IEEE STANDARDS ASSOCIATION

IEEE Standard for Electric Power Systems Communications—Distributed Network Protocol (DNP3)

IEEE Power and Energy Society

Sponsored by the
Transmission and Distribution Committee
and
Substations Committee

IEEE
3 Park Avenue
New York, NY 10016-5997
USA

10 October 2012

IEEE Std 1815™-2012
(Revision of
IEEE Std 1815-2010)

IEEE Std 1815™-2012
(Revision of
IEEE Std 1815-2010)

IEEE Standard for Electric Power Systems Communications—Distributed Network Protocol (DNP3)

Sponsor

Transmission and Distribution Committee

and

Substations Committee

of the

IEEE Power and Energy Society

Approved 8 June 2012
IEEE-SA Standards Board

The Working Group thanks the International Electrotechnical Commission (IEC) for permission to reproduce information from the International Standards IEC/TS 62351-3 ed. 1.0 (2007), IEC/TS 62351-5 ed. 1.0 (2009), and IEC/TS 62351-8 ed. 1.0 (2011).

Abstract: The DNP3 protocol structure, functions, and interoperable application options (subset levels) are specified. The simplest application level is intended for low-cost distribution feeder devices, and the most complex for full-featured systems. The appropriate level is selected to suit the functionality required in each device. The protocol is suitable for operation on a variety of communication media consistent with the makeup of most electric power communication systems.

Keywords: Distributed Network Protocol (DNP3), distribution automation, distribution feeder, electric power communication systems, IEEE 1815, master station, substation automation

The Institute of Electrical and Electronics Engineers, Inc.
3 Park Avenue, New York, NY 10016-5997, USA

PDF: ISBN 978-0-7381-7292-7 STD97267
Print: ISBN 978-0-7381-7344-3 STDPD97267

Notice and Disclaimer of Liability Concerning the Use of IEEE Documents: IEEE Standards documents are developed within the IEEE Societies and the Standards Coordinating Committees of the IEEE Standards Association (IEEE-SA) Standards Board. IEEE develops its standards through a consensus development process, approved by the American National Standards Institute, which brings together volunteers representing varied viewpoints and interests to achieve the final product. Volunteers are not necessarily members of the Institute and serve without compensation. While IEEE administers the process and establishes rules to promote fairness in the consensus development process, IEEE does not independently evaluate, test, or verify the accuracy of any of the information or the soundness of any judgments contained in its standards.

Use of an IEEE Standard is wholly voluntary. IEEE disclaims liability for any personal injury, property or other damage, of any nature whatsoever, whether special, indirect, consequential, or compensatory, directly or indirectly resulting from the publication, use of, or reliance upon any IEEE Standard document.

IEEE does not warrant or represent the accuracy or content of the material contained in its standards, and expressly disclaims any express or implied warranty, including any implied warranty of merchantability or fitness for a specific purpose, or that the use of the material contained in its standards is free from patent infringement. IEEE Standards documents are supplied "**AS IS**."

The existence of an IEEE Standard does not imply that there are no other ways to produce, test, measure, purchase, market, or provide other goods and services related to the scope of the IEEE standard. Furthermore, the viewpoint expressed at the time a standard is approved and issued is subject to change brought about through developments in the state of the art and comments received from users of the standard. Every IEEE standard is subjected to review at least every ten years. When a document is more than ten years old and has not undergone a revision process, it is reasonable to conclude that its contents, although still of some value, do not wholly reflect the present state of the art. Users are cautioned to check to determine that they have the latest edition of any IEEE standard.

In publishing and making its standards available, IEEE is not suggesting or rendering professional or other services for, or on behalf of, any person or entity. Nor is IEEE undertaking to perform any duty owed by any other person or entity to another. Any person utilizing any IEEE Standards document, should rely upon his or her own independent judgment in the exercise of reasonable care in any given circumstances or, as appropriate, seek the advice of a competent professional in determining the appropriateness of a given IEEE standard.

Translations: The IEEE consensus development process involves the review of documents in English only. In the event that an IEEE standard is translated, only the English version published by IEEE should be considered the approved IEEE standard.

Official Statements: A statement, written or oral, that is not processed in accordance with the IEEE-SA Standards Board Operations Manual shall not be considered the official position of IEEE or any of its committees and shall not be considered to be, nor be relied upon as, a formal position of IEEE. At lectures, symposia, seminars, or educational courses, an individual presenting information on IEEE standards shall make it clear that his or her views should be considered the personal views of that individual rather than the formal position of IEEE.

Comments on Standards: Comments for revision of IEEE Standards documents are welcome from any interested party, regardless of membership affiliation with IEEE. However, IEEE does not provide consulting information or advice pertaining to IEEE Standards documents. Suggestions for changes in documents should be in the form of a proposed change of text, together with appropriate supporting comments. Since IEEE standards represent a consensus of concerned interests, it is important to ensure that any responses to comments and questions also receive the concurrence of a balance of interests. For this reason, IEEE and the members of its societies and Standards Coordinating Committees are not able to provide an instant response to comments or questions except in those cases where the matter has previously been addressed. Any person who would like to participate in evaluating comments or revisions to an IEEE standard is welcome to join the relevant IEEE working group at http://standards.ieee.org/develop/wg/.

Comments on standards should be submitted to the following address:

Secretary, IEEE-SA Standards Board
445 Hoes Lane
Piscataway, NJ 08854-4141
USA

Photocopies: Authorization to photocopy portions of any individual standard for internal or personal use is granted by The Institute of Electrical and Electronics Engineers, Inc., provided that the appropriate fee is paid to Copyright Clearance Center. To arrange for payment of licensing fee, please contact Copyright Clearance Center, Customer Service, 222 Rosewood Drive, Danvers, MA 01923 USA; +1 978 750 8400. Permission to photocopy portions of any individual standard for educational classroom use can also be obtained through the Copyright Clearance Center.

Notice to users

Laws and regulations

Users of IEEE Standards documents should consult all applicable laws and regulations. Compliance with the provisions of any IEEE Standards document does not imply compliance to any applicable regulatory requirements. Implementers of the standard are responsible for observing or referring to the applicable regulatory requirements. IEEE does not, by the publication of its standards, intend to urge action that is not in compliance with applicable laws, and these documents may not be construed as doing so.

Copyrights

This document is copyrighted by the IEEE. It is made available for a wide variety of both public and private uses. These include both use, by reference, in laws and regulations, and use in private self-regulation, standardization, and the promotion of engineering practices and methods. By making this document available for use and adoption by public authorities and private users, the IEEE does not waive any rights in copyright to this document.

Updating of IEEE documents

Users of IEEE Standards documents should be aware that these documents may be superseded at any time by the issuance of new editions or may be amended from time to time through the issuance of amendments, corrigenda, or errata. An official IEEE document at any point in time consists of the current edition of the document together with any amendments, corrigenda, or errata then in effect. In order to determine whether a given document is the current edition and whether it has been amended through the issuance of amendments, corrigenda, or errata, visit the IEEE-SA Website at http://standards.ieee.org/index.html or contact the IEEE at the address listed previously. For more information about the IEEE Standards Association or the IEEE standards development process, visit IEEE-SA Website at http://standards.ieee.org/index.html.

Errata

Errata, if any, for this and all other standards can be accessed at the following URL: http://standards.ieee.org/findstds/errata/index.html. Users are encouraged to check this URL for errata periodically.

Patents

Attention is called to the possibility that implementation of this standard may require use of subject matter covered by patent rights. By publication of this standard, no position is taken by the IEEE with respect to the existence or validity of any patent rights in connection therewith. If a patent holder or patent applicant has filed a statement of assurance via an Accepted Letter of Assurance, then the statement is listed on the IEEE-SA Website at http://standards.ieee.org/about/sasb/patcom/patents.html. Letters of Assurance may indicate whether the Submitter is willing or unwilling to grant licenses under patent rights without compensation or under reasonable rates, with reasonable terms and conditions that are demonstrably free of any unfair discrimination to applicants desiring to obtain such licenses.

Essential Patent Claims may exist for which a Letter of Assurance has not been received. The IEEE is not responsible for identifying Essential Patent Claims for which a license may be required, for conducting inquiries into the legal validity or scope of Patents Claims, or determining whether any licensing terms or conditions provided in connection with submission of a Letter of Assurance, if any, or in any licensing agreements are reasonable or non-discriminatory. Users of this standard are expressly advised that determination of the validity of any patent rights, and the risk of infringement of such rights, is entirely their own responsibility. Further information may be obtained from the IEEE Standards Association.

Participants

At the time this standard was submitted to the IEEE-SA Standards Board for approval, the Standard for Electric Power Systems Communications—Distributed Network Protocol (DNP3) Working Group had the following membership:

H. Lee Smith, *Co-Chair*
Ron Farquharson, *Co-Chair*
Andrew West, *Vice Chair*

Bill Ackerman
Demos Andreou
Philip Aubin
Jim Baker
James Bougie
Jake Brodsky
Carlos Bustamante
Ed Cenzon
Mason Clark
Lorene Cunningham
Mike Dood
Chris Francis
Charles Freedman
Dan Friedman
Grant Gilchrist
Randy Kimura
Marc Lacroix
Bob Landman
Parker McCauley
Steve McCoy
Bruce Muschlitz
Craig Preuss
James Recchia
Craig Rodine
Samuel Sciacca
Alan Scott
Barry Shephard
Michael S. Smith
John T. Tengdin
Eric Thibodeau
Tim Tibbals
Jay Vellore
Jack Verson
David Wo[illegible]

The following members of the individual balloting committee voted on this standard. Balloters may have voted for approval, disapproval, or abstention.

Bill Ackerman
Satish Aggarwal
Ali Al Awazi
Saleman Alibhay
Ficheux Arnaud
Jim Baker
Wallace Binder
Paul Bishop
Chris Brooks
William Byrd
Paul Cardinal
Edgar Cenzon
Jerry Corkran
Lorene Cunningham
Ray Davis
Muhammad Dhodhi
Mike Dood
Randall Dotson
Gary Engmann
Ron Farquharson
Dan Friedman
Grant Gilchris[illegible]
Mietek Glin[illegible]
Roman Graf
Stephen Grier
[illegible]roves
Timothy Hayden
Gary Heuston
Gary Hoffman
Yi Hu
Noriyuki Ikeuchi
Piotr Karocki
Yuri Khersonsky
James Kinney
Stanley Klein
Joseph L. Koepfinger
Jim Kulchisky
Marc Lacroix
Chung-Yi [illegible]
G. Luri
Ah[illegible] Mahinfallah
[illegible]anges
P[illegible]re Martin
Jeffery Masters
John McDonald
Gary McNaughton
Willam Moncrief
Jose Morales
Charles Morgan
Adi Mulawarman
R. Muprhy
Bruce Muschlitz
Pratap Mysore
Arthur Neubauer
Michael S. Newman
Charles Ngethe
Gary Nissen
Lorraine Padden
Mirko Palazzo
Donald Parker
Bansi Patel
Craig Preuss
John Randolph
Michael Roberts
Charles Rogers
Bob Saint
Steven Sano
Bartien Sayogo
Samuel Sciacca
Gil Shultz
Mark Simon
Jerry Smith
Joshua Smith
Aaron Snyder
John Spare
Wayne Stec
Gary Stoedter
Walter Struppler
Charles Sufana
William Taylor
David Tepen
Eric Thibodeau
Eric Udren
John Vergis
Jane Verner
Daniel Ward
Andrew West
Janusz Zalewski
Matthew Zeedyk

When the IEEE-SA Standards Board approved this standard on 8 June 2012, it had the following membership:

Richard H. Hulett, *Chair*
John Kulick, *Vice Chair*
Robert Grow, *Past Chair*

Satish Aggarwal	Alexander Gelman	Oleg Logvinov
Masayuki Ariyoshi	Paul Houzé	Ted Olsen
Peter Balma	Jim Hughes	Gary Robinson
William Bartley	Young Kyun Kim	Jon Walter Rosdahl
Ted Burse	Joseph L. Koepfinger*	Mike Seavey
Clint Chaplin	David J. Law	Yatin Trivedi
Wael Diab	Thomas Lee	Phil Winston
Jean-Philippe Faure	Hung Ling	Yu Yuan

*Member Emeritus

Also included are the following nonvoting IEEE-SA Standards Board liaisons:

Richard DeBlasio, *DOE Representative*
Michael Janezic, *NIST Representative*

Don Messina
IEEE Standards Program Manager, Document Development

Matthew J. Ceglia
IEEE Standards Client Services Manager, Professional Services

0 Introduction

This introduction is not part of IEEE Std 1815-2012, IEEE Standard for Electric Power Systems Communications—Distributed Network Protocol (DNP3).

0.1 DNP3 purpose and history

This Introduction discusses the creation and history of DNP3. The structure and operation of the protocol may be easier to understand when taken in the context of the problems the designers of DNP3 intended to solve.

0.1.1 Addressing an impediment to automation

Westronic Incorporated developed DNP3 between 1992 and 1994, intending it to be the first truly open, truly useful protocol standard in the utility industry. Westronic was a manufacturer of remote terminal units and a system integrator based in Calgary, Canada. It had made a reputation converting between the hundreds of proprietary utility protocols in use at the time. This was not an easy task, however, and Westronic management had become frustrated with trying to make its devices compatible with so many proprietary protocols.

A proposal was made that Westronic should develop its own protocol but then release it to the industry. The new protocol would incorporate the best features of the many protocols Westronic had encountered, plus some new ideas. Westronic would place the specification under the control of an independent users' group. Both utilities and vendors would be invited to be members, including Westronic's competitors. Westronic would not receive any money for the sale and distribution of the specification.

0.1.2 Rationale for a new protocol based on standards

Westronic was not the first to propose an open standard for the utility industry, but the designers of DNP3 did not find any of the existing efforts suitable. At the time when Westronic was considering DNP3, there were two main candidates available for an open protocol:

— The Electrical Power Research Institute (EPRI) had recently released the Utility Communications Architecture (UCA) version 1.0. However, version 1.0 listed a choice of protocol profiles only and did not define any object models or services suitable for performing Supervisory Control and Data Acquisition (SCADA) functions. At that point in the development of UCA, very few utilities or vendors had provided input to the specification, and there were some serious concerns about bandwidth usage. These drawbacks and others eventually led to the development of UCA 2.0. UCA 2.0 became an IEEE technical report in 1998 and eventually evolved into IEC 61850.[a]

— The International Electrotechnical Commission (IEC) had developed the first few documents in the IEC 60870-5 series of specifications, including the specifics of the Data Link Layer and general definitions for the Application Layer. (At that time, it was called just 870-5.) Westronic had been participating in this effort but felt that it was progressing too slowly. Furthermore, the IEC had provided many options in the specification, and Westronic was worried the standard would not be restrictive enough to promote interoperability. The IEC eventually released the 60870-5-101-companion standard in 1995 to address these issues.

In 1992, the IEC work seemed to be the more complete of the two efforts and had wider industry support at the time. Westronic therefore decided to base DNP3 on the IEC work already completed. Even now, the

[a] Information on references can be found in Clause 2.

feature sets of IEC 60870-5-101:2003 **[B3]**[b] and DNP3 are very similar because the design teams built them on the same basic research.

UCA was not forgotten. Westronic (by then called Harris Distributed Automation Products) circulated versions of the DNP3 Basic 4 Document Set including a paper called "On the Road to Utility Communications Architecture." The thesis of this paper was that by standardizing on DNP3, utilities would at least be reducing from many protocols to one. This would make it easy for utilities to later change to use UCA. However, very few design elements of UCA found their way into this standard, other than a generally layered architecture.

0.1.3 Need for scalability

The designers of DNP3 built it with several goals in mind, but the one that had the most impact on the final protocol was the industry's desire to limit the amount of bandwidth used. At that time, utilities considered a link running at 1200 bits per second to be fairly quick. (Yes, there are areas where this is still true). Local area networks (LANs) were for office computing only, and the thought of trusting one's SCADA network to a third-party telecom provider was heresy.

Power utilities had heard about layered protocols and the Open System Interconnection (OSI) model, but they were unconvinced of their value in a SCADA protocol. The Internet was beginning to boom, of course, but most utilities considered those protocols for business computing only. They were not for a SCADA network. Those who followed such things may also have heard that there was a backlash against the OSI model brewing. Protocols like Asynchronous Transfer Mode (ATM) and Frame Relay promised higher performance by eliminating layers. No utility at that time would have used these protocols in their network, but they probably heard that "layers are bad."

Therefore, the designers of DNP3 gave themselves a design goal to reduce bandwidth and use as few layers as possible.

This goal combined with the desire to be compliant with IEC 60870-5 resulted in the "Transport Function" as it now exists: a header that is not quite part of the Data Link Layer and yet not quite a complete Transport Layer. A later subclause will discuss the Transport Function in more detail.

0.1.4 Emphasis on reliability

While requesting less bandwidth, utilities refused to compromise on the requirement that a SCADA protocol be extremely reliable. Early bit-oriented protocols had acquired a bad reputation because a change of a single bit could result in a device operating the wrong switch. This led to utilities requiring in bid specifications that vendors build select-before-operate, "I tell you twice," functions into all protocols. A few bad experiences made utilities paranoid about reliability to the point of writing it into contracts.

Therefore, when designing a frame format to use, the DNP3 designers chose the most reliable format they could find. The IEC had done extensive modeling on reliability and had documented the results in IEC 60870-5-1:1990 **[B3]**. Rather than reinvent the wheel, the designers picked the most reliable of the several formats described in that specification, Frame Type 3 (FT3).

In the years that have passed, this decision has proven to be a good one. Many vendors have cursed the calculations necessary for the many cyclic redundancy checks (CRCs). Many system engineers have cursed the extra bandwidth overhead. However, DNP3's reputation for reliability started well and has only improved with the years.

[b] The numbers in brackets correspond to those of the bibliography in **Annex G**.

0.1.5 Feature selection

Because the designers of DNP3 were from a systems integration company, they tried to incorporate into DNP3 the best features of all the utility protocols they had encountered. These features included:

- Broadcasting. The ability to send a single message to multiple devices.
- Select-before-operate—or not. The ability to choose to use extra reliability when operating an output, or to choose not to use it.
- Time-stamped data. Some of the most popular utility protocols, such as Modbus, had no way to accurately time-stamp data. Vendors and utilities were forced to develop proprietary work-around solutions. Other protocols supported time stamps on binary data only. DNP3 permits time stamps on almost all data. This feature has become increasingly more useful as utilities progressively gather other types of historical data beyond the standard binary "sequence of events" log.
- Accurate time synchronization. Many earlier protocols had no way to account for transmission and software delays when synchronizing. The method used in DNP3 is an amalgamation of several different protocols' solutions.
- Quality flags. Representing a maker of data concentrators, the designers provided a mechanism to see whether data was valid, and why. Some protocols, designed by intelligent electronic device (IED) vendors whose data was always online, did not include this feature.
- Multiple data formats. The ability to report data in a variety of formats: 16-bit, 32-bit, with a flag, without a flag, floating-point, binary-coded decimal (BCD), packed, unpacked, and so on.
- Scan groups. The ability to define and ask for a large set of otherwise unrelated data using a single request.
- Layer separation. Separating the function of "getting the data there" from the actual SCADA functions.
- Report-by-exception. More than any other feature, the ability to reliably report only the changes in data has helped make DNP3 successful.
- Internal indications. As several protocol efforts that are more recent than DNP3 have discovered, it is extremely useful to have a global set of flags returned in each response. These flags indicate the health of the device and the results of the last request.

Most of these features had been seen elsewhere, but this was the first time an open utility protocol had attempted to do them all.

0.1.6 Rationale for DNP3 subset definitions

Unfortunately, the "best practices" approach to developing DNP3 was not perfect, causing a number of features to be added that were not really in widespread use. A number of them existed only in Westronic equipment. At various times, vendors have questioned the need for:

- So many different types of counters, particularly delta counters
- So many different types of binary output operations, especially control queuing
- So many different ways to format data (i.e., many qualifier codes)
- Pattern masks
- Binary-coded decimal analogs
- Storage objects

— The ability to either write or operate an output

— So many layers of confirmation and segmentation

0.1.7 Features to support distributed capabilities

Another trend in the early 1990s was the move to put larger processors and more memory in SCADA devices. Marketing and sales people were talking about "the intelligent network." By this, they meant pushing many of the functions previously performed only by master stations into remote devices. These devices would be more independent and make more decisions on their own. Those who join the utility industry these days are sometimes confused by the term "IED" meaning intelligent electronic device. They say, "Aren't all computing devices intelligent?" Yes, but it wasn't always this way.

In terms of DNP3 design, the idea of "the intelligent network" translated to the following features:

— Spontaneous reporting. A device could transmit whenever it wanted, not just when polled by the master. On multi-drop links, this led to the need for a collision avoidance mechanism.

— Meta-data. The DNP3 designers called a spontaneous message an "Unsolicited Response," which shows the mindset in those days. Most devices only sent data in response to a poll request. Therefore, the master always knew what data was coming. For a device to send an unsolicited response, it had to include not only the SCADA data itself but also information describing the data so the master knew what it was. The term these days for such information is meta-data. It appears in such modern technologies as Extensible Markup Language (XML). At that time, though, it was a very new concept for the utility industry.

— Wild-carding. Because the remote device was more intelligent, the designers gave it more choice in the amount and format of the data it reported. A master could ask very simple questions, like "Give me all your data" or "Give me your analog changes" and get very complex answers. Again, because the answer did not exactly match the question, meta-data was required in the response.

— Self-description. The idea that a device could tell the master what data it had available, and how to present it, was already around thanks to UCA 1.0. The DNP3 designers tried to incorporate some of this ability into DNP3. The Device Profile Object and the use of floating-point with the units transmitted were considered very advanced. Perhaps they were too advanced because they appeared in very few implementations.

— Vendor-specific expansion. This standard includes the Private Registration Object, which permits vendors to add proprietary extensions to the basic standard. The Private Registration Object Descriptor permits a standard implementation to parse these extensions even though they are proprietary. These objects, too, have not been very popular, but a few vendors have used them to good effect.

— File transfer. The designers gave DNP3 file transfer capabilities so that an intelligent device could download new configuration or software, or upload oscillography files. At the time DNP3 was developed, few devices had flash memory, and only specialized fault recorder devices performed oscillography. Now both are widespread.

— Program control. The ability to start and stop individual programs and processes on a remote device was common in the factory automation industry. DNP3 provides a rudimentary mechanism to do this.

The dream of the "intelligent network" has had mixed results. Some of these features, like spontaneous reporting, meta-data, prioritization and wild-carding, have worked very well. They are probably some of the main reasons for DNP3's popularity. Other features, like self-description, file transfer, floating-point, program control, and collision avoidance, were not completely thought out. The DNP Technical Committee was forced to revise these and issue technical bulletins clarifying their use. Some features have died a death of obscurity.

However, history should not be a harsh judge. Many people take such features for granted these days, but it is important to remember that DNP3 was there first.

0.1.8 Additional communications features

Because of the intense pressure to reduce bandwidth, and because the DNP3 designers had more expertise in SCADA than in general data communications, a number of common communications features were "left out" of the DNP3 definition. Many designers have subsequently mourned the absence of these features. Some of them the DNP Technical Committee has attempted to "add on" afterward. Others the Committee could only achieve now at the cost of obsoleting all existing implementations.

The following list of missing data communications features illustrates how well the DNP3 architecture works despite the limitations imposed at its birth:

- Network layer. At one point, the designers actually wrote a specification for a DNP3 network layer, but Westronic management did not approve it. In retrospect, this is just as well, because the Internet Protocol (IP) network layer now used is far more popular.
- Application Layer addresses. The ability to select a particular logical device within a physical one would have been useful. Most devices that support this feature have found a way around it through local software mechanisms that use the Data Link address and/or physical port number as a key.
- Application and Transport Layer sequence number initialization. This has caused much grief over the years and has been addressed as well as possible without causing obsolescence. Data communications experts should note, therefore, that DNP3 is not quite connection-oriented and not quite connectionless, but somewhere in between.
- Long sequence numbers. DNP3 sequence numbers are very short, which is good for bandwidth but not for detecting duplicates. This is the reason Transmission Control Protocol (TCP) is required when using DNP3 over wide area networks (WANs), which turns out to be a very robust solution.
- Sequence number in Data Link Confirms. Without a sequence number, it is impossible to determine which Data Link frame a Confirm frame is answering. On a serial point-to-point link, this is not a problem, but on a WAN, Confirm frames could arrive out of order or be lost. Using TCP in WANs addresses the issue on IP networks, but in theory, it could still cause problems in serial radio networks. In practice, it generally works anyway. This problem was inherited from IEC 60870-5 and cannot be changed without obsolescence.
- Sliding window. One constant of DNP3 has been that only one transaction can be outstanding at a time. In theory, a device could send several response fragments very quickly for a particular request, but over the years the DNP Technical Committee has decided that interoperability is best served by enforcing a confirmation between each fragment.
- Access security. The designers of DNP3 purposely avoided dealing with this issue because of its complexity. However, the flexible structure of DNP3 has permitted access security to be integrated as an optional feature. DNP3 Secure Authentication enables a DNP3 master or outstation to unambiguously determine that it is communicating with the correct device and/or authorized user.
- Version control. Most protocols tend to have an octet reserved to show the version of the protocol in use. This was not included in the original DNP3 definition due to bandwidth reasons; however, the introduction of Object Group 0 addresses this problem.
- Overall length field. Segmentation and fragmentation would have been a lot easier and more robust, and the LAN implementation would have been easier if each fragment had a length field at the beginning. It was not included for bandwidth reasons. Again, various software solutions make it work anyway, so perhaps it was the right decision.

0.1.9 Compatibility with IEC protocols

As discussed earlier, there were two reasons why the DNP3 designers wanted it to be compliant with the IEC 60870-5 specifications:

- They wanted to take advantage of the excellent technical work done on reliability in the IEC 60870-5 Data Link Layer specifications.
- They wanted to increase the acceptance of the protocol by showing it was based on standards work that was already well known.

They were so successful in both efforts that even now some people are confused about whether DNP3 and IEC 60870-5 are interoperable.

The answer is that they are not interoperable, although the DNP3 Data Link Layer could be considered compliant to IEC 60870-5-1:1990 **[B3]** and IEC 60870-5-2:1992 **[B5]**. DNP3 was based on the drafts available at the time of IEC 60870-5 Parts 1 through 5. These Parts of the specification described the Data Link Layer in great detail and the Application Layer in general. There were several options specified for the Data Link Layer.

The DNP3 designers chose those options of IEC 60870-5-1:1990 **[B3]** and IEC 60870-5-2:1992 **[B5]** they thought were most appropriate. Unfortunately, when the IEC 60870-5-101:2003 **[B4]** companion standard was released with the details of the Application Layer, it specified *different* Data Link Layer options than those the DNP3 designers had chosen.

Therefore, DNP3 is considered compliant with IEC 60870-5-1:1990 **[B3]** and 60870-5-2:1992 **[B5]** but not with IEC 60870-5-101:2003 **[B4]**.

Table 0-1 shows the differences in the Data Link Layers of the two protocols.

Table 0-1—Comparison of IEC 60870-5 and DNP3 Data Link Layers

Feature	Options permitted in IEC 60870-5-1:1990 [B3] and IEC 60870-5-2:1992 [B5]	Chosen by DNP3	Chosen by IEC 60870-5-101:2003 [B4]
Addressing	Single address, length system-dependent	Two-octet Source address and two-octet Destination address. Considered a single four-octet "structured" address for compliance purposes.	Single address, choice of either zero, one, or two octets in length
Frame Format	Choice of FT1.1, FT1.2, FT2, and FT3	FT3, transmitted asynchronously.	FT1.2
Reliability Mechanism	Varies per frame type	Multiple 16-bit CRCs over each 16 octets of a 255-octet frame. Start and Stop bits, but no parity.	Parity bits and one-octet checksum (not CRC) calculated over 255 octets
Hamming Distance	Varies per frame type	6 for the original FT3. Some debate about the value as currently used. See further discussion in this subclause.	4
Acknowledgments	Either fixed-length or single-octet	Fixed 10-octet only.	Either fixed-length or single-octet
Procedures	Balanced (no master) or Unbalanced (master polls)	Balanced only.	Either Balanced or Unbalanced
Method for Multi-Drop Links	Unbalanced mode	Collision avoidance.	Unbalanced mode

0.1.9.1 Hamming Distance

Some critics of DNP3 have disputed DNP3's right to claim a Hamming Distance of six. The "Hamming Distance" of a protocol is the number of bit errors required in a frame before a receiver could incorrectly identify a corrupted incoming frame as a valid frame. Critics argue that the original calculation was made assuming the FT3 frame was transmitted synchronously, while DNP3 uses the FT3 frame format asynchronously.

The main concern in this debate is inter-character gaps. If a gap is permitted between the octets of a 16-octet block, noise could be introduced that might be misinterpreted as valid data. In fact, it has been shown that there exists at least one case where an inter-character gap of exactly 1-bit time at the end of a message can be misinterpreted, thereby resulting in a Hamming distance of 2. Critics claim that this standard has never *required* that all octets of a block be transmitted together, and this reduces the theoretical reliability of the protocol to below that of the FT1.2 frame.

However, years of use in hundreds of systems have proven DNP3's reliability to be more than sufficient for utility purposes. This may be due to the fact that most DNP3 devices transmit frames without inter-character gaps, and receiving devices tend to start a timer or other mechanism that discards incoming frames when inter-character gaps appear.

The inter-character concern with the DNP3 frame is similar to a problem that occurs in some IEC 608705-101 systems. The FT1.2 frame's reliability relies on the use of parity bits in each octet. However, many utilities mistakenly use the protocol with modems that do not add, or actually remove, such parity bits. The IEC is preparing an IEC 60870-5 standard that clarifies parity bits *shall* be used.

0.1.9.2 Addressing of binary outputs

The other main issue concerning DNP3 compliance to IEC 60870-5 was the structure of the address field. The IEC definition of the address field states that it is a single address, always addressing one end of the link. This is the way IEC 60870-5-101:2003 **[B4]** uses the address field.

By including both a source and a destination in every message, the DNP3 designers permitted the use of multiple masters on the same link, and peer-to-peer communications. This proved to be a powerful argument in the acceptance of DNP3. Furthermore, since IEC 60870-5-2:1992 **[B5]** did not specify a particular length of address, a four-octet address that just happened to be "structured" with two sub-addresses could still be considered compliant.

0.1.9.3 Reality today

Although it was the topic of lively debate when DNP3 was first released, the question of whether DNP3 complies with IEC 60870-5 is essentially a moot point today. DNP3 may be considered compliant to Parts 1 and 2. One could even argue that DNP3 complies with the spirit, if not the letter, of Part 5, the general Application Layer definition. However, the format of the IEC 60870-5-101 **[B2]** Application Layer is very different from that of DNP3. It is clear the two protocols could never interoperate.

It is better to consider the two protocol suites as cousins with a common family tree and leave it at that.

0.1.10 Transport Function

The naming of the Transport Function always confuses newcomers to DNP3. Is it a true Transport Layer, is it a part of the Data Link Layer, or is it something truly different?

The answer is that it really is something different, although it most closely resembles an additional field in the Data Link Layer. It does not have its own addressing or acknowledgments, as a separate layer would. There was no network layer in the original protocol definition, so the transport header was terminated at the end of each physical link, just like the data link header. It does not have the long sequence numbers and other features that would really enforce transmitting frames in sequence. Therefore, it does not seem to be a Transport Layer.

However, if it were a field of the data link header, it would be included in every data link frame, and it is not. Only those frames containing Application Layer data contain a transport header.

The reasons this strange "half-layer" exists are both political and technical. The designers of DNP3 decided they wanted the Application Layer data broken into small segments suitable for passing over noisy links. This capability would at a minimum require a new Data Link Layer field. However, they did not want to add a new field for two reasons:

a) It would eliminate DNP3's chances to be considered compliant to IEC 60870-5. As discussed earlier, this was considered critical to DNP3's acceptance by the industry.

b) Changing the structure of the FT3 frame could possibly compromise the calculated reliability of the frame.

Therefore, the transport header was placed in front of the Application Layer header, in the user data field of the Data Link Layer frame.

However, the next question was, "What to call it?" As noted in this subclause, it had some of the characteristics of a layer but not all of them. Furthermore, the designers knew there would be resistance to any additional layers in the protocol. It was bad enough that they were dedicating *a whole octet* to the segmentation and reassembly functions.

Therefore, the name "Transport Function" was chosen, thus causing years of questions on hotlines, e-mails, and training presentations.

Whatever it is, it makes DNP3 distinct. Along with Application Layer fragmentation, it permits a small, low-powered device to report a nearly unlimited amount of data reliably over a noisy link.

0.1.11 DNP Users Group

One feature of DNP3 that newcomers do not always appreciate is the organization that stands behind it. Over the years, the DNP Users Group has contributed at least as much to the protocol's success as the technical features of the protocol itself.

In roughly chronological order, here are the organizational features that helped DNP3 become popular:

- Membership that included both utilities and vendors
- Low membership fees
- Low cost of the specifications
- A structure consisting of steering, technical, and marketing committees
- The DNP3 Subset Definitions document
- Suggested wordings for utilities to specify DNP3
- Agreement that any non-backward compatible change shall be approved by the General Membership
- The DNP3 hotline, later to become an Internet chat session
- Booths at major trade shows
- Publishing membership lists so utilities could see the lists of vendors
- The DNP3 bulletin board, later to become a Web site
- The DNP3 e-mail mailing list
- The DNP Technical Committee mailing list, which can include non-committee members
- Publishing the technical committee minutes so the process remains open
- Technical bulletins clarifying areas of dispute when interoperability issues arose
- Agreement, first informal and then formal, that the president should always be from a utility
- The DNP3 IED Application Level Conformance Test Documents[c]
- The DNP3 WAN/LAN Specification

0.1.12 Summary

The following design goals, whether formally stated or not at the time, had a major impact on the structure of DNP3:

- Include the best features of the utility protocols in use at the time.
- Push the intelligence in the network toward the remote device.
- Try to comply as much as possible with existing standards efforts, especially IEC 60870-5.

[c] Refer to http//www.dnp.org.

— Use as little bandwidth as possible.

— Make it more reliable than anything that came before.

It is easy to see that these goals are necessarily contradictory. The resulting protocol was not perfect and has been "patched up" over the years. However, it remains popular, open, reliable, and mostly backward-compatible.

0.1.13 Background: Origins of the name "DNP3"

Few people seem to know what to call this protocol. Everyone knows it is DNP3, but is it "DNP 3," "DNP 3.0," "DNP V3.0," or any combination of the above? Also, there are several different subset levels of implementation, and a few non-backward-compatible changes have been made over the years.

As of Technical Bulletin TB2000-003: "Change Management," the official name of the protocol is DNP3-xxxx, where xxxx is the year of release of the Test Procedures to which a device complies. The subset level is specified afterward, as in "DNP3-2000 Level 2."

This naming convention represents an evolution of the name over the years:

— The original Basic 4 documentation referred to the protocol as "DNP V3.00." No one has ever liked saying the "V" part, so that name has never caught on.

— For those who were wondering: DNP V1.00 and DNP V2.00 are proprietary Westronic protocols that were rarely used even at the time DNP3 was released.

— The user's group is called just the "DNP Users Group." That saves it from having to worry about version numbers in marketing information.

— The Subset Definitions defined the format DNP3-Lx, where x was the subset level. That never caught on either, since utilities preferred to spell out the words "Level x" in their bid specs.

— When the Test Procedures were first published in 2000, there had to be some mechanism to distinguish between an implementation that was compliant to the procedures and earlier implementations that were not. The DNP Technical Committee therefore decided to use the year in the specification, similar to the format used by the International Organization for Standardization (ISO), IEEE, and IEC.

— The intent is that there shall never be a DNP 4.0, or even a DNP 3.1. To help illustrate this commitment to backward compatibility, the DNP Users Group changed the name from "DNP V3.00" to "DNP3." The name therefore remains recognizable while eliminating the "software version" impression that the decimal point gave.

Some people rightly complain that it is redundant to say "DNP3 protocol" since the "P" in DNP3 stands for "Protocol" already. However, this truism does not seem to discourage people from using the phrase, and it is likely to be heard for years to come.

Now if one could just figure out how a protocol that originally had no network layer ended up with the name "Distributed *Network* Protocol." One long-standing DNP3 user points out that the networks in question are SCADA networks, which "bear scant resemblance to other things that people usually call networks." Perhaps this is the case.

Ah well, a protocol by any other name is just as interoperable and reliable.

0.2 DNP3 overview

0.2.1 Basic messages and data flow

The document is a brief, but incomplete, overview of DNP3 messages and data flow. Its purpose is to prepare the reader for what follows in the Application Layer, Transport Function, and Data Link Layer specifications for DNP3.

NOTE—Unless otherwise noted, DNP3 makes no representation about how data is processed once it is received.[d]

This initial discussion of DNP3 uses the master–outstation model illustrated in **Figure 0-1**. This subclause purposely omits many details to keep the description straightforward.

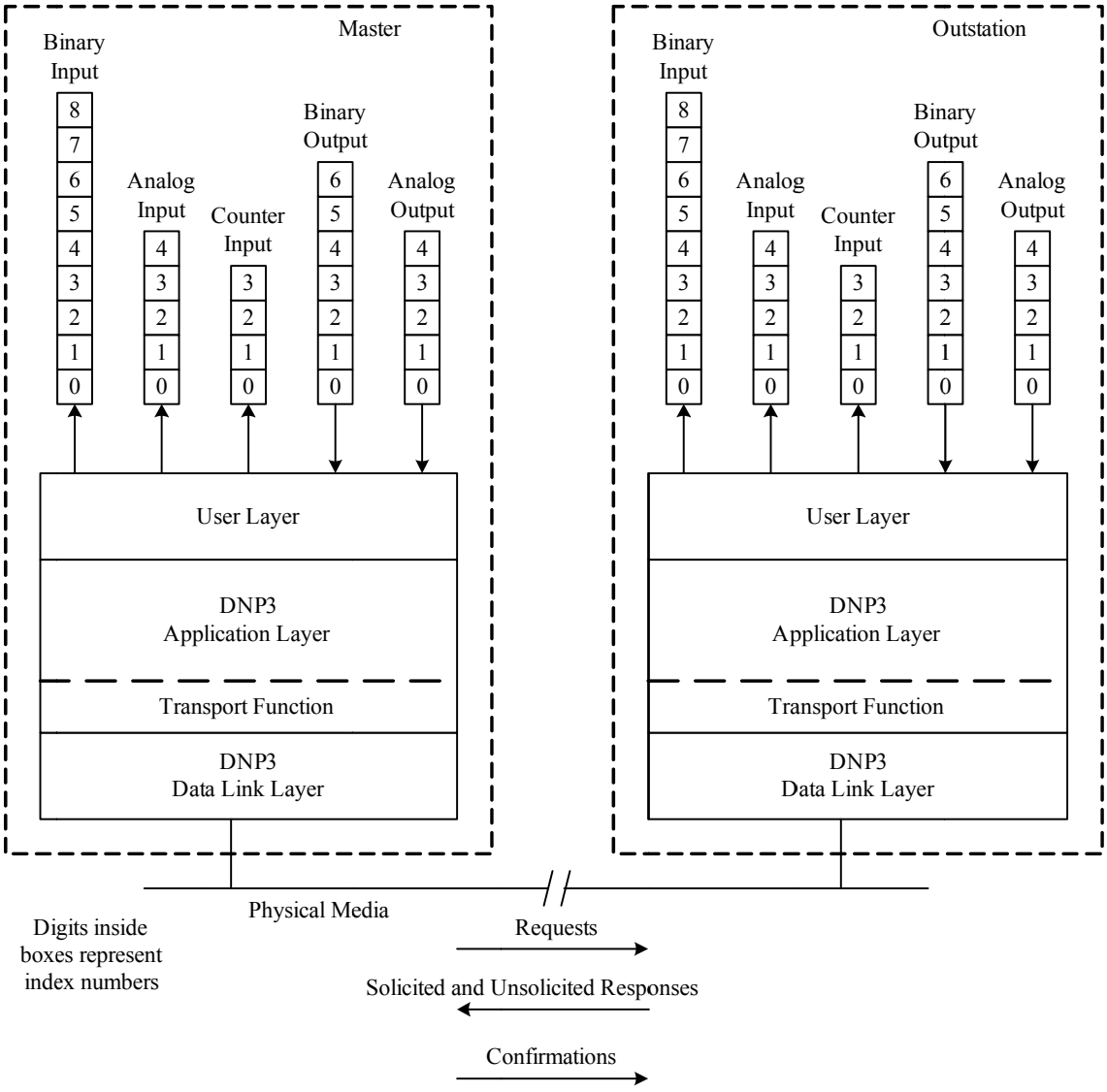

Figure 0-1—DNP3 master–outstation model

[d] Notes in text, tables, and figures of a standard are given for information only and do not contain requirements needed to implement this standard.

The User Layer in the master on the left side of **Figure 0-1** initiates a data transfer by causing its Application Layer to send a request to the outstation. The request contains a function code and zero or more DNP3 objects that specify what data is wanted. The Application Layer passes the request on to the Transport Function for partitioning into transmission-sized units and then on to the Data Link Layer. The Data Link Layer adds addressing and error detection information and transmits the packet to the outstation over the physical media.

At the outstation on the right side, the Data Link Layer receives the octets from the physical layer and checks for errors that were introduced while the packet was in transit. If no errors are detected, the addressing and error detection information added by the transmitting Data Link Layer is stripped from the message, and the remaining octets are passed on to the Application Layer. If necessary, the Transport Function reassembles multiple packets into a complete request. The Application Layer then interprets the function code and DNP3 objects in the message and indicates to the User Layer what data is desired.

The User Layer in the outstation initiates a response based on what data the master requested. It fetches data, classifies it, and presents that data to the Application Layer. The Application Layer creates a message with data formatted into DNP3 objects, passes it through the Transport Function, and then on to the Data Link Layer for transmission to the master using methods similar to those employed by the master to send its request.

Upon receipt of the response at the master, the layers perform address and error checking and reassembly into a complete message for the Application Layer. This layer parses the DNP3 objects in the response and presents the information to the User Layer. The User Layer can then store or operate on that data in a way that is suitable for the end user.

The master always initiates control commands. These actuate device outputs or variables internal to the outstation. The DNP3 User-to-Application Layer interface and transmission procedures are similar to those discussed for data acquisition.

A transaction consists of a single request followed by a single response. A master sends a request and waits for the response, or a timeout, before issuing another request. Multiple transactions may simultaneously occur within a system. For example, consider the case where two masters each make requests to the same outstation.

In some systems the master does not always directly initiate data transfer. DNP3 has provision for the outstation to automatically send data when it detects a condition worthy of transmitting without a specific master request. “Unsolicited Responses” is the terminology applied to this type of operation because the request is implied.

0.2.2 Layering

0.2.2.1 General

ISO defines a communication architecture that separates functions into seven layers called the OSI reference model. DNP3 protocol is based on a simplified model termed the Enhanced Performance Architecture (EPA) that consists of only three layers: Application, Data Link, and Physical. **Figure 0-1** shows how DNP3 fits the EPA structure and communication model.

In theory, each layer in a layer stack performs a set of functions required to communicate with the same layer in another device, relying on the next lower layer for more primitive functions. At the sending device, each layer below the Application Layer receives data from the layer above for transmission. The layer adds more information that enables the equivalent layer in the receiver to properly process the message. At the receiving device, layers examine their layer specific information added by the corresponding layer at the transmission site and process the message appropriately. The layer control information is stripped, and the message is passed to the next higher layer.

The Transport Function within the Application Layer performs a layer-like function of partitioning large messages into smaller messages that the Data Link Layer is capable of handling. The Transport Function is sometimes referred to as a "pseudo layer." In DNP3 the Application Layer, Transport Function, and the Data Link Layer in the transmitter add information to the message for enabling the same layer or pseudo layer in the receiver to process the message.

0.2.2.2 Fragments, segments, and frames

Figure 0-2 illustrates the partitioning of large messages at the Application Layer into smaller units and the addition of header information at each layer.

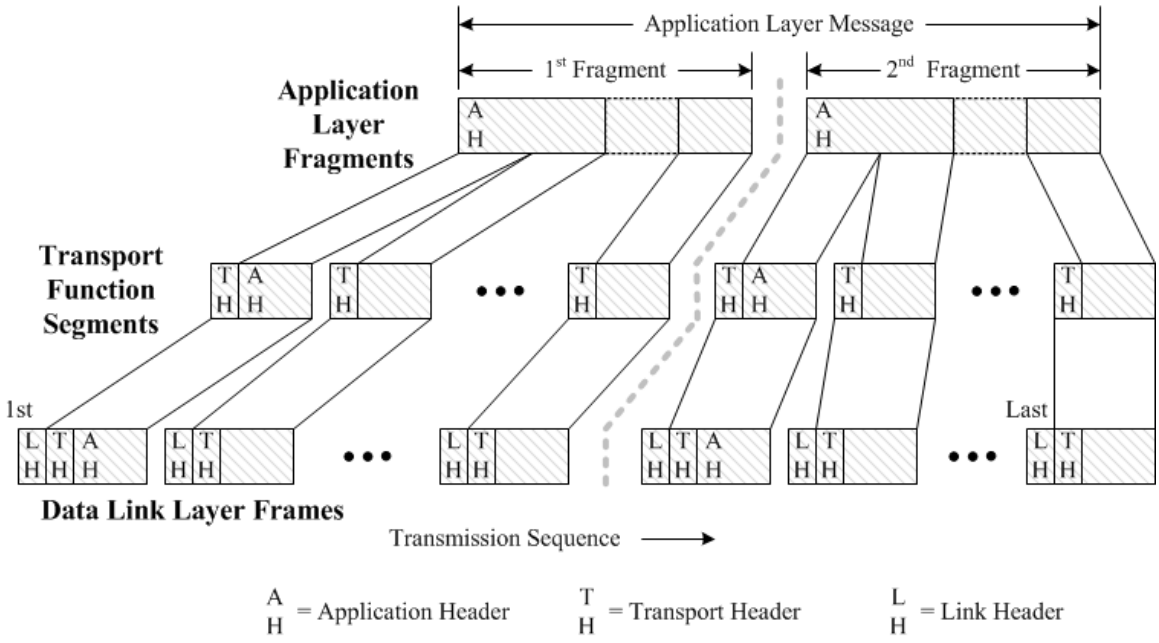

Figure 0-2—Fragmented Application Layer message

Figure 0-2 shows a fragmented Application Layer message, segmentation of each fragment by the Transport Function, and how segments fit into Data Link Layer frames. This diagram does not show timing and confirmation details but serves to demonstrate how the higher level parts nest inside the lower layer structures. It also shows the relative positions of the Application Layer headers, the Transport Function headers, and the Data Link Layer headers.

Table 0-2 provides a summary of the terminology and some brief information associated with each layer or function.

Table 0-2—DNP3 layer summary

Layer or function	Unit name	Information
Application Layer	Application Fragment	Permits the setting of an upper limit on the memory requirements for message reception. Requests shall fit into a single fragment. Responses may require more than one fragment.
Transport Function	Transport Segment	Segmentation breaks a Data Link fragment into pieces that fit into a Data Link frame. Each segment contains a Transport header, but only the first segment of any fragment contains an Application header. Each segment may have a maximum of 250 octets including the Transport header.
Data Link Layer	Data Link Frame	A Frame may have as many as 292 octets including its header and CRC octets. Frames are designed for superior error detection.

0.2.3 Message sequences

Figure 0-3 illustrates a hypothetical sequence and the time relationship of fragments and frames as they move between layers, and between the master and the outstation in a **polled** environment. Readers just beginning to learn this standard are cautioned to only view the diagram as a means of gaining a general overview.

Figure 0-4 illustrates a hypothetical sequence and the time relationship of fragments and frames as they move between layers, and between the master and the outstation in an **unsolicited response** environment. Readers just beginning to learn this standard are cautioned to only view the diagram as a means of gaining a general overview. Later, after studying the details, refer back to this figure when it may be more meaningful.

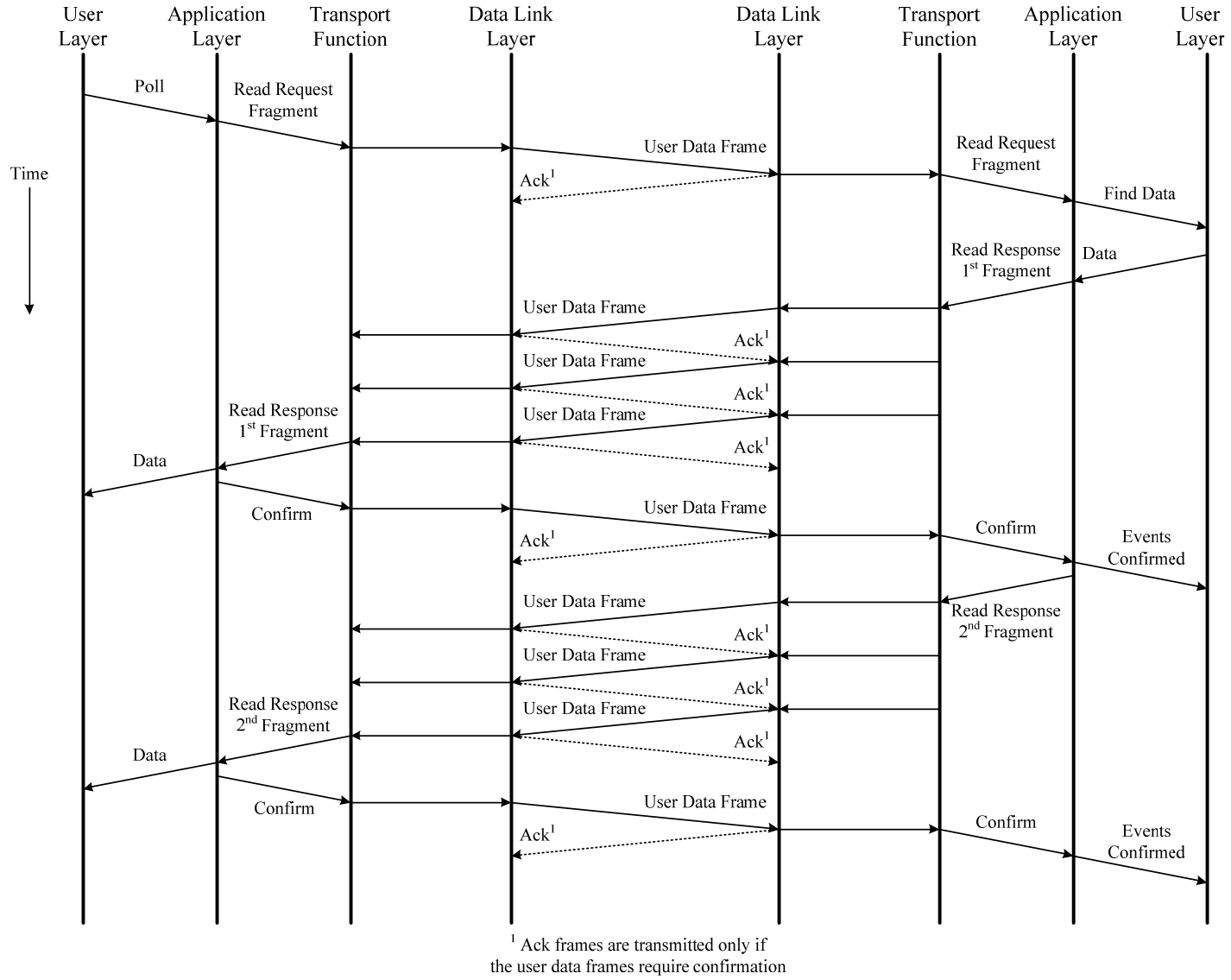

Figure 0-3—Polled sequence with Data Link Layer confirmation

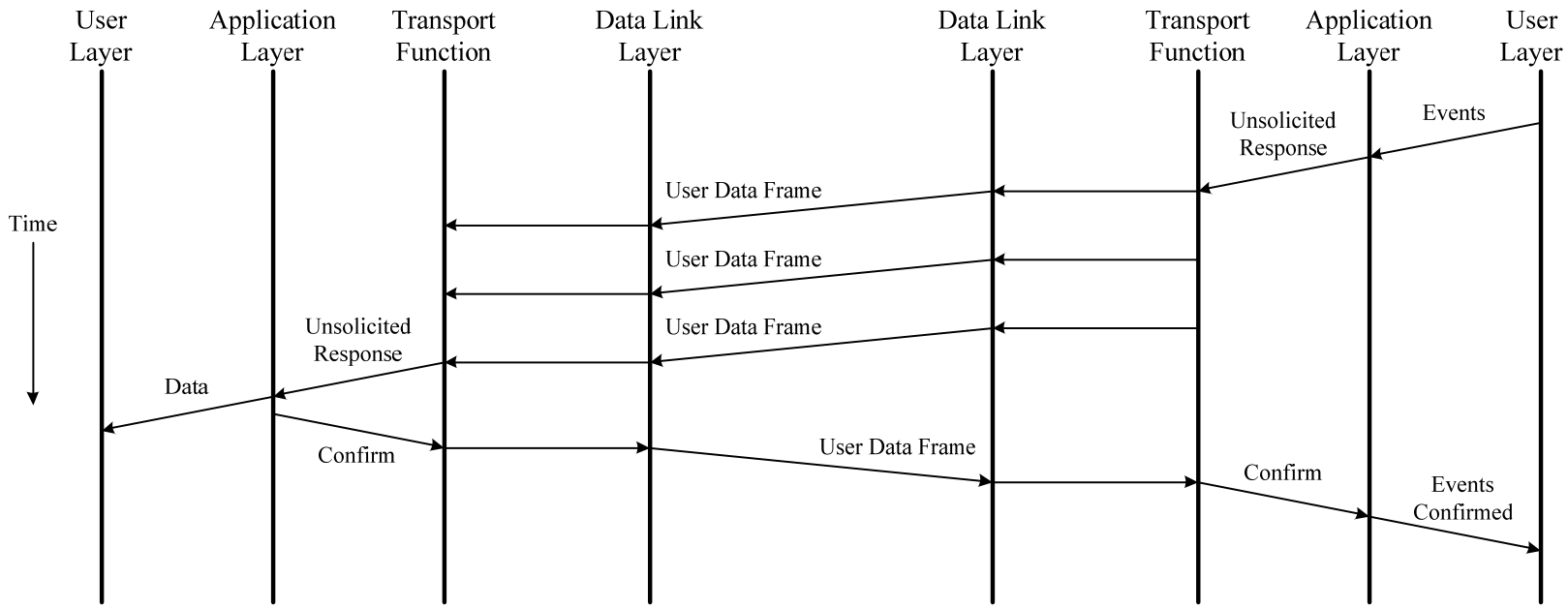

Figure 0-4—Unsolicited response sequence

0.2.4 Data loss and efficiency

One of the fundamental goals of DNP3 is to prevent loss of data transferred from an outstation to the master. Of special concern is the transfer of all binary input states, in time sequence, and without missing any transitions.

To increase the efficiency, DNP3 provides for report-by-exception whereby changes are transmitted soon after they occur, and an occasional integrity poll is issued to synchronize the master and outstation databases. When an outstation transmits changes, it shall request Application Layer confirmation. Only after the master confirms receipt of the changes can the outstation assume the changes arrived at the master.

Outstation devices that are able to report all of their current data in a single frame are not required to support report-by-exception.

0.2.5 Unsolicited responses

Unsolicited responses are messages spontaneously sent from an outstation without a specific request from a master when "something of significance" occurs. The DNP3 protocol includes support for unsolicited responses.

This method of operating has advantages in some applications. In a system with a large number of outstations and a single master, changes at an outstation can reach the master often much faster because there is no delay while waiting for a master poll. The communication costs to achieve faster polling in some installations can be prohibitive, and the quickest notification of changes can occur if most of the messages contain only changes and confirmations. Unsolicited operation may reduce costs where the owners choose a "cost-per-byte" type of service.

On the other hand, equipment that implements unsolicited responses is more complex because the issues of media access and collision avoidance should be dealt with. The DNP3 specification requires that master station software is capable of accepting messages from any of its outstations at any time. Another disadvantage is that system performance may become unpredictable during periods of heavy communication.

Employing unsolicited reporting requires an engineering judgment based on numerous factors for each individual system. There are no guarantees that unsolicited reporting is universally applicable for all systems.

A description and rules for unsolicited responses are provided in Clause **4** through Clause **6** of this standard.

0.2.6 IP networking

DNP3 was originally designed for serial octet streaming, point-to-point communications over voice grade audio links, or hard-wired, multi-drop wire and fiber cabling. As IP networking evolved, users recognized a need for devices to exchange DNP3 messages over these high-speed, packet-based, digital networks. DNP3 now includes this capability.

The approach taken was to place DNP3 as the user of an IP stack and to retain all the same Application Layer, Transport Function, and Data Link Layer structures, objects, and formats as in the original DNP3. Thus, DNP3 Data Link Layer frames are passed transparently across an IP network as TCP or User Datagram Protocol (UDP) packets.

A description and special rules are described in Clause **13** and **Annex C** of this standard.

0.3 Organization of DNP3 Specification

The complete DNP3 Specification is organized into separate clauses wherein details of the DNP3 protocol are documented as follows[e]:

Clause **0** through Clause **3** and **Annex A**:	DNP3 Introduction (formerly known as Volume 1)
Clause **4** through Clause **7**:	Application Layer (formerly known as Volume 2)
Clause **8**:	Transport Function (formerly known as Volume 3)
Clause **9**:	Data Link Layer (formerly known as Volume 4)
Clause **10**:	Layer Independent Topics (formerly known as Volume 5)
Clause **11**, Clause **12**, and **Annex A**:	Data Object Library (formerly known as Volume 6)
Clause **13** and **Annex C**:	IP Networking (formerly known as Volume 7)
Clause **14**:	Interoperability (formerly known as Volume 8)

0.4 Conventions used in this standard

0.4.1 Notes

NOTEs within the text are informative. They are set apart as a separate paragraph beginning with "NOTE—".

0.4.2 Examples

Examples are preceded with a box describing what is illustrated below. The number **EX 0-1** represents the example number.

EX 0-1	This example shows a request for all of the static binary inputs. Assume there are 18 binary inputs.

0.4.3 Single master, single outstation perspective

The DNP3 protocol is suitable for systems with one or more master stations, one or more outstations, and peer-to-peer arrangements. In general, this standard was written from the perspective of a single master and a single outstation to make the documents easier to understand without the additional complexities involved.

A separate subclause (**13.2.3.5**) is devoted to discussion of multi-master systems and their special considerations and requirements. Statements appear elsewhere only when it is necessary to emphasize specific characteristics or behavior for systems with multiple master or outstation devices.

[e] The DNP3 Specification was identified by volumes prior to IEEE Std 1815-2010 standardization.

Contents

Figures

Tables

Examples

IEEE Standard for Electric Power Systems Communications—Distributed Network Protocol (DNP3)

1 Overview

1.1 Scope

This document specifies the DNP3 protocol structure, functions, and interoperable application options (subset levels). The specified subset level defines the functionality implemented in each device. The simplest level is intended for basic devices. More advanced levels support increasing functionality. The protocol is suitable for operation on a variety of communication media consistent with the makeup of most electric power communication systems.

1.2 Purpose

The purpose of this standard is to document and make available the specifications for the DNP3 protocol. While a primary focus of this protocol is the Electric Utility Industry, other industries that deliver Energy and Water are also using DNP3. The intent of this DNP3 standard is to meet the goal established by the National Institute of Standards and Technology (NIST) for a Smart Grid protocol:

- Provides a protocol standard from a recognized standard institution
- Provides interoperability with hundreds of operational systems and thousands of devices
- Provides cyber security based on IEC/TS 62351-5
- Provides Device data profiles in a format that can be mapped to IEC 61850 Object Models

Vendors may use this standard to implement and test the protocol in their products and be assured of interoperability. Users may use the document to specify the features they wish to apply. System Integrators may use this standard to assist in system integration and testing.

1.3 Octet order

Unless specified elsewhere, the least significant octet in multi-octet data values is transmitted first.