudes, it determines dead bus or dead line conditions, or if the voltage difference is within the set point.
- From the frequency data, it determines if each source is within limits and the slip rate (difference between the two frequencies).
- From the zero crossing time tags, it calculates the angular difference between the voltages.

When all conditions are within limits, the synch check relay IED sends a “Synch Close” command to the breaker control IED. Given the performance requirement for the LAN, this command must arrive at the breaker control IED within 4 ms. Coupled with its “Close Permissive” state, on receipt of this message, the breaker IED closes the breaker.

Now the following actions take place:

1. The breaker IED sends a state change multicast message indicating that the breaker is closed, has completed the SBO sequence, and is now de-selected.

2. When the synch check relay IED receives the breaker state change message, it turns off the synch check function for that breaker and issues a state change multicast message to indicate its idle state.

3. The substation host receives the breaker state change message, and it updates its database to show the breaker is closed and de-selected.

4. The substation host sends a multicast message to the VT IEDs, ordering them to stop sending zero crossing information, and to reset to an idle Synch Check new state. Note that the VT IEDs, through the entire synch check process, may also be transmitting a different periodic report to other clients.

5. Other protection IEDs, not shown in Figure B-1~~Figure B-1~~, will receive the breaker state change message and update their relay logic to show this breaker is now closed. For example, in auto-reclosing schemes on ring bus or breaker and a half substations, one breaker is identified as the lead and the other the follower. When the first (lead) breaker closed, the follower would receive its state change message and its relay logic would close the follower breaker. State change multicast messages replace the wiring from breaker “a” and “b” contacts in relay logic and interlocking schemes.

Here one state change multicast message takes the place of two or more individual messages.

B.3.3 Allocation of functions

If the breaker control IED has sufficient capability (hardware and software), then the synch check function might have been loaded into that IED. In that case, the separate synch check relay IED shown in Figure B-1~~Figure B-1~~ would not be required. The scenario would begin with the Substation Host initiating the synch check function in the breaker control IED.

There may be cost implications in making certain that all the breaker control IEDs are capable of the synch check function. There is the additional issue of installing and maintaining the function in multiple breaker control IEDs vs. in only one or two (for redundancy) synch check relay IEDs.

B.4 Object model

Two object models describe the actors that participate in the synch check to close breaker scenario: Remote operation and Substation operation.

B.4.1 Remote operation

All remote operation of substation equipment is executed through the SubstationHost for the synch check scenario. Figure B-2~~Figure B-2~~ shows the interface between the ControlCenterWorkstation and AuthorizedOperator or AuthorizedEngineer. The ControlCenterWorkstation is a specialized RemoteController that include a WAN interface for communication to the SubstationHost.

The WAN interface to the substation may be implemented as a router (shown as a RouterSubstationLAN) if the same communication protocol is used over the WAN as used over the Substation LAN. If the communication protocol is not the same (the more likely case) then the interface is to the SubstationHost, which acts as a gateway for protocol conversion. For the purpose of this scenario the WAN protocol is the same as the LAN protocol. This is preferred because the need for protocol conversion is eliminated and the message load is reduced.

When an AuthorizedOperator or AuthorizedEngineer logs-on the RemoteController (a workstation), their password and authorized security keys to perform the scenario tasks are entered. These security parameters will be used later to verify their authorization to manage a specific action (an AuthorizedEngineer is responsible for system configuration, and an AuthorizedOperator is responsible for control).

A GraphicalUserInterface (GUI) provides both the AuthorizedOperator and the AuthorizedEngineer the capability to ‘point and select’ objects on the screen, ‘select’ software also activates a preset operation within the ControlCenterWorkstation.

Figure B-2~~Figure B-2~~ shows that any virtual device and any measurement unit (not shown) includes a directed association through the LAN interface for Control to the virtual device or measurement unit and Status and Data reported from these devices.

Status and data reported from substation devices are always logged in the SubstationHost database. However, if the LAN and WAN protocol are the same, the status and data can be multicast to both SubstationHost and RemoteController.

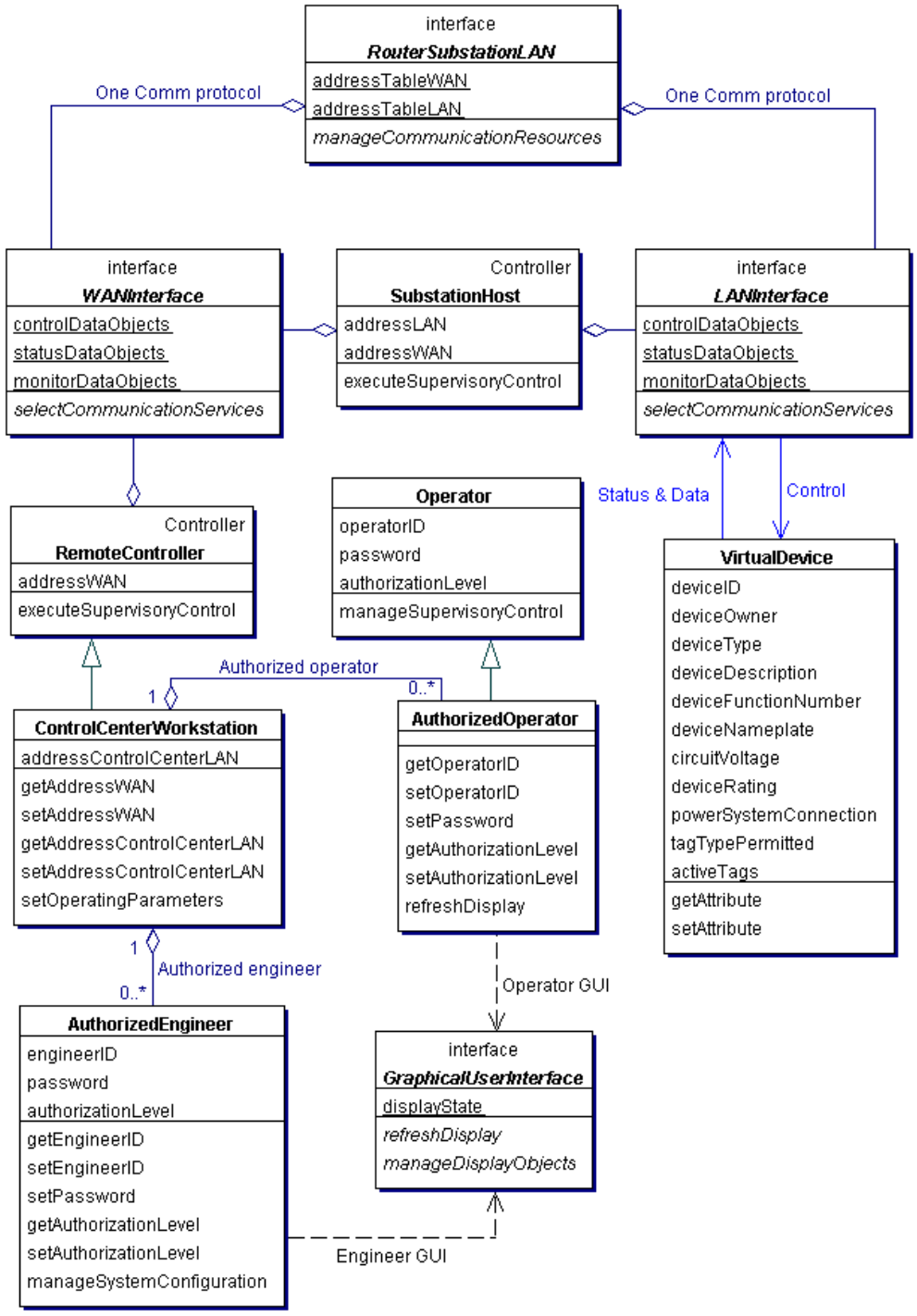

Figure B-2 Remote operation for synch check scenario

B.4.2 Substation operation

For clarity, component models are first described, then a Substation LAN model is described to show the communication relationship between the component IEDs. Control of any substation device (a specialization of VirtualDevice or a specialization of MeasurementUnit) is through the SubstationHost. Status and data is reported from the substation device to the SubstationHost over the Substation LAN, and simultaneously through a substation router over the WAN to the ControlCenterWorkstation, which is a specialization of the RemoteController.

B.4.2.1 Component models

Figure B-3~~Figure B-3~~ shows the VoltageTransformer object model, which defines the LineSideVT and the BusSideVT objects. These VTs are objects of the class VoltageTransformer, which is a specialization of the class InstrumentTransformer. InstrumentTransformer is a specialization of the class MeasurementUnit. All attributes and operations of parent classes are inherited by the subclasses; therefore, each VT object includes all its parent's attributes and operations.

VT_Controller is a specialization of the class EmbeddedDeviceController, which is a specialization of Controller. VT_Controller is defined as part-of the LineSideVT object and the BusSideVT object. EmbeddedDeviceController generally communicates control, and status & data, over the Substation LAN Interface, shown in Figure B-3~~Figure B-3~~ as part-of EmbeddedDeviceController. Specifically, the zero-crossing reports from the VTs are multicast over the LAN.

Figure B-4~~Figure B-4~~ shows the build-out of the VirtualDevice to define the Relay and Breaker IEDs. VirtualDevice defines the parent-class, which is specialized as Switch, then further specialized as a Breaker. Breaker inherits all attributes and behavior of its parent class-objects. Not shown (for clarity) is the EmbeddedDeviceController, which is as part-of class Breaker provides the IED its LAN communication capability.

Relay is also a specialization of VirtualDevice. RelayControlLogic shown as a part-of Relay is responsible for the ***magnitudeParameters*** and ***timeParameters***, which are needed to execute its trip operation. A further specialization of RelayControlLogic is the class SynchCheckRCL (RCL denotes Relay Control Logic). When instantiated, the SynchCheckRCL is responsible for calculating the single-phase angle between voltages and slip rate, and checking for a dead line or a dead bus. For clarity, not shown is the EmbeddedDeviceController, which as part-of class Relay provides the IED for LAN communication.

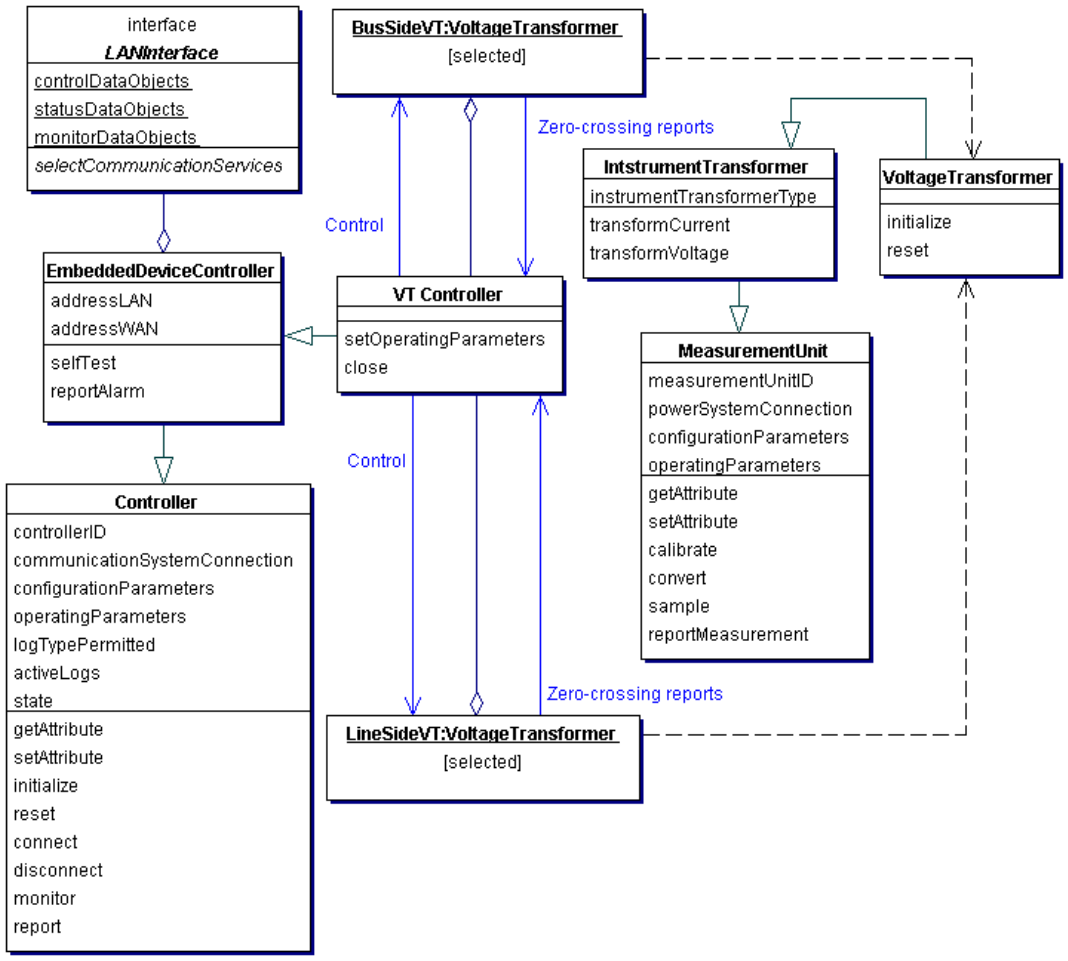

Figure B-3 Voltage transformer object model

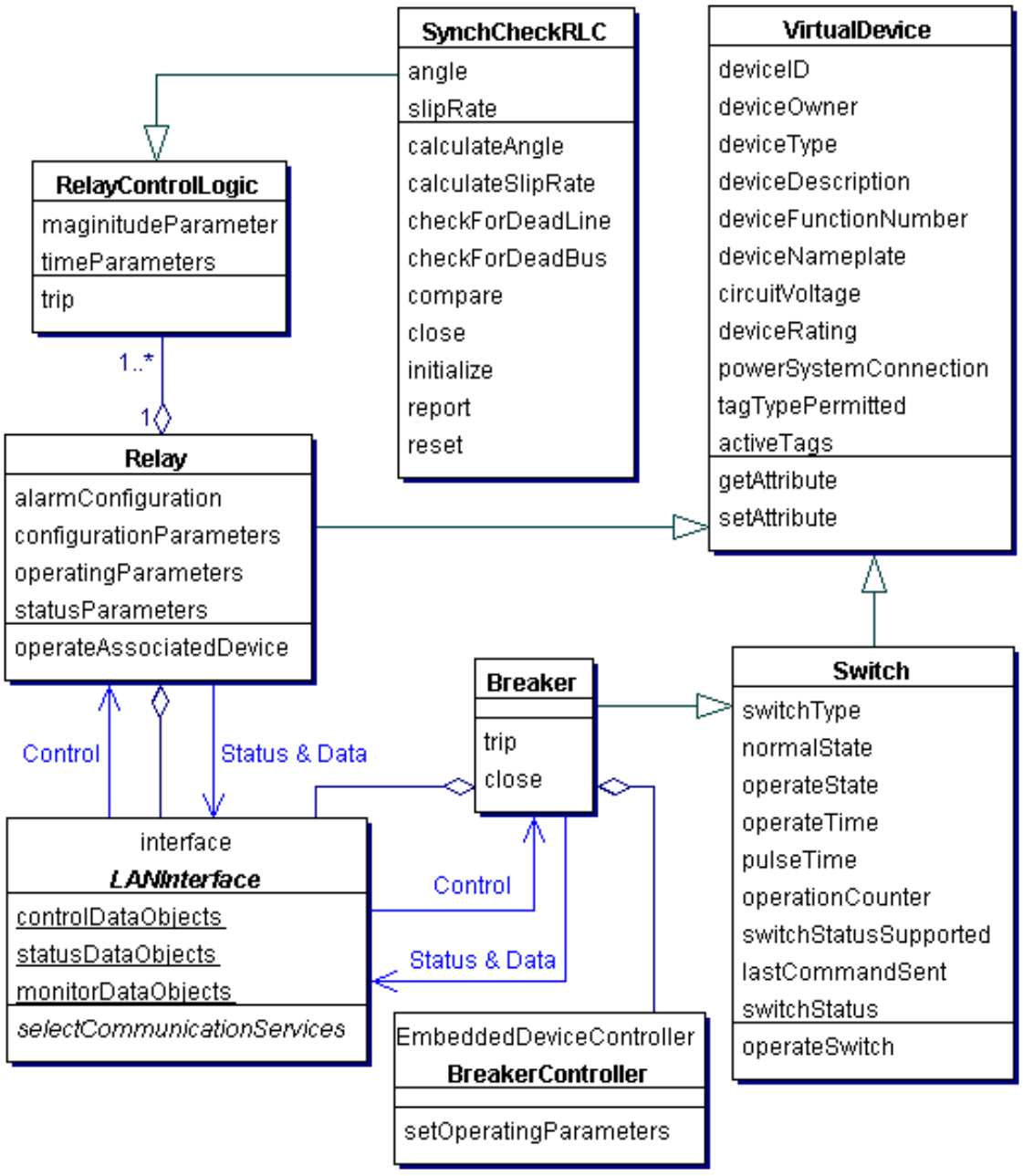

Figure B-4 Virtual device build-out for relay and breaker IEDs

B.4.2.2 LAN model

Figure B-5~~Figure B-5~~ shows the substation LAN operation for the synch check scenario. SubstationHost is required to provide an interface to the WAN, and to maintain all knowledge of the substation's topology, device operating parameters and configuration data. EmbeddedDeviceController is implicitly part-of each BusSideVT, LineSideVT,

Relay, and Breaker to provide the communication interface to the substation LAN for all IEDs.

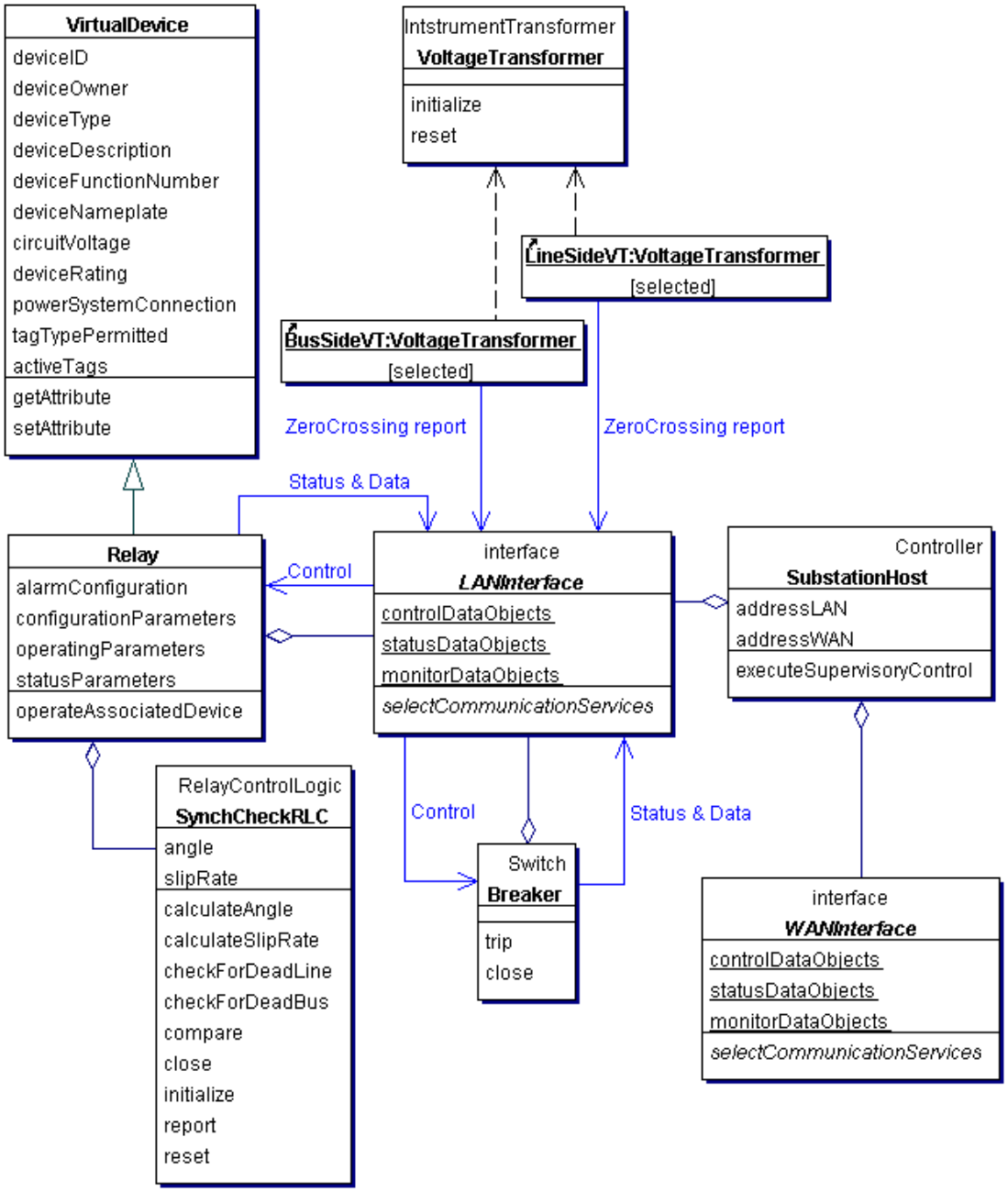

Figure B-5 Substation LAN operation for synch check scenario

The two instrument transformers IEDs (LineSideVT and BusSideVT) are selected by the SubstationHost to provide voltage measurements to the SynchCheckRCL IED to calculate the single-phase **angle** between the voltages and **slipRate**. The Breaker IED

is responsible for responding to the ***close*** command issued from the SynchCheckRCL IED to close the breaker.

B.5 Transaction sequences

Several transaction sequences are described to implement the synch check scenario. The first two sequences show the operator-initiated SBO transactions to select the breaker for close followed by enabling a permissive close.

Select-before-operate (SBO) is a two-step procedure. First, the device must be selected. Second, after confirmation that the device has been selected, the device can be operated. For the synch check scenario, operate is defined as an operator-initiated permissive close. SBO may be implemented in one of two ways: Local SBO and End-to-End SBO.

If Local SBO is implemented, both steps of SBO are processed within RemoteController before a message is sent to SelectedBreaker. Or if End-to-End SBO is implemented, then each step is completed with SelectedBreaker in the loop. For the synch check scenario, End-to-End SBO is used so that after SelectedBreaker receives and validates the ‘select’, all other operators are locked out of operating breaker close. All operator displays will be updated to show that the breaker is selected and cannot be operated except by the validated Operator. This condition will exist until the breaker is closed or the operation has timed-out.

The sequences described next show the synch-check transactions supervised by SubstationHost that result in the synch-check relay sending a command to close the breaker.

B.5.1 Operator-initiated select for breaker close

Figure B-6~~Figure B-6~~ shows the transaction sequence for an operator-initiated ‘select’ for breaker close. From the one-line diagram on the remote workstation computer, Operator selects the breaker to close. For this scenario, the operatorID (and security keys) are predefined as ‘valid’ to execute breaker close.

RemoteController then issues a ‘close.select’ message over the WAN to SubstationHost to select the breaker to be closed. Then SubstationHost sends a ‘close.select’ message over the LAN to SelectedBreaker. After receiving the message from SubstationHost, SelectedBreaker internally verifies that Operator (operatorID) has the authority to close the breaker. If not, SelectedBreaker would return an error message and the breaker would not be selected.

> Note: Another approach is to issue a security password when the operator logs on. Then, rather than passing the operatorID, the password is sent. The receiving IED then verifies the control privilege by checking the password. The advantage of this approach is that the passwords may be predefined, whereas using the operatorID to verify control privilege requires more memory and processing capability in the receiving IED. However, the scenario requires that other operators be locked out when a specific operator has taken control. This can only be done using operatorID because a predefined password does not uniquely identify the operator.

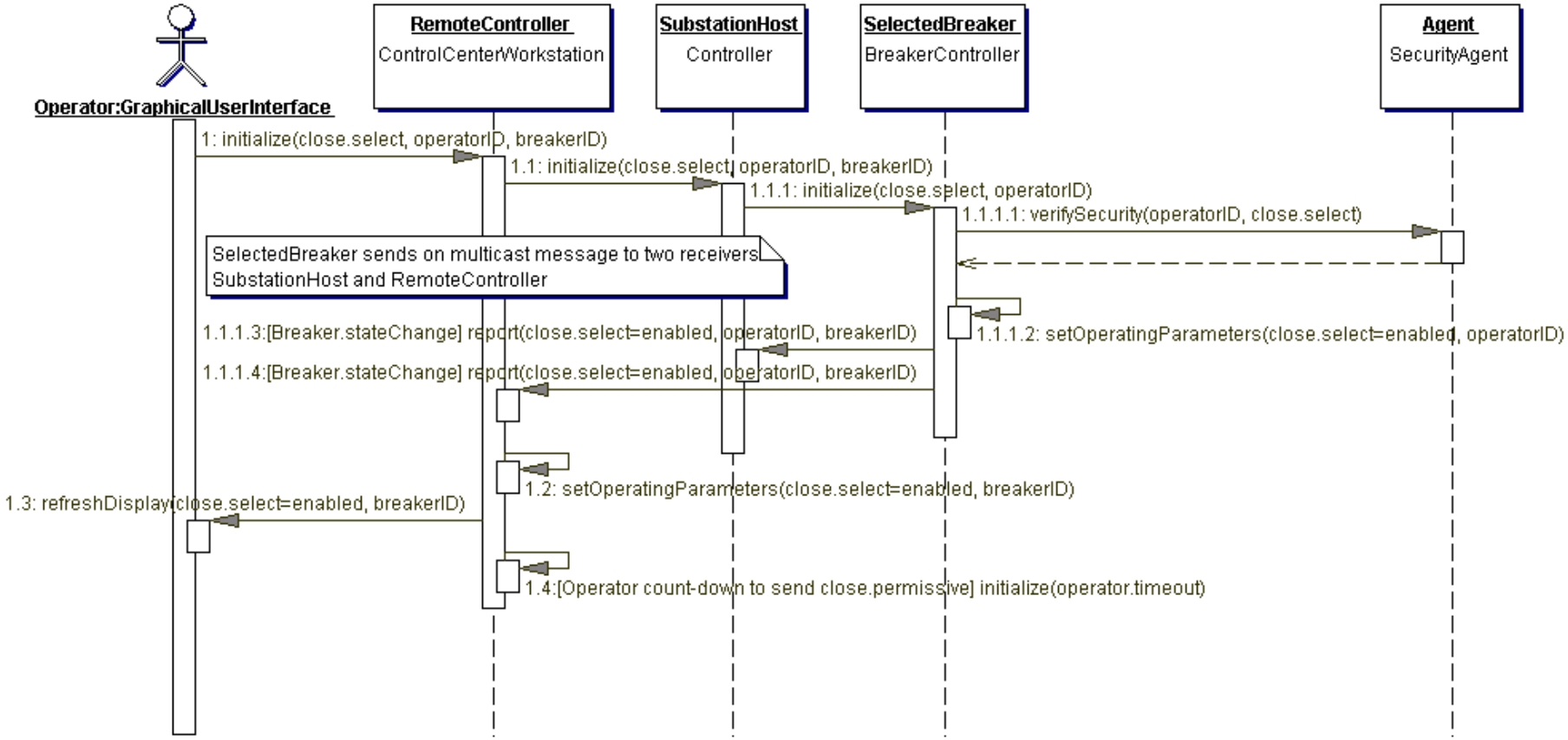

Figure B-6 Operator-Initiated select for breaker close

Operator access is verified in Figure B-6~~Figure B-6~~, and SelectedBreaker sends a multicast message 'report (close.select=enabled)' over the LAN to SubstationHost and over the WAN to RemoteController that Operator selected it for breaker close. When the 'report (close.select=enabled)' message is received, RemoteController internally update the display for Operator. Operator now knows that SelectedBreaker has been selected for breaker close.

Note: EmbeddedDeviceController inherits the **report** method from the class Controller, which is part-of class VirtualDevice. Therefore all specializations of VirtualDevice (e.g., switch, breaker) inherit the method **report**.

Figure B-6~~Figure B-6~~ shows one multicast message to two receivers, SubstationHost and RemoteController. This implies a group address containing two receiver addresses, one is a LAN address and one is a WAN address. In turn the WAN may be connected to sub-networks with one or more remote controllers. Furthermore, only one RemoteController for Operator is included in this scenario. If all operator displays are updated, which they should be, then the group address will need to include all receivers.

Note: Updating all operator displays brings up an interesting question that is not addressed in this scenario. If the operator is not logged on, or for that matter the workstation is turned off, how is the operator notified of new messages that have been multicast to the workstation's address? Some means must be provided to hold these messages in a queue so that the operator is notified of action to be taken when log-on is completed; i.e., notification of messages waiting. When the operator logs on, within one minute the screen should be refreshed with all the IED states maintained by the SubstationHost.

Network management of LAN and WAN addresses is not addressed in this scenario. Furthermore, messages to RemoteController from the SubstationHost or IEDs, which result in operator display updates, should be multicast to all receivers. For clarity, only the RemoteController for Operator is shown in the transaction sequences.

The last step (step 1.4) of the transaction sequence shown in Figure B-6~~Figure B-6~~ is to initialize an internal countdown timer within RemoteController. This countdown is a time-out for the operator to send the close. permissive message to SelectedBreaker. If the operator does not send the close.permissive message within the time-out, RemoteController will automatically send a message to SubstationHost to deselect the breaker. A transaction sequence for this time-out is not shown in Figure B-6~~Figure B-6~~.

B.5.2 Operator-initiated permissive close

Figure B-7~~Figure B-7~~ shows the transaction sequence for an operator-initiated permissive close. When permissive close is enabled, the synch check function is initiated. When specific conditions are satisfied, SynchCheckRCL will issue a close command to the selected breaker.

The transaction sequence to verify and enable the object SelectedBreaker (of class Breaker) for 'close.permissive' is very similar to the sequence described in Figure B-6~~Figure B-6~~. When SelectedBreaker receives the message to 'close.permissive', it must first verify that the Operator is authorized to initiate this action. If not, SelectedBreaker returns an error message. If Operator is authorized, then close.permissive is enabled.

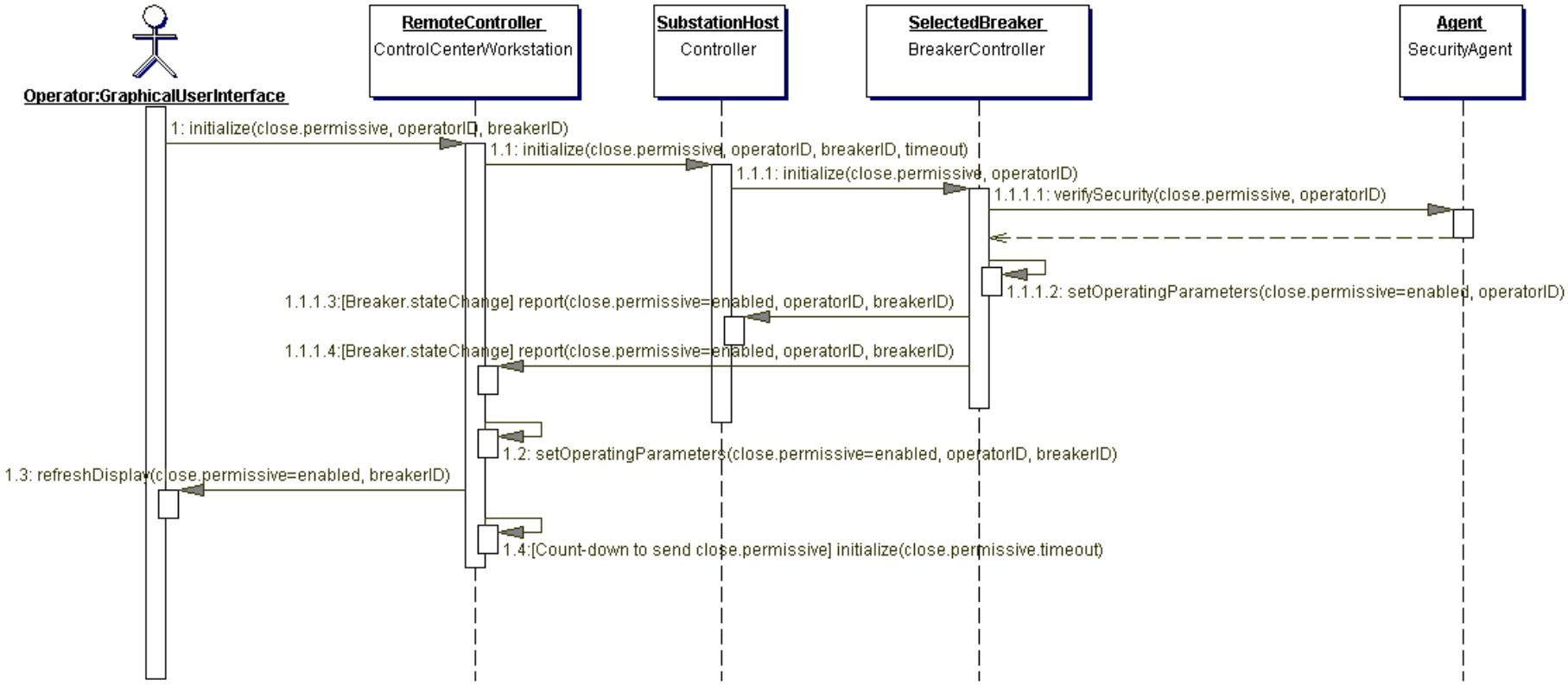

Figure B-7 Operator-Initiated permissive close

Step 1.1 in Figure B-7~~Figure B-7~~ shows that a time-out (predefined in RemoteController) is included in the message to SubstationHost. When SubstationHost selects SynchCheckRelay it will pass the timeout to SynchCheckRelay[19]. After SelectedBreaker enables 'close.permissive', the countdown to close the breaker begins as shown in Step 1.4. If SynchCheckRelay does not send a breaker.close message to SelectedBreaker within the timeout, SubstationHost will cancel the synch check operation by sending the appropriate messages to the selected VTs, SynchCheckRelay and SelectedBreaker. These transaction sequences are not shown in the synch check scenario.

B.5.3 Select synch check parameters, VTs and SynchCheckRelay

Within the substation, only SubstationHost knows the detailed topology and state of all IEDs. Based on predefined conditions, SubstationHost will select the appropriate SynchCheckRelay, LineSideVT and BusSideVT to perform the synch check function. These IEDs will exchange data over the substation LAN only.

Figure B-8~~Figure B-8~~ shows four internal procedures executed by SubstationHost to select the synch check parameters, LineSideVT, BusSideVT and SynchCheckRelay. After selecting these IEDs, SubstationHost will initiate synch check.

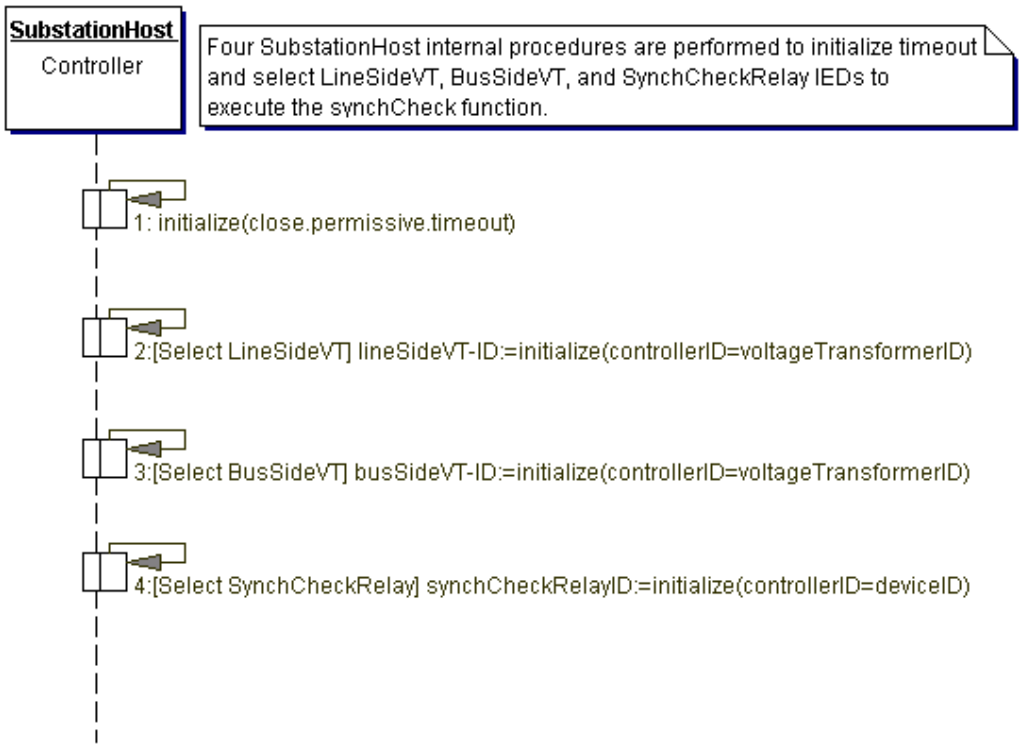

Figure B-8 Select synch check parameters and devices

[19] SynchCheckRelay may need timeout as a parameter in executing its synch check function. If SynchCheckRelay does not require timeout, it will not be passed from SubstationHost. Note, if timeout does occur, SubstationHost has the responsibility to cancel synch check functions; SynchCheckRelay does not have this responsibility.

B.5.4 Initiate synch check

Figure B-9~~Figure B-9~~ shows the transaction sequence initiated by SubstationHost to begin the synch check function. Both voltage transformers begin sending ZeroCrossing information to the SynchCheckRelay as soon as they receive the initiate message from SubstationHost.

The first two messages initialize synchCheck reporting by telling the LineSideVT and BusSideVT to send their ZeroCrossing information to a selected SynchCheckRelay. These two messages also specify the refreshRate for updating the VT ZeroCrossing data.

The third message initializes the SynchCheckRelay by specifying which breaker has been selected to close. The message also tells the SynchCheckRelay which LineSideVT and which BusSideVT will send ZeroCrossing information. Timeout is also passed to SynchCheckRelay if it is needed for executing the synch check function.

When the SynchCheckRelay receives the message to initiate synchCheck from the SubstationHost in step 3, it enables synchCheck and sends a multicast state change message to the SubstationHost, SelectedBreaker, and Remote controller. This multicast message is shown in steps 3.1, 3.2 and 3.3.

Both the LineSideVT (step 4) and BusSideVT (step 5) begin sending ZeroCrossing information (unsolicited reports) to the SynchCheckRelay as soon as they are initiated. Each VT will send ZeroCrossing data periodically in accordance with the refreshRate specified by SubstationHost. The ZeroCrossing data object is a data structure comprised of six data items: controllerID (LineSideVT or the BusSideVT), measurementTime, voltageMagnitude, frequency, frequencyRateOfChange, and positiveZeroCrossingTime.

B.5.5 Check for dead line or dead bus

Figure B-10~~Figure B-10~~ shows a transaction sequence initiated by SynchCheckRelay to check for dead line or dead bus. If dead line or dead bus conditions exist, SynchCheckRelay sends a close message to SelectedBreaker. Because SelectedBreaker has enabled close.permissive, it will respond to the close command from SynchCheckRelay.

After verification, SelectedBreaker closes the breaker and deselects the breaker. This action will disable synchCheck and close.permissive, and deselect close.select. Because its state has changed, Figure B-11~~Figure B-11~~ shows that SelectedBreaker immediately sends SynchCheckRelay and SubstationHost a multicast change of state message. SubstationHost will change the operatingParameters accordingly.

Figure B-11~~Figure B-11~~ shows that SubstationHost then sends a multicast message to the RemoteController (not shown), LineSideVT, and BusSideVT to reset synchCheck=disabled. Each VT will then stop sending zeroCrossing voltage information, and the RemoteController will refresh the operator's display.

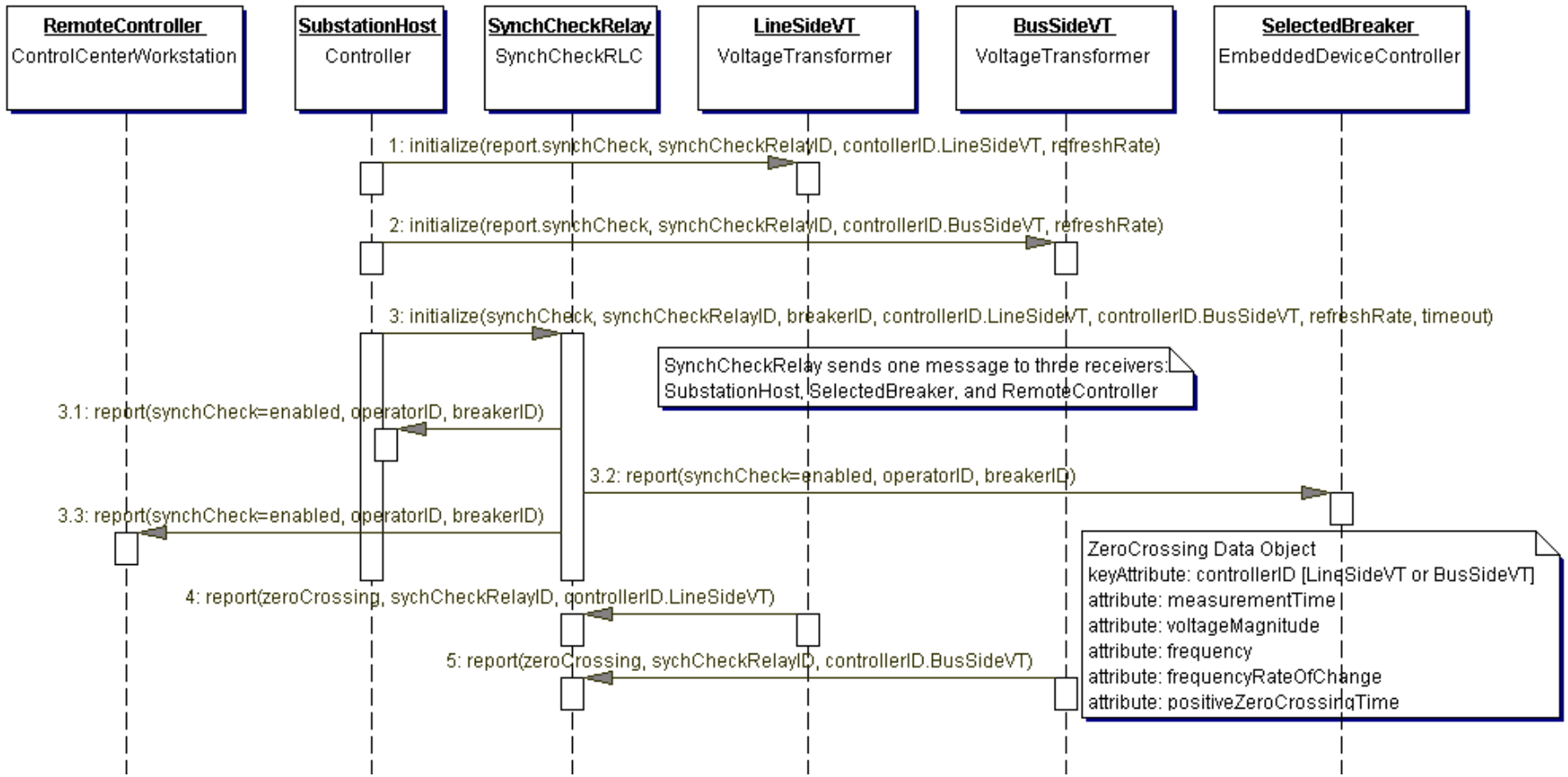

Figure B-9 Initiate synch check

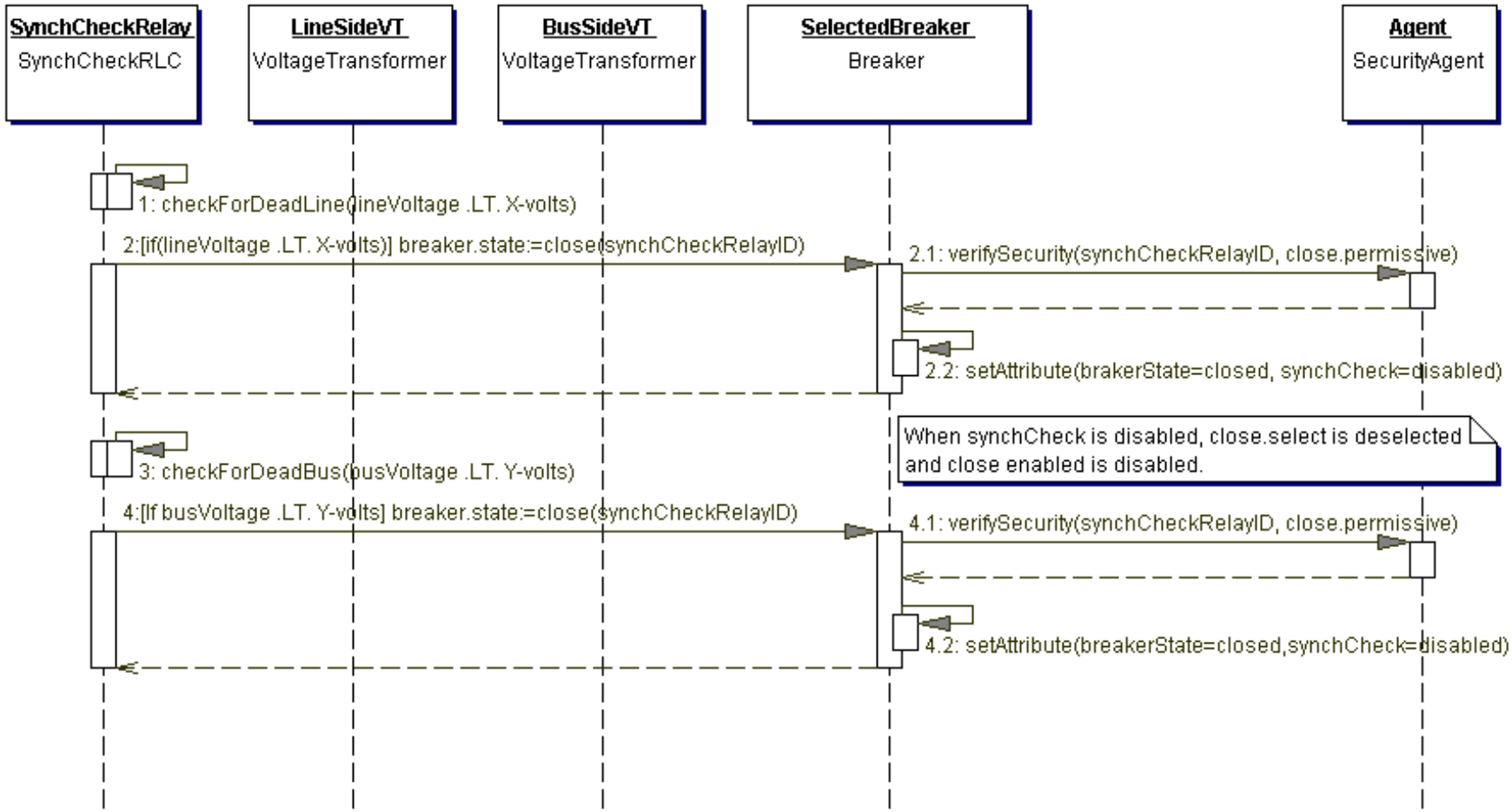

Figure B-10 Check for dead line or dead bus

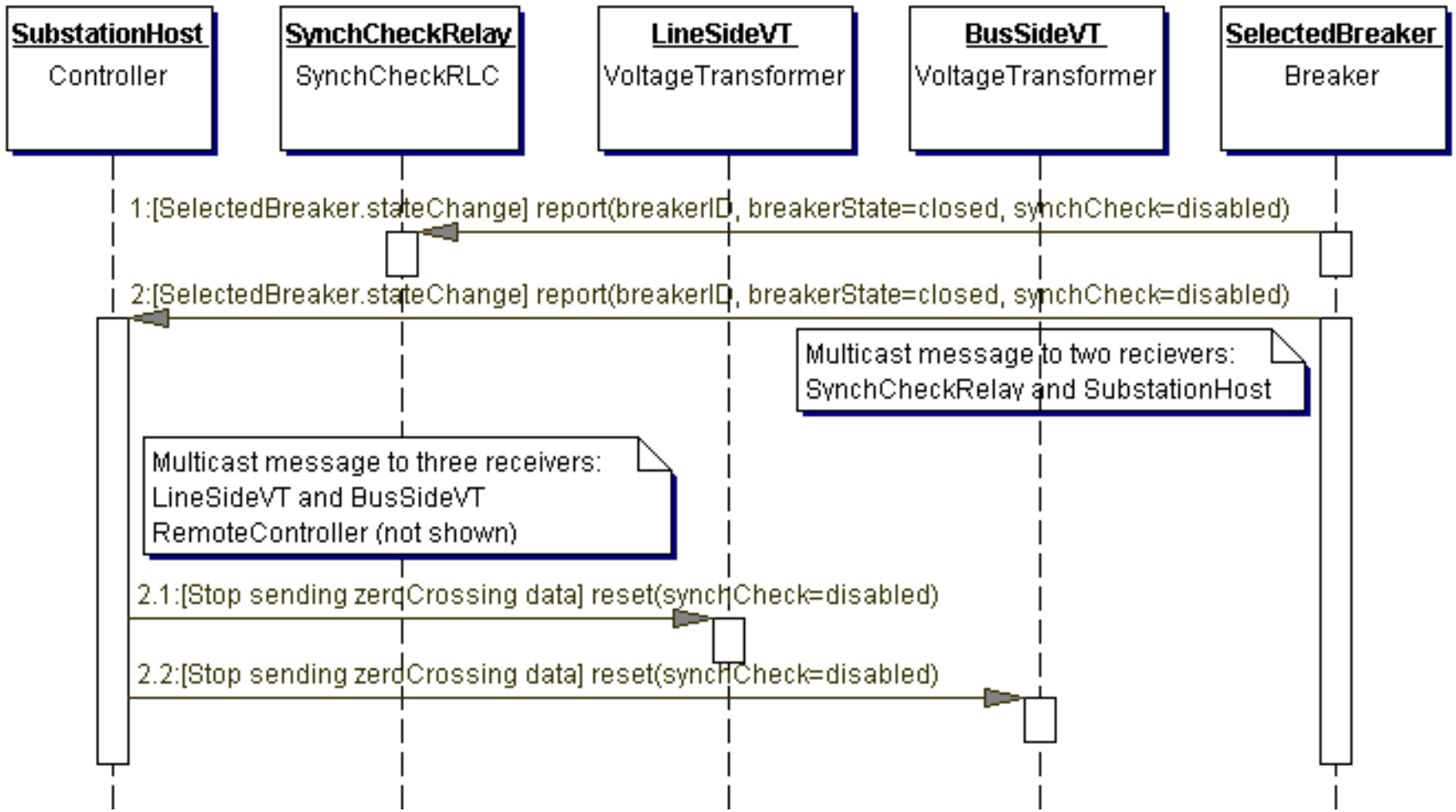

Figure B-11 Report selected breaker state change

B.5.6 Check for high voltage difference

Checking for high voltage difference produces the same transaction sequence shown in Figure B-10~~Figure B-10~~ except for one difference. If SynchCheckRelay detects a highVoltageDifference conditions between the two VTs which persists beyond the preset time out, it sends a message to SelectedBreaker to change the breaker state to close.select=disabled. After verifying the command request, SelectedBreaker deselects the breaker for close (close.select=disabled and close.permissive=disabled), and SelectedBreaker remains in the open state condition (breakerState=open).

B.5.7 Perform synch check

Figure B-12~~Figure B-12~~ shows a transaction sequence to perform the synch check function. First SynchCheckRelay calculates the difference in angle between LineSideVT and BusSideVT from the supplied ZeroCrossing information. When these parameters (angular difference and slipRate are within preset limits, then SynchCheckRelay sends a close message to SelectedBreaker. Again, because SelectedBreaker has enabled ‘close.permissive’, it will respond to the close command from SynchCheckRelay.

When SelectedBreaker verifies that SynchCheckRelay is authorized to close the breaker, it closes the breaker and changes its state to breakerState=closed. This action will also set close.select=deselected and close.permissive=disabled. Figure B-13~~Figure B-13~~ shows that SelectedBreaker then sends a multicast message to SynchCheckRelay and to SubstationHost notifying them of its state change, and that synchCheck is disabled.

Figure B-13~~Figure B-13~~ shows that after SubstationHost receives the message from SelectedBreaker, it sends a multicast message to each VT with the notification that synchCheck=disabled. Each VT will stop sending its zeroCrossing voltage information, and the RemoteController will refresh the operator’s display. Each VT sends a “reset” message to the SubstationHost, confirming that its state has changed to synchCheck=disabled.

B.5.8 Update RemoteController

Transaction sequences were described for selecting synch check parameters and devices, initiating synch check, checking for dead line or dead bus, checking for high voltage difference, verifying that synch check parameters are within limits, and closing the breaker. For each of the transaction sequences SubstationHost sends a multicast message to all instances of RemoteController. When received, operator displays are refreshed and local databases are updated.

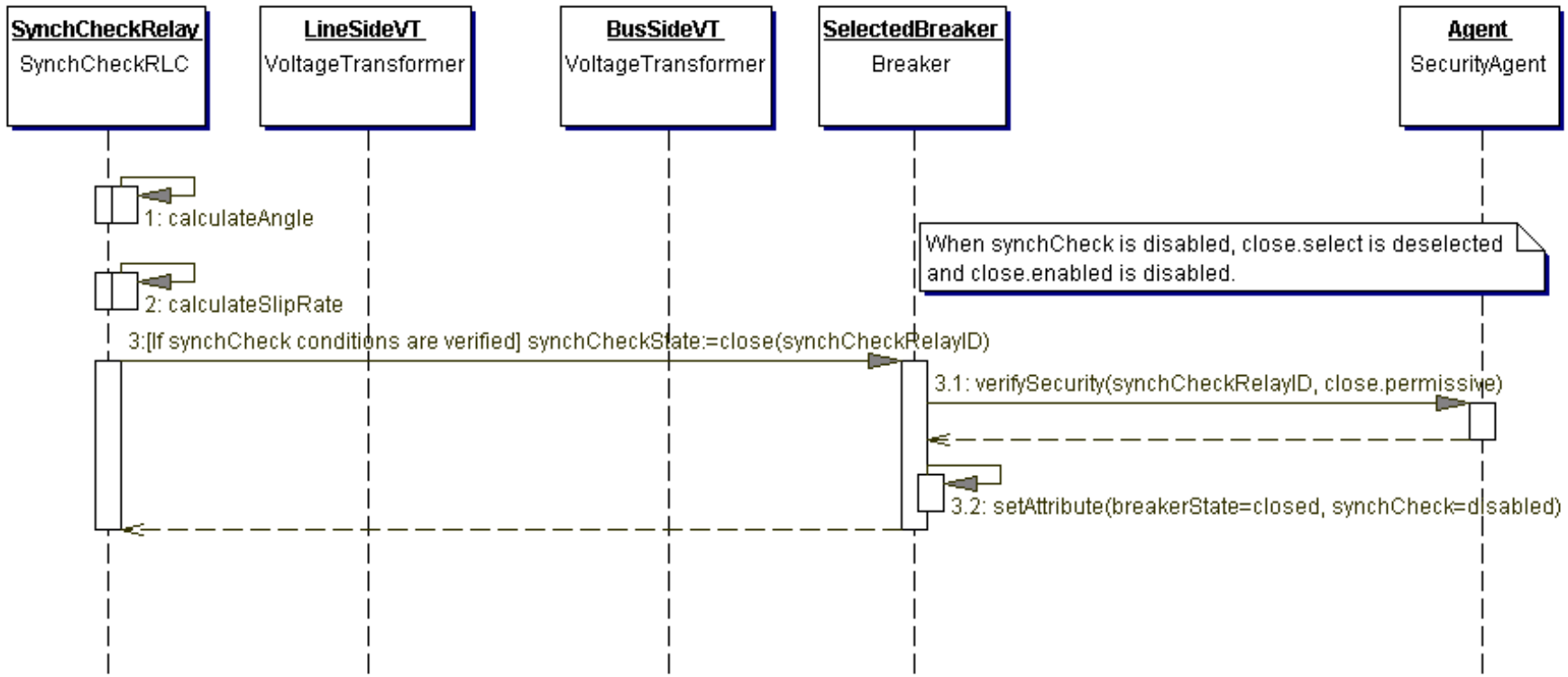

Figure B-12 Verify synch check conditions and close breaker

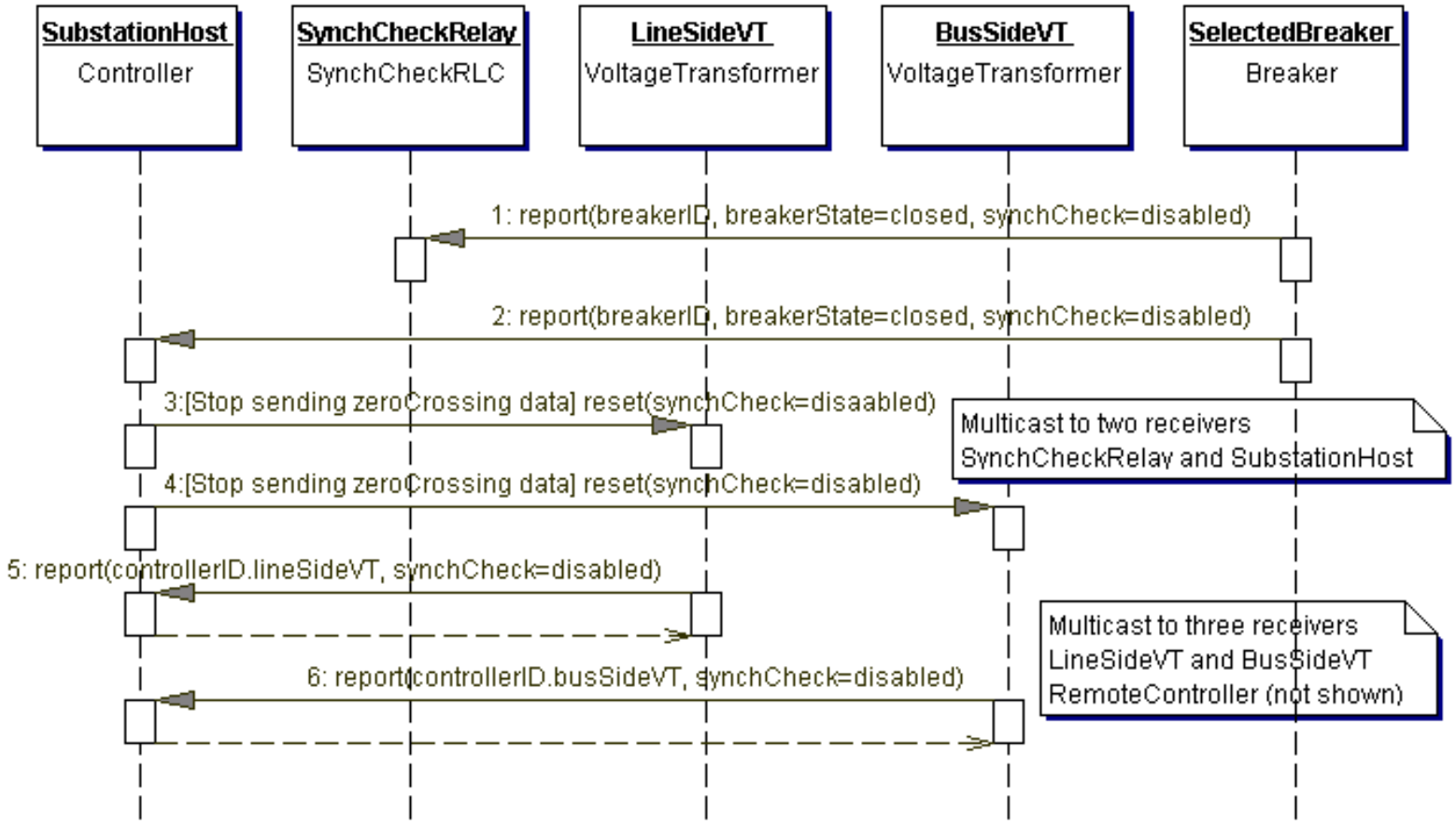

Figure B-13 Report synch check status and stop sending zero crossing data

C Load tap changer control with remote voltage measurements (informative)

Load Tap Changer (LTC) control of a transformer is used to regulate the load voltage. The voltage at the load center is estimated using the local voltage, load current, and impedance of the line. The estimated load voltage is not a true representation of actual load center voltage due to the distribution of the load and line parameter changes, tapped lines, and other variations. The load voltage can be measured by having a remote voltage-measuring device (or may be part of any IED near the load center) and transmitting this information to the LTC control for providing improved load voltage regulation.

The engineer who is configuring the system will first query the substation database to determine which remote Voltage Transformers (VTs) are connected. Then based on load studies, the engineer will select the remote VT that will best represent the load center.

The input to the LTC control and the output from the IED near the load center is the measured load voltage. These two devices are not physically close and may be several miles apart. The measured voltage is usually represented by VT secondary voltage with a nominal voltage of 120V. The voltage dynamic range of 0 to 150V is adequate.

C.1 Performance requirements

The voltage measurement accuracy of ±0.5% and the response time of 0.25 sec for the voltage measurement and communications are adequate for LTC control applications. Either the RMS[20] magnitude or the fundamental magnitude is the voltage information to be sent from the remote VT representing the load center.

C.2 Evaluation criteria

- LTC control should be able to identify the remote IED and set up the IED to receive remote voltage measurement periodically.
- Generate an alarm to the operator when the measured IED voltage does not follow the estimated load center voltage.
- Upon a failure of communication between the remote IED and the LTC control, LTC control will generate an alarm to the operator. Included in with this alarm is the LTC control status, which should be internally set to local control based on line drop compensation calculations.

C.3 Functional configuration

Figure C-1~~Figure C-1~~ shows the functional communication configuration to interface the Utility Control Center, providing an authorized engineer the capability to choose which remote VT will be the Load Center VT. All candidate VT Intelligent Electronic Devices (IEDs) include a WAN communication interface. Also, communication between the control center, and the VTs, with the substation is over the Utility Wide Area Network (WAN).

[20] RMS magnitude is used for this example scenario.

Within the substation, communication between IEDs is over the Substation Local Area Network (LAN). If WAN and LAN communications use the same protocol then communications between substation IEDs and remote IEDs do not need a gateway for protocol conversion[21]. If WAN and LAN communications do not use the same protocol, then the SubstationHost will provide act as the gateway and provide the needed protocol conversion capability.

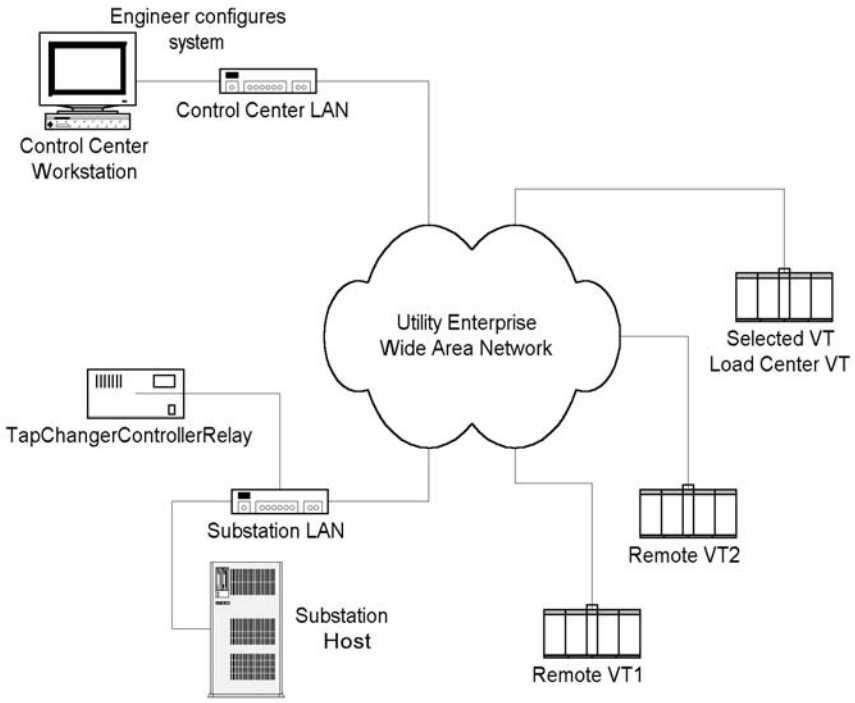

Figure C-1 LTC Control functional configuration

C.4 Object model

Figure C-2~~Figure C-2~~ shows the LTC control object model and its relationship to the SubstationHost. The TapChangerControllerRelay (TCCR) provides the intelligence for operating the LoadTapChanger that is shown hardwired to the TCCR. TCCR is a specialization of Relay (not shown) and therefore inherits all the functionality of Relay.

EmbeddedDeviceController and LoadCenterVoltageTracker are packaged as part of TCCR. EmbeddedDeviceController is a specialization of Controller and inherits all the functionality of Controller. LoadCenterVoltageTracker knows the preset allowable percentage of line drop compensation voltage, which will be compared to the estimated line drop compensation voltage.

[21] Using the same protocol on the WAN and LAN has the advantage in this scenario that all VTs can be designed with the same communication processor. If the WAN and LAN protocols are different, the remote VTs must be designed with a WAN communication processor and the substation IEDs must be designed with a LAN communication processor.

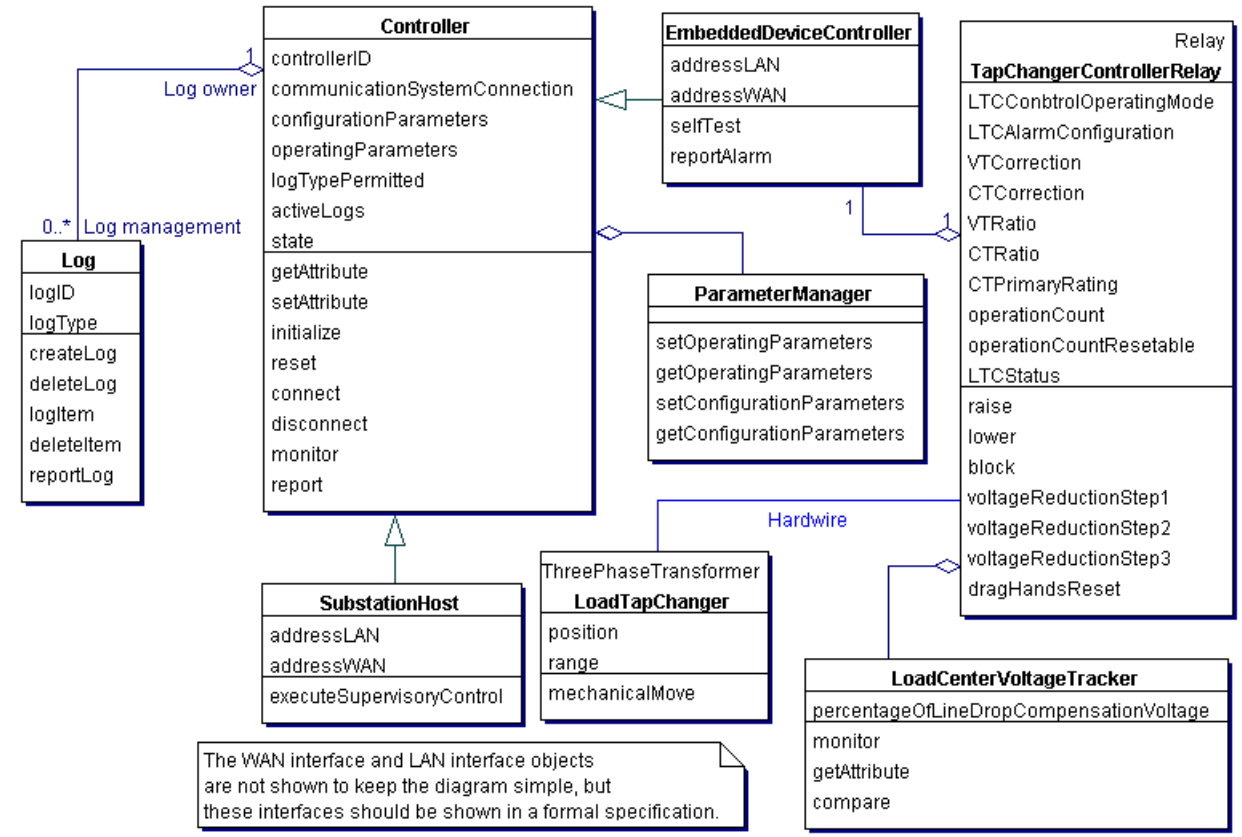

Figure C-2 LTC control object model

Control can be implemented in two ways: If WAN and LAN protocols are the same; the remote IED can communicate directly with the TCCR because it includes the communication processor capability from EmbeddedDeviceController. If the WAN and LAN protocols are not the same, then remote IED must communicate through the SubstationHost which provides the necessary protocol conversion algorithms.

For this scenario, control is implemented through the SubstationHost, and all status and data are reported from the TCCR to the SubstationHost. Control and Status & Data that is exchanged between the TCCR and the SubstationHost uses the Substation LAN communication services. WAN interfaces to the SubstationHost and EmbeddedDeviceController provide the capability for remote operations. The WAN interface and LAN interface objects are not shown to keep the diagram simple, but these interfaces should be shown in a formal specification.

C.4.1 Remote operations

This scenario requires remote engineer control between the control center and the remote LoadCenterVT and the Substation over the Utility Enterprise WAN. LoadCenterVT RMS magnitude is also sent to the Substation over the Utility Enterprise WAN.

C.4.1.1 Remote engineer control

Figure C-3~~Figure C-3~~ shows the object model for remote engineer control. The remote engineer has the capability to select the VT from the connected VTs to represent the load center (LoadCenterVT), and to manipulate the TCCR through the WAN communication services.

The engineer can do those things, and only those things, which he could also accomplish if physically at the control; i.e., change set points, reconfigure the control, and read system operation conditions. The engineer cannot “initialize” the control except by individually changing the set point(s).

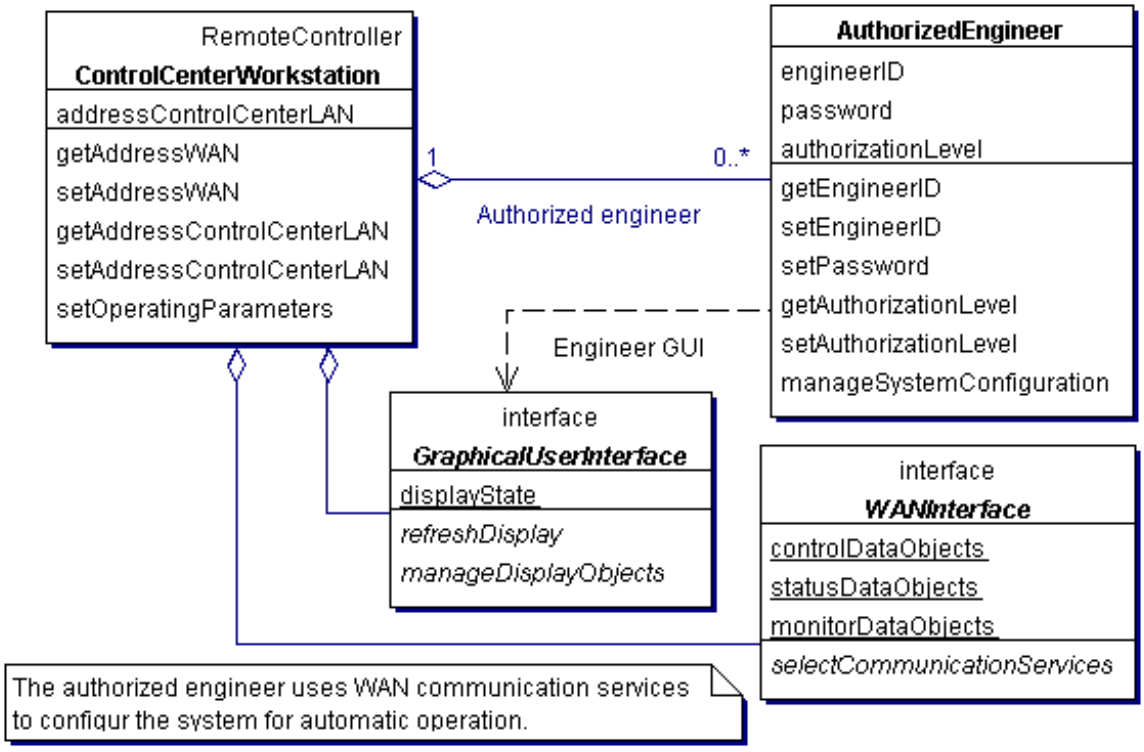

Figure C-3 Remote Engineer Control Object Model

C.4.1.2 Load center VT operation

Figure C-4~~Figure C-4~~ shows the object model for a LoadCenterVT. EmbeddedDeviceController is defined as part of the InstrumentTransformer, which in turn is instantiated as the SelectedLoadCenterVT that reports voltageMagnitudeRMS. Only the WAN Interface is implemented because the SelectedLoadCenterVT IED is not in the substation.

The class InstrumentTransformer is a specialization of the class MeasurementUnit. As a specialization of MeasurementUnit, it inherits all the attributes and behavior and adds its special features.

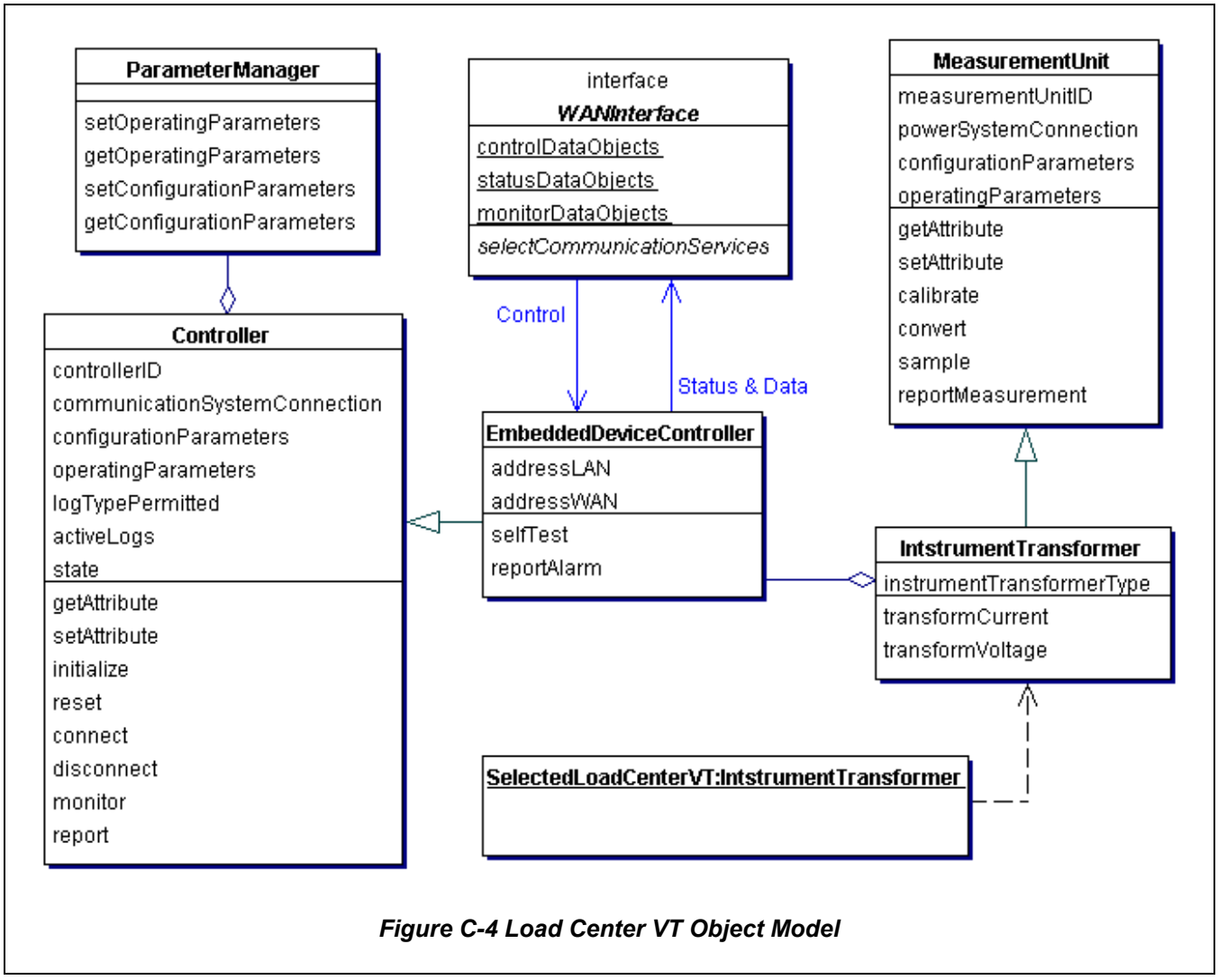

Figure C-4 Load Center VT Object Model

C.4.2 Internal substation operation

LoadCenterVT outputs RMS magnitude, which is used by the TCCR to raise or to lower the LoadTapChanger position. Desired action is specified by the set points controlled by the engineer who established the system configuration. Internal algorithm processing within the TCCR will generate raise and lower tap change commands to the LoadTapChanger. These commands are sent over the hardwire interface as shown in Figure C-2~~Figure C-2~~.

C.5 Transaction sequences

Transaction sequences are described to show the exchange of data required to select a LoadCenterVT, to initialize the selected LoadCenterVT and set its operating parameters, to perform tap changer operations, and to generate alarms.

C.5.1 System configuration engineer selects remote LoadCenterVT

Figure C-5~~Figure C-5~~ shows the transaction sequence needed for the system configuration engineer to query the SubstationHost to get a list of the remote VTs on the feeders connected to the selected transformer. Then based on load studies, the system configuration engineer selects the remote VT that is most representative of the load center. This VT will be called the LoadCenterVT (LCVT).

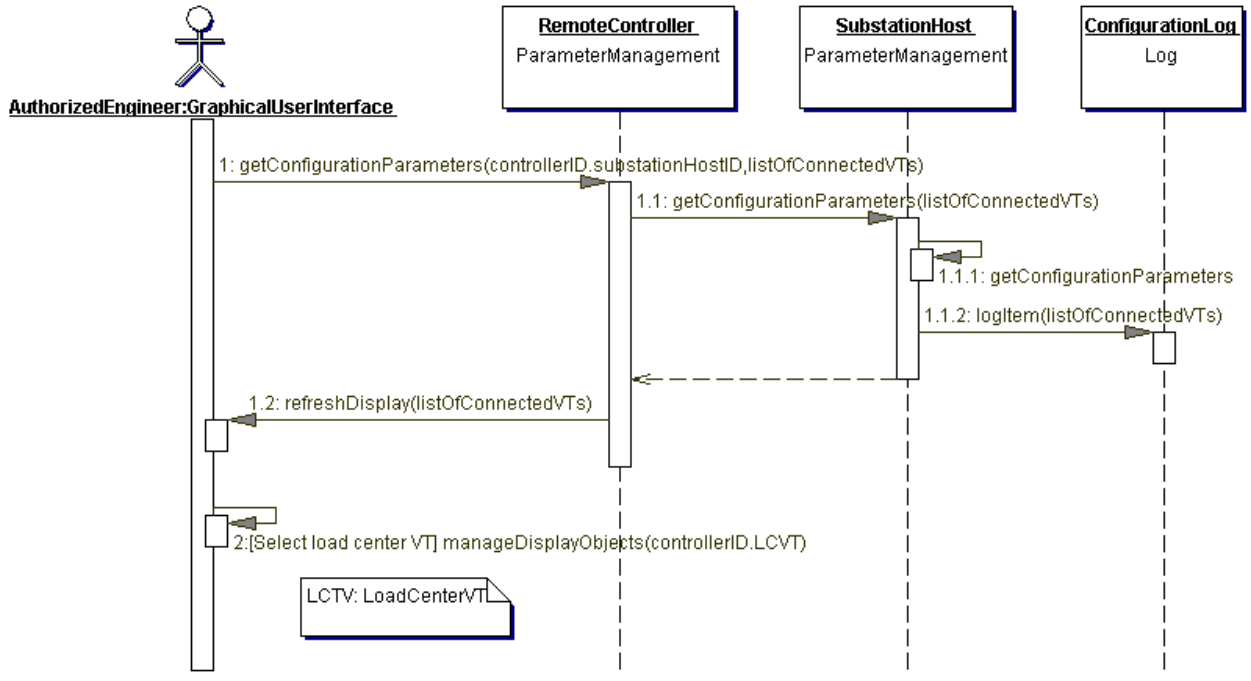

Figure C-5 Select Load Center VT for Tap Changer Control

In response to ***getConfigurationParameters***, the SubstationHost queries its configuration database, then logs the listOfConnectedVTs and reports the list to its

Client, the RemoteController. In turn the RemoteController refreshes the Engineer's display. The Engineer uses the point-click-drag-drop mechanism to select the LoadCenterVT from the list of connected VTs.

C.5.2 Substation host initializes LoadCenterVT input to TCCR

Given the selection of the LoadCenterVT shown in Figure C-5~~Figure C-5~~, the next step is for the SubstationHost to automatically initialize the TCCR as shown in Figure C-6~~Figure C-6~~. First the SubstationHost gets the pre-specified value of X%, which is the allowable voltageMagnitudeRMS deviation from the estimated line drop compensated voltage.

When the request to initialize is received by the TCCR, it must first check for active tags that could prohibit operation of the TCCR. Assuming that there are no active tags, then the TCCR is ready to receive LoadCenterVT information (voltageMagnitudeRMS) from the select VT, and the SubstationHost is ready to monitor these reports from the LoadCenterVT and the state of the TCCR.

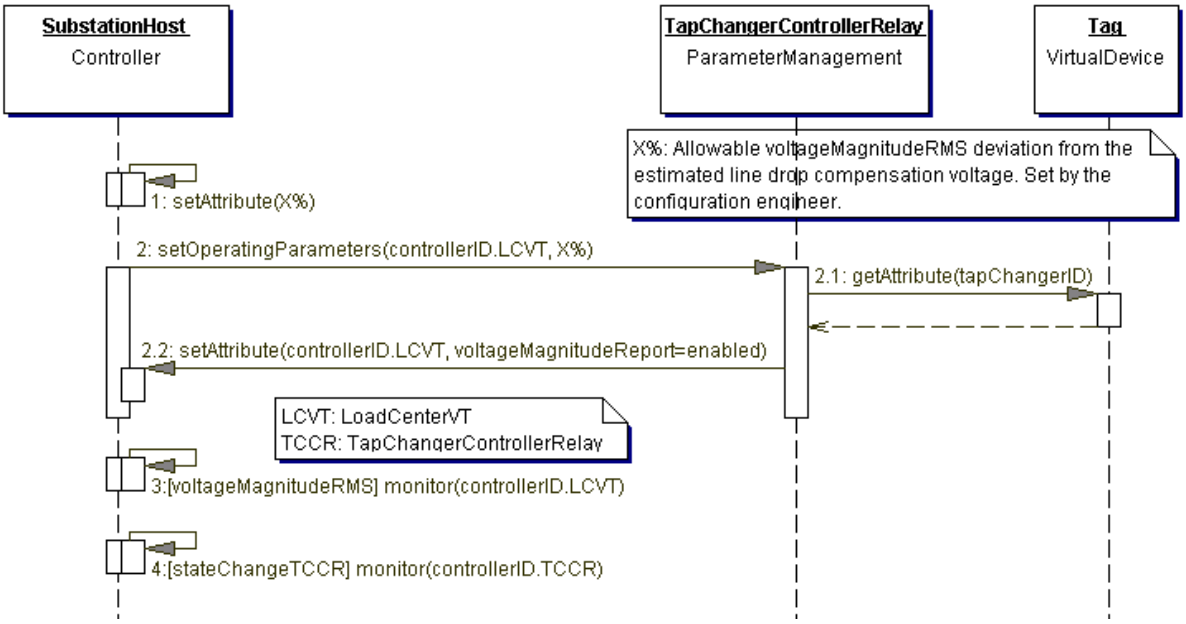

Figure C-6 Substation Host initializes LoadCenterVT Inputs to TapChangerControllerRelay

C.5.3 System configuration engineer sets operating parameters

The system configuration engineer must now set the operating parameters for delivery of voltageMagnitudeRMS data to the TCCR IED. The system configuration engineer must first establish LoadCenterVT operating parameters.

Figure C-7~~Figure C-7~~ shows the transaction sequence to initialize the LoadCenterVT IED. The system configuration engineer must specify the ***startTime***, ***stopTime*** and ***updateInterval*** for sending its voltageMagnitudeRMS to the TapChangerControllerRelay IED. If the ***stopTime*** is set to '0', the LoadCenterVT will send its voltage information at the ***updateInterval*** until it is commanded to stop by the system configuration engineer.

After the LoadCenterVT is initialized, the RemoteController is now ready to monitor reports from the LoadCenterVT and from the TapChangerControllerRelay, and update the system engineer's display with the current status information. After each report is received, the RemoteController refreshes the Engineer's display. Then the GraphicalUserInterface software manipulates the display objects.

Figure C-7 Initialize LoadCenterVT to Send Voltage Information

C.5.4 Tap changer control operations

Figure C-8~~Figure C-8~~ shows the transaction sequence that delivers RMS magnitude voltage information from the LoadCenterVT to the SubstationHost. Because the WAN and LAN protocols are assumed to be different, the SubstationHost performs the gateway function to convert from a WAN protocol to a LAN protocol. After protocol conversion, the SubstationHost logs the receipt of the voltageMagnitudeRMS and reports the voltageMagnitudeRMS to the TapChangerControllerRelay over the substation LAN.

Within the TapChangerControllerRelay, the information is processed to determine if a LoadTapChanger mechanical move should be commanded. If so, the command is sent over the hardwire connection to the LoadTapChanger.

The TapChangerControllerRelay reports all change in status to the SubstationHost, where the status change information is logged. Not shown is the report from the SubstationHost to the RemoteController over the WAN. If these reports were required, then the RemoteController would also log the status change information and refresh the operator's display.

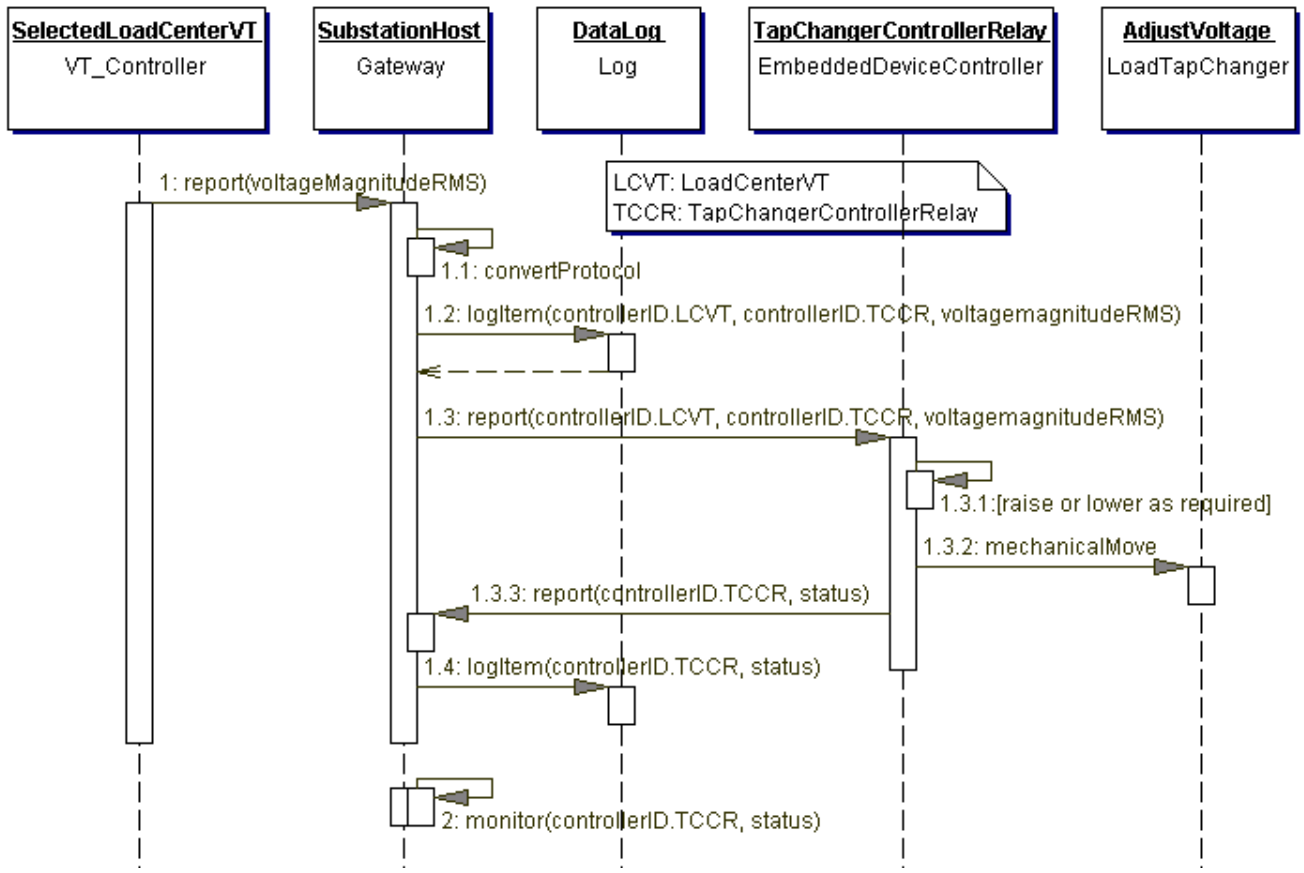

Figure C-8 Tap Changer Operation

C.5.5 Alarm generation

TCCR generates alarms when two situations occur: When the measured voltage (voltageMagnitudeRMS) does not track the estimated load center voltage, and when communication is lost between the selected LoadCenterVT IED and the TCCR IED. Transaction sequences describe the exchange of messages and the corrective action taken by the TCCR.

C.5.5.1 Measured voltage does not track estimated load center voltage

Figure C-9~~Figure C-9~~ shows the transaction sequence to generate an alarm when the voltageMagnitudeRMS exceeds a pre-specified percentage of the line drop compensation voltage (X%). The TCCR monitors the RMS magnitude and compares it with the pre-specified percentage of line drop compensation initially set by the system configuration engineer. When the voltageMagnitudeRMS exceeds X%, then the TapChangerControllerRelay sends an alarm report to the SubstationHost that specifies it has changed state. The state change specifies that voltageMagnitudeRMS exceed the pre-specified percentage. Under these conditions, the TCCR does not change operating mode, it only generates an alarm message.

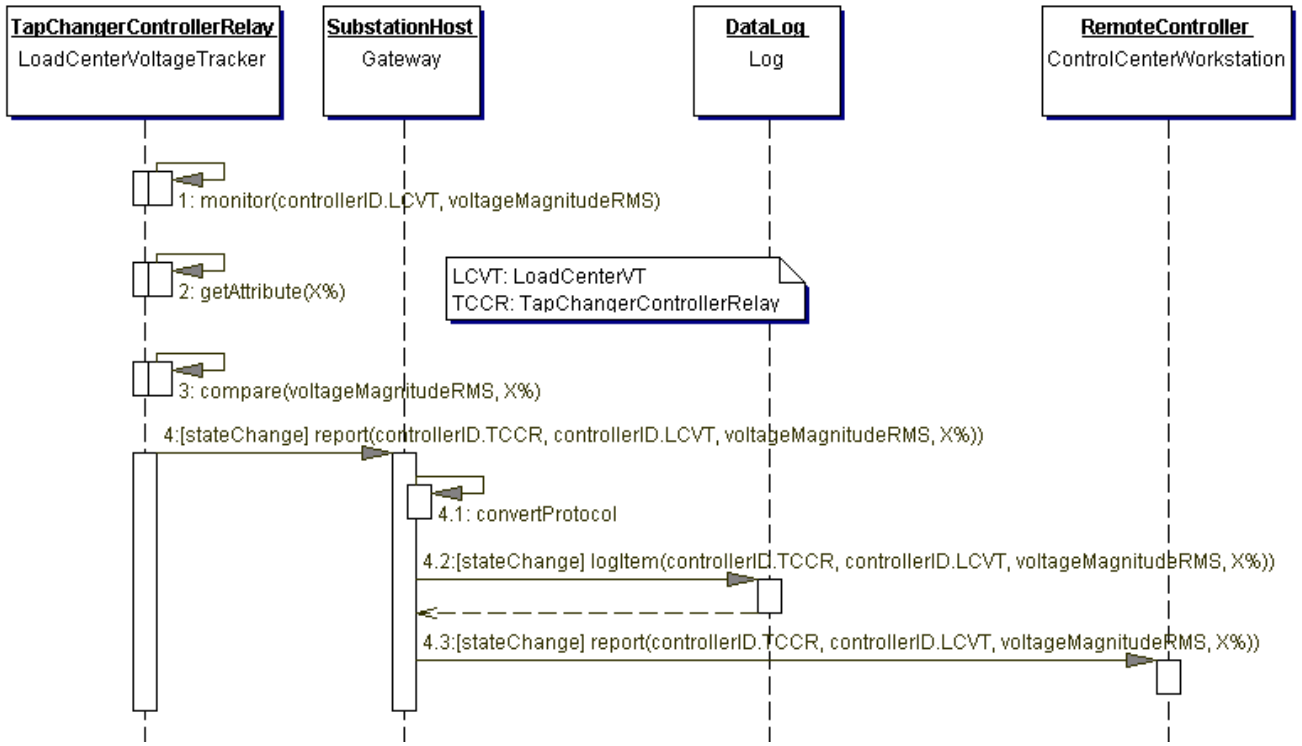

Figure C-9 Alarm Generated when Measured Voltage Exceeds Specified Value

Upon receipt of the state change message from the TCCR, the SubstationHost sets the configuration parameters to reflect the state change, logs the state change, and converts the protocol to send the message over the WAN to the RemoteController.

Figure C-10~~Figure C-10~~ shows that upon receipt of the message at the RemoteController, it refreshes the Operator's display. Then, the Operator will either select a new LoadCenterVT or change X%.

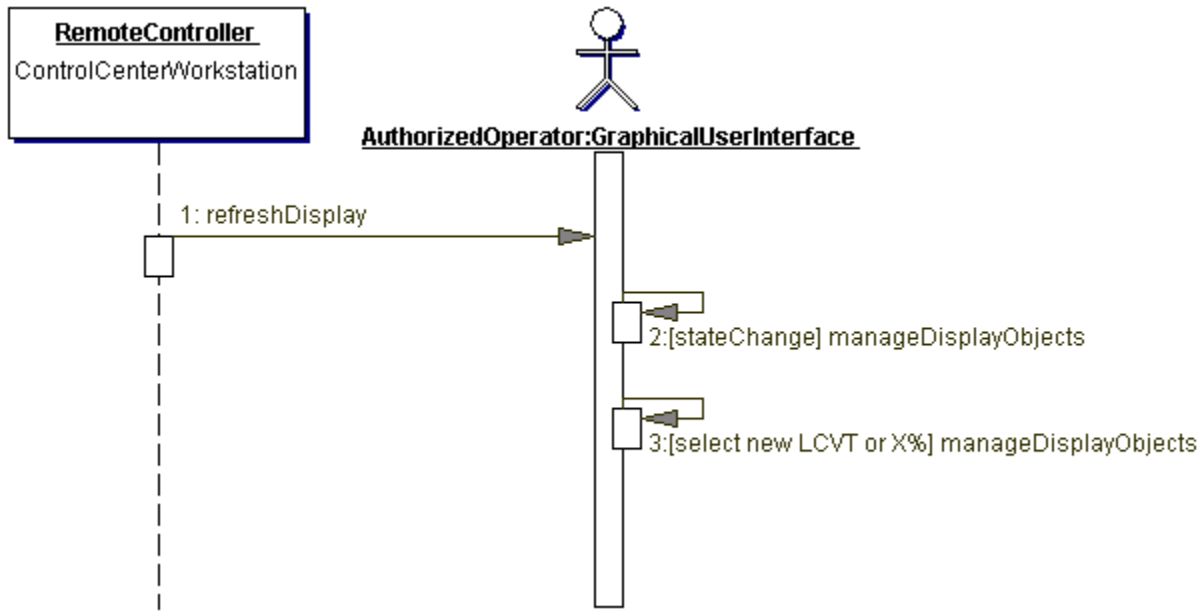

Figure C-10 Operator response to alarm

C.5.5.2 Communication lost between TCCR IED and selected LCVT IED

Figure C-11~~Figure C-11~~ shows the transaction sequence when communication is lost between the TCCR IED and the selected LoadCenterVT IED. When communication is lost, TCCR changes operating modes to use the estimated value of line drop compensation voltage. It will continue to operate in this mode until the Operator reinitializes the process.

The SubstationHost sets the new operating mode, logs the state change and converts the protocol to send the alarm over the WAN to the RemoteController.

Figure C-12~~Figure C-12~~ shows that upon receipt of the alarm, the RemoteController refreshes the Operator's GraphicalUserInterface display. The display software manages all objects on the GraphicalUserInterface.

The operator can then select a new LoadCenterVT, and reinitialize the configuration.

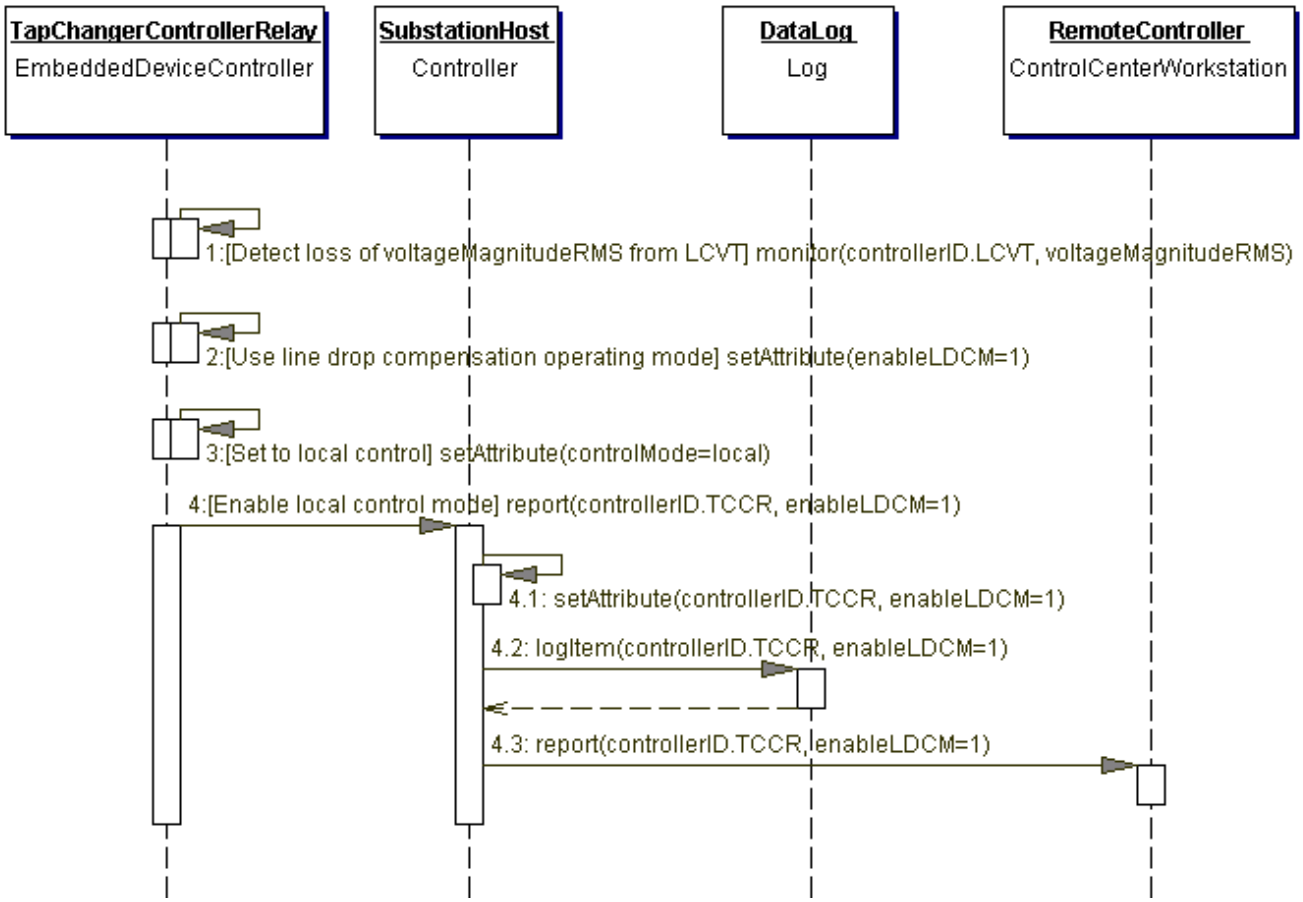

Figure C-11 Alarm generated when communication is lost

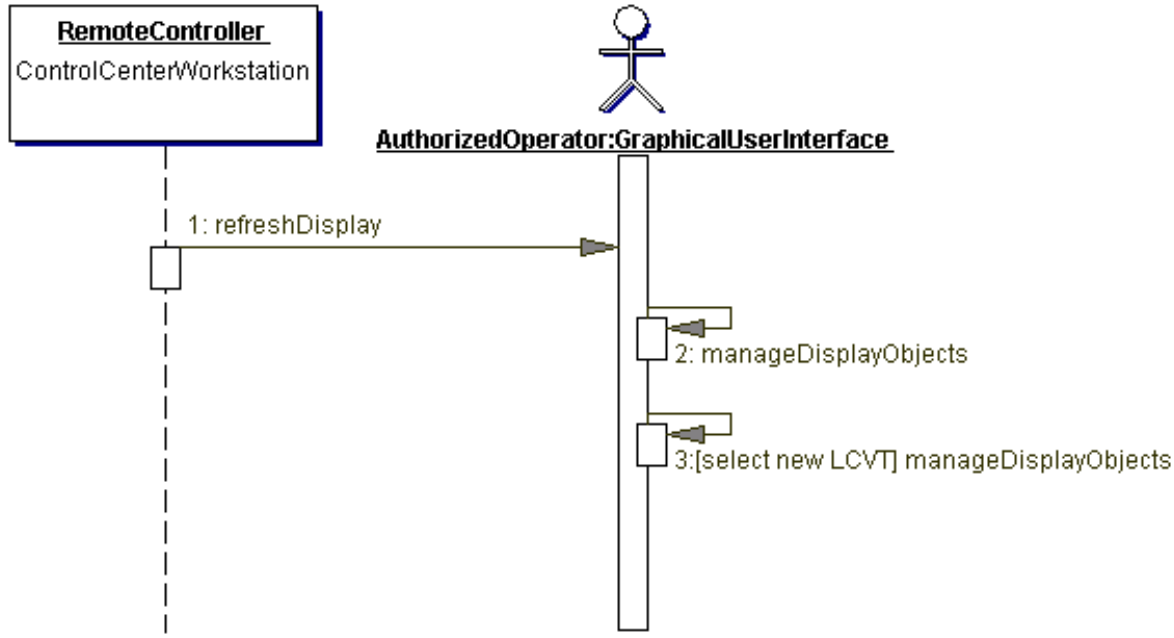

Figure C-12 Operator response to communication loss

D Distributed generation on utility feeders (informative)

A customer, known as a Merchant Generator, has installed generation at their facility and has capacity in excess of their load. The facility is served from a utility feeder that has only a few other customers on it with total load less than available generation from the Merchant Generator.

The utility would like to have access to this extra generating capability at times when capacity resources are low and will request the Merchant Generator to operate its generators. To maximize generation availability, the Merchant Generator must run its generators in parallel with the utility feeder. This parallel operation and the likelihood of back-feeding into the utility's feeder bus suggests that a method of changing the characteristics of the utility's feeder overcurrent relay be adopted in order to backup the feeder fault detecting function of the customer's main circuit breaker protection.

Use of peer-to-peer communications replaces dedicated communication link and I/O interface hardware. Logical elements can reside within overcurrent relays rather than with external logic elements. There is no need to establish line-side voltage source for synchronism check or voltage block closing, and there is no separate SCADA needed at customer's site.

D.1 Performance requirements

All transactions beginning with the state change report from the utility overcurrent relay (U.OCR) to the customer overcurrent relay (C.OCR) through the receipt of the state change report from the C.OCR to the U.OCR should be completed within 200 ms. This includes all communication time over both the utility's communication networks and the customer's communication networks. A transaction sequence view, which reflects all the steps to be included within the 200 ms, is described in Clause D.4.2.

D.2 Functional configuration

Figure D-1~~Figure D-1~~ shows the functional configuration that provides internetworking of the Utility and Merchant Generator communication networks. The Utility Control Center and Utility Substation are networked through the Utility Enterprise Wide Area Network (WAN). The Merchant Generator Substation is networked through the Merchant Generator WAN.

A utility systems engineer will use the engineering workstation to initially configure the system and download all configuration data into the Substation Host. The utility operator will monitor and control the operation using the SCADA/EMS workstation within the Utility Control Center.

Operator control at the Merchant Generator is also remotely provided over their WAN. All substation operation is automatic; i.e., no local operator attended operation. Coordination between the two operators is required to determine the actual settings each will download to their respective IEDs. Thereafter, the control actions are automatic, without operator intervention.

Communication between the Utility WAN and the Merchant Generator WAN requires a Gateway to provide all needed communication protocol conversions[22]. Within each substation, the scenario assumes that an Ethernet Local Area Network (LAN) is provided for substation communications.

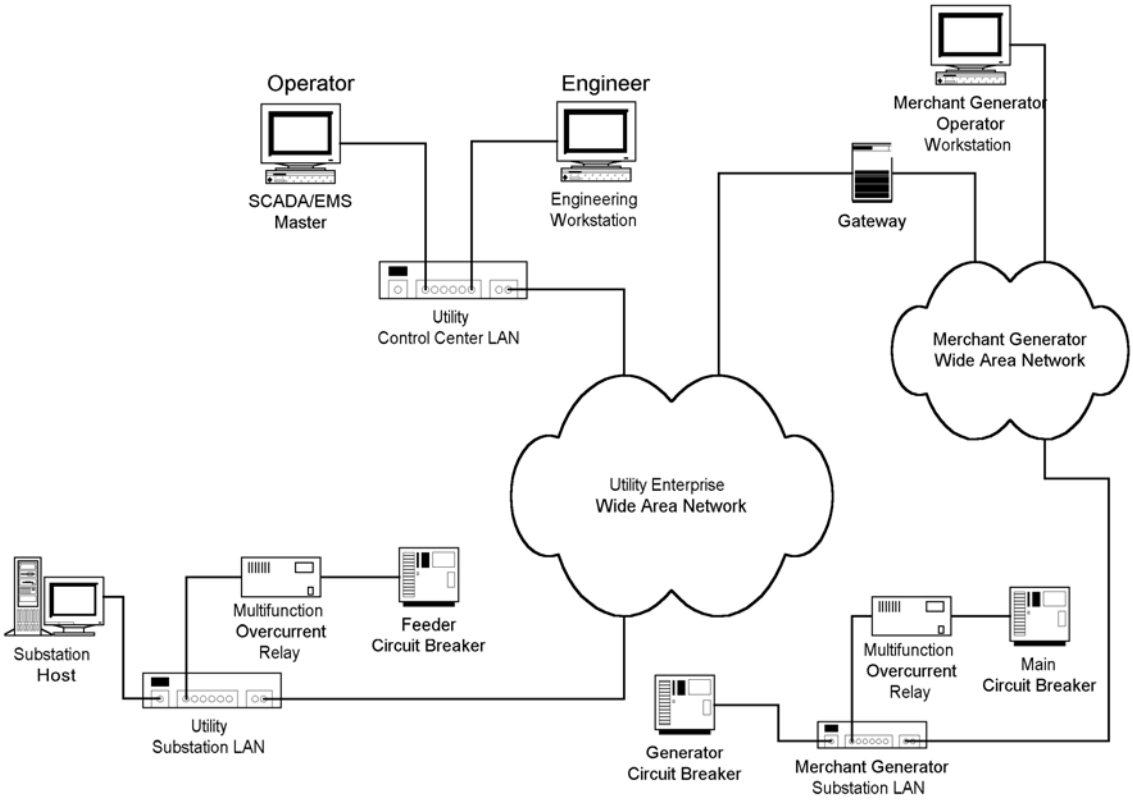

Figure D-1 Distributed generation functional configuration

D.3 Object model

An object model is defined in terms component object models for the communication interface, human interface, and power system.

D.3.1 Communication interface object model

Figure D-2~~Figure D-2~~ shows the communication interface object model. Two specializations of the class Controller are used: Gateway to provide protocol conversion and the Substation Host, which may include a Gateway for protocol conversion if the LAN and WAN protocols are different. WAN and LAN interfaces provide the communication services to transmit control, status, and data between two IEDs.

[22] An alternate architecture is to network the Utility Substation and Merchant Generator Substation without interfacing each substation to their respective WANs. A Gateway is still required to provide protocol conversion.

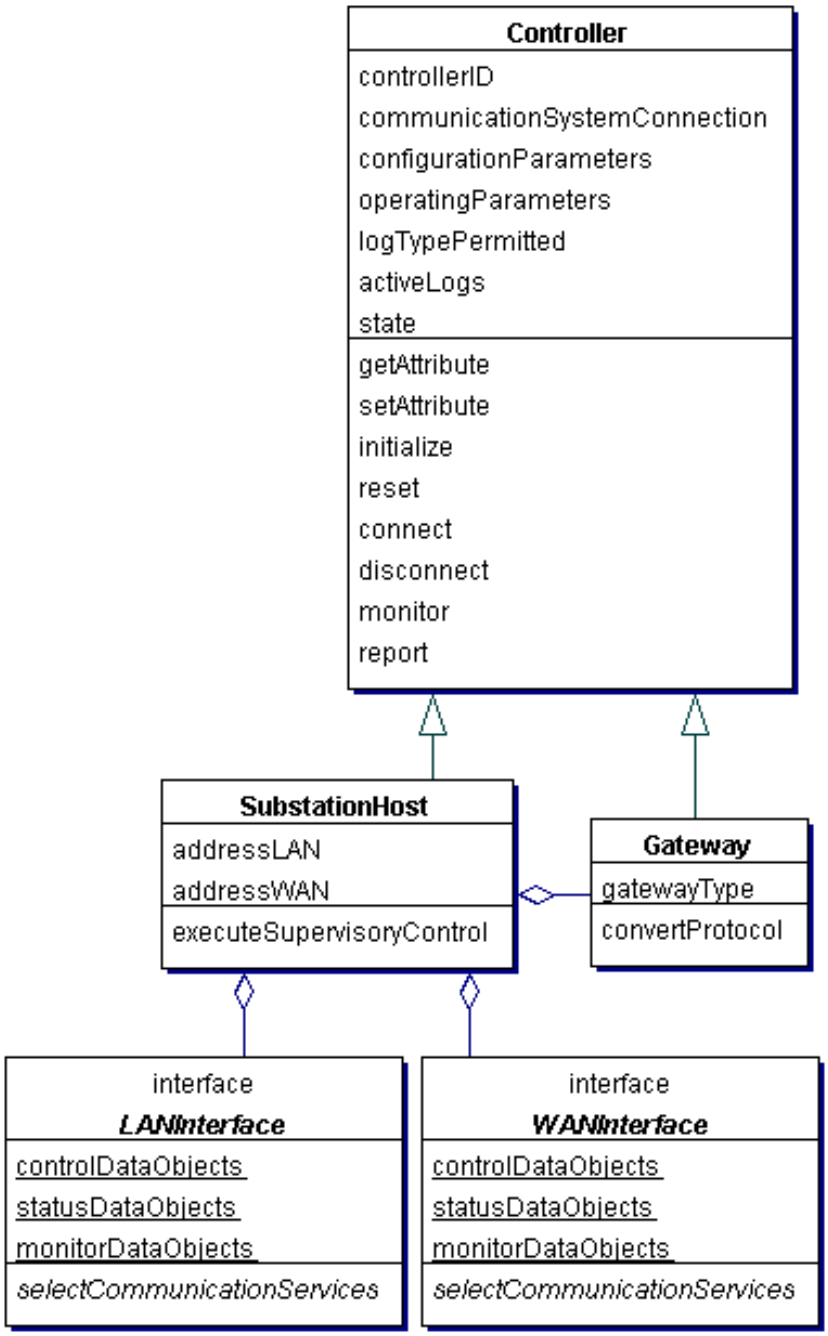

Figure D-2 Communication Interface Object Model

D.3.2 Human interface components

Figure D-3~~Figure D-3~~ shows the object model for the operator and engineer components. A workstation includes two parts: the GraphicalUserInterface and the Controller, where the GraphicalUserInterface is part-of the controller. These components provide all the capability needed for an operator or engineer to use point-click-drag-drop mechanisms to manipulate the objects on the display.

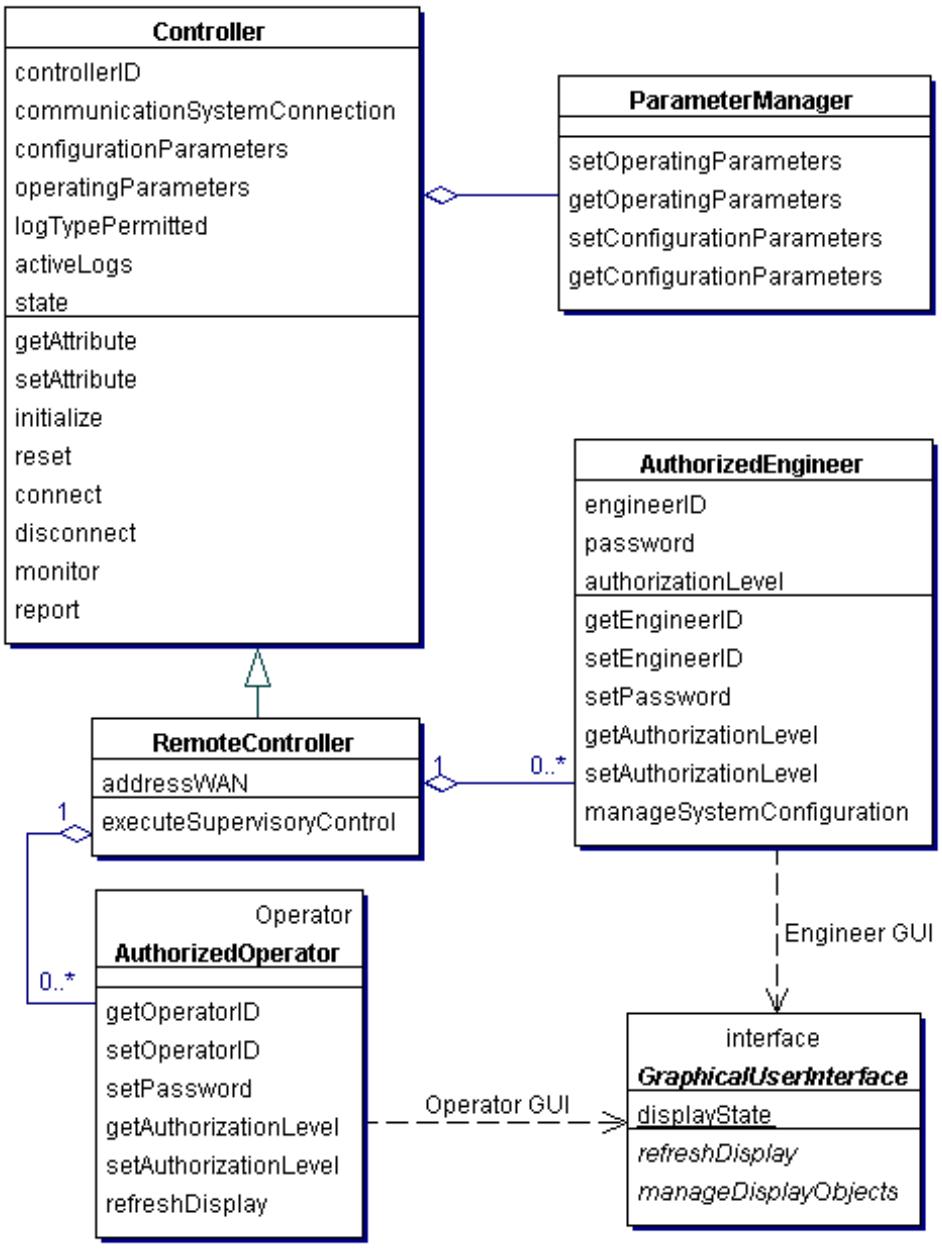

Figure D-3 Human Interface Components

D.3.3 Power system components

Figure D-4~~Figure D-4~~ shows the generalized object model for the power system components required by this scenario.

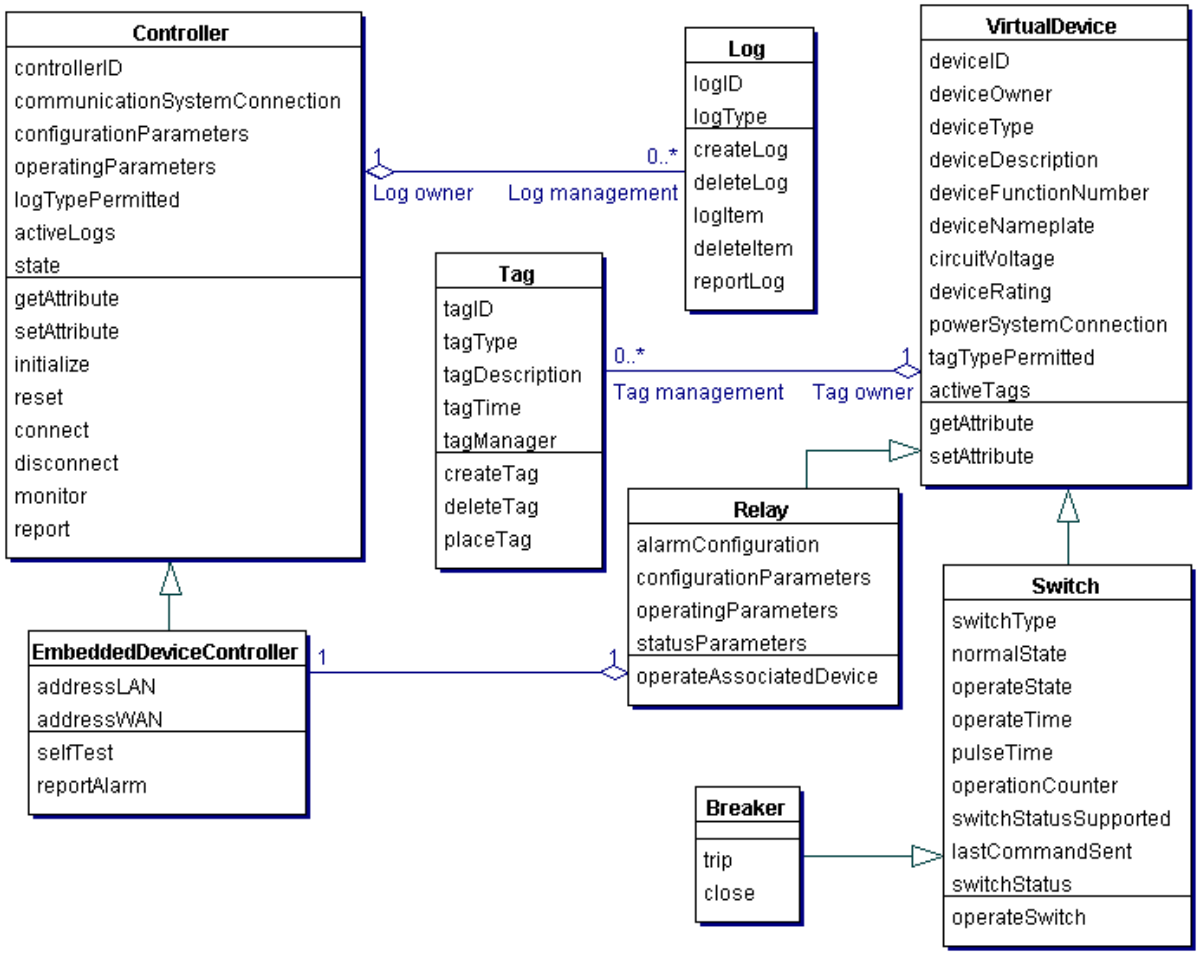

Figure D-4 Power System Components

D.3.3.1 IED communications

Figure D-5~~Figure D-5~~ shows the instances of the power system components required to implement the functional configuration in Figure D-1~~Figure D-1~~. The Merchant Generator's (Customer)GeneratorCircuitBreaker (C.GCB) has an embedded controller to provide a LAN interface within the Merchant Generator Substation. The Customer's MainCircuitBreaker (C.MCB) is not an IED. It is hardwired to the Customer's OverCurrentRelay (C.OCR), which is an IED. Therefore, C.MCB setpoints and status data must be set and fetched from the C.OCR using the Merchant Generator's LAN communication services. For the purpose of this scenario, the Merchant Generator Substation LAN is a combined hub/router to interface on the WAN.

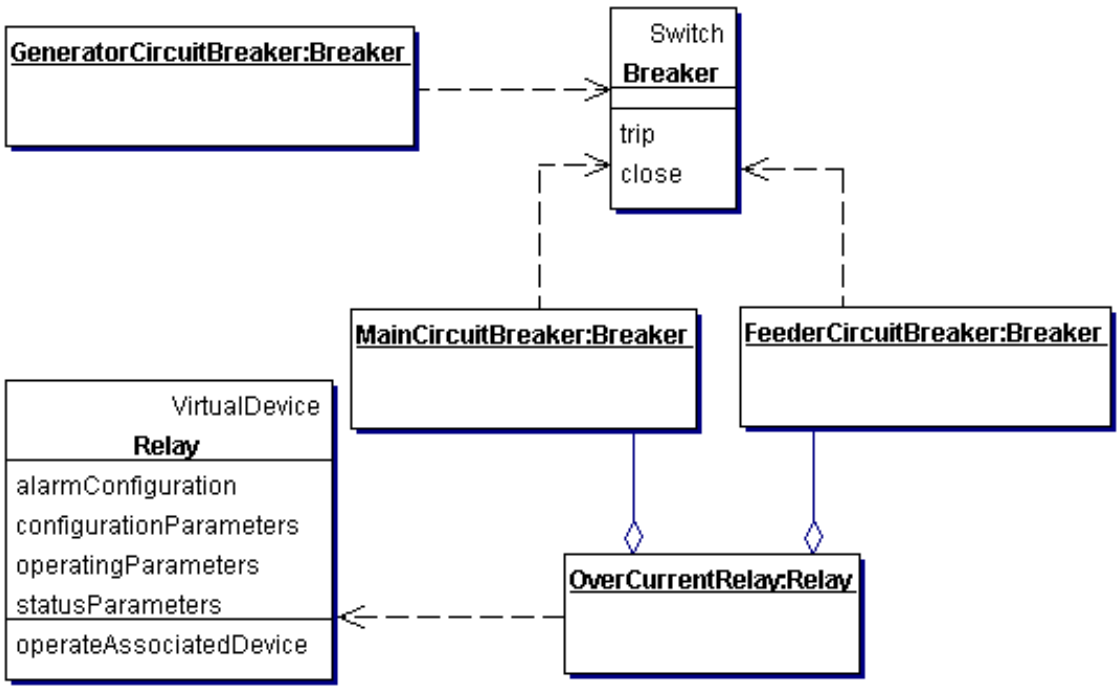

Figure D-5 Instantiation of Breaker and Relay Components

The utility's FeederCircuitBreaker (U.FCB) does not have an embedded controller and is therefore not an IED. It is hardwired to the utility's OverCurrentRelay (U.OCR), which is an IED. Therefore, U.FCB setpoints and status data must set and fetched from the U.OCR using the Utility's LAN communication services.

D.3.3.2 Operating parameters

The operating parameters for this scenario are specified in two setting groups.

D.3.3.2.1 Utility OverCurrentRelay settings group

The settings group that defines the ***operatingParameters*** for the U.OCR is:

instantaneousOverCurrentPickup
timeDelayOverCurrentPickup
timeDelayOverCurrentTimeDial
timeDelayOverCurrentType
reclosingSequence
feederClosingBlock
eventNotification

D.3.3.2.2 Merchant Generator OverCurrentRelay settings group

The settings group that defines the ***operatingParameters*** for the C.OCR is:

instantaneousOverCurrentPickup
timeDelayOverCurrentPickup
timeDelayOverCurrentTimeDial
timeDelayOverCurrentType
eventNotification

D.4 Transaction sequences

Transaction sequences are defined to establish the system configuration by downloading the settings group to each OverCurrentRelay, and for system operation.

D.4.1 System configuration

The utility engineer configures the utility power system components with the cooperation of the Merchant Generator operator who is responsible for the Merchant Generator power system components. Agreed-to procedures are required to implement this cooperative arrangement.

D.4.1.1 Configure U.OCR setting group

Figure D-6~~Figure D-6~~ shows the transaction sequence between the Utility's Engineer and the U.OCR to configure U.OCR setting group.

Using the GraphicalUserInterface the Engineer selects the operatingParameters of the settings group specified in Clause D.3.3.2.1. A message is sent to the SubstationHost over the Utility WAN to set the operatingParameters. When received, the SubstationHost logs the request, converts the protocol if the WAN and LAN protocols are not the same (not shown), and sends a message to the U.OCR to set the operatingParameters.

Figure D-6~~Figure D-6~~ includes several return message dashed-arrows that are not labeled. This notation is used to indicate a confirmation of the original message, and the return message should include any changes in state or the results of an action taken.

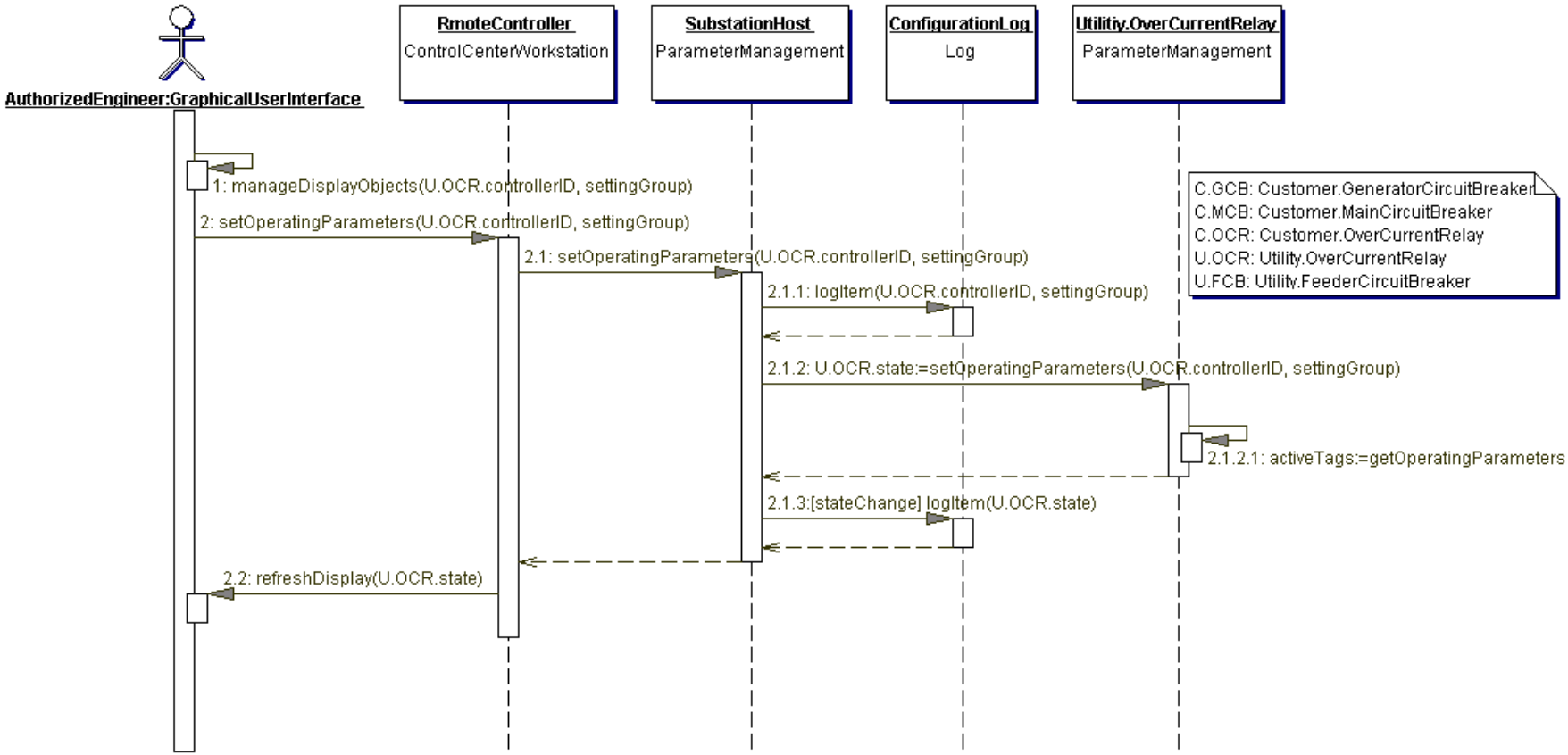

Figure D-6 Configure U.OCR setting group

When the U.OCR receives the request to setOperatingParameters, it first checks to make sure that an operational tag is not active that could inhibit operation. If a Tag exists, the U.OCR will return a message that the U.FCB is tagged and cannot be operated; this sequence is not shown.

If no tags are active, the U.OCR sends a response message to the SubstationHost specifying a change in state that reflects the change in its operatingParameters (settingGroup). The SubstationHost logs the state change, converts the protocol if the WAN and LAN protocols are different (not shown) and sends the state change to the Engineer's workstation where the display is refreshed.

A similar sequence occurs between the Engineer and U.OCR to set the permission parameters to accept reports from the C.OCR if it sends a notification (an unsolicited report) that it has tripped the C.MCB.

A similar transaction sequence (not shown) occurs between the Merchant Generator Operator and the C.OCR in the Customer substation. When this sequence is complete, the C.OCR will have permission to accept notification (an unsolicited report) that the U.OCR has tripped the U.FCB.

All changes in state for the power system components in the Utility's substation and in the Merchant Generator's substation will be indicated on all GraphicalUserInterface displays (Utility Engineer and Operator displays and Merchant Generator Operator display).

D.4.1.2 Configure C.OCR reporting

Figure D-7~~Figure D-7~~ shows the transaction sequence between the Utility's Engineer and the C.OCR to configure C.OCR reporting. Using the GraphicalUserInterface the Engineer selects C.OCR and reportEnable. RemoteController then executes setAttribute to set reportEnable=1. RemoteController sends a message to the SubstationHost over the Utility WAN to set the reportEnable=1 for C.OCR. When received, the SubstationHost logs the request, converts the protocol if the WAN and LAN protocols are not the same (not shown), and sends a message to the C.OCR to set reportEnable=1.

After the SubstationHost receives confirmation that C.OCR has successfully set the reportEnable=1, it logs the C.OCR stateChange. SubstationHost then converts the protocol (not shown) and sends C.OCR state over the Utility WAN to the RemoteController, which in turn refreshes the Engineer's display.

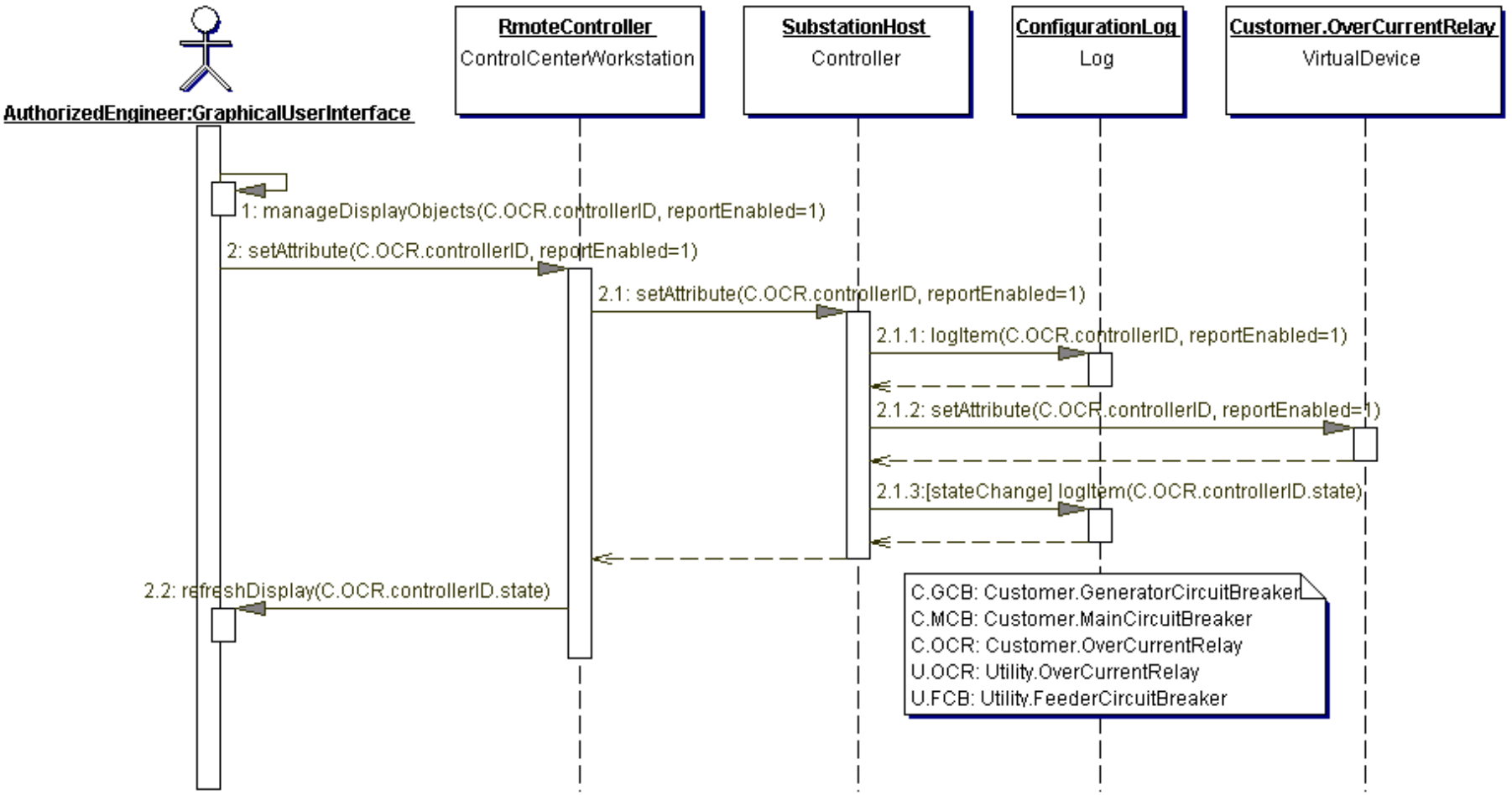

Figure D-7 Configure C.OCR reporting

D.4.2 System operation

System operation transaction sequences to monitor U.GCB and U.FCB states, and to trip C.MCB are described.

D.4.2.1 System operation to monitor U.GCB and U.FCB states

Figure D-8~~Figure D-8~~ shows the system operation transaction sequence at the Customer substation

Step 1 shows that the C.GCB IED is continually monitoring its state (C.GCB.state). Unsolicited reports to the C.OCR are sent at predefined time intervals[23], and when a change of state occurs (step 2).

From these reports, C.OCR knows the C.GCB state, which it will need to respond to a state change report from the U.OCR (described in Clause D.4.2.2).

Step 3 shows the U.OCR is monitoring the U.FCB state (U.FCB.state). If U.FCB has tripped, U.OCR will send an unsolicited report to the C.OCR

D.4.2.2 System operation to trip C.MCB

Figure D-1~~Figure D-1~~ shows the system operation to trip C.MCB. The transaction sequence begins with an unsolicited report to C.OCR from the U.OCR containing U.FCB.state.

The Gateway between the Utility WAN and the Merchant Generator WAN converts the protocol (step 1.1) and forwards the message to C.OCR (step 1.2).

When C.OCR receives the message, it first checks the U.OCR.controllerID to verify that it has received the message from an approved source[24] (step 1.2.1). If verification fails, C.OCR sends an error message to the Customer.Operator and to the Utility.Operator (not shown). It next gets the state of the C.GCB (step 1.2.2). If the verification succeeds and C.GCG is closed (C.GCG.state=closed), C.OCR trips C.BCB (step 1.2.3).

Steps 1.2.4, 1.3 and 1.4 show the transaction sequences to report C.GCB.state and C.MCB.state to C.OCR.

All state changes are logged and the Utility.Operator and Customer.Operator displays are refreshed. These transaction sequences are not shown.

Clause D.1 requires that all transactions be completed within 200 ms. These transactions include all steps shown from the beginning of step 1 through the completion of step 1.4 shown in Figure D-9~~Figure D-9~~. If additional steps (which are not shown in Figure D-9~~Figure D-9~~) are required between step 1 and step 1.4, then these steps must also be completed within the 200 ms.

A transaction sequence similar to that shown in Figure D-8~~Figure D-8~~ and Figure D-9~~Figure D-9~~ may be constructed for the case when C.OCR is monitoring the state of

---

[23] When configuring the Merchant Generator substation, the Customer engineer/operator defined the time interval for sending the C.GCB.state message. Details of configuring the Merchant Generator substation are not shown.

[24] The approved source is part of the access agreement between the Utility and Merchant Generator overcurrent relay IEDs. Details for establishing the access agreement are not shown.

C.MCB. If the C.MCB trips, then the unsolicited reports are sent from C.OCR to U.OCR. This transaction sequence is not shown.

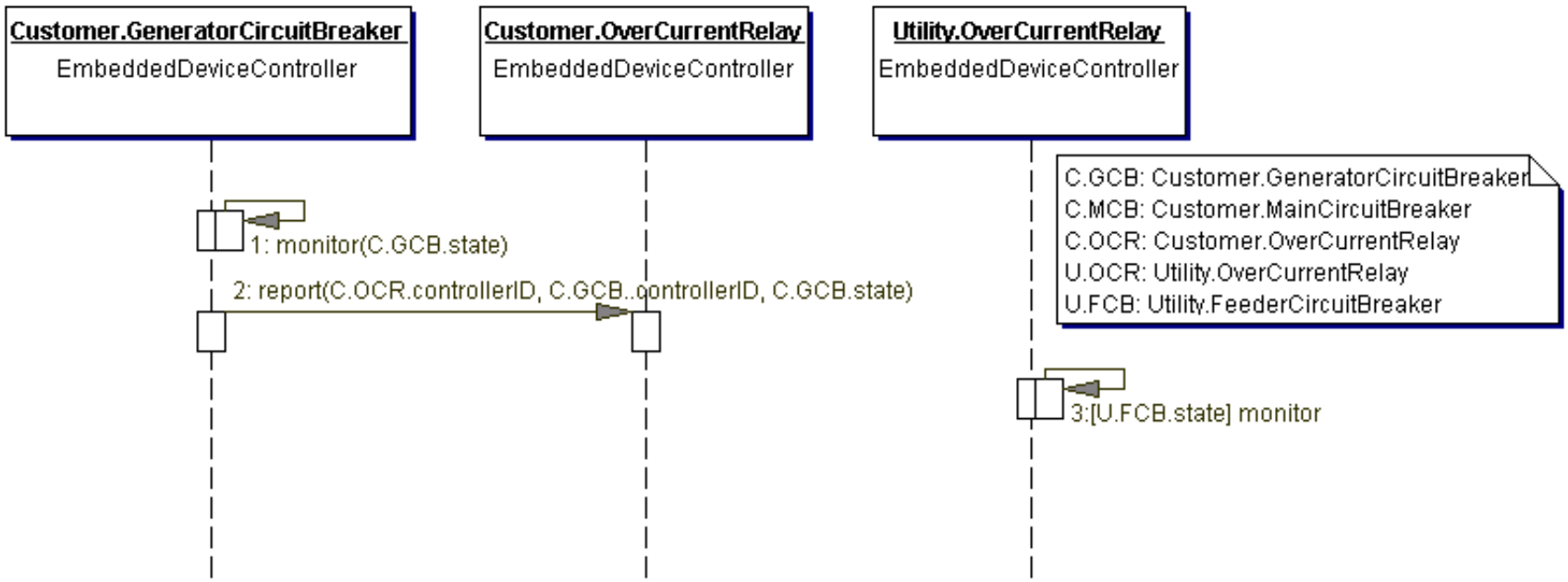

Figure D-8 System operation to monitor U.GCB and U.FCB states

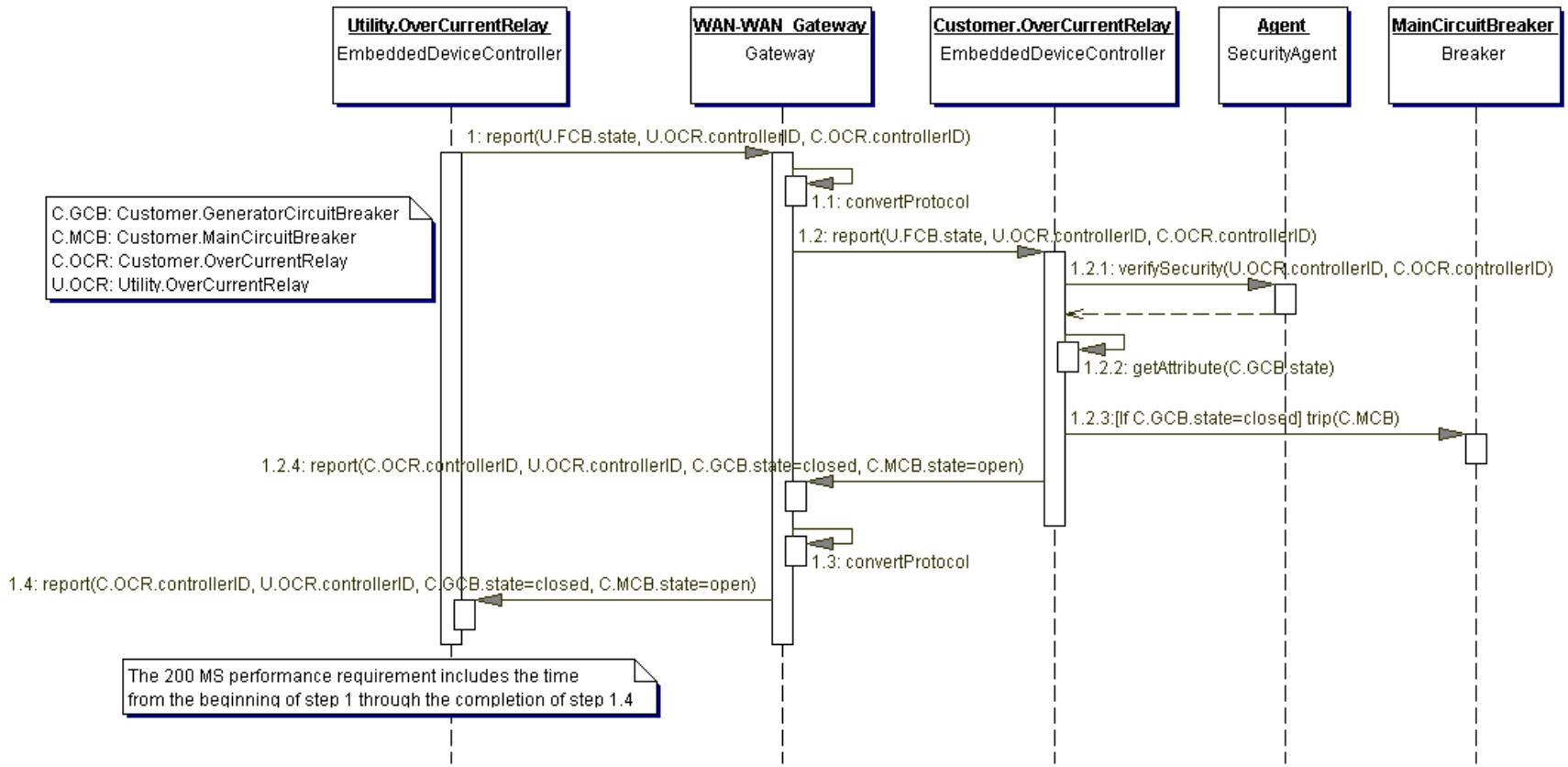

Figure D-9 System operation to trip C.MCB