Guidelines for Inter-Control Center Communications Protocol (ICCP) Implementation

Plant Controls to Dispatch Computer

SINGLE USER LICENSE AGREEMENT

THIS IS A LEGALLY BINDING AGREEMENT BETWEEN YOU AND THE ELECTRIC POWER RESEARCH INSTITUTE, INC. (EPRI). PLEASE READ IT CAREFULLY BEFORE BREAKING OR TEARING THE WARNING LABEL AND OPENING THIS SEALED PACKAGE.

BY OPENING THIS SEALED REPORT YOU ARE AGREEING TO THE TERMS OF THIS AGREEMENT. IF YOU DO NOT AGREE TO THE TERMS OF THIS AGREEMENT, PROMPTLY RETURN THE UNOPENED REPORT TO EPRI AND THE PURCHASE PRICE WILL BE REFUNDED.

1. GRANT OF LICENSE

EPRI grants you the nonexclusive and nontransferable right during the term of this agreement to use this report only for your own benefit and the benefit of your organization. This means that the following may use this report: (I) your company (at any site owned or operated by your company); (II) its subsidiaries or other related entities; and (III) a consultant to your company or related entities, if the consultant has entered into a contract agreeing not to disclose the report outside of its organization or to use the report for its own benefit or the benefit of any party other than your company.

This shrink-wrap license agreement is subordinate to the terms of the Master Utility License Agreement between most U.S. EPRI member utilities and EPRI. Any EPRI member utility that does not have a Master Utility License Agreement may get one on request.

2. COPYRIGHT

This report, including the information contained in it, is either licensed to EPRI or owned by EPRI and is protected by United States and international copyright laws. You may not, without the prior written permission of EPRI, reproduce, translate or modify this report, in any form, in whole or in part, or prepare any derivative work based on this report.

3. RESTRICTIONS

You may not rent, lease, license, disclose or give this report to any person or organization, or use the information contained in this report, for the benefit of any third party or for any purpose other than as specified above unless such use is with the prior written permission of EPRI. You agree to take all reasonable steps to prevent unauthorized disclosure or use of this report. Except as specified above, this agreement does not grant you any right to patents, copyrights, trade secrets, trade names, trademarks or any other intellectual property, rights or licenses in respect of this report.

4. TERM AND TERMINATION

This license and this agreement are effective until terminated. You may terminate them at any time by destroying this report. EPRI has the right to terminate the license and this agreement immediately if you fail to comply with any term or condition of this agreement. Upon any termination you may destroy this report, but all obligations of nondisclosure will remain in effect.

5. DISCLAIMER OF WARRANTIES AND LIMITATION OF LIABILITIES

NEITHER EPRI, ANY MEMBER OF EPRI, ANY COSPONSOR, NOR ANY PERSON OR ORGANIZATION ACTING ON BEHALF OF ANY OF THEM:

(A) MAKES ANY WARRANTY OR REPRESENTATION WHATSOEVER, EXPRESS OR IMPLIED, (I) WITH RESPECT TO THE USE OF ANY INFORMATION, APPARATUS, METHOD, PROCESS OR SIMILAR ITEM DISCLOSED IN THIS REPORT, INCLUDING MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, OR (II) THAT SUCH USE DOES NOT INFRINGE ON OR INTERFERE WITH PRIVATELY OWNED RIGHTS, INCLUDING ANY PARTY'S INTELLECTUAL PROPERTY, OR (III) THAT THIS REPORT IS SUITABLE TO ANY PARTICULAR USER'S CIRCUMSTANCE; OR

(B) ASSUMES RESPONSIBILITY FOR ANY DAMAGES OR OTHER LIABILITY WHATSOEVER (INCLUDING ANY CONSEQUENTIAL DAMAGES, EVEN IF EPRI OR ANY EPRI REPRESENTATIVE HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES) RESULTING FROM YOUR SELECTION OR USE OF THIS REPORT OR ANY INFORMATION, APPARATUS, METHOD, PROCESS OR SIMILAR ITEM DISCLOSED IN THIS REPORT.

6. EXPORT

The laws and regulations of the United States restrict the export and re-export of any portion of this report, and you agree not to export or re-export this report or any related technical data in any form without the appropriate United States and foreign government approvals.

7. CHOICE OF LAW

This agreement will be governed by the laws of the State of California as applied to transactions taking place entirely in California between California residents.

8. INTEGRATION

You have read and understand this agreement, and acknowledge that it is the final, complete and exclusive agreement between you and EPRI concerning its subject matter, superseding any prior related understanding or agreement. No waiver, variation or different terms of this agreement will be enforceable against EPRI unless EPRI gives its prior written consent, signed by an officer of EPRI.

Guidelines for Inter-Control Center Communications Protocol (ICCP) Implementation

Plant Controls to Dispatch Computer

TR-113652

Final Report, September 1999

EPRI Project Manager
R. Shankar

EPRI • 3412 Hillview Avenue, Palo Alto, California 94304 • PO Box 10412, Palo Alto, California 94303 • USA
800.313.3774 • 650.855.2121 • askepri@epri.com • www.epri.com

DISCLAIMER OF WARRANTIES AND LIMITATION OF LIABILITIES

ORGANIZATION(S) THAT PREPARED THIS DOCUMENT

EPRI Instrumentation and Control Center

ORDERING INFORMATION

Requests for copies of this report should be directed to the EPRI Distribution Center, 207 Coggins Drive, P.O. Box 23205, Pleasant Hill, CA 94523, (800) 313-3774.

CITATIONS

This report was prepared by

EPRI Instrumentation and Control Center
714 Swan Pond Road
Harriman, Tennessee 37748

Principal Investigator
Rabon Johnson

This report describes research sponsored by EPRI. The report is a corporate document that should be cited in the literature in the following manner:

Guidelines for Inter-Control Center Communications Protocol (ICCP) Implementation: Plant Controls to Dispatch Computer, EPRI, Palo Alto, CA: 1999. TR-113652.

REPORT SUMMARY

Deregulation of the electric power industry has led to sweeping changes to the manner in which power plants are controlled to meet environmental constraints and to support power transmission system stability and utility loads. Generating stations that were once base-loaded are now being cycled daily to minimum generating capacities, if not taken off-line completely. With generation costs directly impacting operations, higher-cost units are cycled the most often. As a result, an increasing number of power producers are beginning to apply the Inter-Control Center Communications Protocol (ICCP), an international standard for communication and control, to the plants where increased flexibility, accuracy, and throughput are needed for today's power production and distribution.

Background

Although the electric power industry uses real-time data extensively, there are no industry communication standards. The industry has created communication protocols on an as-needed basis only. Previously, dispatching units from central facilities was accomplished primarily by telephone with the use of some digital signals over leased lines via remote terminal units (RTUs). Also, vendors of plant control systems developed proprietary communications protocols that were unique to their system architecture. As a result, a myriad of specialized and proprietary protocols evolved. With the growth of power pools and regional centers, dramatic increases in the amount of utility data communication applications, transfers of plant ownership, and increased energy wheeling over transmission systems, these incompatible protocols pose technical and economic problems. Thus, a standard protocol was needed in the United States to support this evolution of utility communication. Developed under the guidelines of the Utility Communications Architecture (UCA™), ICCP [known as Tele-Control Application Service Element.2 (TASE.2)] bridges a critical technological gap in the industry by interfacing power plant control systems with central dispatch facilities.

Objective

This report provides guidance on ICCP implementation, enabling direct digital communication between power plant control and central dispatch systems.

Approach

EPRI initiated collaborative projects with General Public Utility Generating Company (GPU GenCo) and Potomac Electric Power Company (PEPCO) in response to identified needs in the industry to improve plant control-to-dispatch systems communication. The project team, which consisted of technical leaders and project managers from the two power producers involved in the collaborative effort, the EPRI Instrumentation and Control Center, and EPRI, established the scope, needs, and boundaries of the initiative early. It was decided that the effort would be limited to implementing and demonstrating the enabling technology and that any consequential

initiatives would be considered after proof of concept. To test operability, the parameters in the existing RTUs were used in the ICCP configuration. GPU GenCo incorporated the ICCP nodes into their corporate Local Area Network/Wide Area Network (LAN/WAN) infrastructure, while PEPCO opted to utilize an available, separate, stand-alone fiber optic network. This provided for proof of concept of the operability of ICCP in either structure.

Results

Implementing ICCP on a utility's LAN/WAN infrastructure appears feasible and cost-effective. Sufficient security features have been designed into ICCP to provide for the protection of corporate information and to ensure that the unit control is adequately protected from outside influences. The application and use of bilateral tables provides protection from transmission of undesired or extraneous information. By deploying ICCP in power plant control systems, power producers will benefit from the improved accuracy, speed, and flexibility available in the protocol. ICCP has effectively become the national standard for communication between service areas. Implementing the protocol in power plant control systems improves unit dispatch functionality through enhanced flexibility.

EPRI Perspective

ICCP provides a common way for all power producers to exchange data among control centers, power plants, and substations. Incorporating ICCP into plant controls and communication will enable quicker and more flexible response for load demands and transmission grid stability and support. The enabling technology will assist power producers to be more agile and responsive when business success requires quick and accurate management decisions.

TR-113652

Keywords

Communication
Real-time systems
Utility communication architecture
Dispatching
Generator control
Transmission stability

ACKNOWLEDGMENTS

This project was initiated in mid-1995 by Dave Becker and Joe Weiss, EPRI Palo Alto. Through visits to both GPU and PEPCO, the potential of ICCP was demonstrated. The resulting collaborative project was managed by Joe Weiss. The technical guidance and project management expertise of Dave Becker and Joe Weiss directly contributed to the success of this project.

John Brummer, GPU GenCo, and Frank Bennett, PEPCO, provided technical coordination, while Ken Gray, GPU GenCo provided project management.

Rabon Johnson and Rob Frank, EPRI Instrumentation and Control Center, provided project coordination from an EPRI perspective.

ABSTRACT

This report presents the results of two demonstration projects undertaken to improve digital communications in the electric power utility industry. It also provides guidelines for implementing ICCP within a utility. The enabling technology, Inter-Control Center Communications Protocol (ICCP) Version 6.1, is a digital communications protocol that, in 1996, became a draft international standard under the International Electrotechnical Commission, known as IEC 870-6-802 TASE.2.

General Public Utilities (GPU) and Potomac Electric Power Company (PEPCO) implemented separate but similar projects for demonstrating ICCP as an interface between their dispatch computers and plant control systems. GPU has implemented ICCP in three vendors' systems: Honeywell at Conemaugh, Bailey at Shawville, and Westinghouse at the Portland station. PEPCO implemented ICCP in their MAX Controls system at the Potomac River station and the Foxboro system at Chalk Point. GPU implemented ICCP on their corporate Local Area Network/Wide Area Network (LAN/WAN) infrastructure, and PEPCO implemented ICCP on a separate network, using existing fiber optic circuits for the communication highways. Through the use of selected applications, the project provided the ability to transmit/receive data and objects between their Energy Management System (EMS) and plant controls.

Until recently, direct control of generating units from central dispatch facilities was minimal, consisting primarily of voice telephone communication supplemented by indirect digital signals on leased telephone lines. Algorithms in the central dispatch computers based decisions for cycling units on plant monthly report data used to calculate system economic dispatch needs. Oftentimes, original plant design data was used in the calculations, and real-time plant process information and equipment status was not factored into the calculations. ICCP will enable automation of this functionality and improve flexibility and accuracy of unit dispatch based on more current information than has been used in the past.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

1
INTRODUCTION

1.1 Purpose

This report presents an overview of the application and demonstration of Inter-Control Center Communications Protocol (ICCP) as an interface between leading Distributed Control System (DCS) vendors' systems and the Energy Management Systems at two participating utilities. Section 11 discusses the GPU GenCo project and Section 12 provides an overview of the PEPCO initiative.

This report also provides guidance to users of the Inter-Control Center Communications Protocol (ICCP), otherwise known by its official name of Tele-Control Application Service Element.2 (TASE.2). Throughout this document the name ICCP will be used, except where specific references are made to the International Electrotechnical Commission (IEC) standards. In any case, it should be clear that there is only one protocol and set of specifications that can be referred to as either ICCP or TASE.2.

Although a Draft International Standard (DIS) for ICCP currently exists, it is by necessity written in the style dictated by the IEC, the standards organization sponsoring the DIS. This style has been developed to specific international standards in a precise and unambiguous way so that all implementers will interpret the Standard in the same way and, thus, ensure interoperability between different vendor's ICCP products.

The style of the ICCP Draft International Standard is not necessarily readable for someone who is not intimately familiar with all of the background leading up to the development of ICCP. Furthermore, certain types of information that would be very useful to a user of ICCP, but that are not necessary for specifying the protocol or services provided by ICCP, have been omitted. To provide a general feel for the Standards and for related information, Appendices E through G are reproduced from the Introduction, Scope, and other beginning sections of the Standards.

1.2 Intended Audience

This report is intended as an information source for power producers faced with a need to advance or improve communication between dispatch facilities and their plant control systems. While it discusses issues involved in implementing ICCP, the report is intended for a broad audience of readers. This audience can range from an end-user trying to decide if ICCP is appropriate for their data transfer needs to a vendor planning to implement ICCP. In particular, this guide should be helpful to:

- An end-user, such as an electric utility, with the need for assessing the transfer of real-time data to another utility or utilities, or to another internal control center, who is trying to evaluate which protocol is most appropriate
- An end-user, such as a power producer, with a need for implementing direct digital communication between their plant control systems and the dispatch system
- An end-user who already has decided to use ICCP and now needs guidance in how to procure ICCP
- An end-user who has procured ICCP and now is concerned about exactly how to map their actual data into ICCP data objects
- An end-user who is looking for conventions and answers to practical questions regarding configuring ICCP software and networks
- A vendor with a project to implement the ICCP specification as a specialist

1.3 Organization of the Report

This report first introduces the background and concepts of ICCP, thus providing a framework for understanding the ICCP specification. Following this introduction, the individual server and data objects comprising ICCP are described with cross-references to the specification.

More specifically, Section 1 provides an overview of ICCP implementation. Section 2 covers the basics of ICCP technology and specifications for implementation. Sections 3 through 5 consider implementation issues. Sections 6 and 7 cover network configurations and security issues. Section 8 covers ICCP interconnectivity protocol and Sections 9 and 10 cover procurement and network management issues. Section 11 provides an overview of the GPU GenCo demonstration project, which interfaced the dispatch computer via ICCP to the Bailey, Honeywell, and Westinghouse Distributed Control Systems (DCSs). Section 12 discusses the PEPCO project in which their dispatch system was interfaced to the Foxboro and MAX Controls systems. Appendix A contains definitions and supporting terms used throughout the report. Appendix B is a list of abbreviations; Appendix C lists the report references; Appendix D is an overview of ICCP Specs, and Appendices E through G are excerpts from the ISO Standards.

1.4 ICCP Version Number

This version of the ICCP User Guide applies to ICCP specifications IEC 870-6-503 and 870-6-802 Version 1996-08, which is also informally known as ICCP Version 6.1. See the references in Appendix D for more complete identification of the specifications to which this guide applies, and Appendices E through G for excerpts from the referenced standards.

2
ICCP

2.1 Introduction

Inter-utility real-time data exchange has become critical to the operation of interconnected systems within the electric power utility industry. The ability to exchange power system data with boundary control areas and beyond, provides visibility for disturbance detection and reconstruction, improved modeling capability, and enhanced operation through future security control centers or independent system operators.

Historically, utilities have relied on in-house or proprietary, non-ISO standard protocols such as Western System Coordinating Council (WSCC) Energy Management System Inter-Utility Communications (WEIC), ELCOM, and the Inter-Utility Data Exchange Consortium (IDEC) to exchange real-time data. ICCP began as an effort by power utilities, several major data exchange protocol support groups [WSCC Energy Management Systems Inter-Utility Communications Guidelines (WEICG), IDEC, and ELCOM)], EPRI, consultants, and a number of Supervisor Control and Data Acquisition/Energy Management System (SCADA/EMS) and DCS vendors to develop a comprehensive international standard for real-time data exchange within the electric power utility industry. This included provisions for direct interface to plant control systems, thereby eliminating the need for Remote Terminal Units (RTUs).

By giving all interested parties an opportunity to provide requirement input and to participate in the protocol definition process, it was expected that the final product would both meet the needs of and be accepted by the electric power utility industry. To accomplish this goal, the Utility Communications Specification (UCS) Working Group was formed in September 1991 to:

- Develop the protocol specification
- Develop a prototype implementation to test the specification
- Submit the specification for standardization
- Perform an interoperability test among the developing vendors

UCS submitted ICCP to the IEC Technical Committee (TC)-57 Working Group (WG)-07 as a proposed protocol standard. Another proposed standard based on ELCOM-90 over the Remote Operations Service Element (ROSE) was also being considered by WG-07. TC-57 decided on a multi-standard approach to allow a quick implementation necessary to meet European Common Market requirements by 1992, and to allow long term development of a more comprehensive protocol. The first protocol was designated TASE.1 (Tele-Control Application Service Element.One). The second protocol, based on ICCP over the Manufacturing Messaging Specification (MMS), was designated TASE.2.

Successful first implementations of ICCP between SCADA/EMS control centers led to further expansion to allow communications between control centers and power plants. This expansion did not impact the basic services, but did lead to development of specific power plant objects. These objects have now been incorporated into ICCP.

2.2 Overview of Protocol Concepts

2.2.1 Protocol Architecture

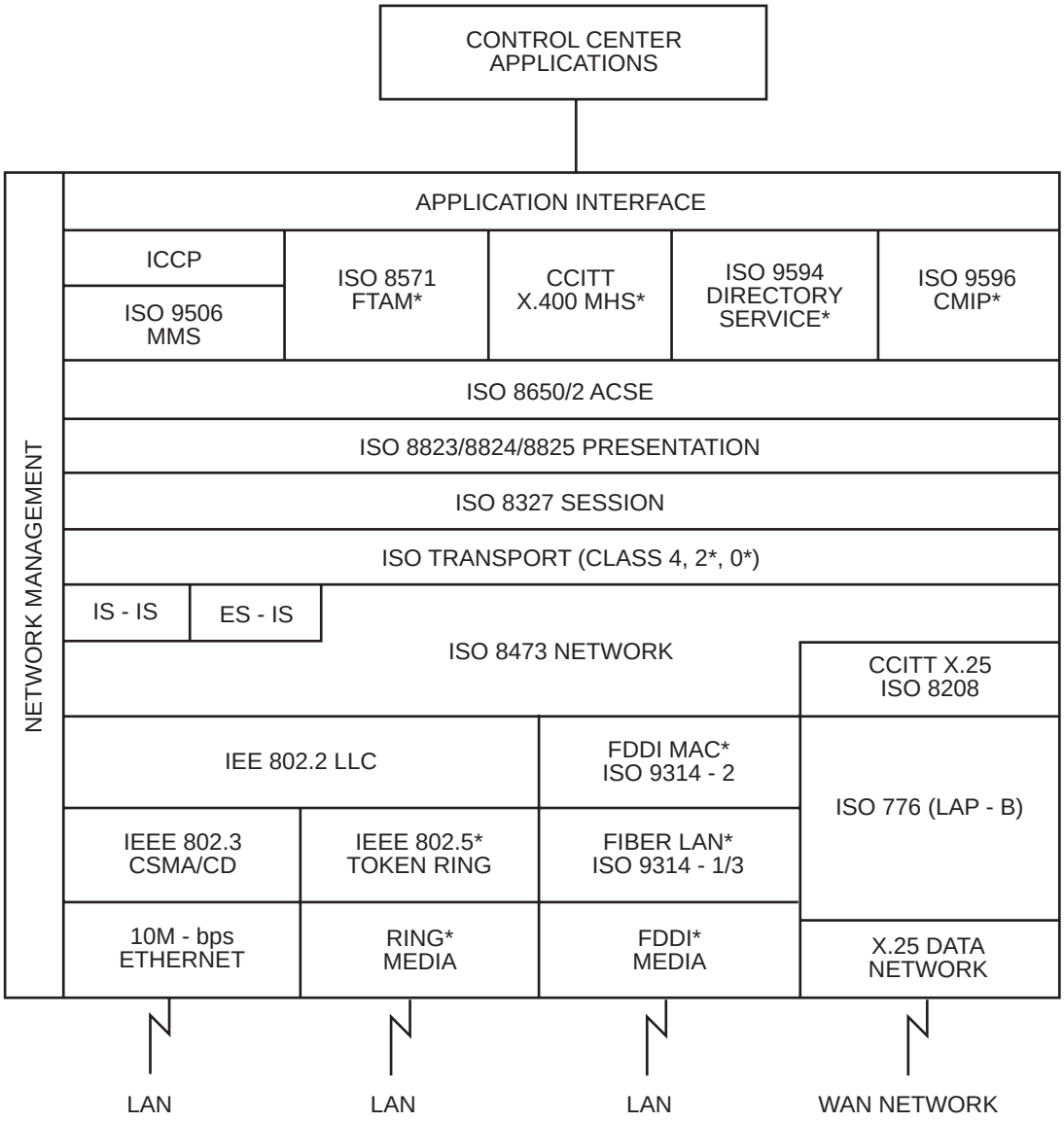
Additional information regarding the specifics about ICCP can be found in Appendix D, reproduced from Section 3 of EPRI Report TR-105800, *Intercontrol Center Communications Protocol (ICCP) Demonstration*. However, the information that follows provides an overview of the ICCP concepts.

ICCP maximizes the use of existing standard protocols in all layers up to and including the lower layers of Layer 7 in the Open Systems Interconnection (OSI) reference model. This has the benefit of requiring new protocol development for ICCP only in the upper sub-layer of Layer 7.

The protocol stack used by ICCP is the standard Utility Communication Architecture (UCA™) Version 1.0 profile with control center applications at the top. ICCP specifies the use of the Manufacturing Messaging Specification (MMS) for the messaging services needed by ICCP in Layer 9 (see Figure 2-1). MMS specifies the mechanics of naming, listing, and addressing variables, and of message control and interpretation, while ICCP specifies such things as the control center object formats and methods for data requests and reporting. Applications at different control centers, possibly written by different vendors, but both conforming to these mechanics, formats, and methods, might inter-operate to share data, control utility devices, output information messages, or define and execute remote programs via an Application Program Interface (API) to ICCP.

ICCP also utilizes the services of the Application Control Service Element (ACSE) in Layer 7 to establish and manage logical associations or connections between sites. ICCP relies on the ISO Presentation Layer 6 and Session Layer 5 as well.

Early drafts of UCA™ Version 2.0 include additional sub-network types in Layers 1-2, including Frame Relay, ATM, and Integrated Services Digital Network (ISDN). It also specifies the use of Transmission Control Protocol/User Datagram Protocol (TCP/UDP) as an alternative transport protocol. Because of the protocol architecture, ICCP is independent of the lower Layers, so that, as new protocols evolve in the lower layers, ICCP will be able to operate over them with only configuration changes. Thus, ICCP is able to operate over either an International Standards Organization (ISO)-compliant Transport Layer or a Transmission Control Protocol/Internet Protocol (TCP/IP) transport service, as long as ISO Layers 5-7 are maintained.



* Indicates optional function

Figure 2-1
UCA™/UCS ICCP Protocol Architecture

2.2.2 Application Program Interface (API)

Although an API is shown in Figure 2-1, the API is not specified in the ICCP specification—only the protocol and service definitions are specified and are the subject of standardization. Each vendor implementing ICCP is free to choose the API most suitable for their product or for their intended customers. Figure 2-2 illustrates this concept.

For example, a SCADA/EMS vendor might choose to provide an API optimized for interfacing with several different types of applications, such as:

- A proprietary real-time SCADA database for storing and retrieving real-time power system data such as analogs, status, and accumulator values, on a periodic basis or when a value changes
- A Relational Database Management System (RDBMS) for storing and retrieving historical or other non-real-time data
- Scheduling and accounting applications to send, for example, C structures containing interchange schedules once an hour or once a day, and binary files containing accounting data spreadsheet files
- Dispatcher console operator message application and/or alarm processor application to send ASCII text messages to be displayed on a dispatcher's console display at another control center

These are just a few examples of the types of APIs that a SCADA/EMS vendor might provide for its ICCP product. How they are implemented is considered a "local implementation issue" in the ICCP specification. As long as the protocol services are implemented according to the specification, interoperability is assured between different ICCP vendors' products.

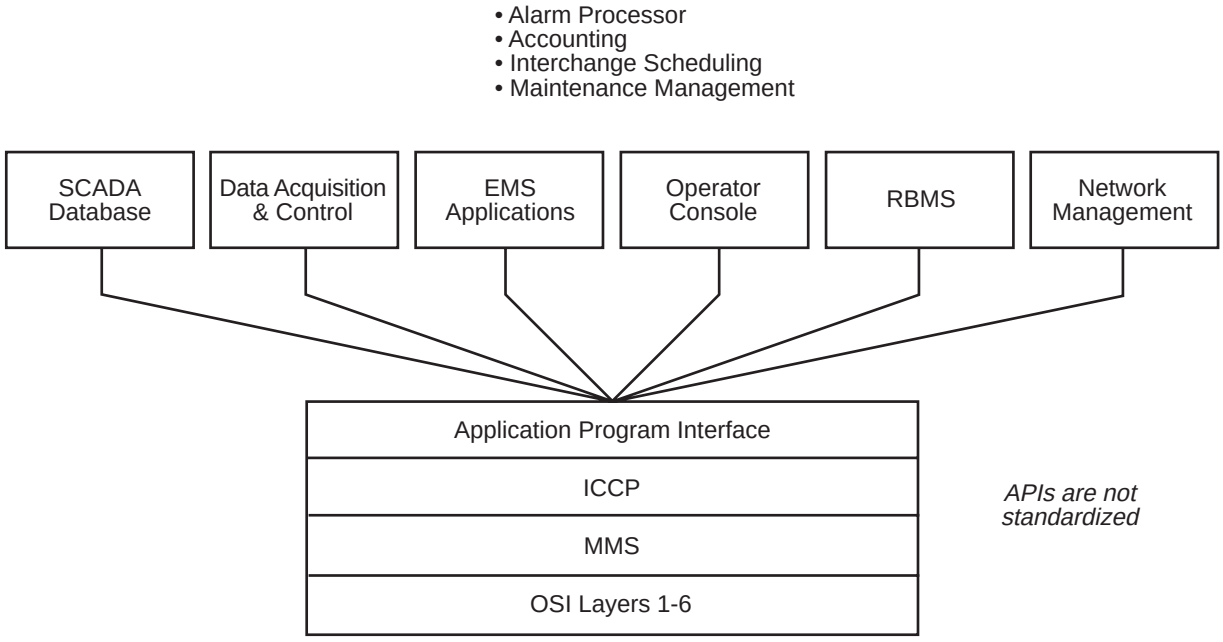


Figure 2-2
Application Program Interface (API)

2.2.3 Client/Server Model

ICCP is based on client/server concepts. All data transfers originate with a request from a control center (the client) to another control center that owns and manages the data (the server). For example, if a Control Center X application needs data from the Control Center Y SCADA database, the Control Center X application, acting as the client, might request Control Center Y, acting as the server, to send the data under conditions specified by the client.

There are various services provided in ICCP to accomplish data transfers, depending on the type of request. For example, if the client makes a one-shot request, the data will be returned as a response to the request. However, if the client makes a request for the periodic transfer of data or the transfer of data only when it changes, then the client will first establish the reporting mechanism with the server (that is, specify reporting conditions such as periodicity for periodic transfers or other trigger conditions such as report-by-exception only), and the server will then send the data as an unsolicited report whenever the reporting conditions are satisfied.

2.2.4 Multiple Associations and Sites

ICCP uses the ISO Association Control Service Element (ACSE) to establish logical associations. Multiple associations can be established from a client to multiple, different control center servers. Although ICCP can be operated over a point-to-point link, it is envisioned that most installations will operate over a router-based Wide Area Network (WAN). As noted previously, ICCP is independent of the underlying transport network, so any combination of sub-networks might comprise the WAN, including the Local Area Networks (LANs) within a site.

Multiple associations can also be established to the same control center for the purpose of providing associations with different Quality of Service (QOS). An ICCP client then uses the association with the appropriate QOS for the operation to be performed. For example, to ensure real-time data messages are not delayed by non-real data transfers, both a High and Low priority association can be established, with a separate message queue for each. ICCP will check the High priority message queue and service any messages queued before serving the Low priority message queue. This permits a common data link to be shared for both the transfer of High priority SCADA data and for lower priority information message transfers.

Figure 2-3 illustrates an ICCP network serving four utilities. As shown, Utility A is a client to server C (Association C1) and a server for four associations: two to client C (Association A1 and A2), one to client B (Association A3), and one to client D (Association A4). The association to client B (A3) would presumably be accomplished via a router at Utility C but could follow any path available if a WAN is provided to interconnect all utilities. Each of the other utilities shown has similar associations established to meet their individual needs. Utility D functions only as a client. Utilities B and C function as both clients and servers. The point made by this diagram is that ICCP provides the capability for any type of interconnectivity needed via configuration of the ICCP software.

ICCP permits either a client or a server to initially establish an association. It further permits an established association to be used by either a client or server application at a site, independent of how the association was established. The Protocol Implementation Conformance Statement (PICS) performance specifies how associations are used in any actual configuration of ICCP.

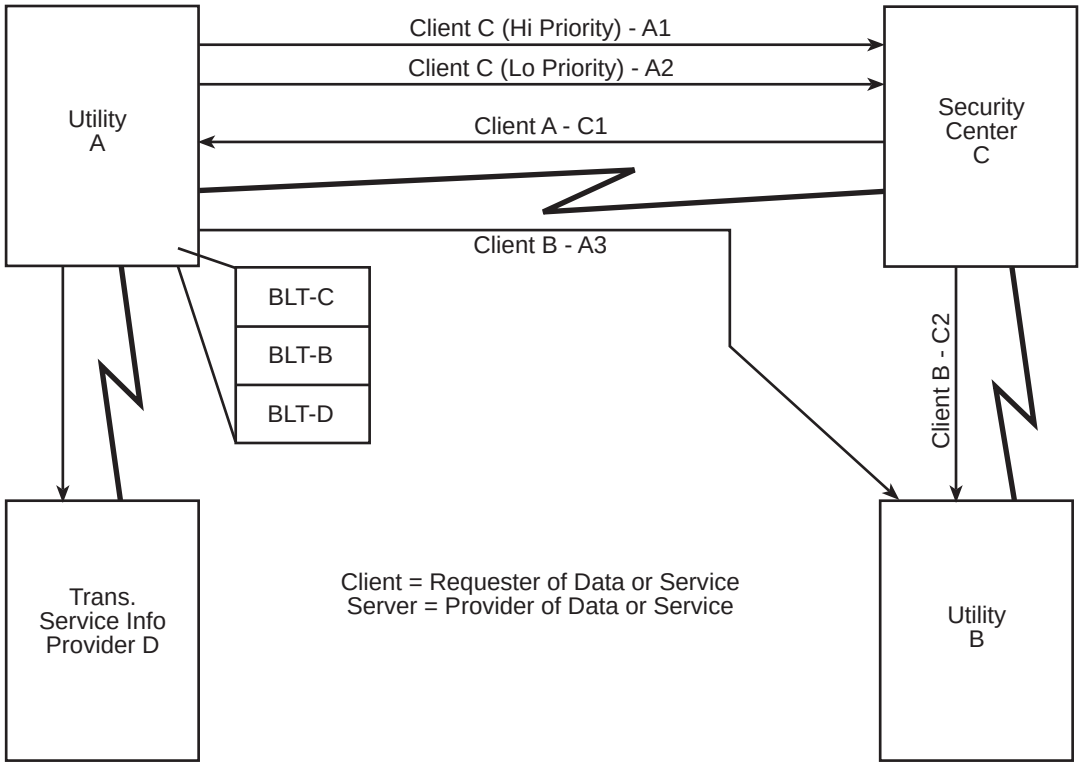


Figure 2-3
ICCP Client/Server Model with Multiple Associations

2.2.5 Access Control via Bilateral Tables

To provide access control, the server checks each client request to ensure that the particular client has access rights to the data or capability requested. Access control is provided through the use of Bilateral Tables (BLTs) defined for each client/server association. BLTs provide execute, read/write, read only, or no access for each item that can be requested by a client.

For example, as shown in Figure 2-3, Utility A maintains a separate BLT for each utility, permitting different access rights for the clients at each utility.

2.2.6 Use of Object Models

Object model concepts are used in two different ways in ICCP. Figure 2-4 illustrates these concepts.

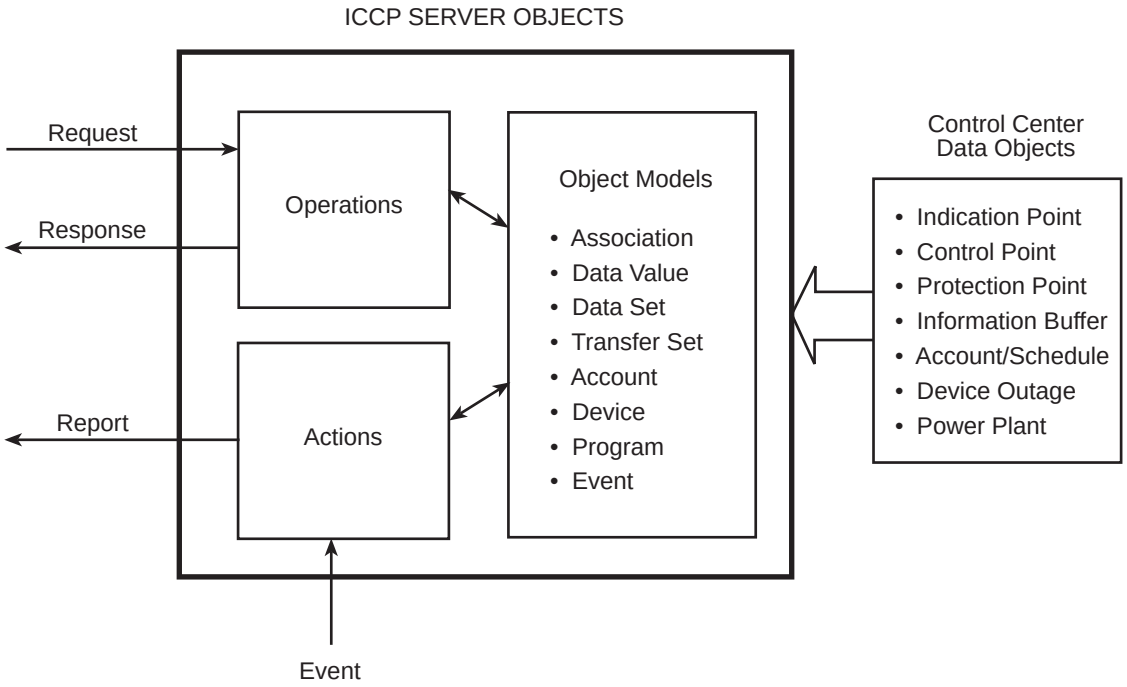


Figure 2-4
ICCP Object Models

2.2.6.1 ICCP Server Objects

First, all ICCP services are provided via ICCP server objects, which can be thought of as classical objects with data attributes and methods as defined in object-oriented design methodologies. There are two basic types of methods in ICCP called operations and actions. An operation is client-initiated via a request to a server, typically followed by a response from the server. An action, on the other hand, is a server-initiated function. An example of an action is the transfer of data via a report to a client in response to a timer expiring or some other external event at the server, such as a change in status of a breaker.

IEC 870-6-503 contains all of the ICCP server object definitions. Sometimes referred to as "internal" objects, these objects are required to implement the ICCP protocol. Explanations of these objects are included in this guide in the ICCP Server Object Description section.

2.2.6.2 ICCP Data Objects

Second, all other data and control elements typically exchanged between control centers are defined as "data objects." These range from simple to complex data structures. In contrast to the "server objects," these objects are not required to implement the ICCP protocol, and so are sometimes referred to as "external" objects.

The standard Control Center Data Objects are defined in IEC 870-6-802. They are also described in this guide in the Conformance Block section. Supported data types include control messages,

status, analogs, quality codes, schedules, text and simple files. Furthermore, additional data objects can be defined by ICCP users and transferred using existing ICCP server objects with no change in the ICCP protocol software contained in IEC 870-6-503. The approach to defining new data objects is described in this guide in the section titled Definition of New Data Objects.

2.2.6.3 Object Model Notation

The ICCP specification uses a formal method of describing objects. The first level is known as an Abstract Object model. This model comprises a Name for the model, followed by a list of Attributes, headed by one attribute known as the Key Attribute. In some cases the attribute listed is actually another object model inherited by the new object model. The meaning of each attribute is provided after the formal object model is presented.

Some object models, especially those used to describe control center data objects, contain Constraints, which provide alternative lists of attributes within a single object model. These constraints provide some flexibility in how the object can be used. All abstract models are described first in the specification.

Abstract object models then have to be mapped to concrete structures with components. Each component is mapped to a data type. The services are mapped to MMS services. This must be specified to ensure that each implementer of ICCP uses the same data types and MMS services to implement the abstract models so that interoperability can be achieved with other vendors' ICCP products.

Section 7 in IEC 870-6-503 describes in more detail the organization of ICCP specification and the use of these models.

2.2.6.4 Conformance Blocks and Services

Conformance blocks are defined for server objects of ICCP in Section 9 of 870-6-503 as a way of grouping ICCP objects together to provide fundamental types of services. A vendor need not implement all defined conformance blocks (nine in all). However, any implementation claiming conformance to ICCP must fully support Block 1, as defined in Section 2.3 of this report. Likewise, an ICCP end user need not procure all ICCP conformance blocks, only the ones actually needed to meet the user's data transfer requirements.

Conformance blocks are also defined for data objects in Section 9 of 870-6-802 as a way of specifying which server object services are needed to transfer each data object.

2.2.6.5 ICCP Specification Organization

The ICCP specification organization is dictated by the rules and guidelines governing International Standards Organization/International Electrotechnical Commission (ISO/IEC) standards documentation. The IEC numbers for the three parts of the specification were assigned by the IEC, basically by assigning the next sequential numbers available in the 870-6-500, –700, and –800 series of documents. The 500 series numbers are reserved for protocol standards and

services specifications. The 700 series is reserved for Application profiles. The 800 series is reserved for Information Structure profiles, also know as Interchange Format and Representation profiles. This follows the classification scheme adopted for OSI functional profiles.

This section explains how the IEC documents are organized and why (that is, separation into parts 503, 702, and 802).

870-6-503

ICCP Part 503, known officially as IEC 870-6-503, TASE.2 Services and Protocol, defines the mechanism for exchanging time-critical data between control centers. The data exchange mechanism is defined in terms of ICCP server object models. It defines a standard way of using the ISO/IEC 9506 MMS services to implement the data exchanges.

A document that defines a standard way to use selected MMS services for exchanging electric utility data is known as an MMS Companion Standard for Electric Utility Data Exchange. And since MMS Companion Standards must follow a consistent format dictated by the MMS standards development groups, 870-6-503 is formatted to the guidelines established for MMS Companion Standards. This means that readability is sometimes sacrificed to follow these guidelines.

The ordering of the information presented in the document follows the “onion skin” analogy. That is, reading this document is like peeling off multiple layers of onionskin, with each new layer taking the reader to a deeper level of specification. This means that the same models are discussed at different levels several times throughout the specification. The order is as follows:

Layer 1

Section 5.1: Informal ICCP Model Description. The informal model of ICCP describes the various ICCP server objects in the context of the utility control center environment using plain English narrative.

Layer 2

Section 5.2: Formal ICCP Model Description. The formal model covers the same ground, but here more formality is introduced. Specifically, the entire control center with its software applications that are involved in data exchange are represented as a Virtual Control Center (VCC), comprising several object models. In this section, formal abstract models with attributes are introduced. Some models are presented as a hierarchy of object models, each of which is described. Each attribute for each object model is defined. Each operation and action is described again in more detail.

ICCP

Layer 3

This layer covers three major sections:

Section 6: Mapping of ICCP Object Models onto MMS Object Models. In this section, the abstract object models are repeated, but this time each attribute is mapped directly to either a basic MMS attribute or to a more complex ICCP data type defined in Section 8, so that standard MMS protocols can be used for the actual transmission of data. For example, in Section 5.2 the Data Set Name attribute of the Data Set object model is defined as "the attribute that uniquely identifies the Data Set." In Section 6, the description for the Data Set Name attribute states that "this attribute shall be represented as the MMS Variable List Name attribute."

Section 7: Mapping of ICCP Operations and Action onto MMS Services. This section does for operations and actions what Section 6 does for attributes. It maps them in only MMS services, describing both the client and server roles in sufficient detail that a software vendor can implement each service in such a way that interoperability with other vendors' ICCP products is assured.

Section 8: Standardized Application-Specific Objects. This section specifies certain ICCP objects and complex data types used in Section 6 and maps them onto MMS standard objects and basic MMS data types. This section deals only with the objects required for ICCP internal use as distinguished from control center data objects, which are the subject of 870-6-802.

The last part of 870-6-503, Section 9, defines the Conformance Blocks, which are described elsewhere in this guide.

870-6-802

ICCP Part 802, known officially as IEC 870-6-802 TASE.2 Object Models, defines the control center data objects, which represent the control center data actually exchanged between control centers. This document is structured in a fashion similar to 870-6-503, but with only two layers:

Layer 1

Section 5: Object Models. This section defines the standard abstract object models for the data to be exchanged with ICCP. This uses the same notation that was used in 870-6-503 to describe the ICCP server object models, defining each attribute for each model. This is the section to browse or read to determine if there are appropriate standard objects available to meet a specific utility's data exchange requirements. It is organized based on the classes of data typically exchanged between control centers. The order is Supervisory Control and Data Acquisition, Transfer Accounts, Device Outage, Information Buffer, and Power Plant objects.

Layer 2

Section 6: MMS Types for Object Exchange. This section defines the data types to be used for exchanging the standard objects. This includes basic types, such as Data_Descrete, which is defined as an integer (width 32), but also includes complex data types based on the abstract

models defined in Section 5. Each abstract model must be mapped to one or more concrete object types, which are defined in terms of structures with components. For example, an Indication Point object, which contains an analog point value with quality and time tag (but not change-of-value counter), is mapped to the Data_DescreteTimeTag type, which is a complex type with a structure containing the components Value, TimeStamp, and Flags. Each component is also mapped to a data type, in this case Data_Discrete, Data_TimeStamp, and Data_Flags, respectively. In all cases, each type maps down to a supported MMS type for data exchange.

This section contains both the basic types and complex structure types, ordered the same as Section 5. The exception is the Matrix Data Type, which is used by several different objects, as described in Section 7.

Section 7: Mapping of Object Models to MMS Types. This section defines the mapping of each object attribute from Section 5 to one or more of the ICCP types defined in Section 6.

Section 8: Use of Supervisory Control Objects. This section provides examples of the use of the Supervisory Control objects in order to introduce some conventions in assigning meaning to certain attributes that are generic in nature.

Section 9: Conformance. This section identifies the 870-6-503 Conformance Block required to provide the necessary services for exchanging each data object described in 870-6-802.

870-6-702

This specification defines the Application Profile (Layers 5-7) for use with ICCP. It is needed for vendors implementing protocol stacks that support the ICCP Application Layer. Most users of ICCP will not be concerned with this specification. Therefore, the present version of this guide does not deal specifically with 870-6-702.

2.2.7 ICCP Server Objects

2.2.7.1 Association

Association objects are used to establish an association, or logical connection, between two ICCP instances. Such an association is typically long running, staying in place as long as both ICCP instances are running and the underlying communications links are maintained.

Three *operations* are defined for Association objects:

- Associate – used by a client to establish an association with a server
- Conclude – used by either a client or server to provide an orderly termination to an association (for example, for planned maintenance)
- Abort – used by either client or server to terminate an association when there are failures in the underlying communications mechanisms

There are no *actions* defined for association objects.

2.2.7.2 Data Value

Data Value objects represent values of control center data elements, including SCADA points such as analog measurements, digital status, and control points, or data structures. Any data element or object that is uniquely identified by a single MMS Named Variable (with persistence) can be represented via the Data Value object. Currently this includes Indication Point, Control Point, and Protection Event objects only.

There are four *operations* defined for Data Value objects:

- Get Data Value – can be used to request the value of a single SCADA point.
- Set Data Value – intended to permit a data value to be written or set at a local control center by a remote control center. In practice, few vendors or utilities are actually permitting the Set capability in an ICCP client because of the desire to keep the ability to change data with the owner of the data, which will be the ICCP server. Note that the Device object defined below is intended to permit remote supervisory control operations.
- Get Data Value Names – allows client to obtain a list of the names of all the Data Value objects at a remote control center for which that client has permission (via the BLT). This operation can be used to determine which points can be viewed by the client to aid in defining data sets or one-shot requests for data, as described later.
- Get Data Value Type – allows client to obtain the Type attribute for a Data Value object.

There are no *actions* defined for Data Value objects.

2.2.7.3 Data Set

Data Set objects are ordered lists of Data Value objects maintained by an ICCP server. This object enables a client to remotely define Data Sets via ICCP. The Data Set object can be used by a client, for example, to remotely define a list of SCADA points to be reported as a group. The establishment of the reporting criteria and the actual transfer of data values are accomplished using the Transfer Set object, as described below.

There are six *operations* defined for Data Set objects:

- Create Data Set – allows a client to create a Data Set object at a remote server. In addition to specifying the list of Data Value objects to be included in the Data Set, the client can also specify which of the following parameters will be included in a Transfer Report containing the actual data values:
  - Transfer Set Name – identifies the Transfer Set object that generated the report
  - Data Set Conditions Detected – identifies the event that triggered the sending of the report. The list of possible trigger events is:
    - Interval time-out
    - Object change
    - Operator request

    - Integrity time-out
    - Other external event
  - Event Code Detected – identifies the event code if the trigger was Other External Event
  - Transfer Set Time Stamp – specifies the time the Transfer Report was generated at the server
- Delete Data Set – allows a client to delete a previously defined Data Set object.
- Get Data Set Element Values – allows a client to obtain the value of each of the Data Value objects included in the referenced Data Set object. This operation permits a one-shot request of all values for the list of Data Value objects included in the referenced Data Set.
- Set Data Set Elements – allows a client to set the value of each of the Data Value objects included in a Data Set. In practice, this is not usually permitted.
- Get Data Set Names – allows a client to get the names of all of the Data Set objects currently defined at a server.
- Get Data Set Element Names – allows a client to obtain the list of names of all of the Data Value objects currently included in a specific Data Set object at a server.

There are no *actions* defined for the Data Set object.

Typical Use

Transfer of SCADA data to and from a real-time SCADA database on an SCADA/EMS system.

2.2.7.4 Transfer Set

Transfer Set objects residing at an ICCP server are used by an ICCP client to establish the actual transfer of data values. While Data Value objects can be individually requested via a one-shot request, receiving the requested value in response, more complex data transfers require the use of a Transfer Set. As mentioned earlier, the transfer of groups of data defined in Data Set objects requires the use of a Transfer Set. The exchange of most all other data in ICCP requires a Transfer Set to be established first.

The Transfer Set object permits information to be exchanged on a periodic basis, on change of state or value, in response to a particular server event, or on operator request. The Transfer Set object provides the operations needed by a client to set up instances of Transfer Sets for each desired data exchange.

Four Transfer Set Object Models

Because of the unique requirements for transferring different types of data between control centers, ICCP provides four types of Transfer Set objects:

- Data Set Transfer Set – used for establishing the transfer of Data Sets defined and created using the Data Set object.
- Time Series Transfer Set – used for transferring the data values of a single Data Value object at different incremental times as specified by a delta time interval.
- Transfer Account Transfer Set – used for transferring many different types of data objects. In ICCP, a Transfer Account is a generic term applied to a whole class of data objects used to represent information on schedules, accounts, device outages, curves, and other entities used by control centers that have only one thing in common—the use of complex data structures to represent data. Initially, the type of data envisioned was accounting or scheduling data, which represent an amount of energy transferred from one utility to another on a periodic basis, hence the name Account or Transfer Account.

  As currently defined in the IEC standard, this transfer set is used to transfer any of the data objects defined as "Block 8 objects". This includes the following:

  - Transfer Account – This is a container type of object that can be used for the exchange of any periodic or profile data for control center energy scheduling, accounting, or monitoring applications.
  - Device Outage – This is a data object designed to exchange information about device outages, either for scheduling outages or reporting actual outages. Devices can include almost any type of physical component in a power system that is routinely monitored for status today.
  - Availability Report – This is the first of five data objects included in a class of data objects labeled Power Plant objects in IEC 870-6-802. It is intended for power plant control systems or DCSs to report on predicted availability of generating units and/or to schedule outages. It is similar to Device Outage object, but differs by having more attributes unique to generation units and power plants, and by not including actual status reports (this is handled by the next data object, Real Time Status).
  - Real Time Status – This object is used by a power plant to report the actual operating status of generating units at the time of the report.
  - Forecast Schedule – This object is intended for use by an EMS or Generation Control System (GCS) to deliver a forecast usage of generating units at a power plant. Similar to the Transfer Account data object, this is another container object with a user-defined matrix to specify the number and meaning of each column in the matrix. Rows are separated by a user-selected delta time increment.
  - Curve – This object is intended for use by a power plant to report various types of curve data, such as heat rate, incremental heat rate, MVAR capacity, opacity SO_x, NO_x, and CO_2 emission curves. The curve is represented as a sequence of curve segments, with each segment defined in terms of a polynomial.

  - Power System Dynamics – This is a collection of data elements (rather than an actual object model) that need to be exchanged between a power plant and a GCS or EMS. These are scalar quantities and can be represented individually as Data Value objects.

These objects are described in detail in this guide under the Block 8 heading in the section titled *Conformance Blocks and Associated Objects.*

- Information Message Transfer Set – used for transferring the Information Buffer data object defined in IEC 870-6-802. The Information Buffer is intended for sending unstructured ASCII test strings or binary data.

There are four *operations* defined for Transfer Set objects:

- Start Transfer – permits a client to request a server to begin to transfer data under the conditions specified by the client in the operation. The capabilities provided differ in important ways for each type of transfer set:
  - For Data Set Transfer Sets, the client provides the name of the Data Set object to use for grouping Data Values for transfer. A separate Transfer Set is used for each Data Set of interest, permitting different transfer conditions for each.
  - For Time Series Transfer Sets, the client names the Data Value object of interest.
  - For Transfer Account Transfer Sets, the client can only enable the transfer of all Transfer Account objects defined in the Bilateral Table. That is, all Block 8 objects get enabled at one time and under one set of conditions.
  - For Information Message Transfer Sets, similar to Transfer Account Transfer Sets, the client can only enable all Information Message objects under the same set of conditions.
- Stop Transfer – used by a client to stop a data transfer operation (that is, disable the transfer). A new Start Transfer operation is required to once again enable the transfer.
- Get Next Data Set Transfer Set Value – used by the client as the first step in starting a Data Set data transfer. The server maintains a "pool" of available Data Set Transfer Sets for a client to use. The client must obtain the name of the next available Transfer Set, and then perform a Start Transfer operation using the name of the Transfer Set to actually start a transfer. Thus, the Start Transfer operation can be thought of as the client "writing" a value of the Transfer Set variable to the server. A Stop Transfer operation actually releases the Transfer Set back into the pool of available Transfer Set names at the server.
- Get Next Time Series Transfer Set Value – similar to the Get Next Data Set Transfer Set Value operation, except that this operation is used to start the reporting of a series of values for the same Data Value Object.

  Note: There is no "Get Next Transfer Set Value" operation for Transfer Accounts (that is, Block 8 objects) or Information Message objects, because the client can only start or stop transfers of all Block 8 objects or Information Message objects, respectively.

There are two *actions* for Transfer Sets:

- Condition Monitoring – performed by the server for each transfer as soon as that set is enabled via a Start Transfer operation. Any and all conditions requested in the Start Transfer operation are monitored by the server until a Stop Transfer operation is performed by the client. Note that for Information Message Transfer Set objects, the conditions used are locally defined only and cannot be specified via the Start Transfer operation.
- Transfer Report – a Transfer Report is generated whenever a condition specified by the client has occurred for an enabled Transfer Set. The Transfer Report is the action used to actually transfer data from the server to the client. The server formats and sends a report with the appropriate data for that type of Transfer Set.

  Associated with the Transfer Report are four additional objects (with no operations or actions) to convey information about the Transfer Report generation process:

  – Transfer Set Name – the name of the Transfer Set object that caused the Transfer Report
  – Transfer Set Conditions – a bitstring indicating which Transfer Condition(s) triggered the transfer
  – Transfer Set Time Stamp – the time of generation of the Transfer report
  – Transfer Set Event Code – indicates the external event that caused the Transfer Report to be sent, if the Other External Event condition was being monitored.

2.2.7.5 Account

Transfer Account objects (that is, Block 8 data objects) are usually transferred via the Transfer Account Transfer Set object. However, there is one special and very useful operation provided, the Query Operation, that permits a client to request a particular account object based on the account reference number and, optionally, start time and duration.

2.2.7.6 Device

Device objects represent actual physical devices in the field for the purpose of providing services that enable the client to control them remotely. Both interlocked (that is, select-before-operate) and non-interlocked devices are represented.

There are four *operations* for Device Objects:

- Select – used by a client to request selection of an interlocked device only. If successful, the Device state is changed from IDLE to ARMED by the server.
- Operate – used by a client to send a command to a Device object to execute a function. For interlocked devices, the Device state must be ARMED.
- Set Tag – used by a client to set the Tag attribute of a Device object.

- Get Tag Value – used by a client to retrieve the current state of the Tag attribute of a Device object.

There are four *actions* defined for Device objects:

- Time-out – results from a time-out after a device has been set to ARMED via a Select operation but not yet operated. This action causes the device state to return to IDLE.
- Local Reset – causes a device state to be reset from ARMED to IDLE by a local action at the server. This might also cause the Tag attribute value to change.
- Success – used to tell the client that a successful Operate operation has been completed.
- Failure – used to tell the client that an Operate operation has failed.

2.2.7.7 Program

A Program object provides a client with remote operation of a program at a server site. The actual program being controlled can be any application program at the server site.

There are six *operations* defined for the Program object:

- Start – starts an IDLE program
- Stop – stops a RUNNING program
- Resume – starts a STOPPED program
- Reset – idles a STOPPED program
- Kill – makes a program UNRUNNABLE
- Get Program Attributes – returns information on a RUNNING program

There are no *actions* defined for a Program object.

2.2.7.8 Event

An Event object represents a system event at a server site, such as a device changing state or the occurrence of a certain data error. Event objects provide a way for a client to be notified of system events at a server. There are actually two objects associated with events: Event Enrollment object and Event Condition object. There is only a minimal description of these objects in the ICCP specification, which map directly to MMS services with the same name.

2.2.7.9 Event Enrollment

Event Enrollment permits a client to express interest in being notified of a particular event when it occurs at a server site. There are three operations associated with an Event Enrollment object:

- Create Event Enrollment – creates an Event Enrollment object that specifies which event is of interest and which condition should be reported. This is accomplished by specifying the name of an Event Condition object as part of creating an Event Enrollment object.
- Delete Event Enrollment – deletes an Event Enrollment object.
- Get Event Enrollment attributes – gets existing Event Enrollment attributes.

There are no *actions* defined.

2.2.7.10 Event Condition

Event Condition objects are predefined at a server for all system events that are to be available to clients for enrollment.

There is one *action* for an Event Enrollment object:

- Event Notification – notifies all clients that have created Event Enrollment objects that specify the particular Event Condition object whenever the event occurs.

It should be noted that the device state change events that are monitored by Event Condition objects can also be reported to a client via SCADA data point changes so that the use of Event objects might not be needed. However, the Event objects provide a mechanism for certain events that might not otherwise be reported to a client.

There are no *operations* defined for the Event Condition object.

2.3 Conformance Blocks and Associated Objects

This section explains the intended use of each conformance block and object. The services and protocols associated with each conformance block and its associated objects are discussed in 870-6-503. The user objects themselves are described in 870-6-802. There are location references at the beginning of each block's description that point to discussions or descriptions in 870-6-503 and 870-6-802.

ICCP was designed from the beginning to be modular. Each conformance block represents a specific function or set of functions that a utility might wish to implement. A utility implementing ICCP for real-time data exchange is only required to purchase Block 1. Additional blocks may be added independently. For example, a utility wishing to exchange power system data by exception and accounting data needs only to purchase Blocks 1, 2 and 8.

Each block can have specific user objects associated with that block. This mapping of objects associated with corresponding conformance blocks is found in 870-6-802, Section 9. When a user decides to purchase a specific block, they should also specify which objects within that block must be supported by the vendor.

2.3.1 Block 1 (Periodic Power System Data)

2.3.1.1 Indication Point Object

Block 1 is slightly different from all the other blocks. Block 1 is the minimum that a developer can implement. It is also the minimum that a user can purchase. There are certain system services that must be supported. In particular, this block includes the following objects:

- Association
- Data Value
- Data Set
- Data Set Transfer Set

Once these objects and associated services are provided in Block 1, they will be utilized whether additional conformance blocks are added or not.

In addition to these special system services, Block 1 provides for the periodic transfer of power system data. Power system data is the database representation of field device status (that is, breaker, MODs, Hot Line Order (HLO) lamps, substation doors, etc.), analog values (that is, megawatt, megavar, voltage, tap settings, phase shifter angles, etc.), and accumulator values (kWh). Each data item can also have a quality code associated with it that provides information about the reliability of the data item itself.

The data object transferred in Block 1 is the Indication Point object. The Indication Point object is used to transfer information about status points (referred to as STATE or DISCRETE) and analog points (referred to as REAL). A formal description of the object can be found in 870-6-802, Section 5.1.1.

An optional data object transferred in Block 1 is the Protection Event object, described here and in 870-6-802, Section 5.1.3.

Status Points

A description of the Status Points foundation types can be found in 870-6-802, Section 6.1.1 as Data_State and Data_Discrete.

The user should decide whether to transfer status point information as STATE or DISCRETE. Using STATE will only allow up to a maximum of four states to be described for each device. Most power system devices are two or three state devices (open, closed, traveling). The choice of STATE allows for the most efficient transfer of status information. Two bits are used to encode the device state. The entire device state and quality are transferred in one octet.

There are, however, multi-state devices and pseudo status points in the SCADA/EMS database that have more than four states. To transfer these status points, the use of DISCRETE is required. Although less efficient, the use of DISCRETE allows the user to transfer a 32-bit integer where each value can represent a different state. Transferring status information using DISCRETE

requires a 32-bit integer for the device states and an additional octet of the associated quality codes.

Analog Points

A description of the Analog Points foundation type can be found in 870-6-802 Section 6.1.1 as Data_Real.

Analog point values are transferred as 32-bit Institute of Electrical and Electronic Engineers (IEEE) format floating point values. Each analog value can have quality codes associated with it that provide information about the reliability of the value itself. Transferring analog information requires a 32-bit integer for the analog value and an additional octet for the associated quality codes.

Quality Codes

A qualitative description of the quality codes that ICCP provides to the user is found in 870-6-802 Section 6.1.1 as Validity.

A description of the Quality Codes foundation type can be found in 870-6-802 Section 6.1.1 as Data_Flags.

Quality codes are derived from the current SCADA/EMS computer system's ability to determine the reliability of a status, analog, or accumulator point that has been stored in the SCADA/EMS database.

A telemetered value within reasonability limits that was updated to the SCADA/EMS database successfully on the last attempted scan has the highest quality. Its quality is derived from the fact that the value is both accurate and current. Quality is also considered high on data points that might not be current, but that have been manually entered by a dispatcher, operator, or program. Because a "conscious" decision has been made to assign a point its particular value, it is considered "good" or of high quality.

ICCP transfers quality codes associated with each data point, however, the assignment of local quality code bits in the receiver's SCADA/EMS database is a local implementation issue. Because each SCADA/EMS database has its own symbols for displaying data quality, each user must determine their own hierarchy of processing and mapping to their own quality symbols.

Time Stamp

A description of the Time Stamp foundation type can be found in 870-6-802 Section 6.1.1 as Data_TimeStamp.

The TimeStamp attribute is used to assess the currency of the data value being transferred. Data can be "old" for a number of different reasons: delays in out going queues at the source SCADA/EMS, delays in transmission across the network, delays due to congestion and re-transmission within the network, and delays in in-coming queues at the receiving SCADA/EMS.

For all of these reasons, the data might need to be time stamped at the source SCADA/EMS at the earliest time following collection of that data from the field device. Values that are calculated from other values in the SCADA/EMS should be time stamped at the time the values are stored in the SCADA/EMS database.

Change of Value (COV) Counter

A description of the Change of Value foundation can be found in 870-6-802, Section 6.1.1 as COV_Counter.

A periodic information report transferring status and analog values will transfer only the current value of the data point. A receiving control center might want to know whether the point had changed and then changed back between information reports. For example, an auto-reclose operation might easily occur between information reports and not be recorded at the receiving site. A COV counter is incremented each time the owner sets a new value of the Indication Point.

Building Complex Data Types

The complex types are created by combining foundation data types. The choice of which complex data type to use is made by the implementer and is a balance between efficiency and the extent to which additional information about the value being transferred is required by the receiving site. For instance, if a client wants to receive status with quality codes and a time tag, the client would specify the use of the Data_StateQTimeTag complex type, described in 870-6-503, Section 6.1.1.

2.3.1.2 Protection Equipment Event Object

The Protection Equipment Event object definition can be found in 870-6-802, Section 5.1.3.

When events occur at the substation, local relay actions can be taken to protect equipment. These events can be phase-to-phase, phase-to-ground, over current, over or under voltage, or other protective relaying schemes. In addition to the name of the event, protection equipment event object reports show the following:

The quality of the information. An underlined value indicates a “yes” answer to the question.

Table 2-1
Information Quality

ElapsedTime Validity	Were the associated times correctly acquired?	VALID INVALID
Blocked	Is the information blocked against further updates until it has been transmitted or safe saved?	NOTBLOCKED BLOCKED
Substituted	Was the information manually entered or entered by an automated source?	NONSUBSTITUTED SUBSTITUTED
Topical	Was the last update of the information successfully completed?	NONTOPICAL TOPICAL
EventValidity	Were no abnormal conditions of the information source detected during the last update?	VALID INVALID

The type of event (SINGLE or PACKED) and information related to the event.

A SINGLE event has its EventState, EventDuration, and EventTime reported.

A PACKED event reports either the cause and involved equipment (START), or the actions taken (TRIP).

START events include the following information. An underlined value indicates a “yes” answer to the question.

Table 2-2
Start Events

StartGeneral	Was this a general start?	START NOSTART
StartPhase1	Was phase 1 involved in the event?	START NOSTART
StartPhase2	Was Phase 2 involved in the event?	START NOSTART
StartPhase3	Was phase 3 involved in the event?	START NOSTART
StartEarth	Was ground current involved in the event?	START NOSTART
StartReverse	Was reverse current involved in the event?	START NOSTART
DurationTime	Event duration in milliseconds	
StartTime	Protection equipment operation start time	

TRIP events include the following information. An underlined value indicates a “yes” answer to the question.

Table 2-3
Trip Events

TripGeneral	Was this a general trip operation?	TRIP NOTRIP
TripPhase1	Was a control operation issued to trip Phase 1?	TRIP NOTRIP
TripPhase2	Was a control operation issued to trip Phase 2?	TRIP NOTRIP
TripPhase#	Was a control operation issued to trip Phase 3?	TRIP NOTRIP
OperationTime	Time in milliseconds from the start of the operation until the first command was issued to an output control circuit	
TripTime	Time of the start of the operation	

2.3.2 Block 2 (Extended Data Set Condition Monitoring)

A description of Data Set condition monitoring can be found in 870-6-503, Section 5.2.9.1.1 and 5.2.9.1.2.

Block 2 is used to provide the capability to transfer power system data in more ways than periodic reports. A periodic report (Block 1) is simple and easy to set up but it has the drawback that, because it reports every value to the client every time the report is generated, it is not very bandwidth-efficient. Block 2 is also referred to as Report-by-Exception, or RBE.

Report-by-exception allows the client to specify that power system objects will be reported only when a change is detected or when an integrity check is performed. ICCP does this by having the server monitor a number of conditions and, when one or more of those conditions occurs, the data that has changed is sent to the client. The client sets the conditions to be monitored in the transfer set at the server.

The conditions that can be monitored are:

- The normal reporting period is due (IntervalTimeOut). This is the same condition that is monitored in Block 1.
- The value, state, or quality code of a value has changed (ObjectChange).
- The operator at the server site has requested that the value be sent to the client (OperatorRequest).
- A periodic report of all values is sent to the client to ensure that the two databases are still synchronized and that no changes have been lost since the last integrity check (IntegrityTimeOut).
- Other, unspecified conditions can be monitored (OtherExternalEvent).

Once the server has determined that a report-by-exception information report is required, it must then determine whether the client has requested that the report be generated as normal MMS named variables, or as blocked data (see next section).

2.3.3 Block 3 (Block Data Transfer)

A description of the rules for encoding block data can be found in 870-6-503, Section 7.1.4.4.2.

Block data with report-by-exception is a very efficient transfer mechanism under certain conditions. It provides the possibility for an ICCP server to send power system data to a client with fewer bytes than required for sending with full Abstract Syntax Notation One (ASN.1) encoding, as required in Blocks 1 and 2. Blocking can be useful where bandwidth is at a premium due to either low data rates or short periodicities (that is, high frequency) of the data reports. However, the consequence of blocking is that the information needed to properly decode the data in a transfer report is not all contained in the report itself.

There are two mechanisms used by Block 3 to achieve efficiency. The first is the dropping of the Tag and Length fields for each data value reported. The second is the creation of an index-based tagging scheme to replace variable names with a one or two byte number. Block 3 provides three rules for encoding. The choice of the proper rule depends on whether the data is all sent periodically or as report-by-exception, and on how many values are sent. These mechanisms and rules are described below.

2.3.3.1 Use of an Octet String MMS Variable

Instead of sending a tag and length along with each data value, as required by the ASN.1 Basic Encoding Rules used in Layer 6, the ICCP server instead utilizes a single long octet-string MMS variable to transfer all of the data values. In order to avoid having to enter the Length field, all primitive data types (and any aggregates based on them) must be encoded using the full length permitted by that type. Variable length fields, therefore, must be padded out to their maximum length. In order for the client to receive and utilize the data, the client must have prior independent knowledge of the location and type of each value in the octet-string. Client knowledge of the type filed is required to permit the dropping of the tag fields.

Then, if the data is sent as provided in Block 1 (that is, not report-by-exception), the data is encoded into the octet-string according to Rule 0, described below:

Rule 0: [rule#, total length, value, …]

This can result in fewer bytes being transferred because the Tag and Length fields for each variable are not transferred. This works best for variables with short data types that require only one byte for the value. However, for longer types, there are cases where this will not result in any savings. For example, transferring an Integer32 variable that happens to equal 0 in value will result in a MMS Protocol Data Unit (PDU) encoding using 3 bytes (tag, length, and value each one byte), whereas blocking would have to expand the integer out to four bytes, actually wasting one byte. Therefore, the type of data to be transferred should be considered before automatically assuming fewer bytes will result just from dropping the Tag and Length fields.

2.3.3.2 Index-Based Tagging

Block data and report-by-exception can be combined to yield a more efficient transfer of data. If block data and report-by-exception are specified, the server has two rules available for constructing the message that will be sent to the client. In each case the database point is identified by an index into the named variable list, followed by the current value of the point. This has the effect of replacing variable names, typically many bytes in length, with a one or two byte index number.

Utilizing Rule 1 below, the header consists of the rule number [1], followed by the total message length in octets. The body of the message consists of a one-octet index (the relative position of the identifier in the names variable list), followed by the value of the identifier. This pairing of index and value is continued to the end of the message.

Rule 1: [rule#, total length, $index_i$ (1-octet), $value_i$ …1]

Rule 2 is similar to Rule 1 except that it utilizes a two-octet index for messages that have more than 255 index-value pairs.

Rule 2: [rule#, total length, $index_i$ (2-octets), $value_i$ …]

Blocking when combined with report-by-exception, thus, provides guaranteed efficiency for transmission by sacrificing inclusion of the information needed to decode the data contained in a message, creating a data maintenance task. If message formats seldom change, this might be a good tradeoff. However, if bandwidth is not a primary concern or report-by-exception is not used, and more flexibility is desired by a client to change involvement at the server, then blocking should probably not be used.

Block 4 (Information Messages)

Block 4 provides a general message transfer mechanism that also includes the ability (by agreement of the two parties) to transfer a simple test of binary files. Block 4 adds the Information Message Transfer Set server object with the associated Information Buffer data object.

One use of this service might be for a utility to notify other utilities within its inter-connection that an event more complex than that represented by simple power system data values, has occurred. For example:

- Notification of a decision to implement an inter-connection wide time error correction action
- Notification of the boundaries of identified electrical islands during a disturbance
- Request for emergency use of pool reserves

These messages might be simple formatted ASCII text messages with data from the SCADA/EMS incorporated into the body of the message. These could be used as alarm text or text reports for display on a receiving operator console or for logging.

The InformationBuffer object provides a unique identifier (InfoReference) and a local identifier (LocalReference). The MessageID identifies the particular instance of a message. The Size attribute is the length in octets of the actual data being transferred.

This object also provides a mechanism for simple, small, binary file transfer. These transfers are limited in size by MMS to 8k-octets. The InfoReference and LocalReference attributes could be used to identify a process that would receive the binary information buffer and store it in a local file. The information stored could, by agreement, be an Excel or Word Perfect file that would later be accessed by the client or server. Individual instances of this file being transferred (the June, July, or August instances) would be distinguished by the MessageID attribute.

An informal description of the Information Message can be found in 870-6-503, Section 5.1.6, and a formal description can be found in Section 5.2.8. The Information Buffer object is

described in 870-6-802, Section 5.4, the type descriptions in Section 6.4, and the mappings to MMS in Section 7.4.

2.3.4 Block 5 (Device Control)

Block 5 adds the Device server object and associated Control Point data object.

Block 5 provides a mechanism for transferring a "request to operate a device" from one ICCP implementation to another. ICCP does not directly control the device, rather it communicates a client's request to operate a device to the server.

ICCP retains some of the important characteristics of RTU device control. Specifically, it retains the select and validate before operation for interlocked devices and the armed-for-execution mode for a selected device.

The ControlPoint object is used to transfer the request. It distinguishes between a device operation (COMMAND) and the transfer of a numeric value (SETPOINT), either floating point (SetpointRealValue) or integer (SetpointDiscreteValue).

A control request can be for non-interlocked (NONINTERLOCKED) or interlocked devices (INTERLOCKED). Both command and setpoint operations can be inter-locked or non-interlocked. Non-interlocked controls are control operations that do not require select-before-operate confirmation. These might include transformer tap changes, raise/lower operations, and digital value setpoint type operations. Interlocked controls on the other hand, require select-before-operate confirmation for critical operations such as breaker trip/closure, reclosure on/off, and HLO lamp on/off.

For interlocked control operations, the client sends a request to operate a specified device to the server. After checking for the existence of the device object, and checking access control in the Bilateral Table, the server then performs a local verification that the device is available for operation. The verification checks that the server actually performs are considered a local implementation issue. The device is SELECTED and a previously agreed to CheckBackName is provided to the client to confirm that the correct device has been selected. A Time-out period is reported to the client giving the length of time that the device will remain selected by the server.

ICCP does provide a mechanism for the server to report to the client whether the desired control point is tagged. The server reports:

0 if the device is not tagged
1 if the device is tagged open and close inhibit
2 if the device is tagged close only inhibit

The client, having received a verification of device operability and a validation of device selected, then sends a final request to have the device operated by the server or to cancel the requested operation.

The server completes the requested control operation and notifies the client of success or failure. Provisions are made so that at any time during this process the client or the server can terminate the operation for a valid reason.

An informal description of device control can be found in 870-6-503, Section 5.1.10, and a formal description can be found in Section 5.2.11. The device object model mapping can be found in Section 6.15. Device operations and action mapping to MMS, including a sequence of device control diagram, can be found in Section 7.1.6.1. The control point object is described in 870-6-802, Section 5.1.2, the type description in Section 6.1.2, and the mappings to MMS in Section 7.1.2.

2.3.5 Block 6 (Program Control)

Block 6 adds the Program Server object and associated services.

Block 6 provides a mechanism for an ICCP client to perform program control at a server ICCP implementation site. Program control is only available by prior agreement between any two ICCP sites.

Implementation of program control is made very straightforward by the fact that MMS provides program invocation and control as part of its basic services. ICCP can then utilize these services with proper interfaces to the SCADA/EMS system to perform remote program control. There are no user objects associated with program control.

An informal description of program control can be found in 870-6-503, Section 5.1.11, and a formal description can be found in Section 5.2.12. The program control object model mapping can be found in Section 6.16. Program operations and action mapping to MMS can be found in Section 7.1.7.

2.3.6 Block 7 (Event Reporting)

Block 7 adds the Event Enrollment and Event Condition objects. Block 7 is not required for any of the other blocks, but instead provides extended reporting of system events occurring at a remote site (that is, the ICCP server).

Block 7 provides two functions to the ICCP client.

- It allows the ICCP client to enroll in two types of events:
  - Specify error condition reporting from the server
    - Time-out conditions
    - Failure conditions
    - Local reset
    - Success conditions
  - State of the device changes at the server

- It allows the ICCP client to receive information about the events that the client has enrolled in and has enabled.

An example of how a utility might utilize event enrollment is in a situation where a utility is required during non-working hours to monitor specific security events at a substation or power plant. During working hours, monitoring and control is performed locally at the site. The enrollment and receipt of the security information could be enabled for only the non-working hours.

An informal description of event reporting and event conditions can be found in 870-6-503, Section 5.1.13, and a formal description can be found in Sections 5.2.13 and 5.2.14. The event enrollment and event conditions object model can be found in Sections 6.17 and 6.18. Event enrollment operations mapping and event conditions action mapping to MMS can be found in Sections 7.1.8 and 7.1.9.

2.3.7 Block 8 (Additional User Objects)

Block 8 adds the Transfer Account Transfer Set server object for transferring Block 8 data objects. An informal description of Transfer Account Transfer Set objects and services can be found in 870-6-503, Section 5.1.7, and a formal description can be found in Section 5.2.9.3. The Transfer Account Transfer Set object model mapping to MMS can be found in Section 7.1.4.

Block 8 also adds the Account server object for requesting Block 8 data objects. Using the Query operation supported by this server object, an ICCP client can specify the following:

- The Transfer Account Reference Number for the account for which information is to be returned
- Start time for the data
- Duration of the data in seconds since the start time
- A RequestID, which is echoed back by the server to permit the client to match incoming data with a specific request
- TAConditions to identify the type of data requested

An informal description of account objects and services can be found in 870-6-503, Section 5.1.5, and a formal description can be found in Section 5.2.7. The Account Object Model mapping is found in Section 6.7. Account operations and action mapping to MMS can be found in Section 7.1.5. However, the details of the attributes included in the Query operation are contained in 870-6-802, Section 5.2.4.

Block 8 provides a utility with a number of additional data objects related to transferring scheduling and accounting information, device outage information, and power plant information. A vendor might offer support for one or more of the data objects in Block 8. Information on these objects can be found in the ICCP specification at the following locations in 870-6-802:

Table 2-4
ICCP Spec Reference Matrix

	Model Object	MMS Types	Object Mapping to MMS
Scheduling & Accounting			
Transfer Accounts	5.2.1	6.2.1	7.2.1
Transmission Segment	5.2.2	6.2.2	7.2.2
Profile Value	5.2.3	6.2.3	7.2.3
Account Request	5.2.4	6.2.4	7.2.4
Device Outage	5.3	6.3	7.3
Power Plant			
Availability	5.5.1	6.5.1	7.5.1
Real-time Status	5.5.2	6.5.2	7.5.2
Forecast Schedule	5.5.3	6.5.3	7.5.3
Curve Mapping	5.5.4	6.5.4	7.5.4

The following subsections describe the ICCP data objects in more detail.

2.3.7.1 Transfer Account Data Object

The ability to transfer scheduling and accounting information between ICCP implementations is a key feature of ICCP. With it, an ICCP client can set up a transfer set that will allow the server to send pre-schedules, next-hour schedules, mid-hour changes, after-the-hour-actuals, and historical information. ICCP generalizes this transfer capability to allow any data that is collected on an hourly (or other period) basis, including such data as generator schedules, interchange schedules, forebay and afterbay elevations, average hourly TOT limits and actual flows, pricing information, delivery point loads, and so on. This transfer capability is accomplished via the Transfer Account data object.

The flexibility of this object is achieved through the use of a matrix data type with the number of rows and columns defined by the user for each type of desired transfer. In addition, the meaning of the column headings are also user-defined, so this standard data object can be used to transfer many different types of schedules and accounts.

An important feature of ICCP is the ability of the client scheduler or dispatcher to specify the time frame for the data to be retrieved and have the server return the specified information corresponding to that time frame. The client specifies this via the TAConditions object described below.

The Transfer Account object definitions can be found in 870-6-802, Section 6.2. A Transfer Account example is provided in the Informative Annex A of 870-6-802.

The Meaning of TAConditions and How to Use Them

TAConditions (Transfer Accounts Conditions) are used to allow the client to set up condition monitoring of specific accounts in the server. Within the electric power utility business, the transfer of scheduling and accounting information is very time-dependent. To illustrate this time dependency, a utility might have the following timeframe requirements:

- Pre-schedules must be completed by 16:00 the day before they are used.
- Next-hour schedules must be completed by 20 minutes prior to the hour in which they will be used.
- Mid-hour changes can occur anytime within the current, already scheduled, hour.
- After-the-hour-actuals are due by 10 minutes past the just-completed, previous hour.
- All 24 hours of historical forebay elevations for the previous day need to be transferred at 10 minutes past midnight of the current day.

The client can instruct the server to monitor specified accounts for specific TAConditions and have the server automatically generate and transfer the account information when the condition is met. The actual timeframes used are a local implementation issue. Since the exact formatting of the Transfer Account data object is user-defined, it is possible to have a different format for data corresponding to each TACondition. The Transfer Account Reference Number can be used to further identify the format of the report being transferred.

Transfer Account Structure

Scheduling and accounting data is stored by utilities in what is, essentially, a matrix structure. ICCP carries forward the matrix concept, but generalizes it to allow the user to define the meaning of the columns. Both floating point and integer matrixes can be transferred. This allows the ICCP client and server to exchange virtually any type of matrix-format data addition to scheduling and accounting data.

Generally speaking, the actual Transfer Report is used to transfer a Transfer Account Data object, and to identify the account to be transferred, the transfer account condition (TACondition), the sending and receiving utilities, the start time (referenced by hour ending, if hourly scheduling and accounting information is being transferred), and the time span of the period used (typically one hour). The message then specifies whether or not this is a wheeling transaction and, if so, the number of wheeling segments is identified. For each segment, the number of floating (or integer) values and the number of periods (the values that will form the matrix of information) are identified.

In the event that a matrix of information is being transferred that has specific meaning to the client, a list of local references can be identified that will provide the client system with the column heading information for the matrix.

Examples

Figure 2-5 illustrates the structure of this object. Up to one floating point matrix and one integer matrix can be sent in one Transfer Account object. Figure 2-6 shows the use of this object to send two matrices of data for a time period covering one hour with a period resolution of 15 minutes.

A more detailed example of the use of the Transfer Account object is provided in the Informative Annex of 870-6-802.

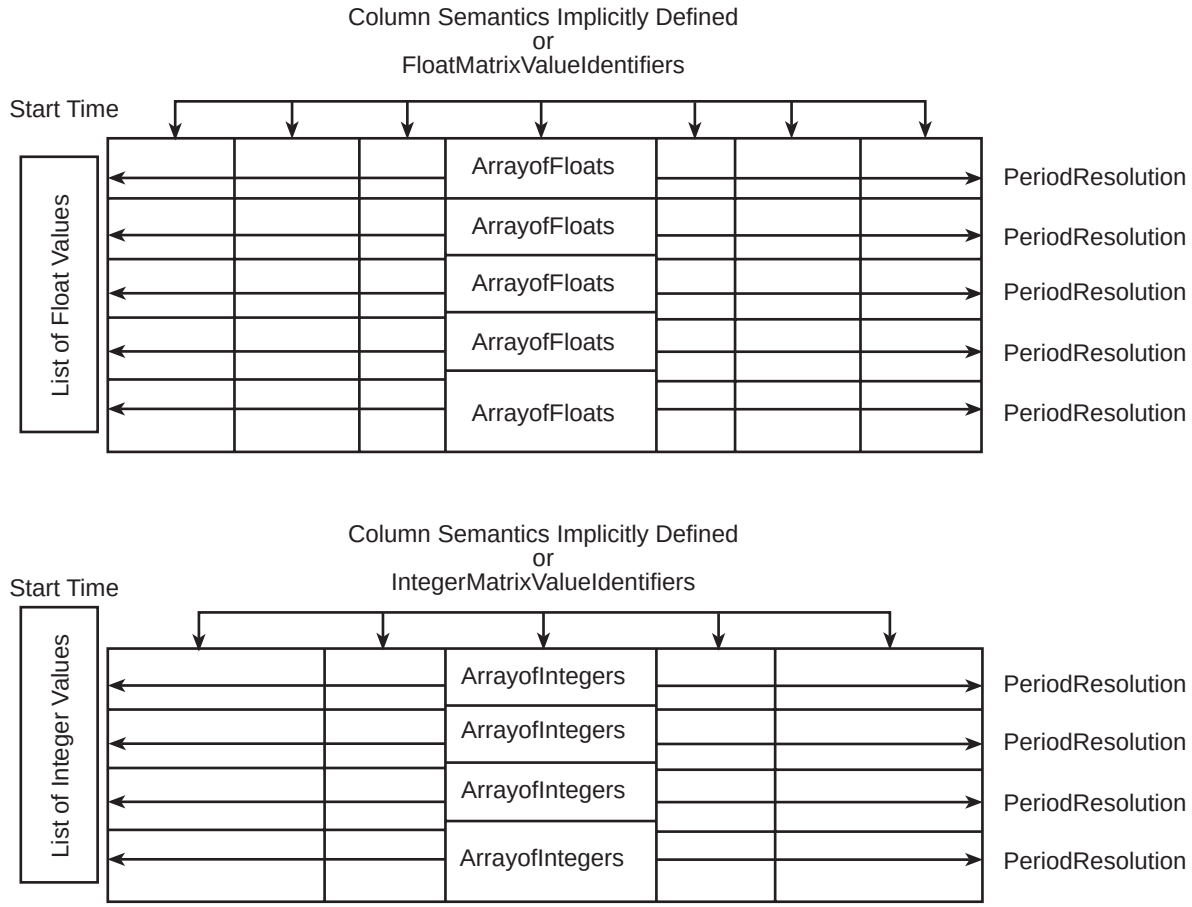


Figure 2-5
Transfer Account Data Object Model Structure

Start Time: 10:00AM
Period Resolution: 00:00:00:15:00
Column Semantics: Implied by Transfer Account Reference

Floating Point Matrix

	Seller Cost	Buyer Cost	Emergency Cost	Tariff Value	Tariff Incurred	Tariff Avoided	Savings
10:00AM							
10:15AM							
10:30AM							
10:45AM							

Integer Matrix

	MWH	Emergency MWH	Area Load
10:00AM			
10:15AM			
10:30AM			
10:45AM			

Figure 2-6
Example of Transfer Account Data Object Use

2.3.7.2 Device Outage

Utilities deal with two types of equipment outages, planned and unplanned. Planned outages are for scheduled maintenance of generators, transmission lines, and line devices. Unplanned outages are due to system disturbances that result in no, partial, or full curtailments of power transfers. ICCP provides an object for the transfer of outage information.

The DeviceOutage object identifies the location of the device (StationName), the device (DeviceType), and provides information about the device (DeviceName, DeviceType, and DeviceRating). If the outage is planned, then the object information can be used to set up a new outage or revise an existing planned outage. In either case, the out-of-service and return-to-service date and times are provided. The outage type can be categorized and, if it is an outage resulting in a partial curtailment, new upper and lower operating limits can be specified.

If the outage is an actual outage as a result of unplanned operation of devices in the electrical system that resulted in a loss of load, the type of action is categorized and the amount of load that was being carried at the time of the service interruption is provided.

The Device Outage object definition can be found in 870-6-802, Section 5.3.

2.3.7.3 Power Plant Objects

ICCP has been extended to include communications between the SCADA/EMS and power plants. Power plants have specific requirements that result in specialized objects. These objects take advantage of the existing underlying ICCP services available such as report-by-exception and condition monitoring.

Power Plant Availability Object

The Power Plant Availability object is used to allow a generation station to inform the control center of the known or scheduled availability of a unit at that site. The object can also be used to schedule a unit outage or curtailment. If it is reporting a unit curtailment, new operating constraints can be reported for the period of the curtailment.

The object identifies the generation station location and the specific unit referenced in the report. It also identifies the start and stop time as well as the duration of the proposed change of status of the unit, either AVAILABLE or UNAVAILABLE. A curtailment is treated as AVAILABLE with a change of operating constraints.

If the unit is AVAILABLE, the report allows the generation station to specify the following items for the duration of the availability:

- A new price
- New maximum ramp rates both up and down
- New gross maximum and minimum capacities
- New net maximum and minimum capacities
- Whether the unit is in standby or on-line mode
  - If the unit is on-line, whether it is available for load following
  - If the unit is on-line but not available for load following, the reason (STARTUP, UNSTABLE)
- If the unit is UNAVAILABLE, the reason it is unavailable (FORCED, SCHEDULED, TESTING)

The reporting generation station can also report whether the unit is providing reserves, and up to 256 characters of user comments.

The Power Plant Availability report object definition can be found in 870-6-802, Section 5.5.1.

Power Plant Real-Time Status Object

The Power Plant Status object is used to allow a generation station to inform the control center of the current status of the plant and each unit at that site.

If the plant is AVAILABLE, the report allows the generation station to report the current operating characteristics of each unit.

- The maximum ramp rates both up and down
- The gross maximum and minimum capacities
- The net maximum and minimum capacities
- Whether the unit is in standby or on-line mode
  - If the unit is on-line, whether it is available for load following
  - If the unit is on-line but not available for load following, the reason (STARTUP, UNSTABLE)
- Whether the unit is externally blocked high or not
- Whether the unit is externally blocked low or not
- If the unit is UNAVAILABLE, the reason it is unavailable (FORCED, SCHEDULED, TESTING, EQUIPMENT)

The Power Plant Real-Time Status object definition can be found in 870-6-802, Section 5.5.2.

Power Plant Forecast Schedule Object

Some power plant units are operated with pre-scheduled base points. These units are either not used for load following or have pre-defined load following periods. This object allows base points and operating modes to be transferred to the generation stations. Modes of operation can be user-defined in the event that load following does not adequately describe the required modes.

The scheduled period for which a megawatt base point value applies can be specified in the object. Normally, one hour (hour ending) would be used but the user can define other scheduled period durations. The only constraint is that all of the referenced periods have to be of equal duration.

The Power Plant Forecast Schedule object definition can be found in 870-6-802, Section 5.5.3.

Power Plant Curve Object

The power plant might want to transfer data in the form of two-dimensional curves. These curves could be incremental heat rate, hydro-head-dependent efficiency curves, cost curves, or other power plant-related curves.

The curve is defined as an nth order polynomial. All segments in the curve are of the same order. Each segment of the curve is represented by a start-of-segment and an end-of-segment, and mathematically by the nth order polynomial: $A_0 + A_1X + A_2 X^2 + \ldots A_nX^n$. In representing each segment's polynomial, only the coefficients, $A_0 + A_1 \ldots A_n$ are transferred. The client can reconstruct the curve by knowing the order of the polynomials and these coefficients.

2.3.8 Block 9 (Time Series Data)

Block 9 adds the Time Series Transfer Set server object.

Block 9 provides an ICCP client with the ability to receive time series data. Time series data might be data that has a required sampling time too fast to conveniently transfer it continuously between ICCP implementations and that is not needed at the client site in real-time. Examples of this type of data might be a 200-millisecond sample on key analog values on the backbone transmission system during a disturbance. Once collected, the values might then be transferred as historical data to a disturbance analysis center. Real-time trending of values with longer reporting periods for more efficient use of communications bandwidth is another potential time series data application.

In setting up a time series data transfer, the client establishes the begin-time and the end-time. Both the begin-time and end-time might be in the past, in which case the server immediately generates a report of historical values based on the timeframe specified. If the begin-time is the current time or 0, the server begins to immediately collect values until the end-time occurs. At the designated end-time, the server stops collecting values and generates a report for the client. If the end-time is the current time or 0, the server assumes current time and stops collecting values and generates the report. For a planned collection of values in the future, the client can specify both a future begin- and end-time.

Two intervals are specified to make the collect and transfer of information as efficient as possible. The sample interval specifies the collection rate at the server. The reporting interval specifies the time between reports.

Several options exist to fine-tune the reporting functions of the server. Since the end-time might not fall exactly on one of the reporting period boundaries, the server can be instructed to generate a report at the specified end-time. Similarly, the server can be instructed to generate a report at the initial enabling of the transfer set and at each reporting period. The server can also be instructed to allow an operator at the server system to initiate on demand a transfer of the time series data collected since the last reporting period.

The Time Series Transfer Set object model can be found in 870-6-503, Section 5.2.9.2 and the Time Series Transfer Set object model mapping can be found in Section 6.9.2.

2.4 Mapping Utility Data to Conformance Blocks and Control Center Data Objects

A utility planning to use ICCP must perform several tasks before knowing exactly which conformance blocks are required to meet its needs. The following issues will need to be addressed:

- The data to be transferred needs to be identified and performance requirements established. This will uncover which ICCP services are needed and, hence, which ICCP server objects and conformance blocks are needed. For instance, if SCADA data, schedule/accounting data,

and text reports are to be transferred, then Conformance Blocks 1, 2, 4, and 8 will be required.

- Analyzing data to be transferred will also identify which data objects are needed. For instance, if in addition to scheduling/accounting data, outage information for scheduling and reporting outages is needed, then either the Device Outage object or a combination of the Availability and Real-Time Status objects will probably be required.
- Each data element will need to be mapped to an ICCP object attribute. This mapping will need to be documented. Every attempt should be made to map to existing standard objects specified in the ICCP specifications because this will ensure interoperability with other ICCP vendor products, without additional software development.
- Some types of data reports will not map 1:1 to standard ICCP objects. The choice will then be one of the following:
  - "Force fit" a data element to an attribute with a different meaning. This eliminates any ICCP software changes but creates the opportunity for misunderstandings as to the meanings of certain attributes.
  - Add a new attribute to a standard object, thus, customizing it. If no new data types are introduced, this will result in minimal change but will ensure there is no misunderstanding. Eventually common usage might result in a modification of the standard object.
  - Create a whole new data object. Sometimes, this is the only choice. This process is described in the next section.
- Any choices made will need to be documented. A Network Interface Control Document (NICD) is commonly used for this purpose. This document is not defined anywhere since it goes beyond what is specified in the ICCP specifications but, by common practice, it includes the mappings, the common conventions decided on for assigning numbers or codes to reference numbers, the definition of any new data objects, and so on. For data objects that use the Matrix data type, the meanings of column headings and the number of rows for different uses of this object will also be documented here.

2.5 Definition of New Data Objects

The designers of ICCP expected that new objects would be needed from time to time. Not all possible uses of ICCP could be envisioned during the creation of the initial version of the specification. If the process of mapping actual data requirements to ICCP objects described in the previous sections requires a new object, then the process described in this section should be followed.

The process for creating a new data object is as follows:

1. Develop an abstract data model, creating a name for the object and deciding which attributes are needed, along with defining each attribute. The object model definitions in the ICCP specification provide examples of how this is done. This can be done by end-users with little or no knowledge of MMS.

2. The abstract model then needs to be mapped to concrete structures with components. Part of this process involves assigning data types to each attribute. The goal is to reuse the data types already defined in the ICCP specification, thus minimizing the implementation effort for the new object.

3
BILATERAL TABLE ISSUES

ICCP specifies access control through the use of Bilateral Tables. The functionality required is clearly presented in 870-6-503. The type of access for each ICCP data object is defined via these tables. However, implementation is left as a "local implementation issue." This includes the management and maintenance of these tables. As a result, each vendor is free to choose how to implement the functionality for Bilateral Tables, including what type of operator interface to provide. This means that an actual physical table is not required, as long as the functionality is implemented according to the ICCP specification.

A common request from end-users is for an ICCP client to have the capability to view the Bilateral Table at an ICCP server to determine which objects it has access to, and to be notified whenever that table is updated. The ICCP specifications do not specify a way to accomplish this although there is a Data Value operation, Get Data Value Names, which an ICCP client can use to obtain a list of all the Data Value objects accessible to that client. A "browser" capability that would allow a client to view the objects it has access to could be quite useful for a user acting as an ICCP client. The user could then simply point and click the objects it wants to receive. Part of the functionality could include creating data sets by pointing and clicking. This would minimize the potential for operator error on entering data values.

Some users might not desire to use the security mechanisms provided through Bilateral Tables, for instance, where ICCP is used between two regional control centers within the same utility. One way to handle this is to just provide the same access to all control center objects in the Virtual Control Center (VCC) to any client. However, the protocol operations and actions specified in the ICCP specifications still must be implemented to ensure interoperability.

4
USER INTERFACE ISSUES

The ICCP specifications do not specify a user interface for managing and maintaining ICCP. This is left as another “local implementation issue”. Each vendor is free to choose an appropriate interface.

The following areas might need a user interface:

- Displaying ICCP performance data such as status of each association and data link, last error detected, throughput statistics, and so on.
- Control of data link associations, data sets, or other ICCP objects to enable or disable selected capabilities
- Creation and editing of Bilateral Tables
- Creation and editing of Data Sets
- Setting up and managing broadcast groups for information messages when Conformance Block 4 is implemented. Because ICCP does not provide a broadcast capability, it might be desirable to have the ability to create broadcast groups that would specify groups of destinations (that is, other ICCP sites or operator consoles) to receive information messages.

5
OTHER LOCAL IMPLEMENTATION ISSUES

ICCP is a standard real-time data exchange protocol. It provides numerous features for the delivery of data, monitoring of values, program control, and device control. All the protocol specifics needed to ensure interoperability between different vendors' ICCP products have been included in the specifications.

The ICCP specifications, however, do not attempt to specify other areas that will need to be implemented in an ICCP software product, but that do not affect interoperability. These areas are referred to as "local implementation issues" in the specification. ICCP implementers have the freedom to handle these in different ways and can, therefore, differentiate their products by the way they handle these issues. For example, one vendor might have a graphic-oriented user interface permitting point and click operations for creating data sets or controlling ICCP data links, while another might provide only programmer's editing tools to accomplish these tasks.

Local implementation issues in the specification include, but are not limited to, the following:

- The API through which local applications interface to ICCP to send or receive data
- A user interface to ICCP for user-management of ICCP data links
- Management functions for controlling and monitoring ICCP data links
- Failover schemes where redundant ICCP servers are required to meet stringent availability requirements, such as those typically experienced in an SCADA/EMS system environment
- How data, programs, or devices will be controlled or managed in the local SCADA/EMS to respond to requests received via an ICCP data link

These responsibilities fall to the SCADA/EMS vendor and the implementing utility. This section will attempt to address some of the areas that have been identified as local implementation issues in the ICCP specifications, and that have not been covered elsewhere in this guide.

5.1 Client Server Association Management

The client always initiates the association establishment procedure with an ICCP server. A single ICCP site can act as both client and server to one or more ICCP sites. It can also simultaneously be just a client or just a server to other sites. In the case where it is both client and server with another site, the use of the associations between the two sites is a local implementation issue. The simpler method to implement is where each client uses a different association with its server. However, it is possible to utilize the same association for client server pairs in both directions. This implementation is more complex but is also more resource-efficient. If a site that can utilize

one association for both client-server directions (dual-use) attempts to establish an association with a site that does not support dual-use, it is the responsibility of the dual-use site to fall back to single-use associations. It is a local implementation issue whether or not to support dual-use associations.

5.2 Local Implementation Setup Issues

When a utility implementing ICCP joins an existing network or begins communicating with another ICCP implementation, there are a number of issues that should be decided among the data exchange members.

- Pre-defined Data Set object names must be published to appropriate data exchange partners.
- The maximum number of associations that will be allowed.
- The maximum exchange frequency of data should be agreed to in order to avoid overloading a SCADA/EMS with data requests.
- Which data types can be specified as critical data.
- The use and specification of retry counters.
- The assignment of values to the Information Reference Number used by most data objects.

5.3 Specific Conformance Block Issues

Individual conformance blocks in ICCP have specific user considerations. Some of these issues are local to the utility and some are issues that should be discussed with the ICCP and SCADA/EMS vendors prior to procuring an ICCP implementation.

5.3.1 Block 1 (Data Set Definition Management)

A concern among ICCP users is how to ensure synchronization of data set definitions at both the client and server sites.

5.3.1.1 Data Set Definition

The approach assumed by ICCP is for the client to create all data sets each time the association for transferring the data defined in data sets is established with another ICCP server. This would ensure data set definitions are synchronized at least each time an association is restored after being brought down for whatever reason. This means that the ICCP server would not retain any data set definitions after an association with a remote client is brought down. The main drawback for sites with large amounts of data seems to be the time required to create all data sets before any data is actually transferred, but this approach must be used if interoperability is to be guaranteed.

A second approach is for the server to always retain data set definitions whether or not associations exist with an ICCP client. Then it would be up to the client, either periodically or on

request, to verify the definitions at the server, perhaps using the Get Data Set Names and Get Data Set Element Names operations to compare the lists of Data Value objects at the server with the lists known at the client. However, this requires that the client and server expect to operate in this manner ahead of time.

5.3.1.2 Data Set Updates

How does an ICCP client know when a server site changes the list of available Data Value objects? ICCP does not provide a mechanism for database management that would alert a client to such changes. Therefore, some scheme needs to be defined outside of ICCP. The simplest approach is to have an ICCP server site operator agree to email, phone, or FAX notices of any changes affecting a client (that is, addition, deletion, or modification of a point). Perhaps changes could be sent as low priority alarms. The optimum approach would be to have the client poll the server periodically using the ICCP Get Data Value Names operation and then locally display and highlight any changes.

5.3.2 Block 2 (Extended Data Set Condition Monitoring)

When using report-by-exception, there is always a small chance that, due to either the client or the server system being down, or due to communications problems, that the two databases will not be identical. The integrity scan (analogous to an RTU integrity scan) is used by ICCP to resynchronize databases. The use and frequency of integrity scans should be decided by data exchange members who have implemented Block 2.

When using report-by-exception for analog values, the use and specification of deadbands that reduce unnecessary transmission of minor changes should be decided by data exchange members.

An issue for discussion between the ICCP implementers and the utility is where the deadbands for analog points and the database for status points will be monitored. Is this a SCADA/EMS or an ICCP implementation responsibility?

5.3.3 Block 4 (Information Messages)

5.3.3.1 Operator Messages

Block 4 can be used to send operator messages. After an ICCP implementation has received an operator message, it must be passed to the SCADA/EMS for presentation to the dispatchers or operators. How will the SCADA/EMS display, save, retrieve and purge the resulting message?

5.3.3.2 Binary File Transfers

Block 4 allows for the transfer of small, binary files. These files will need to be stored in SCADA/EMS directories and the end-user notified that they have been received. Will existing files automatically create new copies of the file? How will end-users be notified? What convention will be used to identify the binary files as EXCEL, MSWord, Word Perfect, and so on?

5.3.3.3 Requesting an Information Message Object

ICCP does not support the request of a specific information message or object. Some utilities have solved this problem through establishing a convention for the use of the Information Reference number, which is a 32-bit integer. This number can be broken down into 9 bytes or fields. Each byte (or combination of two or more bytes) can be assigned a meaning. One byte can be reserved for indicating whether the information message is a request for a specific information message object or whether it is the actual object. For example, byte 4 could be encoded as follows:

- Information Message data object
- Request for Information Message object identified by the rest of the Information Reference number

The rest of the Information Reference number would remain unchanged. The Info Stream attribute would be empty for the request.

Segmenting Long Information Messages

Information messages must fit within the maximum length MMS Protocol Data Unit (PDU), which is 8000 bytes. If messages longer than this need to be transferred, an application above ICCP in the protocol stack needs to perform this function at both the client and server end of the association. Data fields provided by the ICCP information object can be used to convey to the receiving site either information necessary to reassemble multiple segments, but ICCP simply forwards this data without interpreting it.

One solution adopted by a power pool is to use the LocalReference or MessageId field in the InformationBuffer object described in 870-6-802, Section 6.4 to indicate if the message is completely contained in the PDU or if it is segmented into two or more segments. If more than one, a value is assigned corresponding to the order of the segment in the total sequence and whether or not it is the last segment.

5.3.4 Block 5 (Device Control)

During a Device Control operation the server provides a CheckBackName to the client to allow the client to verify that the server has selected the expected device. The content of the CheckBackName is by agreement between the two sites.

After receiving the select request from the client, the server is required to make local checks to verify that the device is operational. The checks that will be performed (for example, communications to the device available, status of the device is current, device is free of blocks or inhibiting tags, etc.) are determined by the server implementation.

5.3.5 Block 6 (Program Control)

The invocation of programs requested by an ICCP client will use other SCADA/EMS services to initiate and control the programs. Local implementation issues include program scheduling, execution monitoring of scheduled programs, priority of execution, to which process the program will be assigned, and exception and abort processing.

5.3.6 Block 8 (Transfer Accounts)

5.3.6.1 Meaning of TAConditions

The TAConditions refer to general periods of time before, during, and after a schedule is in effect. All parties sharing schedules and account data need to agree on specific time periods to associate with each TACondition and also agree to the format of the data reported under each condition. An alternative used by some utilities is to make all transfers occur as a result of Object Change or Operator Request only. The types of reports to be sent are agreed to and assigned unique Transfer Account Reference numbers. Then, as the data becomes available (or changes) for each report type, it is sent with the appropriate number so that the client can interpret the data contained.

5.3.6.2 Complex Scheduling Transactions

For many utilities a transfer of a schedule is just one step in a more complex transaction. For instance, a member company in a power pool might submit a proposed schedule to the power pool operator, who first acknowledges receipt then reviews and either accepts or rejects. If rejected, the member company then needs to submit a revised schedule. If accepted, the member company then needs to confirm. ICCP itself contains no provision for maintaining a “memory” of each step of the transaction. Such “memory” needs to be implemented in an application above the ICCP Application Program Interface (API).

Some utilities have solved this problem through establishing a convention for the use of the Transfer Reference Number, which is a 32-bit integer. This number can be broken down into 9 bytes or fields. Each byte (or combination of two or more bytes) can be assigned a meaning. One byte can be reserved for indicating the status of a schedule in the approval process. For example, Byte 4 could be encoded as follows:

1. Original submittal
2. Received (acknowledged)
3. Approved
4. Rejected
5. Revised
6. Confirmed

The rest of the Transfer Account Reference number would remain unchanged and the entire data object would be retransmitted each time with any needed revisions to attribute values.

Note that this approach could also be used to provide just a simple acknowledgment that any Block 8 object was successfully received by a client. ICCP does not provide an application-level acknowledgment. It relies instead on the Transport layer to deliver an error-free message or retransmit it transparently to ICCP in the Application layer. If it is unable, the Transport layer would take down the connection, thus notifying both the ICCP client and server of a problem. Otherwise, ICCP assumes the message is received error-free. The application above ICCP implementing this capability could then also maintain records of all messages sent to ensure no data is lost.

5.3.7 Block 9 (Time Series Data)

A request can be made for Time Series data to be collected for a specified point at a specific sampling interval. The reporting request might then expire or be terminated. A subsequent request for data from the same point might specify a different sampling interval with a begin-time that includes historical data with the first sampling time. The historical data must then be extrapolated such that the new sampling interval is identical for all reported data. How the server will extrapolate that historical data (linear, best fit, etc.) is an implementation issue for the server system.

6
NETWORK CONFIGURATION

One of the first issues to be addressed by a potential user of ICCP is whether the ICCP communications processor should be integrated into the SCADA/EMS LAN or function as a stand-alone gateway processor. Legacy systems are the most likely candidates for stand-alone processors. Both implementation configurations have advantages and disadvantages. It should be noted that GPU elected to implement their configuration on the corporate LAN/WAN infrastructure, while PEPCO determined their configuration would function more effectively in a stand-alone network.

In the case of an integrated processor, access to the SCADA database is direct, without any intervening protocol. It is easier to implement the functionality provided in ICCP if the ICCP client or server has direct access to the SCADA database and operating system. Functions such as program initiation, monitoring of database points for report-by-exception, control operations, and operator messages are all simplified with direct access. Security, however, becomes more of a concern when the processor communicates with outside the controlled environment of the SCADA/EMS system, thus providing a potential path of access. Firewalls and other security might be needed at the connections to other entities whose systems might be open to the Internet or other outside networks.

Users considering the use of ICCP need to decide how to acquire the ICCP software. ICCP products available commercially from vendors are typically packaged in one of three ways:

As a Protocol Native to the SCADA/EMS System. This type of ICCP product is typically offered by SCADA/EMS vendors as a standard product associated with their standard SCADA/EMS system (hence, the use of the term “native”). The software will run on one of the standard hardware/software platforms used for other SCADA/EMS applications and, thus, might be considered to be closely integrated into the SCADA/EMS operating environment. Typically, the ICCP software features the same API as other SCADA/EMS applications. This approach is sometimes referred to as an integrated processor approach.

Not all SCADA/EMS system vendors that offer ICCP actually use this approach. Some provide only a stand-alone gateway processor (described below).

The *advantages* of this approach are:

- All SCADA/EMS applications have direct access to the ICCP API, providing full functionality and possibly better performance. Operator messaging and device control might be simpler to implement with this approach.
- SCADA can be retrieved from (and deposited into) the real-time SCADA database directly. Monitoring and notification of SCADA data changes is direct.

- A separate relational database is not required to buffer data—the relational database associated with the SCADA/EMS system can be used directly.
- System administration and maintenance is accomplished using the standard SCADA/EMS system/network administration tools.
- User interfaces will have the same look and feel as other SCADA/EMS user interfaces.
- Since the platform is the same as for other SCADA/EMS applications, common servers can be shared between ICCP and other applications. This also helps the spares and maintenance problem.

The *disadvantages* are:

- This approach might not be available for legacy systems.
- If a proprietary API is used, it might prevent open access to other possible users of ICCP outside the SCADA/EMS environment.
- Security might be a concern where the processor that communicates with systems outside the controlled environment of the SCADA/EMS system provides a potential path of access. Routers, firewalls, and other security might be needed where the connections are to other entities whose systems may be open to the Internet or other outside networks.

As a Tool Kit from a Third Party Supplier. This approach typically provides ICCP software on a stand-alone platform (that is, Windows NT on a PC), but provides an API with full functionality for an SCADA/EMS system or other control center computer that has Transmission Control Protocol/Internet Protocol (TCP/IP), NETBIOS, DDE, SQL, or other industry-standard communications networking capability.

The *advantages* of this approach are:

- Provides the same (or nearly the same) capability as a native ICCP implementation for an EMS/SCADA system whose vendor does not offer a native ICCP implementation.
- Might provide a more open API to enable open access to the ICCP messaging services.
- Should be a low-cost solution.

The *disadvantages* are:

- Might have a different look and feel for the user interface.
- Might not use the same network administration tools as SCADA/EMS systems.
- Might require custom development work to interface to the SCADA/EMS system.

As a Standalone Gateway Processor or Communications Node Processor (CNP). This approach is for providing ICCP capability to a legacy system with limited communications networking capability. Typical offerings permit connection over either a serial line or a LAN, using TCP/IP as the transport protocol. Some typical messaging protocols offered include the following:

- Inter-Utility Data Exchange Consortium (IDEC) Host-to-CNP protocol, developed for the same application as a CNP implementing the IDEC protocol. This is a simple block transfer protocol that has been implemented by several vendors. This requires that the legacy SCADA/EMS system have software running that also talks the Host-to-CNP protocol.
- File Transfer Protocol (FTP). This requires that data to be transferred, be formatted as flat files. Custom parsing software is required at both the host processor and the ICCP gateway processor. If only limited SCADA data (that is, analogs and status) are to be sent at low periodicities, this might be an acceptable approach.
- Emulation of a legacy system protocol, such as Western System Coordinating Council (WSCC) or some existing proprietary protocol. This would have no impact on the host system, but would require custom emulation software in the ICCP gateway. Some vendors already provide WSCC and IDEC emulation. Others provide ICP gateways that emulate their existing proprietary EMS data link protocols for their own legacy systems.
- New custom protocol between the gateway and the legacy system. This might be acceptable for limited uses of ICCP but would probably require excessive development to utilize all of the features of ICCP.

The *advantages* of this approach are:

- Might be the only way to implement ICCP for a legacy system.
- Might have minimal impact on the legacy host computer.
- Might provide additional security.

The *disadvantages* are:

- A second protocol is required between the gateway processor and the host computer.
- Limited functionality of ICCP via the restricted host-to-gateway protocol and serial connection. Implementing object transfers, control operations, accounting information transfers, and operator messages across an intervening protocol requires that the two databases be maintained and that additional applications be implemented to carry the ICCP functionality all the way to the SCADA/EMS.
- A separate database is required on the ICCP gateway processor to store and buffer ICCP data, which might be different from that used on the SCADA/EMS system, thus creating training and maintenance issues.
- Might be different operating system and processor hardware, requiring different system/network administration, additional licenses and spares.
- Lower performance and throughput with greater time delays in transferring data is likely.

7
SECURITY

ICCP provides access control via the Associate operations to establish an association. The client must identify itself to the server. The server must have a Bilateral Table in place for that client and the Bilateral Table version numbers must match. Otherwise, the server must conclude the association. As previously described, the Bilateral Table identifies all objects that the client is authorized to access and the level of access permitted for each object. In both the GPU GenCo and PEPCO Demonstration projects, system and data security were optimized through the use of Bilateral Tables.

ICCP does not provide mechanisms for authorization or for encryption. These would normally be provided by lower layer protocols.

8
PROFILES

8.1 Open Systems Interconnection (OSI)

As stated earlier, ICCP was originally designed for operation of OSI/ISO-compliant protocols, specifically the protocols identified in UCA™ Version 1.0. This has been the norm for vendor implementations up to the current time. The current effort to standardize ICCP also assumes a fully compliant OSI protocol stack.

8.2 TCP/IP

Depending on the vendor providing ICCP, it might be possible to operate ICCP over TCP/IP. There are two possibilities proposed for UCA™ Version 2.0 to accomplish this:

- OSI Layers 5-7, including ICCP, directly over TCP/IP. This approach replaces ISO TP4/CLNS in Layers 3-4 with TPO over TCP/IP. This approach is specified in RFC 1006 and is the most widely supported by protocol vendors. However, TPO over TCP/IP does not have the same capability as TP4 regarding Quality of Service (QOS) parameters and the automatic checking via "keep alive" messages to ascertain that a data link has not gone down. To provide an equivalent capability, the Application layer using ICCP would need to generate periodic test messages. In practice, this might not be a problem since most ICCP links are expected to support the transfer of periodic SCADA data at rates as often as every 4 seconds; any attempt to send data over a failed link would get reported immediately.

  An unresolved issue identified is the reporting of error messages from the Transport layer. It is not clear if the MMS maps error messages from TCP and TP4 to the same error codes for reporting to ICCP. If not, the Transport layer will not be truly transparent to ICCP and the handling of different error codes would become a local implementation issue.

- OSI layers 3-7 encapsulated in User Datagram Protocol/Internet Protocol (UDP/IP) messages. This approach uses RFC 1070. It retains the QOS and automatic detection of outages but does require the maintenance of two address spaces (that is, the ISO CLNS network layer addresses and the TCP IP layer addresses).

The consensus regarding ICCP and MMS is that the transport layer should be mostly transparent so that either TP4/CLNS or TP0/TCP/IP can be used. The idea of TCP use is that, if a utility already uses TCP/IP, then it is probably preferable to use ICCP over TCP rather than introducing a full OSI stack just for ICCP. There is nothing, however, to prevent a utility from running both stacks in parallel. The vendor used by the utility should be consulted to determine the actual choices available.

9
PROCUREMENT OF ICCP

Because of the acceptance of ICCP by the utility industry, there are a number of ICCP products on the market. These can be obtained from either EMS or SCADA vendors as well as third party suppliers on a variety of hardware platforms and operating systems. As a result, there should be no need for a user of ICCP to have to develop new software to implement the protocol. Because of the extensive interoperability testing either accomplished or planned in the near future, interoperability of these products is not a high risk area, although data link testing is obviously required as part of acceptance testing.

There are, however, a number of areas identified through the ICCP specifications and in this guide that are referred to as "local implementation issues" that a vendor is free to handle in a variety of ways. There are also other system considerations and configuration issues that need to be clearly specified. As a result, it is recommended that a procurement specification be prepared to capture any requirements in these areas.

The initial intent at GPU was to have each of the DCS vendors procure and implement an ICCP interface to their systems. However, partway through the project it was determined that the optimum approach was to contract with a single provider for all interface activities. GPU competitively bid that activity and Honeywell was contracted to provide the interface for all three vendors. GPU purchased the necessary interface hardware and Honeywell provided the ICCP nodes and interface software. PEPCO elected to contract with each of the two DCS vendors, therefore interfacing ICCP to their control systems.

The purpose of this section is to help guide a prospective user in preparing a procurement specification for use either as a single supplier or as general guidelines for competitively bidding the activity.

9.1 Preparing a Procurement Specification

A procurement specification should address the following areas:

- A network diagram showing all networking requirements between ICCP nodes.
- System configuration at each ICCP node to identify all computer system interfaces from the SCADA database, RDBMS, power applications, or operator consoles to each ICCP server. This would require a choice of integrated processor or stand-alone gateway implementation of ICCP.

- System requirements, such as sizing, performance, availability, backup, and recover (including whether redundant ICCP servers are needed and failover schemes are required), and the use of certain corporate standards.
- Functional requirements, such as which ICCP conformance blocks are needed, specific ICCP data objects needed, the definition of any new data objects to handle the required data transfers, API needs, security required, the number of associations and intended use of each, user interface (such as for Bilateral Table creation and editing, Data Set creation and editing, and network management).
- Hardware requirements, such as the use of specific platforms, routers, LAN hub technologies, and so on.
- Support software requirements, such as standards to be followed, operating system, programming languages, editor and program development support tools, and so on.
- Project implementation requirements, such as project deliverables, customer and vendor responsibilities, project management guidelines, quality assurance provisions, testing, commissioning, warranty, maintenance support, documentation, and training.

9.2 Network Interface Control Document

In addition to a procurement specification, which serves to document all specific requirements of the ICCP vendor, there is usually a need for a document often referred to as a Network Interface Control Document (NICD). The NICD is needed to document agreements and conventions about such issues as:

- Mapping of specific data to ICCP objects and attributes.
- Variable naming, as for SCADA point names.
- Definition of the key attributes used to uniquely identify instances of ICCP objects, such as the Information Reference Number for Information Messages or the Transaction Reference Number for Transfer Account objects.
- Definition of requirements for any special applications developed to handle unique messaging needs beyond the ICCP specifications, such as complex transactions involving several ICCP message transfers, segmenting of Information Messages longer than 8000 bytes, or creation of any new data objects.
- Any other agreements between multiple parties all connected to the same ICCP network.

10
CONFIGURATION MANAGEMENT OF AN ICCP NETWORK

There is very little in the IEC specification dealing with management of an ICCP network. This section attempts to address common issues a user will be faced with in using ICCP. Actual capabilities provided with an ICCP software product will be vendor-dependent. Management of an integration of ICCP into a power producer's LAN/WAN infrastructure will be somewhat unique to the facility's infrastructure.

10.1 Naming of Data Value Objects

Naming of Data Value Objects is a local matter. The names chosen should have meaning to all ICCP clients and servers, wherever they are located. If ICCP is used internal to a single utility between control centers, then it might be that all sites use the same names locally. These names could be maintained for naming ICCP objects as well.

If, however, ICCP is used in a power pool setting for example, then one utility's names will not necessarily be used by another utility, even for the same substation (for instance, where a substation is between two utilities and, therefore, has many points monitored by both utilities). In this case, there are two choices:

- A global network name for each point could be defined and used by all parties. Each utility would then map that name into a local name. This is the most common approach and probably the easiest to maintain.
- The name used by the owner (source/server) of the data could be used by all, with a mapping done at the client from the server name to the client name, if necessary. This approach has the disadvantage that any local name changes by the owner/server of the data would require a change at every client location.

10.2 Creation of Data Sets

The creation of data sets is described as a client function in IEC 870-6-503. However, that document does not specify how they are to be created or what tools should be provided. This is a local implementation matter and will depend on the ICCP vendor. Special requirements should be specified in a procurement specification.

As a minimum, the vendor should provide an editor capability to permit an operator to define the contents of each data set and name it. It would be helpful if the operator could browse a list of the Data Value objects that a server site permits the client to see and access. This list can be

obtained via the Get Data Value Names operation. This might require custom development beyond an ICCP vendor's standard offering.

An alternative approach is to automatically define and name data sets based on the same list of Data Value objects desired by the client for each ICCP server.

10.3 Association Management

ICCP assumes that associations will be brought up as part of an initialization procedure implemented in the ICCP software. However, the particular method used is a local implementation issue. Furthermore, ICCP assumes that associations are long lasting and, once up, will remain up until a data link is lost.

Therefore, any operator capabilities to control associations and/or data links (that is, to enable/disable an association or data link) are defined by the vendor. If specific capabilities are required, they should be included in an ICCP procurement specification.

10.3.1 Performance Management

The ICCP specifications do not address monitoring of ICCP data link performance. Any features that an ICCP user requires must be specified separately.

10.3.2 Fault Management

ICCP assumes that an error-free transport mechanism is available for transferring data unless an error message is received from a lower layer protocol. Therefore, there are no test capabilities or fault management capabilities specified for ICCP.

As a result, any maintenance of error statistics or lost connections, as well as an operator display of these features, is considered a local implementation issue and will be up to the vendor to decide what capabilities, if any, to include with its standard ICCP product. If specific capabilities are needed, they should be included in the procurement specification.

11
INTEROPERABILITY—CASE STUDY RESULTS: GPU DEMONSTRATION PROJECT

11.1 Project Overview

This section provides an overview of one of the two demonstration projects initiated to deploy ICCP for connecting the power plant controls systems to the utility's energy management system. The problem addressed in the GPU initiative is the connectivity between the various control systems in a major electric power company portfolio and its enterprise systems.

Most of the DCSs used in various process industries consist of many proprietary components. Therefore, it is not a simple task to establish communications between two different control systems. It is these proprietary elements that have allowed the various control system vendors to survive by continuing to develop a unique set of tools that its customer base can expand upon. The process industry is bound legally to the copyrights of these proprietary elements and therefore, must focus on a widely accepted standard that various vendors can mutually agree upon to accomplish connectivity.

EPRI funded the development of a communications protocol specifically to address connectivity of various Supervisory Control and Data Acquisitions systems used in electric power dispatching. Detailed in earlier sections of this report, the protocol is published by EPRI as the Inter-Control Center Communications Protocol (ICCP). Because the protocol is applicable to any electric dispatching center, EPRI has submitted it as an International Standard known as IEC 870-06.TASE.2 as detailed earlier in this report, and is involved in migrating the protocol for power plant communications and control.

For an electric power producer to succeed in the future deregulated power generation market, new technologies must be considered and deployed to reduce the unit operating costs through improved operating efficiency and reduced maintenance expenditures. A key component of success will be the integration of the companies computing capabilities to achieve maximum data utilization. Use of the companies Information Technology (IT) LAN/WAN infrastructure will play a vital role in connecting the unit control systems to the dispatching control system, or Energy Management System (EMS), for exchanging data and control signals. Also, as ownership of power plants change in the industry, corporate Information System infrastructures are being changed drastically. In many instances, ties to the corporate LAN/WAN are being severed as ownership changes and new or modified networks are established. Unit scheduling and cost data are frequently changing; it is this continuous transmitting of sensitive data that requires adequate security be in place. Leakage of this data to a competitor could adversely affect the company's profitability in a rapidly changing market.

The demand for electrical generation varies hourly depending on a number of factors. The major factors influencing the pricing of electrical generation are:

- Number of generating units available at the time
- Weather conditions, especially outside temperature and humidity
- Time of day, typically evenings are off-peak, low-demand times
- Type of day, holiday, weekend, and normal business day
- Availability of transmission access from generating unit

For these reasons, embarking on a plan such as this requires a detailed risk analysis to validate the selection of the LAN/WAN as a secure media. The security that ICCP provides becomes increasingly more important than past protocols because now unit control and financial data is transmitted throughout the organization and potentially over public data paths. Availability to this data by any of its competitors would jeopardize the company's competitive edge.

The EPRI/Host utility demonstration project was initiated at GPU to deploy and demonstrate an ICCP digital communications link to connect the host electric utility EMS and selected station DCSs. The project demonstrated through the use of selected applications the ability to transmit/receive data and objects for information exchange and unit dispatch control between the two dissimilar systems.

The GPU EMS is located in one of the corporate facilities separate from all of the electric generating stations. This project links the EMS to three different plants with the following DCSs:

- Conemaugh Station with a Honeywell DCS
- Portland Station with a Westinghouse DCS
- Shawville Station with a Bailey DCS

Figure 11-1 shows the network configuration for Portland Station where ICCP was integrated in the corporate LAN/WAN station communications. This configuration is typical for the other GPU stations.

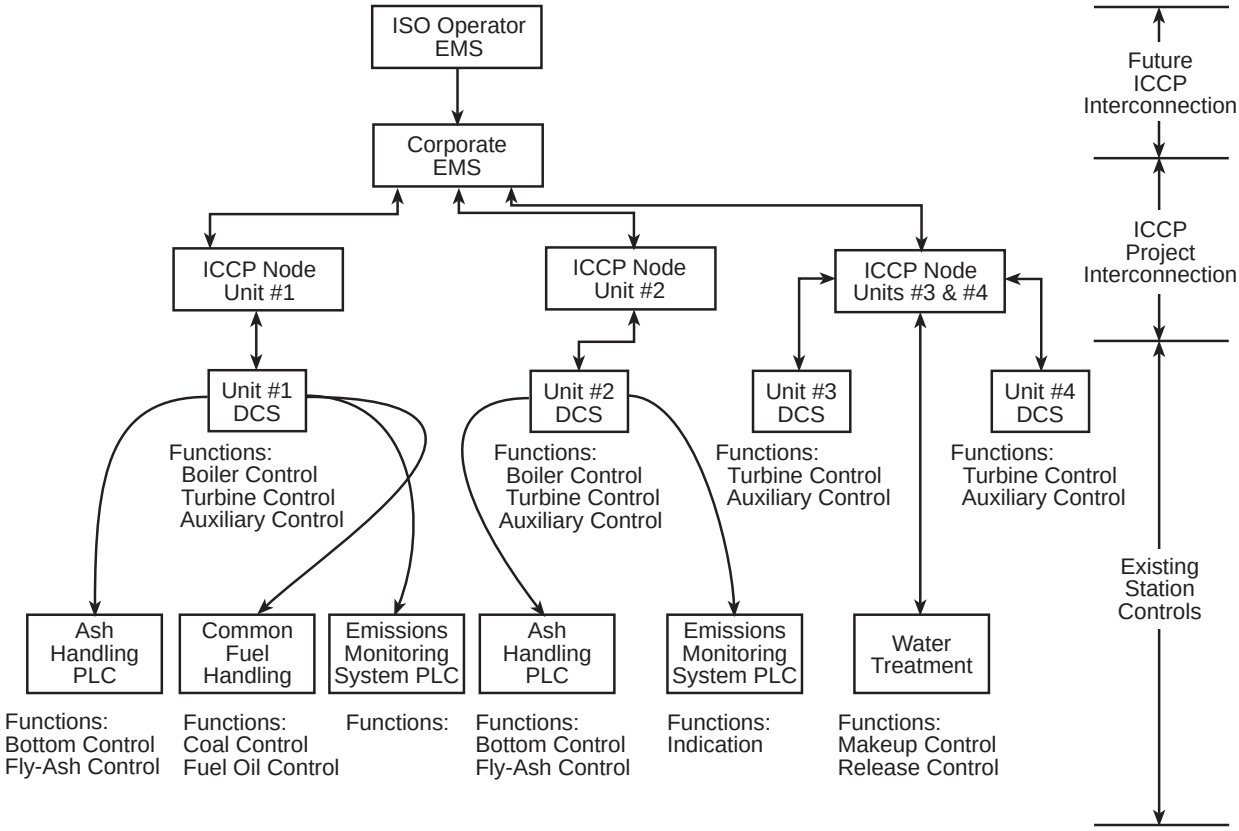


Figure 11-1
Portland Station ICCP Network

The data types that can be exchanged between the two or more ICCP nodes are:

- Real-time process analog and discrete control and data acquisition signals as used by the existing Remote Terminal Units (RTUs)
- Historical data such as averaged values or accumulations over time
- Predefined reports from either system
- Unit Forecast Schedule data
- Unit Commitment data
- Simultaneous updating of generation reports requirements through utilization of a common distributed database

A primary objective of the project was the integration of the corporate power plant process data to the client server environment.

Another objective of the project was to establish and demonstrate the ability to exchange data between two different vendor DCS systems over the WAN. The project successfully achieved this objective.

The initial phase of the project established the ICCP digital link between the EMS and DCS. The project team developed and established all necessary interfaces for exchanging the required data over the ICCP links, whether analog, discrete, or object format. This was followed by implementation of the selected applications to demonstrate viability of the enabling technology.

Results expected from the project include:

- Reduced dependence on verbal communication between the dispatching center and unit control rooms, resulting in an expected cost-saving of five man-hours per month during peak system coordination
- Elimination of RTUs at the stations, which is expected to reduce the station book value by approximately $65,000 and reduce associated O&M costs by 40 man-hours per year
- Reduction in Lost Generation Report Accounting development time by two man-hours per month
- Relief of 45 man-hours per month for the Group Shift Supervisor due to automation of the Daily Status Report

The risk analysis deployed for the project identified threats to the secure operation of the ICCP from the following categories for the demonstration project:

- Intentional Human Intervention - the deliberate disruption of control of the data link. This type of intrusion is most likely done from within the organization; however, knowledgeable outside hackers also fall into this category.
- Accidental Human Intervention - the accidental or procedural failure by which an individual has the ability to access the data link.
- Natural Threats – the category that most likely results in interruptions in the data link. Examples would be storms, fires, and so on.
- Physical Threats – a category that might cause faulty data or physical loss of equipment and services.

From the analysis, it was concluded that the existing LAN/WAN is the only cost-justifiable communication topology for this project at GPU. Installation of a separate fiber system to support this system while increasing security was cost-prohibitive.

11.2 Summary Analysis

The ICCP Demonstration Project was justified within GPU in that the project strives to solve a lack of a standard communications link between the generating units digital control systems, the central dispatching systems energy management control system, the corporate client server environment, and the legacy corporate mainframe. Each of these systems contains some format and quantity of data generated by one or more of the other systems. Integration through a standard protocol enables greater efficiency, improving production while reducing costs.

11.2.1 Methodology: Functional Module Decomposition

The methodology that was utilized for the ICCP EMS/DCS demonstration project was based on functional decomposition of the existing system and a projection of future needs in a competitive environment. The system implemented was based on the hierarchical control functions performed in each of the modules. A spatial model was developed based on the objective tree of the project. The spatial model established the short, medium, and long-term deliverables of the project.

The short-term objective focused on delivering an operational communications system based on the long-term needs of the organization. This prototype system delivered capability for data exchange functionality between the various DCSs and EMS. This functional operation was selected to validate the life-cycle methodology and provides immediate benefit to the organization.

Figure 11-2 shows the functional organization of the GPU ICCP Implementation Team. During the project, the team established the necessary hardware, protocol modules to be implemented in the DCS and EMS hardware, and the topology for each of the sites.

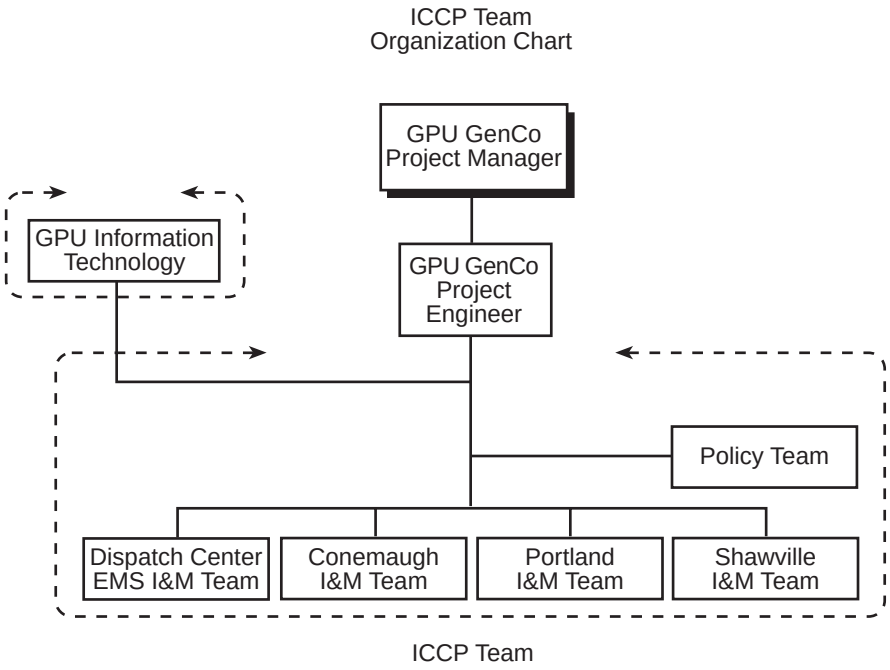


Figure 11-2
GPU ICCP Demonstration Implementation Project Team

11.2.2 Results

In summary, the project reports that:

The resulting ICCP system is easy to install. With NT-based PCs, Honeywell's PHD and a few RDIs, and Siemens's ICCPNT, the team found that they could provide passing of data to almost any system with minimal effort. When one configures an ICCP connection, it must be ensured that spelling (including caps) is precise. Most of the problems encountered during installation were due to minor errors in configuration. Databases on both the Honeywell PHD and Siemens's ICCPNT were easy to configure and maintain. The Honeywell RDIs were easily configured into the system and provided a good method of exacting data from the various DCS. The LAN connection can either be Ethernet or Token-Ring-based.

12
INTEROPERABILITY—CASE STUDY RESULTS: PEPCO DEMONSTRATION PROJECT

12.1 Introduction

This section provides an overview of the PEPCO ICCP Demonstration Project and summarizes preliminary results available at the time of publication. Under this project, EPRI's Inter Control Center Protocol, ICCP was deployed in a demonstration initiative at PEPCO for improved unit dispatch between PEPCO's dispatch computer and the plant control systems at the Potomac River and Chalk Point stations. As the electric power industry evolves into deregulation, PEPCO recognized that improved communication between energy management facilities and power plant control systems was necessary for optimal unit dispatch and control of the generating units. It was this need that prompted PEPCO's involvement in the demonstration project.

12.2 Background

As detailed earlier in this report, EPRI has championed a digital communications protocol for transmission of data, costs, and control exchange between power sectors in this country. The enabling technology is Inter-Control Center Communications Protocol (ICCP) Version 6.0. Recognizing the need for further deployment of the protocol into the generating plant control systems, to achieve improved unit control based on real-time economic dispatch, EPRI established demonstration projects to implement ICCP in each of the major DCS vendor systems. This project links the PEPCO System Energy Management System (EMS) computer to the Foxboro Distributed Control Systems (DCSs) at Chalk Point and the MAX Controls DCS at the Potomac River Station. The link will enable timely data and objects exchange between the systems and includes the ability for unit dispatch.

12.3 Objective

The objective of this project was to improve direct communications between two of the DCS vendors' systems in operation in PEPCO and in the EMS system. Currently, communications for generator load control between PEPCO's central load dispatch and power plants is primarily microwave via remote terminal units (RTUs) with minimal telephone voice communications. This project was intended to demonstrate ICCP as a protocol that replaces the RTUs and provides enhanced functionality.

12.4 Technical Approach

The project was implemented under a collaborative effort between PEPCO and EPRI.

PEPCO functioned as prime contractor for the project. A project champion and overall team leader was established within PEPCO to be responsible for delivery of business results and overall project success. A project team of representatives from EPRI, PEPCO, and the control system vendors ensured focus on maintaining unity, consistency between the two vendors, EPRI, and PEPCO, and compliance to the protocol.

Each vendor provided a communications gateway for exchanging services between their proprietary data highway and the ICCP node(s) located at each station. This gateway complies with the International Organization for Standardization (ISO) standards for Open Systems Interconnection (OSI) ISO and other standards agreed upon. Since ICCP conforms to EPRI Data Acquisition and Information Services (DAIS) and Utility Communications Architecture (UCA™) standards as described in EPRI reports TR-101706s Vols. 1 and 2, the gateways were designed accordingly. ICCP, as a non-proprietary protocol, provides open-system communication utilizing the Manufacturing Message Specification (MMS) ISO 9506 and Remote Database Access (RDA) ISO 9579 protocols. The gateways enable transmitting and receiving live data, historical data, object data, reports, files, alarm data, and control data between the EMS and DCSs. It was planned that any data highway point available on the plant data highway could be mapped through the gateways.

The gateways map the DCS data highway points through calls made by a process automation protocol (MMS) that is part of the ICCP structure. Data flow control is through bilateral tables, or parameter lists maintained in both the EMS and DCS systems. The DCS gateway maps the DCS data highway points through calls made by MMS to the bilateral table for ICCP connectivity to PEPCO's data warehouse. See Section 2 of this report for more details.

It was planned from the beginning that after the enabling technology was demonstrated in Phase I, PEPCO and EPRI would consider continuation of the initiative in Phase II. If EPRI and PEPCO agree to execute a separate follow-on TC, Phase II will integrate the data now present on the corporate LAN/WAN infrastructure, merging with the business aspects of PEPCO's client server environment. Applications developed during Phase II will focus on the daily operating cost and analysis of the units and the what-if modeling to provide station and corporate management with the tools needed to optimize dispatch and marketing of station output.

12.5 Project Scope

The project scope defined at the start of the initiative included:

- Developing and implementing new ICCP communications nodes to replace the current Automatic Generation Control (AGC) Console at the Potomac River and Chalk Point stations
- Demonstrating new direct digital communication between the plant DCS' and Energy Management System, enabling improved communications

- Publishing guidelines describing implementing ICCP between proprietary DCS systems and the EMS; documenting the project results, quantifying the benefits / improvements provided by the project, and detailing enhanced dispatch applications and advanced DCS automation implementations that the ICCP enables.

12.6 Project Activities

The tasks of the project provide an overview of the activities of the project.

Tasks included:

- *Implement ICCP nodes to replace the station's RTUs and AGC consoles*

  The Control Room Operators currently dial-in unit operations data (generator load, VAR, etc.) into thumbwheel switches on the AGC panel. This data is transmitted via the RTUs over microwave to the dispatch computer. Unit dispatch signals are transmitted over microwave to the RTUs at the station, which generates 4-20ma signal demands to the plant control system. This project will enable replacement of the AGC Console and RTU communications with direct digital connectivity between the station control systems and the dispatch computer, utilizing ICCP to eliminate data mismatches and improve accuracy.

The steps to implement this task included:

1. Determined the need and installed equipment for ICCP nodes at each station.
   ICCP Node hardware
   Foxboro – Chalk Point Station
   MAX Controls Systems – Potomac River Station

2. Determined and implemented changes to Network Hardware
   Foxboro
   Max Controls
   EMS

3. Implemented ICCP Interface Software
   Foxboro
   MAX Controls

4. Test

- *Demonstrate direct digital communications between EMS and the plant DCS*

Extensive testing and verification is needed to gain confidence in the new infrastructure and prove its functionality and to develop a level of confidence before considering abandoning the existing RTUs. This activity was planned to establish a mechanism to thoroughly test and verify system operability. Reliability of the ICCP infrastructure, will be verified prior to abandoning the existing RTU infrastructure.

The activities of this task included:

1. Developing a test plan to prove operability of the new communications infrastructure. The following parameters were used to verify operability.

   MW Demand
   MW Actuals
   High Limits
   Low Limits
   Rate of Change
   AGC Set/reset
   Etc.

2. Identifying data to be transmitted to the stations from the EMS

3. Scheduling and administering testing of the new infrastructure

12.7 Conclusions

As the industry evolves into competition, improved communications between various systems will be needed and guidelines, specifications, and reports depicting methodologies and approaches for effecting improved communications will be valuable. The expected results from this project include improved accuracy, throughput, and flexibility of data transmittal enabled through the direct connectivity provided by ICCP. Also expected was the reduced maintenance and overhead through eventual elimination of the RTU technology.

13
CONCLUSIONS

Preliminary results from the GPU effort suggest that, through the use of discrete alarming from the EMS to the station DCS, a projected savings of five man-hours per month at each utility during peak system coordination is expected. Also, eliminating the need for unit dispatch RTUs at the stations is expected to result in equipment savings and reduce the associated O&M cost by 40 man-hours per year. Initial results at GPU indicate a reduction in development time of the Lost Generation Reports by 2 man-hours per month, and that automation of the Daily Status Report relieves the station Group Shift Supervisor of approximately 45 man-hours per month.

A less tangible, but critical benefit, will be an improved communications infrastructure that will enable quicker and more flexible response to supporting transmission grid stability and control. This capability will also support needed improved information flow for quicker determination and control of generation costs as power producers evolve into the competitive arena. This improved communications infrastructure will become more crucial, with needs for improved speed and flexibility of unit control and corporate-wide communication as new owners and marketeers come onto the scene. Technology, such as that demonstrated by the ICCP demonstration projects, that provides connectivity not only between dispatch systems but also directly into the plant controls, will be an enabling factor in support of deregulation.

The preliminary results of the demonstration projects indicate that implementing ICCP on a utility's LAN/WAN infrastructure is feasible and cost-effective. Network communications security is paramount and sufficient security features have been designed into ICCP protocol to provide ample protection of corporate information and to ensure that the unit control is adequately protected from outside influences. The application and use of bilateral tables, which define data points through an ICCP node, provide protection from transmittal of undesired or extraneous information. Keep-alive functions of the nodes provide protection from losses due to node and/or communication line failures.

By deploying ICCP in power plant control systems, power producers will benefit from the improved accuracy, speed, and flexibility available in the protocol. ICCP has effectively become the national standard for communication between service areas and implementing the protocol in power plant control systems will improve their unit dispatch functionality through enhanced flexibility. This will become increasingly valuable as the industry evolves into deregulation where greater swings of the existing units are expected and enhanced needs for supporting transmission grid stability will be necessary and important with increased energy wheeling across systems.

A
DEFINITIONS

Common terms relative to ICCP and network administration are defined below.

API – Application Program Interface.

Action – An activity performed by the ICCP server under some defined circumstances.

Accounting Information – A set of information that describes an account for a utility. See IEC 870-6-802 for more details.

Analog Data - Data represented by a physical quantity that is considered to be continuously variable and whose magnitude is made directly proportional to the data or to a suitable function of the data.

Asynchronous Transmission - A method of data transmission that allows characters to be sent at irregular intervals by preceding each character with a start-bit and following it with a stop-bit.

Availability - The ratio, expressed as a percentage, of the total time a functional unit or service is capable of being used or is available to be used during a given interval to the length of the interval; for example, if the unit is not capable of being used for 20 minutes in a week, the availability is 99.8 percent (10080 - 20 minutes/10080 minutes X 100).

Backhaul - Point-to-point transmission from a remote site back to a central site for further distribution.

Baud - A unit measuring the rate of information flow, with five baud roughly equivalent to one alphanumeric character.

Bilateral Agreement – An agreement between two control centers that identifies the data elements and objects that can be accessed and the level of access permitted.

Bilateral Table – The computer representation of the Bilateral Agreement. The representation used as a local matter.

Binary Phase Shift Keying (BPSK) - A digital modulation scheme used in transmission communications.

Byte - A sequence of eight adjacent binary digits usually treated as a unit.

Call - (1) Any demand to set up a connection. (2) A unit of traffic measurement.

Circuit mode - A circuit-switched operational mode for transferring (transporting and switching) user information through a network. Contrast with packet switching mode.

Client – An ICCP user that requests services or objects owned by another ICCP user acting as a server. The client is a communicating entity that makes use of the VCC for the lifetime of an association via one or more ICCP service requests.

CNP – Communication Network Processor.

Data - Representation of facts, concepts, or instructions in a formalized manner suitable for communication, interpretation, or processing by humans or by automatic means.

Data Encryption Standard (DES) - A cryptographic algorithm for the protection of unclassified computer data, issued as Federal Information Processing Standard Publication 46-1.

Data Set – An object that provides services to group data values for singular operations by an ICCP client.

Data Value – An object that represents some alphanumeric quantity that is part of the VCC, which is visible to an ICCP user. Data Values exist as part of the implementation of the control center and represent either real entities within the utility such as current, or derived values calculated in the control center. Data Value objects include services for accessing and managing them.

dB Decibel - An analog unit of measure of signal strength, volume, or signal loss due to resistance as expressed in logarithmic form.

Demand Assigned Multiple Access (DAMA) - Refers to contention access schemes that allow multiple communications users to share a discrete portion of the bandwidth.

Digital Data - Data represented by discrete values or conditions (that is, “0” or “1”), as opposed to analog data.

Electronic Access - The capability to access information via on-line access (dedicated or dial-up), E-Mail, and FAX.

Electronic Data Interchange (EDI) - The exchange of routine business transactions in a computer-processable format, covering such traditional applications as inquiries, planning, purchasing, acknowledgments, pricing, order status, scheduling, test results, shipping and receiving, invoices, payments, and financial reporting.

EMS – Energy Management System.

Encrypt - To convert plain text into an unintelligible form by means of a cryptosystem.

Full-Duplex Operation - A mode of operation in which simultaneous communication in both directions can occur between two terminals. Contrast with half-duplex or simplex operation, in which communications occur in only one direction at a time.

Gateway - In a communication network, one of the network nodes equipped for interfacing with a network using different protocols. Note: A gateway might contain devices such as protocol translators, impedance matching devices, rate converters, fault isolation, or signal translators as necessary to provide system interoperability.

Half-Duplex Operation - That mode of operation in which communication between two terminals occurs in either direction but in only one direction at a time. Contrast with duplex or simplex operation. Note: Half-Duplex operation can occur on half-duplex circuits or on duplex circuits, but it cannot occur on simplex circuits.

Hertz (Hz) – Cycle per second; a measure of electromagnetic frequency that represents the number of complete electrical waves in a second. One kilohertz (kHz) is one thousand cycles per second; one megahertz (MHz) is one million; one gigahertz (gHz) is one billion.

Instance – An implementation of ICCP executed in either the client or the server role.

Integrated Services Digital Network (ISDN) - A network that provides end-to-end digital connectivity to support a wide range of services, including voice and non-voice services, to which users have access by a limited set of standard multi-purpose user network interfaces, as defined in the ITU-T1 series.

Internetworking - The process of interconnecting a number of individual networks to provide a path from a terminal or a host on one network to a terminal or a host on another network. The networks involved can be of the same type or they can be of different types, however, each network is distinct, with its own addresses, internal protocols, access methods, and administration.

Interchange Schedule – A set of information that specifies how energy is transferred from one system to another. See IEC 870-6-802 for more details.

Ka-band - A higher frequency than Ku-band, operating from 18 to 31gHz.

k/bs - Kilobit per second.

Ku-band - The range of frequencies between 11 and 14gHz, used increasingly by communications satellites.

Local Area Network (LAN) - A data communications system that (a) lies within a limited spatial area, (b) has a specific user group, (c) has a specific topology, and (d) is not a public switched telecommunications network, but can be connected to one.

Modem - Acronym for MOdulator-DEModulator. A device that modulates and demodulates signals. Note: Modems are primarily used for converting digital signals into quasi-analog signals

for transmission over analog communication channels and for reconverting the quasi-analog signals into digital signals.

Modulation - The process of superimposing an information signal onto a carrier for transmission.

Narrowcasting - Using the electronic media to reach a specific audience.

NICD – Network Interface Control Document. Used to clarify use of a standard and resolve specification issues.

Object – An abstract entity used to implement the ICCP protocol and represent data, and to optionally provide services for accessing that data within a VCC.

Object Model – An abstract representation that is used for real data, devices, operator stations, programs, event conditions, and event enrollments.

Operation – An activity that is performed by the ICCP server at the request of the ICCP client.

Packet - In data communication, a grouping of a sequence of binary digits, including data and control signals, that is transmitted and switched as a composite whole. The data, control signals, and possibly error control information, are arranged in a specific format. The packet can be of either fixed or variable length.

Packet Mode - A packet-switched operational mode for transferring (transporting and switching) user information through a network without establishing a connection. The packets do not necessarily arrive at their destination in the order they were sent, unlike the circuit mode of transmission. See packet switching.

Protocol - Any set of standard procedures that permit devices to intercommunicate.

Public-Switched Telephone Network (PSTN) - Any common carrier network that provides circuit switching among public users. Note: The term is usually applied to the public switched telephone network, but it could be applied more generally to other switched networks, for example, packet-switched public data networks.

Quadrature Phase Shift Keying (QPSK) - Digital modulation scheme used in transmission communications that allows increased sending capacity.

Server – An ICCP user that is the source of data and provides services for accessing that data. An ICCP server behaves as a VCC over the lifetime of an association.

Service – An activity that is either an ICCP action or operation.

Simplex Operation - That mode of operation in which communication between two points occurs in only one direction at a time. Contrast with half-duplex or duplex operation.

Synchronous Transmission - Digital transmission in which the time interval between any two similar significant instants in the overall bit stream is always an integral number of unit intervals. Note: Isochronous and anisochronous are characteristics, while synchronous and asynchronous are relationships.

T1 - TDM digital channel carrier (1.544 MBPS).

TAL – Time allowed to live.

Tagged – The term "tagged" is derived from the practice of putting a physical tag on a device as it is turned off for servicing or locked out from network access as a safety measure. The ICCP term "tagged" is used to signal such a condition to the ICCP user.

Time Series – A set of values of a given element that is taken at different times as specified by a single time interval. A time series is implemented through the transfer set mechanism as defined within this specification.

Transfer Account – A set of information that associates interchange scheduling information with either hourly or profile data.

Transfer Conditions – The events or circumstances under which an ICCP server reports the values of a data set, values in a time series, or all transfer account information.

Transfer Set – An object used to control data exchange by associating data values with transmission parameters such as time intervals, for example. There are four types of Transfer Sets: Data Set Transfer Sets, Time Series Transfer Sets, Transfer Account Transfer Sets, and Information Message Transfer Sets.

UCA™ - Utility Communication Architecture.

User – An implementation of ICCP executed in either the client or the server role.

Virtual Control Center (VCC) – An abstract representation of a real control center that describes a set of behavior with regard to communication and data management functionality limitations. VCC is a concept taken from the underlying MMS services.

WAN - Wide Area Network.

B
ABBREVIATIONS

General abbreviations of terms relative to ICCP and network administration.

ACSE – Association Control Service Element

AM – Amplitude Modulation

API – Application Program Interface

BCD – Binary Coded Decimal

BPSK – Binary Phase Shift Keying

CNP – Communication Network Processor

COV – Change of Value

DES – Data Encryption Standard

DCS – Distributed Control System

DIS – Draft International Standard

EDI – Electronic Data Interchange

EMS – Energy Management System

EPRI – Electric Power Research Institute

HLO – Hot Line Order

ICC – Inter-Control Center

ICCP – Inter-Control Center Communications Protocol

IDEC – Inter-Utility Data Exchange Consortium

IEC – International Electrotechnical Commission

IP – Internet Protocol

ISDN – Integrated Services Digital Network

KQH – Kilovar Hour Readings

KWH – Kilowatt Hour Readings

LAN – Local Area Network

LFC – Load Frequency Control

MMS – Manufacturing Messaging Specification

NICD – Network Interface Control Document

MOD – Motor Operated Disconnect

PDU – Protocol Data Unit

QOS – Quality of Service

RBE – Report-by-Exception

ROSE – Remote Operations Service Element

TAL – Time Allowed to Live

TASE – Tele-Control Application Service Element, IEC's designation of an international standard protocol for utility data exchange.

TASE.1 – TASE based on the ELCOM-90 Protocol

TASE.2 – TASE based on the ICCP Protocol

TCP – Transmission Control Protocol

TLE – Time Limit for Execution

TOD – Time of Day

UCA™ – Utility Communications Architecture

UCS – Utility Communications Standards Working Group

UDP – User Datagram Protocol

VCC – Virtual Control Center

VMD – Virtual Manufacturing Device

WAN – Wide Area Network

WSCC – Western System Coordinating Council

WEIC – WSCC Energy Management System Inter-Utility Communications

WEICG – WSCC Energy Management Systems Inter-Utility Communications Guidelines

C
REFERENCES

Utility Communication Architecture (1.0), EL-7547, December 1991.

MMS Feasibility Report, ECC, Inc./ONE, December 1991.

UCS Benchmark Study, ECC, Inc./ONE, August 1992.

ICCP Scope Document, RP3355-02, ECC, Inc., November 1992.

ICCP Specification (3 Volumes), December 1994, (Version 5.1).

- IEC 870-6-503, Service and Protocol
- IEC 870-6-702, Application Profile
- IEC 870-6-802, Object Models

Ohio Edison Workshop and Demonstration, August 1994.

WAPA Workshop and Demonstration, September 1994.

Inter-Control Center Communications Protocol (ICCP) Demonstration, TR-105800, Project 3830-01, November 1995.

Inter-Control Center Communications Protocol (ICCP) User's Guide, TR-107176, Project 4379-01, December 1996.

"Inter-Control Center Communications Protocol (ICCP): A Useful Tool to Provide Connectivity of Digital Systems and Energy Management Systems," a paper presented at the Honeywell User Group and the ISA/POWID Symposiums, June, 1999, by John Brummer and Ken Gray, GPU GenCo and Mike Stapley, Honeywell, and Rabon Johnson, EPRI I&C Center.

D
ICCP SPECS

D.1 Specifications Overview

The Inter-Control Center Communications Protocol (ICCP) allows for data exchange of Wide Area Networks (WANs) between a utility control center and other utilities, power pools, regional control centers, power plants, substation computers, and Non-Utility Generators. Data exchange information consists of real-time and historical power system monitoring and control data, including measured values, scheduling data, energy accounting data, and operator messages. This data exchange occurs between one control center's SCADA/EMS host and another center's host, often through one or more intervening communications processors. ICCP defines a mechanism for exchanging time critical data between control centers. In addition, it provides support for device control, operator station messaging, and control of programs at a remote control center.

The ICP protocol relies on the use of MMS services (and hence, the underlying MMS protocol) to implement the control center data exchange. Figure D-1 shows the relationship of ICCP, the MMS provider, and the rest of the OSI stack. In most cases, the values of objects being transferred are translated from/to the local machine representation automatically by the local MMS provider. Some ICCP objects require a common syntax (representation) and meaning (interpretation) by both communicating ICCP systems. This common representation and interpretation constitutes a form of protocol.

The control center applications are not part of ICCP. It is assumed that these applications request ICCP operations, and supply control center data and functions to the ICCP implementation as needed. (The specific interface between ICCP and the control center applications is a local issue.) In some cases, the control center applications are distributed and might reside in a processor that is part of an Energy Management System (EMS), a redundant configuration that uses communications servers or Communication Network Processors (CNPs) to provide the ICCP protocol. Interfaces to such processors are not part of ICCP, but ICCP does support this distributed concept (local issue).

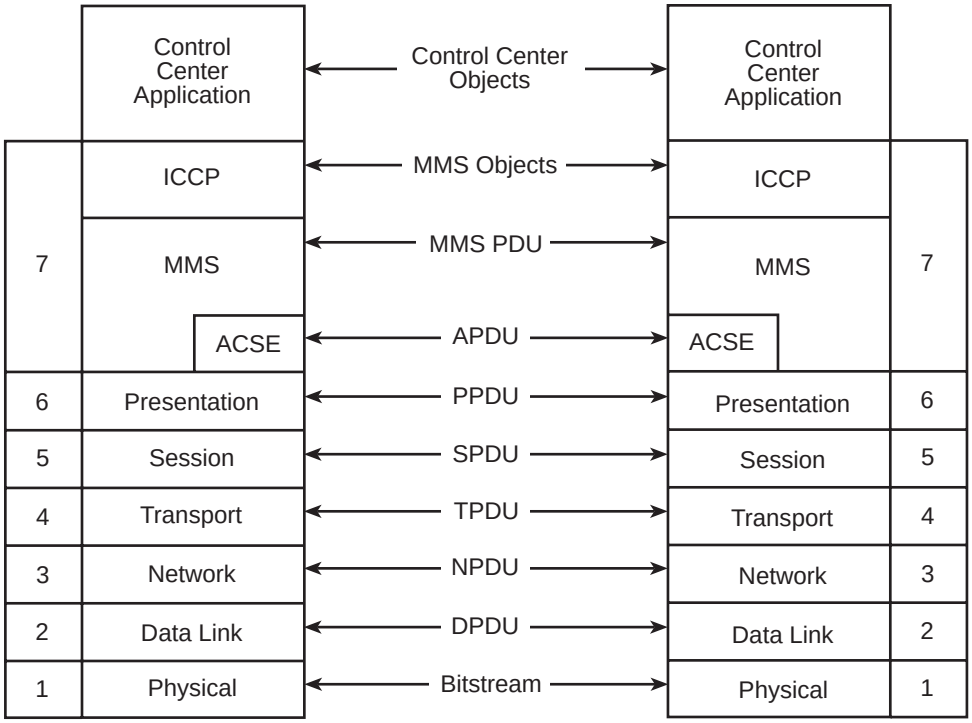


Figure D-1
Protocol Relationships

D.2 Guidelines for ICCP Development

The development of the USC/ICCP architecture and protocol suite was based on the following guidelines:

- Where possible, existing standards and concepts were used such as UCA™, IEC, WG07 TASE profiles, ISO, and MAP/TOP. This results in lower-cost implementation and reduces duplication of effort.
- Media independence (allows the best media possible for the intended application).
- The architecture facilitates the implementation of front-end processors.
- Focus was on the requirements of the electric power control center communications as determined by the Utility Communication Specification Working Group. Key requirements include:
  - Ability to handle real-time, high priority messages and mix with larger, low priority messages such as file transfers.
  - High reliability for critical data.
  - Shared communication media to lower costs.
  - High security (data exchange shall be controlled by Bilateral Agreements and encryption shall be possible).

  - Ability to handle a wide variety of control center data types.
  - Failover/Restart to recover from control Center system or front-end processor failures (local issue).

- Functionality levels (called Conformance Blocks) are defined to allow for simple (and therefore, low-cost) implementations. See Table D-1 for the Conformance Blocks.

Table D-1
ICCP Conformance Blocks

Block 1 – Basic Services

Association Objects

Initiate
Conclude
Abort

Data Value Objects

Get Data Value
Set Data Value
Get Data Value Names
Get Data Value Type

Data Set Objects

Create Data Set
Delete Data Set
Get Data Set Element Values
Set Data Set Element Values
Get Data Set Names
Get Data Set Element Names

Transfer Set Objects

Start Transfer
Stop Transfer
Data Set Transfer Set Condition Monitoring

Next Transfer Set Object

Get Next Transfer Set Value

Block 2 – Extended Data Set Condition Monitoring

Integrity TimeOut Conditions
Object Change (Report-by-Exception)

Block 3 – Blocked Transfers

Server generates Block Data (encoded as octet data, understood by both ends for efficiency purposes)

Block 4 – Operator Stations

Output (transfer of text messages to logical stations)

Block 5 – Device Control

Select
Operate
Timeout
Local Reset
Success
Failure

Block 6 – Programs

Start
Stop
Resume
Reset
Kill
Get Program Attributes

Block 7 – Events

Device Objects
Success
Failure

Event Condition Objects
Event Notification

Event Enrollment Objects
Create Event Enrollment
Delete Event Enrollment
Get Event Enrollment Attributes

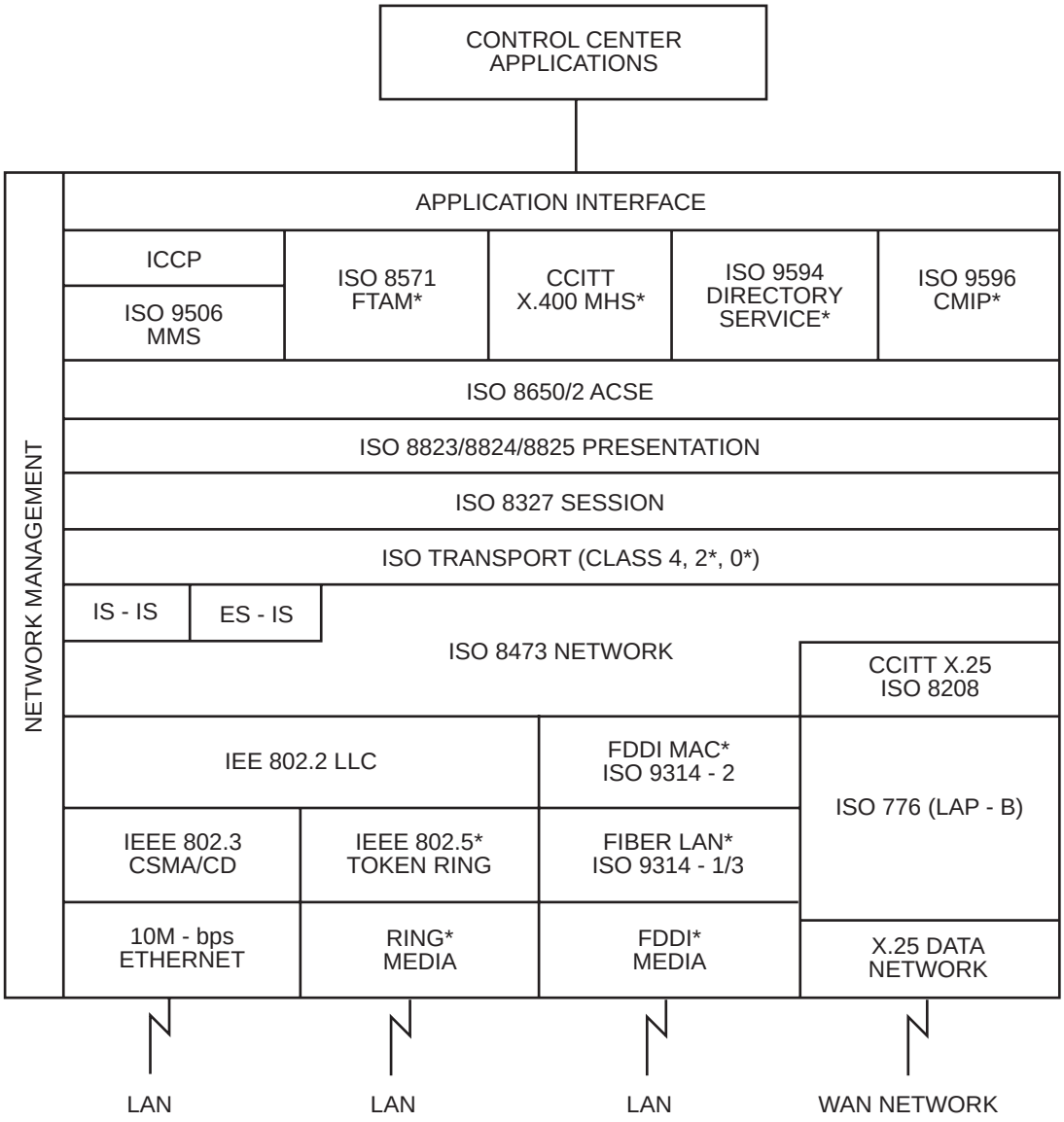
Block 8 – Accounts

Interchange Schedule Objects
Accounting Information Objects
Transfer Account Objects

Block 9 – Time Series Data

Series Data Points (Sampled Over Time Interval)

- Depending on the functions and media, ICCP can run over simple, direct links or LANs, or over more complex network structures supporting dynamic, adaptive routing (see Figure D-2)
- For ease of implementation and software maintenance, the ICCP supports object-oriented definitions of data and functions (See Table D-2 for the list of objects).



* Indicates optional function

Figure D-2
UCA™/UCS Protocol Architecture

Table D-2
ICCP Objects

Bilateral Table and Protocol Objects (see Clause 8 of ICCP, Section 503, Version 5.1)

Supervisory Control and Data Acquisition
- IndicationPoint Object (Values, Quality, Timestamp, and Change Counter)
- ControlPoint Object (Commands, Values, Tags)
- Protection Equipment Event Object Model (Events, Times, Class)

Transfer Accounts (Schedules, Accounts, Times, Wheeling, Ramps)
- TransferAccount Object
- TransmissionSegment Object
- HourlyValue Object
- ProfileValue Object
- AccountRequest Object

Miscellaneous Messages
- InformationBuffer Object (Text or Binary Data for Generic Application)

- Network Management shall be provided to allow for ease of expanding the architecture, changing bilateral agreements, modifying objects, exchanges and data structures, adding circuits or media types, and gathering information on network operation and performance. (This is a local implementation issue.)
- The architecture shall be flexible to allow for the future addition of new protocols (such as the EPRI Database Access Integrated Service). To this end, the profiles and layers shall adhere to transparency principles to allow modification of lower layers without impacting applications.
- The Client/Server architecture supports a variety of transmission sequences including:
  - Asynchronous, event-oriented, messages can be sent at any time over a previously established connection.
  - Applications can request data using a “one-shot” message.
  - Periodic data exchanges can be set up by writing the periodic parameters (such as start time, periodicity, and data identification) to an ICCP server at the data owner. This might be over a connection established by two applications or might be set up for data set transfers where applications access the information from a database. The latter will reduce the number of transfers where applications might share the same information.
  - Data (periodic or on change event) can be in report-by-exception (RBE) format where only the changed data is sent.
  - Data values are identified using MMS object names or, for time critical data, can be binary blocks that are understood by the applications that might use definition messages to coordinate the data value message formats.

- There are a number of local issues (for example, network addressing, and human-machine interface) that are required in a complete working ICCP system. These were deliberately left to ICCP implementers and users.

D.3 Rationale for MMS Selection

The MMS standard (ISO/IEC 9506) was designed to facilitate the exchange of application data among manufacturing and process control systems. MMS development work occurred within the International Organization for Standardization (ISO) committees with broad representation across industries. The objective was to develop a messaging system that was robust and broad enough in scope to address the needs of many industry sectors, thereby reducing overall costs and leveraging product investment across industries.

MMS was selected for the messaging service for ICCP because:

- MMS is supported by a variety of computer and control device manufacturers. This support includes multiple network media and profiles such as ISO/IEC8802.3, 8802.4, TCP/IP and X.25.
- MMS is evolving in a number of areas relevant to making MMS a robust, widely adopted standard across multiple industries. Work is progressing in the areas of conformance and interoperability testing, database standards, efficient encoding methods, and companion standards.
- MMS is expected to be used in other areas of electric power utilities, such as SCADA applications and control applications. The use of a common protocol throughout these functions will reduce costs.
- MMS provides a standardized service to meet all of the identified message exchange sequences (including control sequences) for support of ICCP.
- MMS is object-oriented, satisfying one of the key guidelines for ICCP.
- The benchmark results showed that real-time data could be exchanged via MMS in a timely manner with minimal overhead using data-by-exception or blocked data. As improved encoding standards become available, they can easily be incorporated into MMS.

D.4 ICCP Protocol Details

The protocol stack proposed is the standard UCA™ profile with UCS applications at the top and MMS providing the messaging service for the ICCP (which handles real-time messages) Table D-3 summarizes the key features of ICCP.

ICCP is based on client/server concepts. For example, if a client (application) runs periodically and needs data form the server (owner of the data), the client can establish an association and request the periodic transfer of the data.

Table D-3
ICCP Key Features

Negotiation	At connection time, each node automatically negotiates with the calling mode to determine the features that will be supported and utilized at that node.
Object-Oriented	Data is modeled in terms of objects, grouping all relevant data into a single object.
Real-Time Support	Allows for automatic periodic or even-drive data (with a single "poll"), discard of old data (based on a time-allowed-to live).
Periodic Exception Data	At an established periodicity, only the data that has changed is sent (except for a periodic integrity check) to reduce the amount of traffic on the network.
Immediate Exception Data	Data that changes is immediately exchanged (except for a periodic integrity check) to reduce the amount of traffic on the network.
Event Driven Data	The transmission of selected data and the identification of the trigger that caused the transmission of that data.
Event Enrollment	A receiving utility can choose to enroll in the specific event at the originating utility. When that event occurs, pre-arranged data and identifiers will be transmitted.
Program Initiation	A program can be initiated in the remote system to perform an agreed-upon task.
End-to-End Acknowledgment	A positive acknowledgment of the receipt of data is sent from the receiving end to the transmitting end.
Priority Messaging	The ability to define priority levels at the transport and network layer (four will be implemented initially) for control, exception reporting, periodic, report, file transfer, and so on.
Extended Messages	The exchange of a message exceeding 128 characters in length (disturbance descriptions, file transfer, and so on).
Security	Authentication of connection applications is assured and access to the data functions is controlled by bilateral tables.
Congestion Control	Congestion on the network is detected and corrective actions are automatically taken to alleviate the congestion.
Dynamic Routing	Alternate routes in a mesh network are selected based on priority, channel traffic, cost, and security. The availability of each node will be known to the other nodes on the network.

Table D-3 (cont.)
ICCP Key Features

Time Sequence Data	A message that contains a sequence of values for a single database point. For example, a time sequence trend/graph of an analog point.
Encryption	Encoded data for additional security (pending standardization).
Application Level Broadcast	Simultaneous transmission of a message to all or selected groups of utilities.
Media Independence	Any network and media type (TCP/IP, LAN, X.25, Ethernet, WAN, etc.) can be used. LAN to WAN routing is automatic.
Efficiency Blocking	Data can be blocked to improve efficiency. Applications handle the translation.
OSI and UCA™ Compliance	Puts this protocol in the mainstream and allows for future enhancements such as DAIS.

At the Network Layer, ICCP specifies the Information System to Information System (IS-IS) protocol to allow for dynamic adaptive link-state routing. The IS-IS software creates the forwarding database, which is then used by the Connectionless Network Layer Protocol to route packets to the proper system. IS-IS allows ICCP networks to reconfigure themselves automatically. In this way, messages from one utility might traverse many different paths over time, depending on what the physical topology of the network is at the time the communication takes place. Different data paths through the network might occur, even on a single connection without the ICCP application being affected by this dynamic network reconfiguration. It should be noted that ICCP can also be used in configurations that use standard off-the-shelf routers to provide the IS-IS routing protocol.

Object modeling techniques were effectively applied in the design of ICCP. All ICCP operations and actions run from protocol objects. For example, the Bilateral Table, which defines what data and functions can be accessed by the remote control center, is an ICCP Protocol object. Also, all data and control elements are defined as objects. Using these techniques, ICCP is easier to implement and maintain. New data objects can be added and exchanged without modifying the protocol services.

Supported Data Types include control messages, status, analogs, quality codes, schedules, text and simple files. In addition to data exchange, other functions are Remote Control, Operator Station Output, Events, and Remote Program Execution.

E
IEC 870-6-503 TASE.2 SERVICES AND PROTOCOL, VERSION 1996-08 EXCERPTS

E.1 Introduction

The Tele-Control Application Service Element (TASE.2) Protocol (also known as Inter-Control Center Communications Protocol, ICCP) allows for data exchange over Wide Area Networks (WANs) between a utility control center and other control centers, other utilities, power pools, regional control centers, and Non-Utility Generators. Data exchange information consists of real-time and historical power system monitoring and control data, including measured values, scheduling data, energy accounting data, and operator messages. This data exchange occurs between one control center's SCADA/EMS host and another center's host, often through one or more intervening communications processors.

This section of IEC 870-6 defines a mechanism for exchanging time-critical data between control centers. In addition, it provides support for device control, general messaging and control of programs at a remote control center. It defines a standardized method of using the ISO/IEC 9506 Manufacturing Message Specification (MMS) services to implement the exchange of data. The definition of TASE.2 consists of three documents. This section of IEC 870-6 (future IEC 870-6-503) defines the TASE.2 application modeling and service definitions. The future IEC 870-6-702 will define the Application Profile for use with TASE.2. The future IEC 870-6-802 will define a set of standardized object definitions to be supported.

The TASE.2 describes real control centers with respect to their external visible data and behavior using an object-oriented approach. The objects are abstract in nature and can be used for a wide variety of applications. The use of TASE.2 goes far beyond the application in control center-to-control-center communications. The specification must be understood as a tool box for any application domain with comparable requirements, that is, the TASE.2 can be applied in areas like substation automation, power plants, factory automation, chemical plants, or other areas that have comparable requirements. It provides a generic solution for advanced Information and Communication Technology.

The TASE.2 version number for this standard is 1996-08. See IEC 870-6-503, Section 8.2.3 for more details.

E.2 Scope

This section of IEC 870-6 specifies a method of exchanging time-critical control center data through wide- and local-area networks using a full ISO-compliant protocol stack. It contains provisions for supporting both centralized and distributed architectures. This standard includes the exchange of real-time data indications, control operations, timeseries data, scheduling and accounting information, remote program control and event notification.

Though the primary objective of TASE.2 is to provide control center (tele-control) data exchange, its use is not restricted to control center data exchange. It can be applied in any other domain having comparable requirements. Examples of such domains are power plants, factory automation, process control automation, and others.

This standard does not specify individual implementations or products, nor does it constrain the implementation of entities and interfaces within a computer system. This standard specifies the externally visible functionality of implementations, together with conformance requirements for such functionalities.

E.3 Control Center

The model of a control center includes four primary classes of host processors: SCADA/EMS, DSM/Load Management, Distributed Applications, and Display Processors. The SCADA/EMS host is the primary processor, utilizing analog and digital monitoring data collected at power plants, Non-Utility Generators, and transmission and distribution substations via Data Acquisition Units (DAUs) and Remote Terminal Units (RTUs). The control center typically contains redundant SCADA/EMS hosts in a “hot standby” configuration. The DSM/Load Management host(s) are used by either an operator or an EMS application to initiate load management activities. The Distributed Application host(s) perform miscellaneous analysis, scheduling, or forecasting functions. Display Processors allow for local operator and dispatcher display and control. Typically, the control center will contain one or more Local Area Networks (LANs) to connect these various hosts. The control center will also access several WANs, often through intermediate communications processors. These WAN connections might include the company-wide area network for communications with the corporate host and a distinct real-time SCADA network. Each control center will also have one or more TASE.2 instances to handle data exchange with remote control centers.

Other classes of host processors, like archive systems, engineering stations, or quality control systems (for example, for data recording according to ISO 9000) might also be included. The application of the TASE.2 control center model is, in principle, unlimited. This model provides a common and abstract definition applicable for any real systems that have comparable requirements.

E.4 Architecture

The TASE.2 protocol relies on the use of MMS services (and hence, the underlying MMS protocol) to implement the control center data exchange. Figure E-1 shows the relationship of TASE.2, the MMS provider, and the rest of the OSI stack. In most cases, the values of objects being transferred are translated from/to the local machine representation automatically by the local MMS provider. Some TASE.2 objects require a common syntax (representation) and meaning (interpretation) by both communicating TASE.2 systems. This common representation and interpretation constitutes a form of protocol. The control center applications are not part of this standard. It is assumed that these applications request TASE.2 operations and supply control center data and functions to the TASE.2 implementation as needed. The specific interface between TASE.2 and the control center applications is a local issue and not part of this standard.

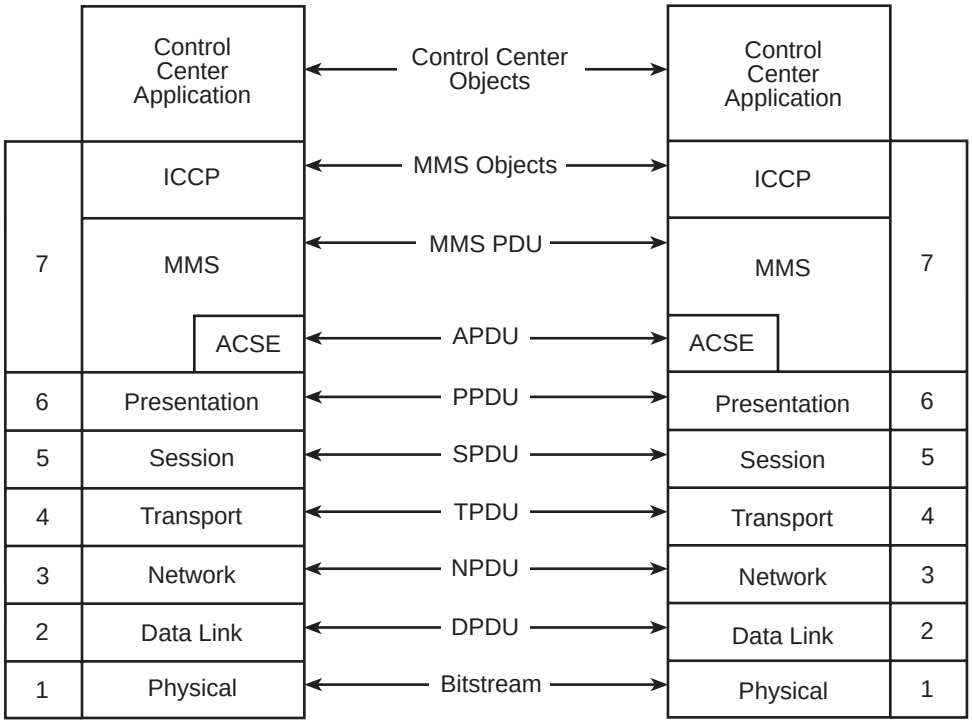


Figure E-1
Protocol Relationships

E.5 Network Model

The TASE.2 Data Exchange network can be either a private or public packet-switched or mesh network connecting communications processors that provide adequate routing functionality to allow for redundant paths and reliable service.

Figure E-2 shows a typical network topology using a packet-switched configuration. The packet-switch provides routing and reliable service between control centers (which might include internal networks and routing capabilities).

The mesh network shown in Figure E-3 demonstrates the concept of redundant paths for a mesh network. Each control center maintains its own series of direct circuits, and also provides a mechanism for routing between those direct circuits. Control Center C provides an alternate routing path for network traffic going from Control Center A to B. This network configuration requires key control centers to provide significant routing capabilities.

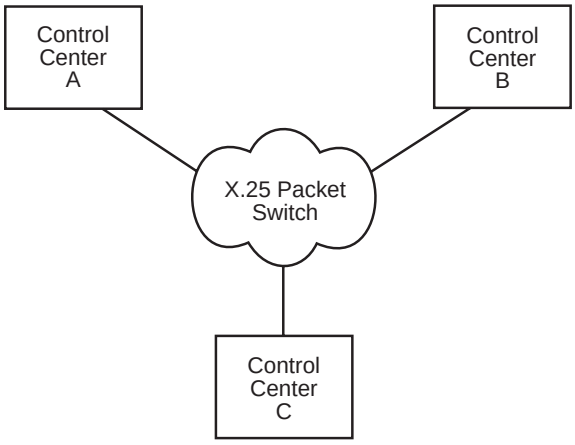


Figure E-2
Packet-Switched Network

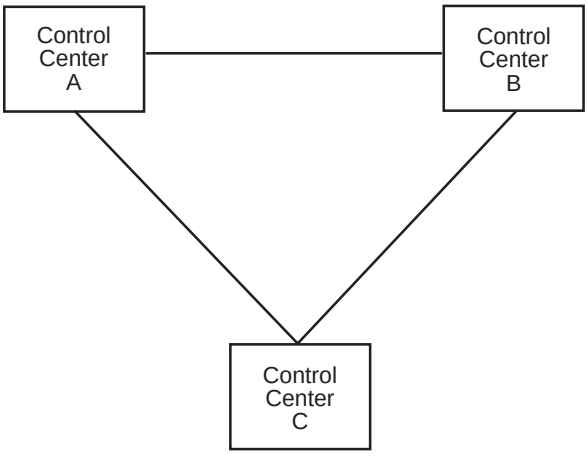


Figure E-3
Mesh Network

E.6 Relation between TASE.2 and MMS

The TASE.2 resides on top of MMS. It describes a standardized application of MMS using the MMS services and protocol. TASE.2 enhances the functionality of MMS by specifying structured data mapped to MMS objects and assigning specific semantics to it. As an example for pure MMS services, MMS allows reading data from a remote system. The data will be responded without any specific condition. If these data shall be read depending on very specific conditions (for example, on change only), then TASE.2 provides appropriate services that are not provided by MMS.

Though the specific requirements agreed upon within IEC TC 57 have led to the definition of TASE.2, there are several other application domains (outside the control centers) with less, very limited, or mixed requirements that might use the TASE.2 services. These other areas are outside the scope of this standard, but the use of TASE.2 goes far beyond the specific scope of this standard.

TASE.2 provides an independent and scalable set of services to allow efficient implementations optimized for the respective requirements of a control center. It does this by defining several conformance building blocks (CBBs). MMS also offers a scalability of its services specifying MMS CBBs. A simple TASE.2 implementation requires only a simple MMS implementation.

TASE.2 and MMS provide their services to their respective users. MMS provides its services to TASE.2 and TASE.2 provides its services to the control center application. MMS is an independent standard that can also provide its services to users other than TASE.2—it might serve directly to specific control center applications or to any other application. This means that the use of MMS is not restricted to TASE.2.

For requirements outside the scope of this standard or for future requirements (for example, journaling of data, downloading and uploading of mass data such as programs), additional MMS models and services (that is, Journaling and Domain Loading) can be applied by a real system in addition to TASE.2. This is possible because the additional application of MMS objects and services is independent of the use of TASE.2 and the use of MMS by TASE.2.

E.7 Normative References

The following normative documents contain provisions that, through reference in this text, constitute provisions of this section of IEC 870-6. At the time of publication, the indicated editions were valid. All normative documents are subject to revision and parties to agreements based on this section of IEC 870-6 are encouraged to investigate the possibility of applying the most recent editions of the normative documents indicated below. Members of IEC and ISO maintain registers of currently valid International Standards.

57/229/CDV: Tele-Control Equipment and Systems - Part 6: Tele-Control protocols compatible with ISO standards and ITU-T recommendations—Section 702: TASE.2 application profile (future 870-6-702, in preparation)

57/230/CDV: Tele-Control Equipment and Systems—Part 6: Tele-Control protocols compatible with ISO standards and ITU-T recommendations—Section 802: TASE.2 object models (future 870-6-802, in preparation)

ISO/IEC 8073: Information Processing Systems—Open systems interconnection—Connection-oriented transport protocol specification

ISO/IEC 8473: Information Processing Systems—Data Communications Protocol for providing the connectionless-mode network service

ISO/IEC 8649: Information Processing Systems—Open systems interconnection—Service definition for the association control service element

ISO/IEC 8802-3: Information Processing Systems—Local area networks—Part 3: carrier sense multiple access with collision detection (CSMA/CD) access method and physical layer specifications

ISO/IEC 9506-1: 1990 Industrial Automation Systems—Manufacturing Message Specification—Part 1: Service Definition

ISO/IEC 9506-2: 1990 Industrial Automation Systems—Manufacturing Message Specification—Part 2: Protocol Specification

ISO/IEC 9542: Information Processing Systems—Telecommunications and information exchange between systems—End system to intermediate system routing exchange protocol for use in conjunction with the protocol.

ISO/IEC 10589: Information Technology—Telecommunications and information exchange between systems—Intermediate system to intermediate system intro-domain routing exchange protocol.

ISO/DISP 10613-1: Information Technology—International Standardized Profile RA—Relaying the Connectionless-Mode Network Service—Part 1: Relay Function General Overview and Sub-network-Independent Requirements

ISO/DISP 10613-2: Information Technology—International Standardized Profile RA—Relaying the Connectionless-Mode Network Service—Part 2: LAN Sub-network Dependent Media Independent Requirements

ISO/DISP 10608-1: Information Technology—International Standardized Profile TA—Connection-Mode Transport Service over Connectionless-Mode Network Service—Part 1: General Overview and Sub-network-independent requirements

ISO/DISP 10608-2: Information Technology—International Standardized Profile TA—Connection-Mode Transport Service over Connectionless-Mode Network Service—Part 2: TA51 profile including sub-network-dependent requirements for CSMA/CD Local Area Networks (LANs)

F
IEC 870-6-702 TASE.2 PROFILES, VERSION 1996-11 EXCERPTS

F.1 Introduction

This section of IEC 870-6 is one of the IEC 870-6 series defining functional profiles to be used in telecommunication networks for electric power systems. It is largely based on existing ISO/IEC International Standards and International Standardized Profiles (ISP).

The notion of Functional Profiles is fundamental in the organization of IEC 870-6. A description of Functional Profiles, their classification scheme, and the manner of defining them are outlined in IEC 870-6-1.

This profile for Tele-Control Application Service Element (TASE.2, also known as Inter-Control Center Communications Protocol, ICCP) is an Application-class Profile providing communications capabilities to Control Center applications. The TASE.2 in the Application Layer is specified in the future IEC 870-6-503. The present standard refines the Application Layer protocol to meet interoperability requirements and specifies requirements on the Presentation and Session Layers' support for TASE.2. TASE.2 operates in a connection mode, so this A-profile needs to interface to a Transport-class profile of the T-profile variety.

Because the TASE.2 is an MMS-based protocol, this Functional Profile (FP) is based on MMS profiles. In the OSI International Standardized Profile taxonomy, there is a category for MMS A-profiles. The present standard makes frequent use of the AMM11 profile.

F.2 Scope

This section of IEC 870-6 is a Functional Profile (FP) and defines the provision of the TASE.2 communications services between two Control Center End Systems. It is supported by the Transport Services implemented in accordance with Transport-profiles defined for the type of network that interconnects the Control Center End Systems. This is demonstrated in Figure F-1.

This FP also defines the provision of the OSI Connection-mode Presentation and Session Services between the End Systems.

DISP 14226 specifies the AMM11 profiles for MMS. The parts of ISP 14226 that cover the profile used as a basis for this FP are ISP 14226-1 and ISP 14225-2. This FP is in alignment with ISP 14226, as far as possible and maintains this compatibility by reference. There are TASE.2 requirements in addition to ISP 14226. Such requirements are specified in this FP.

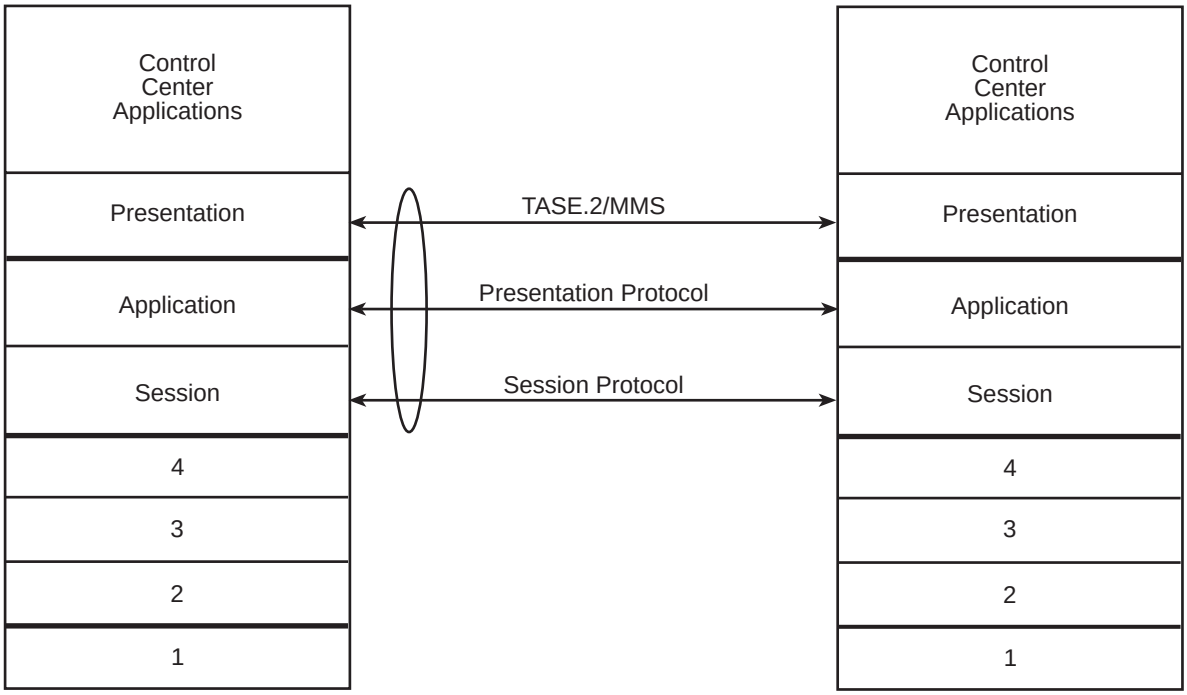


Figure F-1
Applicability of Functional Profile

F.3 Normative References

The following normative documents contain provisions that, through reference in this text, constitute provisions of this section of IEC 870-6. At the time of publication, the indicated editions were valid. All normative documents are subject to revision, and parties to agreements based on this section of IEC 870-6 are encouraged to investigate the possibility of applying the more recent editions of the normative documents indicated below. Members of IEC and ISO maintain registers of currently valid International Standards.

ISO/IEC 8650-2: 1995 Information Technology—Open Systems Interconnection—Protocol Specification for the Association Control Service Element—Part 2: Protocol Implementation Conformance Statement (PICS) Proforma

ISO/IEC 8822: 1994 Information Technology—Open Systems Interconnection—Presentation service definition

ISO/IEC 8823-7: 1994 Information Technology—Open Systems Interconnection—Connection-Oriented Presentation Protocol: Protocol Specification

ISO/IEC 8823-2: 1995	Information Technology—Open Systems Interconnection—Connection-Oriented Presentation Protocol—Part 2: Protocol Implementation Conformance Statement (PICS) Proforma
ISO/IEC 8824: 1990	Information Processing Systems—Open Systems Interconnection—Specification of Abstract Syntax Notation One (ASN.1)
ISO/IEC 8825: 1990	Information Technology—Open Systems Interconnection—Specification of Basic Encoding Rules for Abstract Syntax Notation One (ASN.1)
ISO/IEC 9506-1: 1990	Industrial Automation Systems—Manufacturing Message Specification—Part 1: Service definition
ISO/IEC 9506-2: 1990	Industrial Automation Systems—Manufacturing Message Specification—Part 2: Protocol specification
ISO/IEC TR 10000-1:1992	Information Technology—Framework and Taxonomy of International Standardized Profiles—Part 1: Framework
ISO/IEC TR 10000-2:1994	Information Technology—Framework and Taxonomy of International Standardized Profiles—Part 2: Principles and Taxonomy for OSI Profiles
ISO/IEC DISP 14226-1	Information Technology—International Standardized Profile AMM11: MMS General Application Base Profile—Part 1: Specification of ACSE, presentation and session protocols for the use by MMS (in preparation)
ISO/IEC DISP 14226-2	Information Technology—International Standardized Profile AMM11: MMS General Application Base Profile—Part 2: Common MMS requirements (in preparation)
ISO 8326: 1987	Information Technology—Open Systems Interconnection—Basic connection-oriented session service definition
ISO 8327: 1987	Information Technology—Open Systems Interconnection—Basic connection-oriented session protocol specification
ISO/DIS 8327-2	Information Technology—Open Systems Interconnection—Basic connection-oriented session protocol specification—Part 2: Protocol implementation conformance statement (PICS) Proforma

ISO 8649: 1988	Information Processing Systems—Open Systems Interconnection—Service Definition for the Association Control Service Element (ACSE)
ISO 8650: 1988	Information Processing Systems—Open Systems Interconnection—Protocol Specification for the Association Control Service Element (ACSE)
57/228/CDV	Tele-Control Equipment and Systems—Part 6: Tele-Control protocols compatible with ISO standards and ITU-T recommendations—Section 503: TASE.2 services and protocol (future IEC 870-6-503, in preparation)

G
IEC 870-6-802 TASE.2 OBJECT MODELS, VERSION 1996-08 EXCERPTS

G.1 Introduction

The primary purpose of Tele-Control Application Service Element (TASE.2) is to transfer data between control systems and to initiate control actions. Data is represented by object instances. This section of IEC 870-6 proposes object models from which to define object instances. The object models represent objects for transfer. The local system might not maintain a copy of every attribute of an object instance.

The object models presented herein are specific to “control center” or “utility” operations and applications; objects required to implement the TASE.2 protocol and services are found in the future IEC 870-6-503. Since needs will vary, the object models presented here provide only a base; extensions or additional models might be necessary for two systems to exchange data not defined within this standard.

It is by definition that the attribute values (that is, data) are managed by the owner (that is, source) of an object instance. The method of acquiring the values is implementation-dependent; therefore, accuracy is a local matter.

The notation of the object modeling used for the objects specified in Clause 5 of this section of 870-6 is defined in the future IEC 870-6-503. It should be noted that this section of 870-6 is based on the TASE.2 services and protocol. To understand the modeling and semantics of this standard, some basic knowledge of the future IEC 870-6-503 is recommended.

Clause 5 describes the control center-specific object models and their application. They are intended to provide information to explain the function of the data.

Clause 6 defines a set of MMS-type descriptions for use in exchanging the values of instances of the defined object models. It is important to note that not all attributes of the object models are mapped to types. Some attributes are described simply to define the processing required by the owner of the data and are never exchanged between control centers. Other attributes are used to determine the specific types of MMS variables used for the mapping and, therefore, do not appear as exchanged values themselves. A single object model can also be mapped onto several distinct MMS variables, based on the type of access and the TASE.2 services required.

Clause 7 describes the mapping of instances of each object type MMS variables and named variable lists for implementing the exchange.

Clause 8 describes device-specific codes and semantics to be used with the general objects.

An Informative Annex is included which describes some typical interchange scheduling scenarios, along with the use of TASE.2 objects to implement the schedule exchange.

G.2 Scope

This section of IEC 870-6 specifies a method of exchanging time-critical control center data through wide- and local-area networks using a full ISO-compliant protocol stack. It contains provisions for supporting both centralized and distributed architectures. The standard includes the exchange of real-time data indications, control operations, time series data, scheduling and accounting information, remote program control and event notification.

G.3 Normative References

The following normative documents contain provisions that, through reference in this text, constitute provisions of this section of IEC 870-6. At the time of publication, the indicated editions were valid. All normative documents are subject to revision and parties to agreements based on this section of IEC 870-6 are encouraged to investigate the possibility of applying the most recent editions of the normative documents indicated below. Members of IEC and ISO maintain registers of currently valid International Standards.

ISO/IEC 9506-1: 1990	Industrial Automation Systems—Manufacturing Message Specification—Part 1: Service definition
ISO/IEC 9506-2: 1990	Industrial Automation Systems—Manufacturing Message Specification—Part 2: Protocol specification
57/228/CDV:	Tele-Control Equipment and Systems—Part 6: Tele-Control protocols compatible with ISO standards and ITU-T recommendations—Section 503: TASE.2 services and protocol (future IEC 870-6-503, in preparation)