INTERNATIONAL STANDARD

ISO 14827-2

First edition
2005-11-01

Transport information and control systems — Data interfaces between centres for transport information and control systems —

Part 2: DATEX-ASN

Systèmes de commande et d'information des transports — Interfaces de données entre les centres pour systèmes de commande et d'information des transports —

Partie 2: DATEX-ASN

Reference number
ISO 14827-2:2005(E)

PDF disclaimer

This PDF file may contain embedded typefaces. In accordance with Adobe's licensing policy, this file may be printed or viewed but shall not be edited unless the typefaces which are embedded are licensed to and installed on the computer performing the editing. In downloading this file, parties accept therein the responsibility of not infringing Adobe's licensing policy. The ISO Central Secretariat accepts no liability in this area.

Adobe is a trademark of Adobe Systems Incorporated.

Details of the software products used to create this PDF file can be found in the General Info relative to the file; the PDF-creation parameters were optimized for printing. Every care has been taken to ensure that the file is suitable for use by ISO member bodies. In the unlikely event that a problem relating to it is found, please inform the Central Secretariat at the address given below.

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Published in Switzerland

Contents

Page

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of technical committees is to prepare International Standards. Draft International Standards adopted by the technical committees are circulated to the member bodies for voting. Publication as an International Standard requires approval by at least 75 % of the member bodies casting a vote.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO shall not be held responsible for identifying any or all such patent rights.

ISO 14827-2 was prepared by Technical Committee ISO/TC 204, *Intelligent transport systems*, Working Group 9, with the collaboration of:

— European Road Transport Telematics Implementation Coordination Organization (ERTICO);

— Comité Européen de Normalisation (CEN);

— American Association of State Highway and Transportation Officials (AASHTO);

— Institute of Transportation Engineers (ITE);

— National Electrical Manufacturers Association (NEMA).

ISO 14827 consists of the following parts, under the general title *Transport information and control systems — Data interfaces between centres for transport information and control systems*:

— *Part 1: Message definition requirements*

— *Part 2: DATEX-ASN*

Introduction

In the 1980s and 1990s, transport networks became increasinigly congested and computer technologies were deployed to more efficiently manage the limited transport network. As these systems were deployed, it became more important to integrate nearby systems to properly provide the required services.

One of the first efforts to standardize the interface between transport control centres was a European Union effort led by the DATEX Task Force. In May 1993, this group was established as a horizontal activity to coordinate the diverging developments which were ongoing within the framework of the Advanced Transport Telematics (ATT) Programme. Within the ATT Programme, three different data exchange systems were developed: INTERCHANGE, EURO-TRIANGLE and STRADA. The group produced a set of basic tools to meet existing needs, including a common data dictionary, a common set of EDIFACT messages and a common geographical location referencing system.

The initial solution provided a common interface which satisfied the basic requirements of existing systems, and was named the Data Exchange Network (DATEX-Net) Specifications for Interoperability. During the initial efforts to deploy this International Standard, there was a growing sense that the message structure should be better organized and should be defined using Abstract Syntax Notation One (ASN.1) rather than EDIFACT.

ASN.1 presents a standard notation for the definition of data types and values. A data type is a class of information (e.g. numeric, textual, still image or video information). A data value is an instance of such a class. ASN.1 defines several basic types and their corresponding values, and rules for combining them into more complex types and values. These types and values can then be encoded into a byte stream according to any of several standardised encoding rules.

Efforts to standardize communications between transport control centres were also underway in other parts of the world. In 1997, all of these efforts began to merge, with the United States developing the initial draft of the ASN.1 structures for the Data Exchange in Abstract Syntax Notation (DATEX-ASN). These structures, called data packets, were then placed within a procedural context and submitted to the ISO standardization process.

A portion of the submittal dealt with the specification of messages. As this portion of the document could apply to various protocols, it was placed in ISO 14827-1 — *Message definition requirements*. The remainder of the original submittal formed the basis of the application layer protocol and was placed in this part of ISO 14827. Thus, this part defines only one way to implement the messages that are specified in the format defined by ISO 14827-1. This resulting International Standard supports existing and foreseen data exchange needs using modern design concepts.

Due to the flexibility required by the rapidly developing transport information and control systems (TICS) environment, this part of ISO 14827 uses a very generic structure. Thus, although initially intended to be an International Standard for TICS, it is flexible enough to be used for virtually any data exchange.

ISO 14827-1 explains how to define end-application messages that are to be exchanged between centres for TICS. This definition has been designed to be relatively generic to the selected protocol (e.g. DATEX-ASN, CORBA, etc.) This part of ISO 14827 provides the specification of the Data Exchange protocol in ASN.1 (DATEX-ASN) used to exchange data between central systems. DATEX-ASN was the first protocol standardized because:

— the development of DATEX-Net could be leveraged, and

— there was sufficient market interest to perform the required technical work.

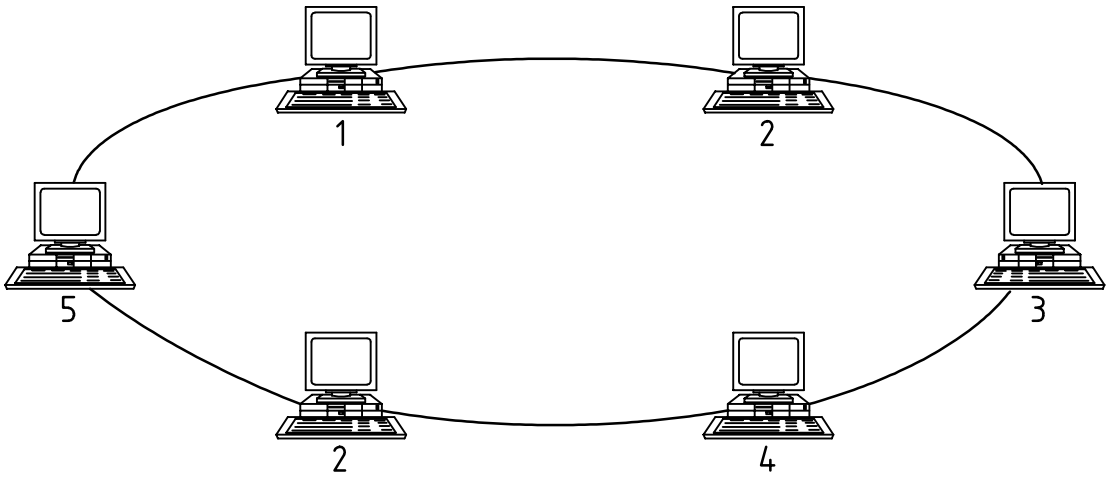
Transport information and control systems — Data interfaces between centres for transport information and control systems —

Part 2: DATEX-ASN

1 Scope

DATEX-ASN allows different systems to exchange relevant data. This is contained in end-application messages. Each end-application message is defined as either a “subscription” or a “publication” according to the format as specified in ISO 14827-1. DATEX-ASN defines how these end-application messages are packaged to form a complete data packet and also defines the rules and procedures for exchanging these data packets. Systems using DATEX-ASN are free to implement additional end-application functionalities according to the user requirements.

A DATEX-ASN network comprises a certain number of systems, an example of which is provided in Figure 1.

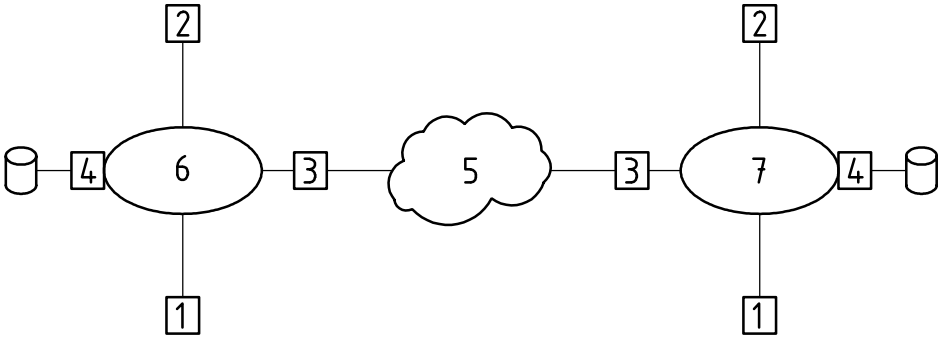


Key

1 weather system
2 traffic management system
3 transit management system
4 emergency management system
5 information service provider

Figure 1 — An example of a DATEX-ASN network

Each system can be viewed as consisting of the interfaces, as shown in Figure 2:



Key

1 application interface
2 operator interface
3 communication interface
4 database interface
5 communications cloud
6 client system
7 server system

Figure 2 — System interfaces

This part of ISO 14827 deals only with the communications interface. It has been designed to meet the unique requirements of TICS; however, it has been designed in a generic fashion and thus could be used for other data exchanges as well.

Systems implementing this part of 14827 sometimes operate simultaneously as a client and server, using multiple sessions. The communications cloud between the two systems may be complex or simple.

2 Normative references

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO 4217, *Codes for the representation of currencies and funds*

ISO 8824-1, *Information technology — Abstract Syntax Notation One (ASN.1) — Part 1: Specification of basic notation*

ISO 8825-2, *Information technology — ASN.1 encoding rules — Part 2: Specification of Packed Encoding Rules (PER)*

ISO 14827-1, *Transport information and control systems — Data interfaces between centres for transport information and control systems — Part 1: Message definition requirements*

3 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO 14827-1 and the following apply.

3.1
connectionless transport profile
service that provides end-system to end-system communications without any connection set-up

EXAMPLE UDP/IP.

3.2
connection-oriented transport profile
service that allows one end-system to exchange a continuous stream of data with another end-system, the data of which is guaranteed to be delivered in the same order in which it was sent without any duplication

NOTE This service is typically achieved by first establishing a connection, then sending the data, and finally terminating the connection.

EXAMPLE TCP/IP.

3.3
data element
syntactically formal representation of some single unit of information of interest (such as a fact, proposition, observation, etc) about some (entity) class of interest (e.g. a person, place, process, property, concept, association, state, event, etc.)

3.4
datagram
entity of data containing enough information to be routed from source to destination without relying on previous network configuration

EXAMPLE IP datagram.

3.5
datagram publication
DATEX-ASN publication (reply) that is sent directly over the given transport profile, in contrast with a file publication

3.6
destination
system or device to which the information in the data packet is intended to be sent

3.7
encoding rules
rules which specify the representation during transfer of the values of ASN.1 types

NOTE 1 Encoding rules also enable the values to be recovered from the representation, given knowledge of the type.

NOTE 2 For the purpose of specifying encoding rules, the various referenced type (and value) notations, which can provide alternative notations for built-in types (and values), are not relevant.

3.8
Ethernet
specific combination of physical and data link layer protocols as defined in IEEE 802.3 that allow multiple systems to gain access to a shared medium and communicate with one another

3.9
file
data storage object, which may be located on any file system such as a hard-disk, a floppy-disk, a RAM-drive, etc.

3.10
file publication
DATEX-ASN publication (reply) that is stored on the server's file system until the client has an opportunity to retrieve it via a file transfer protocol, in contrast with datagram publication

3.11
guaranteed delivery
DATEX-ASN mechanism in which the client acknowledges the receipt of a publication (reply)

3.12
heartbeat
data packet sent to indicate that the sending system is still alive and communicating

3.13
maximum turn-around time
maximum amount of time a system is given to provide an appropriate response to the incoming data packet

3.14
origin
system or device which was the source for all of the information in the data packet

NOTE In many cases, this will be the same as the sender, but could be different. For example, a bridge (or proxy agent) may translate between protocols; in this case the bridge (or proxy agent) would be the sender, while the system generating the data would be the origin.

3.15
port
logical channel in a communications system

NOTE UDP and TCP use port numbers to multiplex data packets from a variety of applications onto a single communications system.

3.16
response time-out period
maximum duration a system is required to wait for a response data packet prior to assuming that the previously sent data packet was never received by the other application

3.17
sender
system which created and sent the DATEX-ASN data packet

3.18
session
period of time during which a client and a server exchange multiple data packets

3.19
silently drop
to ignore a data packet

NOTE A data packet that is silently dropped does not cause any action to occur within the receiving system, nor is any response sent to the subject data packet.

3.20
transport profile
set of services which are responsible for providing a virtually error-free, point-to-point connection so that host A can send data packets to host B and they will arrive uncorrupted

NOTE Connection-oriented transport profiles may also ensure that the data packets arrive in the correct order.

3.21
turn-around time
period of time it takes a client or server to produce and transmit a response data packet, measured starting from the point at which the last byte of data is received from the other system to the point when the last response byte is transmitted

4 Symbols and abbreviated terms

For the purposes of this document, the abbreviated terms of ISO 14827-1 and the following apply.

CMIP Common Management Information Protocol (RFC 1189)

CORBA Common Object Request Broker Architecture

D-COM Distributed Communications Object Model

FDDI Fibre Distributed Data Interface (ANSI X3T9.5)

FrED Friendly Exchange of Data

FTP File Transfer Protocol (RFC 959)

HTML Hyper Text Mark-up Language (RFC 2854)

HTTP Hyper Text Transfer Protocol (RFC 2616)

IP Internet Protocol (RFC 791)

ISDN Integrated Services Digital Network

NTCIP National Transportation Communications for Intelligent Transportation Systems (ITS) Protocol

PPP Point-to-Point Protocol (RFC 1661)

SNMP Simple Network Management Protocol (RFC 1157)

SQL Structured Query Language

TCP Transmission Control Protocol (RFC 793)

TCIP Transit Communications Interface Profiles

TFTP Trivial File Transfer Protocol (RFC 1350)

TICS Transport Information and Control Systems

UDP User Datagram Protocol (RFC 768)

5 Implementation considerations

Before exchanging data, transportation centres must agree on the specific issues that are described in the list below.

NOTE Some of these issues (e.g. lower layer protocols) may be specified elsewhere. For example, Annex D provides a definition of these traits for standardized IP implementations.

a) General:

   1) time period throughout which the overall agreement is valid;

   2) rules for terminating the agreement before the expiry time of the agreement;

   3) server and client domain names and off-line contact addresses, telephone, fax and e-mail details.

b) Access (DATEX-ASN requires a user-name and associated password.):

   1) IP address of the client, assigned by the Internet Assigned Number Authority if the link uses a public network;

   2) IP address of the server, assigned by the Internet Assigned Number Authority if the link uses a public network;

   3) a list of authorized client user-names, referred to as the user-name used throughout this part of ISO 14827;

   4) a password associated with each client user-name.

c) Protocols:

   1) selection of lower layer protocols, including:

      i) presentation (e.g. BER, EDIFACT, or others) and session layers;

      ii) transport and network layers (e.g. UDP/IP, TCP/IP, etc.);

      iii) data link and physical layers (e.g. Ethernet, FDDI, PPP over ISDN).

   2) maximum datagram size;

   3) selection of preferred file transfer protocols.

d) Management of background information:

   1) specification of the Data Registry to be used.

e) Message management:

   1) messages which must be supported, which may include messages that are standardized in other documents and/or messages unique to the specific implementation.

6 Data exchange procedures

This part of ISO 14827 defines an application layer protocol by which data elements are exchanged between a client and server. Communication between client and server shall be accomplished by the exchange of data packets and files as defined in this section.

6.1 General data packet procedures

DATEX-ASN data packets shall be constructed according to the formally defined ASN.1 data structures defined in Annex A.

6.1.1 Sessions

This part of ISO 14827 requires all data packets to be transmitted in an application session. Within each session, one system shall act as a client and the other shall act as the server.

NOTE Multiple sessions may exist simultaneously. Thus, a pair of systems may have two concurrent sessions, one where System A acts as the client and System B acts as the server and the other where System A acts as the server and System B acts as the client. These sessions would be distinguished by lower layer protocols (e.g. TCP or UDP port numbers).

6.1.2 Transport requirements

Data may be exchanged over connection-less or connection-oriented transport profiles, but a single transport profile shall be used for all data packets exchanged within a session.

EXAMPLE If the first data packet in establishing a session is transmitted using UDP, then all data packets within that session will use UDP. Likewise, if the initial transmission is TCP, then all data packets will be TCP.

6.1.3 Response time-outs

The client and server shall negotiate the response time-out period for each session. The response time-out period should be long enough to accommodate the network propagation delays for both data packets as well as the turn-around time required to handle the message on the receiving end. In theory, this should be measured from when the last byte is transmitted to when the last response byte is received; however, it is expected that most implementations will measure the time from the return from the system write call to the return from the system read call.

NOTE A typical implementation is to set the time-out to be an integral multiple of the turn-around time and the multiplier is typically set to three. However, as some communications media and networks may experience significant delays, the system should allow this multiplier to be set at run-time.

6.1.4 Retransmission

If a specific data packet requires a response and an appropriate response is not received within the response time-out period, the identical data packet (e.g. same data packet number, same time stamp, etc.) shall be retransmitted one time only. If no response is received to the second data packet, prior to a subsequent response time-out period, the data packet transmission shall be considered unsuccessful. If a response is received after the time-out period, it may be ignored.

6.1.5 Duplicate data packets

Any time a client or server receives a data packet that requires a response, a new response data packet shall be prepared and transmitted as soon as possible, even if the received data packet was a duplicate data packet.

6.2 General file procedures

The client may request the publication (reply) data to be sent within the publication data packet, or it may request the publication data to be stored in a file on the server with the publication data packet indicating the file name of the publication file. The file can then be retrieved by the client within the constraints set by the server. Such a publication file shall only contain the “TICS information” as defined by the PublicationData structure as defined in A.9.

6.3 Sessions

Within each session, one system is a client and the other is a server. A server with a given domain name shall not accept more than one session with any client domain name with a given transport profile; however, as a single system may have multiple domain names, multiple sessions could exist between a given client system and server system pair.

NOTE 1 Multiple sessions may exist on a single physical link simultaneously. For example, system A may act as a server in one session with system B while acting as a client in a second session.

NOTE 2 A single client may have sessions with multiple servers simultaneously; thus, the complete session number over any given transport profile is defined by the server domain name followed by the client domain name.

NOTE 3 Some implementations may have a need to frequently publish relatively large data packets. There are various ways to achieve this, including: (1) increasing the UDP/IP datagram size to support the required size; or (2) maintaining a prolonged TCP connection over which the large data packets are periodically sent. The preferred solution will depend on a number of implementation-specific issues such as media quality and required reliability of transmission.

NOTE 4 Simultaneous sessions between a single client and server pair may exist if the sessions use different transport profiles (e.g. one UDP and one TCP).

6.3.1 Establishing a session

The server may wish to establish a session. For example, this may be in order to publish information for a registered subscription (request) or allow a receipt of a subscription if the server is protected by a firewall. In this case, the server shall transmit an “Initiate” data packet, as defined in A.3, with the datex-Destination-txt and datex-Sender-txt fields set to the proper name.

A server should not terminate a session it initiated for a period of one heartbeat duration after final publication.

If the client receives an “Initiate” data packet or if the client wishes to establish a session, the client shall transmit a “Login” data packet, as defined in A.4.

Upon receiving a “Login” data packet, a server shall determine if the domain names, user-name, password, maximum heartbeat duration, response time-out period, allowed encoding rules, datagram size and login reason are valid for the request. The server shall also ensure that a session with the given domain name and transport profile does not already exist. If the request is found to be invalid, the server shall either:

— respond with a “reject” data packet, as defined in A.12, with the “error-code” set to the most appropriate code number which applies to the denial, or

— not respond if the server determines this is appropriate due to security reasons.

If the request is valid, the server shall respond with an “accept” data packet, as defined in A.11, and shall identify the selected encoding rules from the list of options in the login request. This completes the procedures to establish a session.

The procedure to establish a session is summarized in Figure 3. All data packets exchanged during this procedure shall use the encoding rules that were agreed to off-line. All data packets exchanged after the successful completion of this procedure shall use the encoding rules, as negotiated within the “Login” and “Accept” data packets.

EXAMPLE Per Annex D, if the session is established over TCP/IP on Port 355, data packets exchanged during this procedure shall use BER encoding; data packets exchanged after the successful completion of the login process would then use the encoding rules negotiated by the “Login” and “Accept” data packets.

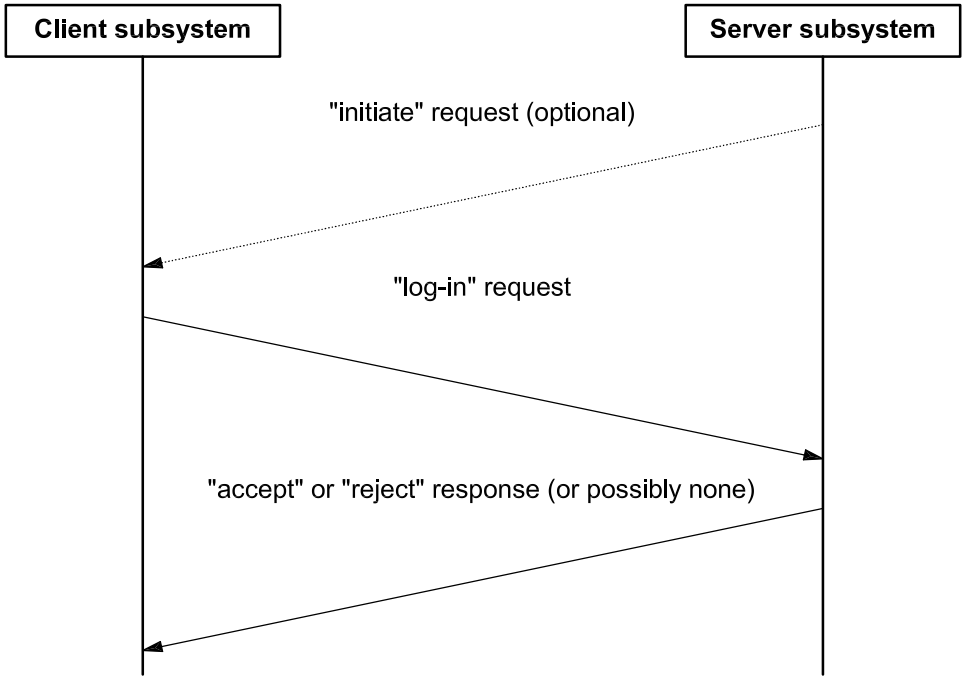


Figure 3 — Establishing a session

6.3.2 Maintaining a session

Sessions are maintained by the client and server exchanging “FrED” data packets. If, at any point during a session, no data packets are received from the other system for a period exceeding the maximum heartbeat duration, as specified in the login request, the session shall be immediately terminated by both the client and the server without exchanging any data. This type of termination should only be encountered due to unusual circumstances, e.g. a system crash.

NOTE 1 FrED stands for a “Friendly Exchange of Data”. The data packet is generally used as an acknowledgement data packet, but it is also used as a system heartbeat when there has been a prolonged period of silence. Thus, the term “ack” did not truly apply to this data packet and the committee determined that it should be termed a “FrED”.

NOTE 2 A session may be kept open permanently by meeting the requirements of this subclause.

The client shall maintain the session until the termination procedures are initiated as indicated by 6.3.3. The client shall keep track of the elapsed time since it received a data packet from the server and shall ensure that this time does not exceed the maximum heartbeat duration by generating “FrED” data packets, as defined in A.5, as needed. The DATEX.FrED_ConfirmPacket_number-ulong shall be zero (0) for such “FrED” data packets, hereinafter referred to as “FrED” heartbeat data packets. It is recommended that the client transmit “FrED” heartbeat data packets roughly three times more often than the time specified by the maximum heartbeat duration.

The server shall acknowledge a “FrED” heartbeat data packet by transmitting a “FrED” data packet with the DATEX-FrED_ConfirmPacket_number-ulong set to the packet number of the “FrED” heartbeat data packet which is being acknowledged. This shall complete the session maintenance procedure.

When desired, the session shall be terminated according to the procedure described in 6.3.3.

The procedure to maintain a session is summarized in Figure 4.

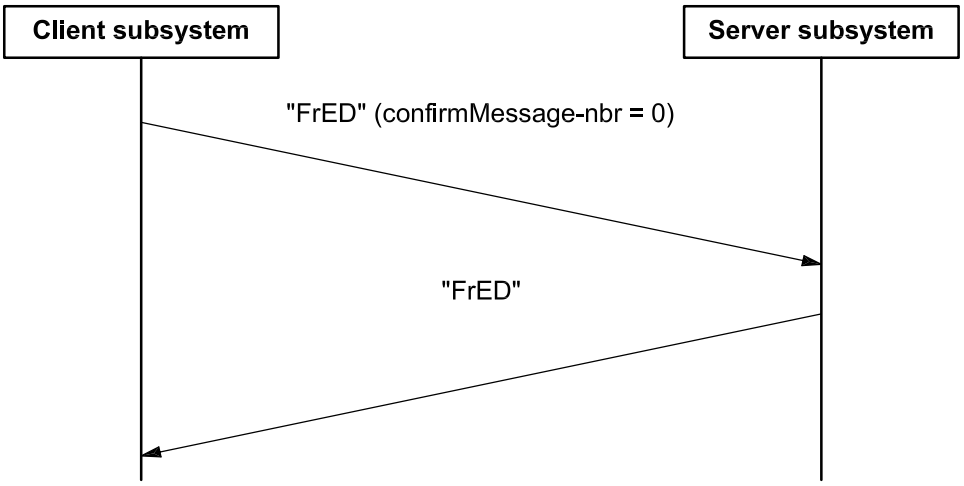


Figure 4 — Maintaining a session

6.3.3 Terminating a session

A session may be actively terminated by either the client or the server. If the server wishes to terminate a session, it shall transmit a “Terminate” data packet, as defined in A.6. If the server does not receive any response after two tries, the server shall terminate the session on its end.

If the client receives a valid “Terminate” data packet, or if the client wishes to terminate a session, it shall transmit zero or more subscription cancellations, as defined in 6.4 and A.8, if it wishes to cancel any persistent subscriptions, followed by a “Logout” data packet, as defined in A.7.

NOTE 1 Registered subscriptions do not expire with the termination of a session if the “Persistent” flag was set in the subscription. This allows systems to keep subscriptions active when the session is not active. For example, this may be useful for dial-up connections or to minimize the impact of system crashes.

NOTE 2 A server does not need to wait for a FrED for a guaranteed publication to a non-persistent subscription.

Upon receipt of a valid “Logout” data packet, the server shall terminate the associated session and issue a ”FrED” data packet, as defined in A.5. The client shall terminate the associated session upon receipt of the “FrED”. This shall complete the session termination procedure.

NOTE If the client does not receive the “FrED”, despite following the retransmission rules, it will terminate the session according to the rules of 6.3.2.

The procedure to terminate a session is summarized in Figure 5.

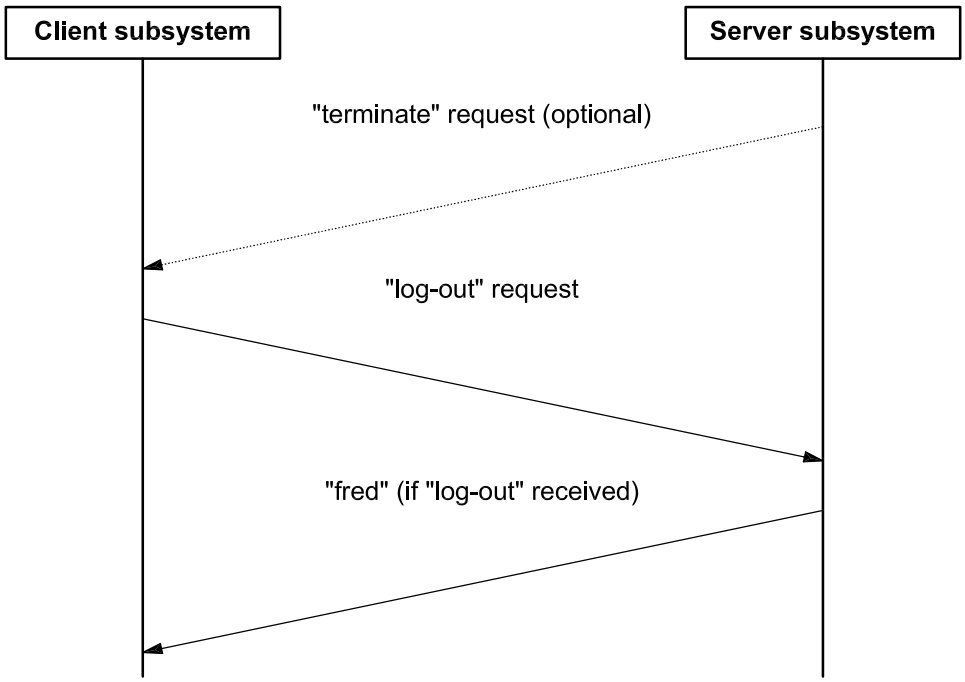


Figure 5 — Terminating a session

6.4 Requesting information

Clients and servers shall provide for off-line subscriptions (requests), as defined in 6.4.1, on-line subscriptions as defined in 6.4.2, or both.

NOTE The subscription process may take place off-line or on-line. This allows a server to transmit publications (replies) (e.g. accident publications) without having to support the associated subscription data packet. This may be desirable in order to bring legacy systems into compliance or for security purposes.

6.4.1 Off-line subscriptions

Servers may provide local mechanisms to register any and all subscriptions to which the server claims compliance. This feature shall be supported for all clients known to the server.

Clients may provide local configuration mechanisms to accept publications from a remote server so that the client may accept publications related to an off-line subscription.

6.4.2 On-line subscriptions

A client may support the ability to transmit “subscription” data packets, as defined in A.8. If the client claims compliance for on-line subscriptions, it shall support this service for all subscriptions to which it claims compliance.

A server may accept “subscription” data packets in order to allow for on-line requests to be processed. If the server claims compliance for on-line subscriptions, it shall support “subscription” data packets for all subscriptions to which it claims compliance.

Upon receipt of a subscription data packet, the server shall respond with either an ”accept” or ”reject” data packet, as defined in A.11 and A.12. An “accept” shall only indicate that the data was properly received and understood by the system; it does not guarantee that the end-application will accept the subscription. For example, if a valid subscription is received, but the owner of the given session is not authorized to receive the

requested data, an “accept” data packet would be transmitted, but the end-application would immediately transmit a “publication”, as defined in 6.5, indicating that the subscription has been terminated with a reason of accessDenied.

This shall complete the subscription procedure.

If the subscription was accepted, the server shall publish data according to 6.5. A subscription can be cancelled by setting the datexSubscribe-CancelReason-cd field to one of the cancel reasons.

The procedure to request information is summarized in Figure 6.

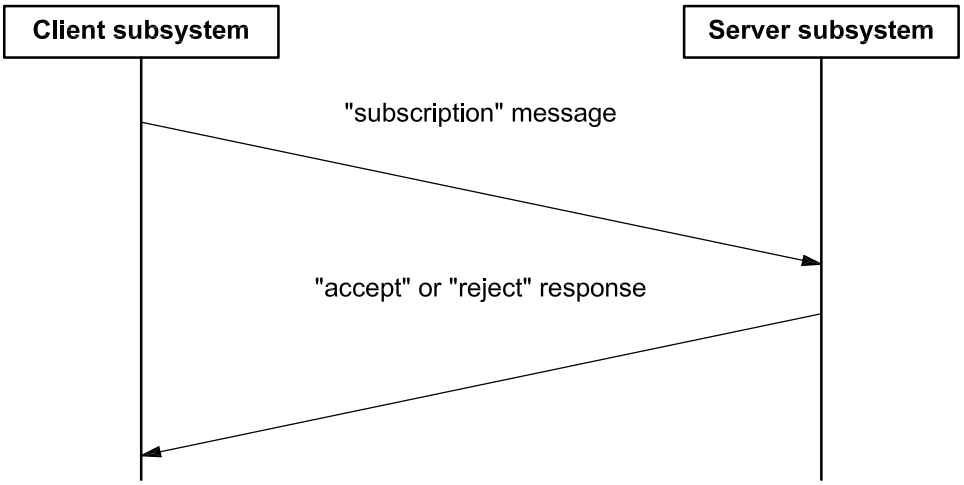


Figure 6 — Subscribing a session

6.5 Publication of information

Publication of information is partially dependent upon the type of request. The general procedure is described in 6.5.1; the specific procedures for the different types of requests are described in:

— 6.5.2 for single subscriptions, and

— 6.5.3 for registered subscriptions.

For each subscription to which a system claims compliance, the associated publication shall be supported. Support of the publication data packet is mandatory for all data packets less than the maximum datagram size. If the system claims compliance to publications that exceed the maximum datagram size, the system shall support the “publication-notice” (see 6.5.1.3) and the file transfer mechanism (see 6.5.1.5); otherwise, support for the “publication-notice” and file transfer mechanism are optional. If a system claims compliance to the “publication-notice” and file transfer mechanism, the system shall support these features for all subscriptions to which it claims compliance. Support for multiple publications within a single data packet or single file is mandatory for clients to receive and optional for servers to send.

6.5.1 General procedures

The server shall generate a “publication” data packet, as defined in A.9, at times specified in 6.5.2-6.5.3.

6.5.1.1 Guaranteed flag

The datexPublish-Guaranteed-bool shall be set to “true” if the associated “subscription” data packet requested guaranteed delivery; otherwise, it shall be set to “false”.

6.5.1.2 Datagram publications

If the client subscribed for datagram publications, the publication data packet shall contain the publication data if the resulting datagram is smaller than the maximum datagram size defined in the Interchange agreement.

NOTE A memory table would most typically be transmitted as a datagram, although it could also be transferred as a file by creating a virtual disk in memory.

6.5.1.3 Publication-notice datagram

If the client subscribed for file publications, or if the resulting datagram is larger than the maximum datagram size, the data shall be stored in a file and the publication data packet shall indicate the path and filename of the publication data.

NOTE The file transfer uses standard file transfer protocols; thus, the file is exchanged in a separate session from the one used for the DATEX-ASN data packets described in this part of ISO 14827.

6.5.1.4 Client response

If the client determines that the publication is improperly encoded, except for any PublicationData structure that may be present, it shall issue a “reject” data packet, as defined in A.12. Otherwise, if datexPublish-Guaranteed-bool is set to true, the client shall issue an “accept” data packet, and if datexPublish-Guaranteed-bool is set to false, no response shall be sent.

NOTE Any errors within the PublicationData structure are handled by the procedures in 6.5.1.6.

This shall complete the publication procedures if the “publication” data packet was invalid or if the “publication” data packet indicated a filename and that file was previously downloaded and not yet acknowledged with a transfer done data packet. If the “publication” data packet indicated a file that has not been previously downloaded, the procedures of 6.5.1.5 and 6.5.1.6 shall follow in order. If the “publication” data packet contained the publication data, the procedures of 6.5.1.6 shall follow.

NOTE A duplicate file name may be received due to a duplicate message being sent because of a communications error or due to a recycling of message names. Servers may wish to name files with large sequential numbers to avoid duplicate file names and prevent any loss of data. If the client has already been notified of the publication, there is no reason to start another file transfer process.

6.5.1.5 File transfer mechanism

If the “publication” data packet indicated a new filename (i.e. it was not a duplicate message), the client shall immediately retrieve the indicated file after sending the accept data packet, if required. The file transfer shall be via one of the supported file transfer mechanisms, as negotiated in the “subscription” and “publication” data packets. Once the file has been transferred (or the client has exceeded its maximum number of tries to download the file), the client shall transmit a “transfer-done” data packet, as defined in A.10, to verify the receipt (or notify the server of failure). The server shall acknowledge the “transfer-done” data packet with an “FrED” data packet, as defined in A.5. The server shall attempt to keep the file available for downloading until the transfer done notice is received.

6.5.1.6 Reject invalid publication data

For each invalid PublicationData structure contained within the publication (due to invalid encoding), the client shall send a reject data packet. This shall complete these procedures.

The procedure to publish information is summarized in Figure 7. A server may cancel any subscription at any time by sending a publication data packet with the publicationType field of the PublicationData structure set to one of the terminate codes.

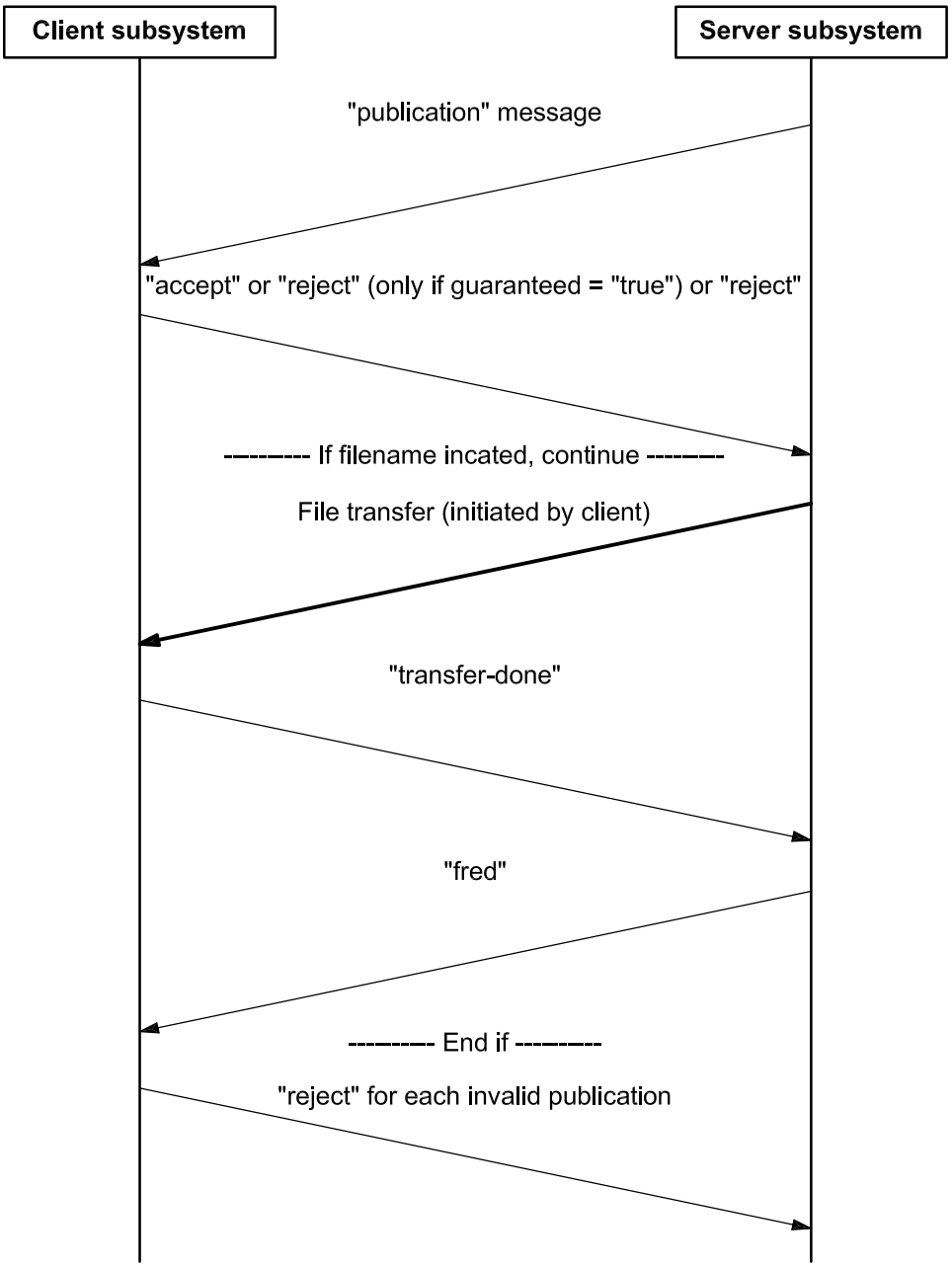


Figure 7 — Publishing information

NOTE The file transfer in the above exchange is a complex procedure, as defined by the associated file transfer standard. The file transfer is requested by the client, but is shown as an arrow from the server to indicate that the file is on the server and is being sent to the client.

6.5.2 Single subscriptions

For one-time, or “single” requests, the server shall publish the requested data as soon as possible after completion of the “subscription” process. The publication shall contain all data fulfilling the subscription request.

NOTE Historical data are considered to be separate data elements; thus, historical data can be retrieved as any other information.

6.5.3 Registered subscriptions

Subscriptions can also be “registered” in order to request information as it becomes available or on a periodic basis.

6.5.3.1 Registered subscriptions shall either be continuous or daily (i.e. activated on specific days of the week).

a) If the subscription is continuous, the subscription shall activate at the datexRegistered-StartTime and remain activated 24 hours a day, seven days a week, until the subscription expires (i.e. reaches the datexRegistered-EndTime) or is explicitly cancelled by a subsequent subscription. If the datexRegistered-EndTime is less than or equal to the datexRegistered-StartTime, then no publication shall be made.

b) If the subscription is “daily”, the subscription shall become active within the server at the start time (datexRegistered-StartTime) on each valid day of week (datexRegistered-DaysOfWeek-cd) occurring on or after the specified start date (datexRegistered-StartDate) and before or on the specified end date (datexRegistered-EndDate). It shall be immediately activated if the current time is valid when the subscription is registered. The subscription shall deactivate at the end of the period defined by the start time (datexRegistered-StartTimeOfDay) plus the duration (datexRegistered-Duration). The end date shall not be earlier than the start date. A daily subscription may also be deactivated by a request that cancels or modifies the subscription. If the datexRegistered-EndDate is less than the datexRegistered-StartDate, then no publication shall be made.

6.5.3.2 Upon subscription activation (or re-activation), the server shall publish an initial publication message. If the server can not provide information at the indicated start time, it shall provide information as soon as possible.

6.5.3.3 Registered subscriptions shall also be classified as either: a) event-driven (i.e. provides information when a specific event occurs); or b) periodic (i.e. provides information at a defined frequency).

6.5.3.4 After the initial publication is produced, the server shall attempt to produce a subsequent publication message as follows.

6.5.3.4.1 Ifthe mode is “periodic”, the server shall attempt to produce a new publication periodically at a frequency as defined by datexRegistered-UpdateDelay-qty. If the subscription is sent after the start time, the cycle shall be synchronized with the datexRegistered-StartTime.

NOTE In this case, the initial publication will be at a random point in the cycle, and the second publication may follow at any fraction of the cycle later, but will occur on a cycle point as measured from the datexRegistered-StartTime.

In the periodic mode, a server should publish information at every cycle point. If the server is unable to publish the information within a period of 60 % of a cycle beyond the cycle point, the publication should not be transmitted. Both the server and the client should terminate less important subscriptions (e.g. as reflected in the datexSubscription-Priority field) to minimize the probability of this occuring.

NOTE For example, it is assumed that there is a periodic subscription for data every second. The server responds originally at the start time, the next publication (containing information about conditions at 1,0 s) is sent at 1,25 s, the next publication (indicating conditions at 2,0 s) is sent at 2,5 s, and the next publication (containing data valid for 3,0 s) is not ready until 3,75 s. This last publication should be ignored (i.e. it is more than 0,6 s late) and the server and client should consider cancelling less important subscriptions. The intent is to not send old data and having the systems build up an infinite backlog of messages to send while the system is clearly at maximum capacity. Depending on message content and system design, this may or may not be a practical problem. Thus, this clause is provisional and the implementation should take whatever appropriate actions are necessary to solve these problems.

6.5.3.4.2 If the mode is “event-driven”, the server shall produce a publication within a period of datexRegistered-UpdateDelay-qty after the server is notified of an event. Thus, in this case, the datexRegistered-UpdateDelay-qty parameter serves as a maximum latency value for event reporting. The subscription message shall define the term “event” in the definition and/or message body. If the maximum latency is exceeded, the data shall be published as soon as possible and the datexPublish-LatePublicationFlag shall be set. Servers should terminate less important subscriptions (as reflected in the datexSubscription-Priority field) to minimize the probability of this occuring.

Annex A
(normative)

Data packet structures

A.1 General

DATEX-ASN data packets are defined in ASN.1 as Application Layer data packets and can be exchanged using any compatible lower-layer combination. All DATEX-ASN data packets shall conform to the DatexDataPacket structure (and appropriate substructures) as defined in the following ASN.1 module. Each field identified in this module is formally defined in Annex B of this part of ISO 14827.

```
ISO14827-2 {iso(1) standards(0) std14827(14827) part2(2)} DEFINITIONS AUTOMATIC TAGS ::=
BEGIN

DatexDataPacket ::= SEQUENCE {
       datex-Version-cd              ENUMERATED {
                                                 experimental,
                                                 version-1,
                                                 ...},
       datex-Data-txt            OCTET STRING,
              -- an optionally encrypted C2CAuthenticatedMessage
       datex-Crc-id              OCTET STRING (SIZE (2))
       }

C2CAuthenticatedMessage ::= SEQUENCE {
       datex-AuthenticationInfo-txt      OCTET STRING (SIZE (0..255)),
       datex-DataPacket-nbr          INTEGER (0..4294967295),
       datex-DataPacketPriority-cd       INTEGER (0..10),
       options                HeaderOptions,
       pdu                    PDUs
       }

HeaderOptions ::= SEQUENCE {
       datex-Origin-txt              UTF8String (SIZE (0..40)) OPTIONAL,
       datex-OriginAddress-loc           OCTET STRING          OPTIONAL,
       datex-Sender-txt              UTF8String (SIZE (0..40)) OPTIONAL,
       datex-SenderAddress-loc           OCTET STRING          OPTIONAL,
       datex-Destination-txt         UTF8String (SIZE (0..40)) OPTIONAL,
       datex-DestinationAddress-loc      OCTET STRING          OPTIONAL,
       cost               Cost                    OPTIONAL,
       datex-DataPacketTime          Time                  OPTIONAL
       }

Cost ::= SEQUENCE {
       amount-Currency-cd        OCTET STRING (SIZE (3)),
       amount-Factor-qty             INTEGER,
       amount-Quantity-qty       INTEGER
       }
```

A.2 Protocol data unit

The PDU structure allows multiple types of data packets to be sent in the same overall structure as defined above. The various types of structures that can be contained in the PDU are described below:

— initiate: allows the sever to request a new session;

— login: checks passwords, etc., and manages who is on-line;

— FrED: a “friendly exchange of data” used to confirm receipt of data packets and to maintain a session when there are periods of silence;

— terminate: when the server has to discontinue the session;

— logout: allows the client to discontinue the session;

— subscription: requests data (may be one-time only or registered);

— publication: provides the requested data;

— transfer-done: allows the client to notify the server that a publication file has been retrieved;

— accept: accepts a login, subscription or publication;

— reject: rejects a login, subscription or publication.

```
PDUs ::= CHOICE {
               initiate          Initiate,
               login             Login,
               fred              FrED,
               terminate         Terminate,
               logout            Logout,
               subscription      Subscription,
               publication       Publication,
               transfer-done     TransferDone,
               accept            Accept,
               reject            Reject
               }
```

A.3 Initiate data packet structure

```
Initiate ::= SEQUENCE {
       datex-Sender-txt              UTF8String (0..40),
       datex-Destination-txt         UTF8String (0..40)

                        }
```

A.4 Login data packet structure

```
Login ::= SEQUENCE {
       datex-Sender-txt                      UTF8String (0..40),
       datex-Destination-txt                 UTF8String (0..40),
       datexLogin-UserName-txt               OCTET STRING,
       datexLogin-Password-txt               OCTET STRING,
       datexLogin-EncodingRules-id           SEQUENCE OF OBJECT IDENTIFIER,
       datexLogin-HeartbeatDurationMax-qty   INTEGER (0..65535),
       datexLogin-ResponseTimeOut-qty        INTEGER (0..255),
       datexLogin-Initiator-cd               ENUMERATED {
                                                    serverInitiated,
                                                    clientInitiated,
```

```
                                                    ...}
        datexLogin-DatagramSize-qty       INTEGER (0..65535)
        }
```

A.5 FrED data packet structure

```
FrED ::= INTEGER (0..4294967295) -- datexFrED-ConfirmPacket-nbr
```

A.6 Terminate data packet structure

```
Terminate ::= ENUMERATED {    -- datexTerminate-Reason-cd
                    other,
                    serverRequested,
                    clientRequested,
                    serverShutdown,
                    clientShutdown,
                    serverCommProblems,
                    clientCommProblems,
                    ...}
```

A.7 Logout data packet structure

```
Logout ::= ENUMERATED { -- datexLogout-Reason-cd
                    other,
                    serverRequested,
                    clientRequested,
                    serverShutdown,
                    clientShutdown,
                    serverCommProblems,
                    clientCommProblems,
                    ...}
```

A.8 Subscription data packet structure

```
Subscription ::= SEQUENCE {
        datexSubscribe-Serial-nbr          INTEGER (0..4294967295),
        type                         SubscriptionType,
        ...}

SubscriptionType ::= CHOICE {
        subscription                   SubscriptionData,
        datexSubscribe-CancelReason-cd     ENUMERATED {
                                                        other,
                                                        dataNotNeeded,
                                                        errorsInPublication,
                                                        pendingLogout,
                                                        processingMgmt,
                                                        bandwidthMgmt,
                                                        ...}
        }

SubscriptionData ::= SEQUENCE {
        datexSubscribe-Persistent-bool     BOOLEAN,
        datexSubscribe-Status-cd           ENUMERATED {
```

```
                                                        new,
                                                        update},
        mode                        SubscriptionMode,
        datexSubscribe-PublishFormat-cd       ENUMERATED {
                                                        other,
                                                        ftp,
                                                        tftp,
                                                        dataPacket,
                                                        ...},
        datexSubscribe-Priority-cd        INTEGER(1..10),
        datexSubscribe-Guarantee-bool         BOOLEAN,
        message                     EndApplicationMessage
        }

SubscriptionMode ::= CHOICE {
        single                      Null,
        event-driven                    Registered,
        periodic                    Registered
        }

Registered::= CHOICE {
        continuous                  SEQUENCE {
         datexRegistered-UpdateDelay-qty   INTEGER (0..4294967295)   DEFAULT 0,
                -- 0 means as soon as possible
              datexRegistered-StartTime      Time               OPTIONAL,
            -- defaults to immediate
              datexRegistered-EndTime        Time               OPTIONAL
            -- defaults to "until cancelled"
         },
        daily                       SEQUENCE {
         datexRegistered-UpdateDelay-qty   INTEGER (0..4294967295)   DEFAULT 0,
                -- 0 means as soon as possible
              datexRegistered-DaysOfWeek-cd      BIT STRING {
                                                        other,
                                                        sunday,
                                                        monday,
                                                        tuesday,
                                                        wednesday,
                                                        thursday,
                                                        friday,
                                                        saturday}
                                                        (SIZE (8)),
         datexRegistered-StartDate     Time               OPTIONAL,
                -- defaults to immediate
              datexRegistered-EndDate        Time               OPTIONAL,
                -- defaults to "until cancelled"
              datexRegistered-StartTime      Time               OPTIONAL,
                    -- defaults to midnight
              datexRegistered-Duration-qty       INTEGER (0..65535) OPTIONAL
            -- defaults to 1440 (i.e., 24 hours)
              }
        }

Time ::= SEQUENCE {
        time-Year-qty                   INTEGER (-32768..32767)   OPTIONAL,
         -- defaults to current year unless otherwise specified
        time-Month-qty                  INTEGER (1..12)       OPTIONAL,
         -- defaults to current month unless otherwise specified
        time-Day-qty                    INTEGER (1..31)       OPTIONAL,
         -- defaults to current day unless otherwise specified
```

```
        time-Hour-qty                   INTEGER (0..23)         DEFAULT 0,
        time-Minute-qty                 INTEGER (0..59)         DEFAULT 0,
        time-Second-qty                 INTEGER (0..60)         DEFAULT 0,
        secondFractions                 CHOICE {
               time-Deciseconds-qty             INTEGER (0..9),
               time-Centiseconds-qty                INTEGER (0..99),
               time-Milliseconds-qty                INTEGER (0..999),
               ...}                                     DEFAULT 0,
               timezone                     SEQUENCE {
               time-TimeZoneHour-qty                INTEGER (-13..13)  DEFAULT 0,
               time-TimeZoneMinute-qty              INTEGER (0..59)    DEFAULT 0
               }                                    OPTIONAL
                      -- defaults to UTC
        }
```

A.9 Publication data packet structure

```
Publication ::= SEQUENCE {
        datexPublish-Guaranteed-bool            BOOLEAN,
        format                      Publish-Format
        }

Publish-Format ::= CHOICE {
        data                        SEQUENCE OFPublicationData,
        datexPublish-FileName-txt           UTF8String (SIZE (0..2000))
        }
-- Support for sending/receiving multiple publications in a single data packet or file is
optional;         however,         the         'data'         field         of         the
-- Publish-Format structure must still be coded as a SEQUENCE OF structure.


PublicationData ::= SEQUENCE {
        datexPublish-SubscribeSerial-nbr        INTEGER (0..4294967295),
        datexPublish-Serial-nbr                 INTEGER (0..4294967295),
        datexPublish-LatePublicationFlag-bool BOOLEAN,
        publicationType                 PublicationType
        }


PublicationType : := CHOICE {
        datexPublish-Management-cd      ENUMERATED {
                                                temporarilySuspended,
                                                resume,
                                                terminate-other,
                                                terminate-dataNoLongerAvailable,
                                                terminate-publicationsBeingRejected,
                                                terminate-PendingShutdown,
                                                terminate-processingMgmt,
                                                terminate-bandwidthMgmt,
                                                terminate-accessDenied,
                                                unknownRequest,
                                                . . .},
        publicationData                 EndApplicationMessage
        }


EndApplicationMessage ::= SEQUENCE {
        endApplication-Message-id           ISO14827-MESSAGE.&id,
        endApplication-Message-msg          ISO14827-MESSAGE.&MessageBody
```

```
    }
```

A.10 Transfer done data packet structure

```
TransferDone ::= SEQUENCE {
    datexTransferDone-FileName-txt       UTF8String (SIZE (0..2000)),
       datexTransferDone-Success-bool    BOOLEAN
       }
```

A.11 Accept data packet structure

```
Accept ::= SEQUENCE {
    datexAccept-Packet-nbr           INTEGER (0..4294967295),
    acceptType                    CHOICE {
              datexAccept-Login-id          OBJECT IDENTIFIER,
                -- encoding rules
              single-subscription           NULL,
              datexAccept-Registered-nbr           INTEGER (0..4294967295),
            -- the accepted value for the UpdateDelay parameter
              publication                   NULL
              }
       }
```

A.12 Reject data packet structure

```
Reject ::= SEQUENCE {
    datexReject-Packet-nbr           INTEGER (0..4294967295),
    rejectType                       RejectType,
    alternateRequest                 AlternateRequest   OPTIONAL
       }

RejectType ::= CHOICE {
    datexReject-Login-cd             ENUMERATED {
                                                    other,
                                                    unknownDomainName,
                                                    accessDenied,
                                                    invalidNamePassword,
                                                    timeoutTooSmall,
                                                    timeoutTooLarge,
                                                    heartbeatTooSmall,
                                                    heartbeatTooLarge,
                                                    sessionExists,
                                                    maxSessionsReached,
                                                    ...},
    datexReject-Subscription-cd      ENUMERATED {
                                                    other,
                                                    unknownSubscriptionNbr,
                                                    invalidTimes,
                                                    frequencyTooSmall,
                                                    frequencyTooLarge,
                                                    invalidMode,
                                                    publishFormatNotSupported,
                                                    unknowSubscriptionMsgId,
                                                    invalidSubscriptionMsgId,
                                                    invalidSubscriptionContent,
                                                    ...},
```

```
    datexReject-Publication-cd          ENUMERATED {
                                other,
                                invalidPublishFormat,
                                ...},
       rejectPublicationData            SEQUENCE {
              datexReject-SubscriptionSerial-nbr    INTEGER (0..4294967295),
              datexReject-PublicationSerial-nbr     INTEGER (0..4294967295),
              datexReject-PublicationData-cd        ENUMERATED {
                                                        other,
                                                        unknownSubscription,
                                                        unknownPublicationNbr,
                                                        unknownPublicationMsgId,
                                                        invalidPublicationMsgId,
                                                        invalidPublicationMsgContent,
                                                        repeatedPublicationNbr,
                                                        ...}
        }

       }

AlternateRequest ::= SubscriptionType
```

Annex B
(normative)

Data dictionary

The data elements defined within this section are defined using the following ASN.1 Information Object Specification:

```
DatexDataDictionary ::= BEGIN

DATA-ELEMENT ::= CLASS {
&name          UTF8String (SIZE (0..255)),
&namecontext       UTF8String (SIZE (0..40)),
&definition    UTF8String (SIZE (0..65535)),
&class         UTF8String (SIZE (0..40)),
&classScheme       UTF8String (SIZE (0..40)),
&classSchemeVer    UTF8String (SIZE (0..40)),
&keyword       UTF8String (SIZE (0..255))       OPTIONAL,
&remarks       UTF8String (SIZE (0..2000))      OPTIONAL,
&valueDomain       UTF8String (SIZE (0..255)),
&DataType,
&valueRule     UTF8String (SIZE (0..255)),
&constraints       UTF8String (SIZE (0..2000))      OPTIONAL}
WITH SYNTAX {
DESCRIPTIVE-NAME       &name
CONTEXT        &namecontext
DEFINITION         &definition
CLASS              &class
CLASS SCHEME       &classScheme
SCHEME VERSION     &classSchemeVer
[KEYWORDS          &keyword]
[REMARKS           &remarks]
VALUE DOMAIN       &valueDomain
DATA TYPE          &DataType
VALUE RULE         &validValueRule
[CONSTRAINTS       &constraints]
}
```

The included fields are defined to conform with those as specified in IEEE P1489-1999. Some fields required by IEEE 1489-1999 are not included in this standard due to their redundancy.

B.1 AMOUNT_Currency_code-datex1

```
amount-Currency-cd  DATA-ELEMENT ::= {
DESCRIPTIVE-NAME    "AMOUNT_Currency_code-datex1"
CONTEXT             "ITS"
DEFINITION          "The three letter currency code associated with the quantity."
CLASS               "Communication Networks"
CLASS SCHEME        "ITS Classification Scheme"
SCHEME VERSION      "980201"
VALUE DOMAIN        "Code-Currency"
DATA TYPE           OCTET STRING (SIZE (3))
VALUE RULE          "ISO 4217"
}
```

B.2 AMOUNT_Factor_quantity

```
amount-Factor-qty   DATA-ELEMENT ::= {
DESCRIPTIVE-NAME    "AMOUNT_Factor_quantity"
CONTEXT             "ITS"
DEFINITION          "The 10^x factor applied to the value given in AMOUNT_Quantity-
                    quantity. For example, if the Curency is USD, the Factor is -3, and
                    the Quatity is 11; the amount being specified would be US$0.011 or 1.1
                    US cents."
CLASS               "Communication Networks"
CLASS SCHEME        "ITS Classification Scheme"
SCHEME VERSION      "980201"
VALUE DOMAIN        "Qty-unlimited"
DATA TYPE           INTEGER
VALUE RULE          "INTEGER"
}
```

B.3 AMOUNT_Quantity_quantity

```
amount-Quantity-qty DATA-ELEMENT ::= {
DESCRIPTIVE-NAME    "AMOUNT_Quantity_quantity"
CONTEXT             "ITS"
DEFINITION          "The quantity of units as specified by currency and factor."
CLASS               "Communication Networks"
CLASS SCHEME        "ITS Classification Scheme"
SCHEME VERSION      "980201"
VALUE DOMAIN        "Qty-unlimited"
DATA TYPE           INTEGER
VALUE RULE          "INTEGER"
}
```

B.4 DATEX.ACCEPT_Login_id-oid

```
datexAccept-Login-id  DATA-ELEMENT ::= {
DESCRIPTIVE-NAME    "DATEX.ACCEPT_Login_id-oid"
CONTEXT             "ITS"
DEFINITION          "Indicates the OID of the encoding rules being accepted for the
                    session."
CLASS               "Communication Networks"
CLASS SCHEME        "ITS Classification Scheme"
SCHEME VERSION      "980201"
VALUE DOMAIN        "Id-Object Identifier"
DATA TYPE           OBJECT IDENTIFIER
VALUE RULE          "OBJECT IDENTIFIER"
}
```

B.5 DATEX.ACCEPT_Packet_number-ulong

```
datexAccept-Packet-nbrDATA-ELEMENT ::= {
DESCRIPTIVE-NAME    "DATEX.ACCEPT_Packet_number-ulong"
CONTEXT             "ITS"
DEFINITION          "The DATEX.MESSAGE_DataPacket_number-ulong, as defined in B.19, of the
                    request which is being accepted."
CLASS               "Communication Networks"
CLASS SCHEME        "ITS Classification Scheme"
SCHEME VERSION      "980201"
VALUE DOMAIN        "Number-ULONG"
DATA TYPE           INTEGER (0..4294967295)
VALUE RULE          "INTEGER (0..4294967295)"
}
```

B.6 DATEX.ACCEPT_Registered_number-ulong

```
datexAccept-Registered-nbr    DATA-ELEMENT ::= {
DESCRIPTIVE-NAME     "DATEX.ACCEPT_Registered_number-ulong"
CONTEXT              "ITS"
DEFINITION           "Indicates the accepted value for the datexRegistered-UpdateDelay-qty
                     Parameter from the associated request."
CLASS                "Communication Networks"
CLASS SCHEME         "ITS Classification Scheme"
SCHEME VERSION       "980201"
VALUE DOMAIN         "Number-ULONG"
DATA TYPE            INTEGER (0..4294967295)
VALUE RULE           "INTEGER (0..4294967295)"
}
```

B.7 DATEX.FrED_ConfirmPacket_number-ulong

```
datexFrED-ConfirmPacket-nbr   DATA-ELEMENT ::= {
DESCRIPTIVE-NAME     "DATEX.FrED_ConfirmPacket_number-ulong"
CONTEXT              "ITS"
DEFINITION           "The data packet number being confirmed by the 'friendly exchange of
                     data'data packet. If the FrED is being used as a heartbeat, the value
                     of this field shall be zero (0)."
CLASS                "Communication Networks"
CLASS SCHEME         "ITS Classification Scheme"
SCHEME VERSION       "980201"
VALUE DOMAIN         "Number-ULONG"
DATA TYPE            INTEGER (0..4294967295)
VALUE RULE           "INTEGER (0..4294967295)"
}
```

B.8 DATEX.LOGIN_DatagramSize_quantity-ushort

```
datexLogin-DatagramSize-qty   DATA-ELEMENT ::= {
DESCRIPTIVE-NAME     "DATEX.LOGIN_DatagramSize_quantity-ushort"
CONTEXT              "ITS"
DEFINITION           "The maximum datagram size that will be supported during the session."
CLASS                "Communication Networks"
CLASS SCHEME         "ITS Classification Scheme"
SCHEME VERSION       "980201"
VALUE DOMAIN         "Qty-USHORT"
DATA TYPE            INTEGER (0..65535)
VALUE RULE           "INTEGER (0..65535)"
}
```

B.9 DATEX.LOGIN_EncodingRules_id-oids

```
datexLogin-EncodingRules-id   DATA-ELEMENT ::= {
DESCRIPTIVE-NAME     "DATEX.LOGIN_EncodingRules_id-oids"
CONTEXT              "ITS"
DEFINITION           "A listing of the encoding rules that the Client supports."
CLASS                "Communication Networks"
CLASS SCHEME         "ITS Classification Scheme"
SCHEME VERSION       "980201"
VALUE DOMAIN         "Id-Object Identifiers"
DATA TYPE            SEQUENCE OF OBJECT IDENTIFIER
VALUE RULE           "SEQUENCE OF OBJECT IDENTIFIER"
}
```

B.10 DATEX.LOGIN_HeartbeatDurationMax_quantity-ushort

```
datexLogin-HeartbeatDurationMax-qty  DATA-ELEMENT ::= {
DESCRIPTIVE-NAME     "DATEX.LOGIN_HeartbeatDurationMax_quantity-ushort"
CONTEXT              "ITS"
DEFINITION           "The maximum duration, in seconds, allowed during the session without
                     the exchange of data packets. If this time is exceeded without any
                     data packets being received from the other system, the session shall
                     be locally terminated without exchanging any data packets. The value
                     zero shall indicate that heartbeats are not being used."
CLASS                "Communication Networks"
CLASS SCHEME         "ITS Classification Scheme"
SCHEME VERSION       "980201"
VALUE DOMAIN         "Qty-USHORT"
DATA TYPE            INTEGER (0..65535)
VALUE RULE           "INTEGER (0..65535)"
}
```

B.11 DATEX.LOGIN_Initiator_code-datex2

```
datexLogin-Initiator-cd   DATA-ELEMENT ::= {
DESCRIPTIVE-NAME     "DATEX.LOGIN_Initiator_code-datex2"
CONTEXT              "ITS"
DEFINITION           "Indicates who initiated the Login request."
CLASS                "Communication Networks"
CLASS SCHEME         "ITS Classification Scheme"
SCHEME VERSION       "980201"
VALUE DOMAIN         "Code-DATEX Initiator"
DATA TYPE            ENUMERATED {serverInitiated, clientInitiated, …}
VALUE RULE           "ENUMERATED {serverInitiated, clientInitiated, …}"
}
```

B.12 DATEX.LOGIN_Password_text-general

```
datexLogin-Password-txt   DATA-ELEMENT ::= {
DESCRIPTIVE-NAME     "DATEX.LOGIN_Password_text-general"
CONTEXT              "ITS"
DEFINITION           "An optionally encrypted password for the login request. The password
                     and user name shall be verified by the end application to ensure that
                     the user has appropriate rights. The encryption algorithm is not
                     specified in this standard."
CLASS                "Communication Networks"
CLASS SCHEME         "ITS Classification Scheme"
SCHEME VERSION       "980201"
VALUE DOMAIN         "Text-OctetString Unlimited"
DATA TYPE            OCTET STRING
VALUE RULE           ”OCTET STRING”
}
```

B.13 DATEX.LOGIN_ResponseTimeOut_quantity-ubyte

```
datexLogin-ResponseTimeOut-qty   DATA-ELEMENT ::= {
DESCRIPTIVE-NAME     "DATEX.LOGIN_ResponseTimeOut_quantity-ubyte"
CONTEXT              "ITS"
DEFINITION           "The time in seconds within which a system will expect to receive a
                     response to the transmision of a data packet. This same timer will
                     apply to both the Client and the Server and will apply to all data
                     packets transmitted within the session where a response is required.
                     The time is measured from the return from the system write () call. If
                     a system has not received the appropriate response prior to this timer
```

```
                    expiring, it will assume the transmitted data packet was not received
                    by the other end and proceed as defined in Clause 6.1.4."
CLASS               "Communication Networks"
CLASS SCHEME        "ITS Classification Scheme"
SCHEME VERSION      "980201"
VALUE DOMAIN        "Qty-UBYTE"
DATA TYPE           INTEGER (0..255)
VALUE RULE          "INTEGER (0..255)"
CONSTRAINTS         "The value zero (0) is not allowed."
}
```

B.14 DATEX.LOGIN_UserName_text-general

```
datexLogin-UserName-txt   DATA-ELEMENT ::= {
DESCRIPTIVE-NAME    "DATEX.LOGIN_UserName_text-general"
CONTEXT             "ITS"
DEFINITION          "An optionally encrypted user-name for the Login request. The
                    encryption algorithm is not specified in this standard."
CLASS               "Communication Networks"
CLASS SCHEME        "ITS Classification Scheme"
SCHEME VERSION      "980201"
VALUE DOMAIN        "Text-OctetString Unlimited"
DATA TYPE           OCTET STRING
VALUE RULE          ”OCTET STRING”
}
```

B.15 DATEX.LOGOUT_Reason_code-datex14

```
datexLogout-Reason-cd DATA-ELEMENT ::= {
DESCRIPTIVE-NAME    "DATEX.LOGOUT_Reason_code-datex14"
CONTEXT             "ITS"
DEFINITION          "The reason the logout is occuring."
CLASS               "Communication Networks"
CLASS SCHEME        "ITS Classification Scheme"
SCHEME VERSION      "980201"
VALUE DOMAIN        "Code-DATEX Terminate"
DATA TYPE           ENUMERATED {other, serverRequested, clientRequested, serverShutdown,
                    clientShutdown, serverCommProblems, clientCommProblems, …}
VALUE RULE          "ENUMERATED {other, serverRequested, clientRequested, serverShutdown,
                    clientShutdown, serverCommProblems, clientCommProblems, …}"
}
```

B.16 DATEX.MESSAGE_AuthenticationInformation_text-general255

```
datex-AuthenticationInfo-txt  DATA-ELEMENT ::= {
DESCRIPTIVE-NAME    "DATEX.MESSAGE_AuthenticationInformation_text-general255"
CONTEXT             "ITS"
DEFINITION          "Authentication information which the two interconnected systems have
                    agreed to include within each message. This can be used to exchange a
                    fixed code or algorithmic code in order to provide additional password
                    protection. This may be zero length octet string, a fixed octet
                    string, or a variable octet string according to rules defined in the
                    Interchange Agreement (e.g., the parameters defined in Clause 5)."
CLASS               "Communication Networks"
CLASS SCHEME        "ITS Classification Scheme"
SCHEME VERSION      "980201"
KEYWORDS            "Authentication, DATEX"
VALUE DOMAIN        "Text-OctetString 255"
DATA TYPE           OCTET STRING (SIZE (0..255))
VALUE RULE          "OCTET STRING (SIZE (0..255))"
}
```

B.17 DATEX.MESSAGE_Crc_id-crc16

```
datex-Crc-id        DATA-ELEMENT ::= {
DESCRIPTIVE-NAME    "DATEX.MESSAGE_Crc_id-crc16"
CONTEXT             "ITS"
DEFINITION          "The reletivaly unique code for the message byte stream which is used
                    for error checking. The value of the code is determined by applying
                    the CRC16 algorithm defined in ISO 3309 on the byte stream encoding of
                    datex-Data-txt. For example, if BER encoding was used, the CRC would
                    be calculated on the identifier, length, and contents octets of the
                    encoding."
CLASS               "Communication Networks"
CLASS SCHEME        "ITS Classification Scheme"
SCHEME VERSION      "980201"
KEYWORDS            "CRC, DATEX"
VALUE DOMAIN  "Id-Crc16"
DATA TYPE           OCTET STRING (SIZE (2))
VALUE RULE          "OCTET STRING (SIZE (2))"
}
```

B.18 DATEX.MESSAGE_Data_text-general

```
datex-Data-txt      DATA-ELEMENT ::= {
DESCRIPTIVE-NAME    "DATEX.MESSAGE_Data_text-general"
CONTEXT             "ITS"
DEFINITION          "The optionally encrypted contents of the DATEX Data Packet as defined
                    by the structure C2CAuthenticatedMessage."
CLASS               "Communication Networks"
CLASS SCHEME        "ITS Classification Scheme"
SCHEME VERSION      "980201"
VALUE DOMAIN        "Text-OctetString Unlimited"
DATA TYPE           OCTET STRING
VALUE RULE          "OCTET STRING"
CONSTRAINTS         "The initial layout for this octet string is given by
                    C2CAuthenticatedMessage; however, this data may be encrypted prior to
                    transmission over the communications media."
}
```

B.19 DATEX.MESSAGE_DataPacket_number-ulong

```
datex-DataPacket-nbr  DATA-ELEMENT ::= {
DESCRIPTIVE-NAME    "DATEX.MESSAGE_DataPacket_number-ulong"
CONTEXT             "ITS"
DEFINITION          "Indicates the data packet number for the data packet being sent. The
                    first data packet sent during session establishment shall have a
                    number of zero. Each subsequent data packet shall have its number
                    incremented by one. The client and server shall maintain separate
                    counters such that the first data packet sent by the server and the
                    first data packet sent by the client shall both be zero."
CLASS               "Communication Networks"
CLASS SCHEME        "ITS Classification Scheme"
SCHEME VERSION      "980201"
VALUE DOMAIN        "Number-ULONG"
DATA TYPE           INTEGER (0..4294967295)
VALUE RULE          "INTEGER (0..4294967295)"
}
```

B.20 DATEX.MESSAGE_DataPacketPriority_code-datex11

```
datex-DataPacketPriority-cd   DATA-ELEMENT ::= {
```

```
DESCRIPTIVE-NAME    "DATEX.MESSAGE_DataPacketPriority_code-datex11"
CONTEXT             "ITS"
DEFINITION          "The priority of the message. A system must process messages in the
                    order in which they are received, unless a later message has a higher
                    priority. Messages of higher priority may be processed before messages
                    of lower priority at the option of the implementation."
CLASS               "Communication Networks"
CLASS SCHEME        "ITS Classification Scheme"
SCHEME VERSION      "980201"
VALUE DOMAIN        "Code-DATEX Priority"
DATA TYPE           INTEGER (1..10)
VALUE RULE          " INTEGER (1..10)"
}
```

B.21 DATEX.MESSAGE_DataPacketTime_frame

```
datex-DataPacketTime  DATA-ELEMENT ::= {
DESCRIPTIVE-NAME    "DATEX.MESSAGE_DataPacketTime_frame"
CONTEXT             "ITS"
DEFINITION          "The time at which the data packet is being generated."
CLASS               "Communication Networks"
CLASS SCHEME        "ITS Classification Scheme"
SCHEME VERSION      "980201"
VALUE DOMAIN        "Time data structure per A.8"
DATA TYPE           Time
VALUE RULE          "Time data structure per A.8"
}
```

B.22 DATEX.MESSAGE_Destination_text-name

```
datex-Destination-txt DATA-ELEMENT ::= {
DESCRIPTIVE-NAME    "DATEX.MESSAGE_Destination_text-name"
CONTEXT             "ITS"
DEFINITION          "The domain name of the system which is supposed to receive the
                    message."
CLASS               "Communication Networks"
CLASS SCHEME        "ITS Classification Scheme"
SCHEME VERSION      "980201"
KEYWORDS            "DATEX, Destination"
VALUE DOMAIN        "Text-Name"
DATA TYPE           UTF8String (SIZE (0..40))
VALUE RULE          "UTF8String (SIZE (0..40))"
}
```

B.23 DATEX.MESSAGE_DestinationAddress_location-address

```
datex-DestinationAddress-loc  DATA-ELEMENT ::= {
DESCRIPTIVE-NAME    "DATEX.MESSAGE_DestinationAddress_location-address"
CONTEXT             "ITS"
DEFINITION          "A unique address for the computer which is the intended recipient of
                    the message."
CLASS               "Communication Networks"
CLASS SCHEME        "ITS Classification Scheme"
SCHEME VERSION      "980201"
VALUE DOMAIN        "Location-Address"
DATA TYPE           OCTET STRING
VALUE RULE          "AddressCode ::= CHOICE
                    {
                          gis                   IMPLICIT [1] NumericString,
                          mhORName              IMPLICIT [2] MhORName, --X.400
```

```
                        dn                IMPLICIT [3] DistinguishedName, -- X.500
                        isdnOrPhonenumber         IMPLICIT [4] E164Form,
                        rfc822Address             IMPLICIT [5] PrintableString,
                        pstnAddress               IMPLICIT [6] NumericString
                  }"
}
```

B.24 DATEX.MESSAGE_Origin_text-name

```
datex-Origin-txt    DATA-ELEMENT ::= {
DESCRIPTIVE-NAME    "DATEX.MESSAGE_Origin_text-name"
CONTEXT             "ITS"
DEFINITION          "The domain name of the system that collected the data contained in
                    the end application message."
CLASS               "Communication Networks"
CLASS SCHEME        "ITS Classification Scheme"
SCHEME VERSION      "980201"
VALUE DOMAIN        "Text-Name"
DATA TYPE           UTF8String (SIZE (0..40))
VALUE RULE          "UTF8String (SIZE (0..40))"
}
```

B.25 DATEX.MESSAGE_OriginAddress_location-address

```
datex-OriginAddress-loc   DATA-ELEMENT ::= {
DESCRIPTIVE-NAME    "DATEX.MESSAGE_OriginAddress_location-address"
CONTEXT             "ITS"
DEFINITION          "A unique address of the system that collected the data contained in
                    the end-application."
CLASS               "Communication Networks"
CLASS SCHEME        "ITS Classification Scheme"
SCHEME VERSION      "980201"
VALUE DOMAIN        "Location-Address"
DATA TYPE           OCTET STRING
VALUE RULE          "AddressCode ::= CHOICE
                    {
                        gis                   IMPLICIT [1] NumericString,
                        mhORName              IMPLICIT [2] MhORName, --X.400
                        dn                IMPLICIT [3] DistinguishedName, -- X.500
                        isdnOrPhonenumber         IMPLICIT [4] E164Form,
                        rfc822Address             IMPLICIT [5] PrintableString,
                        pstnAddress               IMPLICIT [6] NumericString
                    }"
}
```

B.26 DATEX.MESSAGE_Sender_text-name

```
datex-Sender-txt    DATA-ELEMENT ::= {
DESCRIPTIVE-NAME    "DATEX.MESSAGE_Sender_text-name"
CONTEXT             "ITS"
DEFINITION          "The domain name of the system which is sending the message."
CLASS               "Communication Networks"
CLASS SCHEME        "ITS Classification Scheme"
SCHEME VERSION      "980201"
VALUE DOMAIN        "Text-Name"
DATA TYPE           UTF8String (SIZE (0..40))
VALUE RULE          "UTF8String (SIZE (0..40))"
}
```

B.27 DATEX.MESSAGE_SenderAddress_location-address

```
datex-SenderAddress-loc   DATA-ELEMENT ::= {
DESCRIPTIVE-NAME     "DATEX.MESSAGE_SenderAddress_location-address"
CONTEXT              "ITS"
DEFINITION           "A unique address for the computer which is sending the message."
CLASS                "Communication Networks"
CLASS SCHEME         "ITS Classification Scheme"
SCHEME VERSION       "980201"
VALUE DOMAIN         "Location-Address"
DATA TYPE            OCTET STRING
VALUE RULE           "AddressCode ::= CHOICE
                     {
                           gis                    IMPLICIT [1] NumericString,
                           mhORName               IMPLICIT [2] MhORName, --X.400
                           dn                 IMPLICIT [3] DistinguishedName, -- X.500
                           isdnOrPhonenumber          IMPLICIT [4] E164Form,
                           rfc822Address              IMPLICIT [5] PrintableString,
                           pstnAddress                IMPLICIT [6] NumericString
                     }"
}
```

B.28 DATEX.MESSAGE_Version_code-datex15

```
datex-Version-cd     DATA-ELEMENT ::= {
DESCRIPTIVE-NAME     "DATEX.MESSAGE_Version_code-datex15"
CONTEXT              "ITS"
DEFINITION           "This is the DATEX-ASN version to which this message conforms"
CLASS                "Communication Networks"
CLASS SCHEME         "ITS Classification Scheme"
SCHEME VERSION       "980201"
VALUE DOMAIN         "Code-DATEX Version"
DATA TYPE            ENUMERATED { experimental, version-1, …}
VALUE RULE           "ENUMERATED { experimental, version-1, …}"
}
```

B.29 DATEX.PUBLISH_FileName_text-memo

```
datexPublish-FileName-txt DATA-ELEMENT ::= {
DESCRIPTIVE-NAME     "DATEX.PUBLISH_FileName_text-memo"
CONTEXT              "ITS"
DEFINITION           "The name of the file, including the path, which contains the SEQUENCE
                     OF PublicationData that the server is attempting to publish. It is
                     recommended that file names be sequentially numbered to ensure a high
                     degree of uniqueness."
CLASS                "Communication Networks"
CLASS SCHEME         "ITS Classification Scheme"
SCHEME VERSION       "980201"
VALUE DOMAIN         "Text-Memo"
DATA TYPE            UTF8String (SIZE (0..2000))
VALUE RULE           "UTF8String (SIZE (0..2000))"
}
```

B.30 DATEX.PUBLISH_Guaranteed_boolean

```
datexPublish-Guaranteed-bool  DATA-ELEMENT ::= {
DESCRIPTIVE-NAME     "DATEX.PUBLISH_Guaranteed_boolean"
CONTEXT              "ITS"
```

```
DEFINITION          "Indicates whether or not the client receiving the publication (reply)
                    is required to acknowledge receipt of the data packet by issuing an
                    'Accept' data packet. This shall be set to 'true' if
                    DATEX.SUBSCRIPTION_Guarantee_bool was set to 'true' in the associated
                    subscription (request)."
CLASS               "Communication Networks"
CLASS SCHEME        "ITS Classification Scheme"
SCHEME VERSION      "980201"
VALUE DOMAIN        "Boolean"
DATA TYPE           BOOLEAN
VALUE RULE          "BOOLEAN"
}
```

B.31 DATEX.PUBLISH_LatePublicationFlag_boolean

```
datexPublish-LatePublicationFlag-boolDATA-ELEMENT ::= {
DESCRIPTIVE-NAME    "DATEX.PUBLISH_LatePublicationFlag_boolean"
CONTEXT             "ITS"
DEFINITION          "Indicates whether the publication is being sent after the expiration
                    of the UpdateDelay timer."
CLASS               "Communication Networks"
CLASS SCHEME        "ITS Classification Scheme"
SCHEME VERSION      "980201"
VALUE DOMAIN        "Boolean"
DATA TYPE           BOOLEAN
VALUE RULE          "BOOLEAN"
}
```

B.32 DATEX.PUBLISH_Management_code-datex3

```
datexPublish_Management_cd    DATA-ELEMENT ::= {
DESCRIPTIVE-NAME    "DATEX.PUBLISH_Management_code-datex3"
CONTEXT             "ITS"
DEFINITION          "Indicates the status of the publication feature per one of the
                    defined enumerated values."
CLASS               "Communication Networks"
CLASS SCHEME        "ITS Classification Scheme"
SCHEME VERSION      "980201"
VALUE DOMAIN        "Code-DATEX Publication Type"
DATA TYPE           ENUMERATED { temporarilySuspended, resume, terminate-other, terminate-
                    dataNoLongerAvailable, terminate-publicationsBeingRejected, terminate-
                    PendingShutdown, terminate-processingMgmt, terminate-bandwidthMgmt,
                    terminate-accessDenied, unknownRequest, . . .}
VALUE RULE          " ENUMERATED {temporarilySuspended, resume, terminate-other,
                    terminate-dataNoLongerAvailable, terminate-publicationsBeingRejected,
                    terminate-PendingShutdown, terminate-processingMgmt, terminate-
                    bandwidthMgmt, terminate-accessDenied, unknownRequest, . . .}"
}
```

B.33 DATEX.PUBLISH_Serial_number-ulong

```
datexPublish-Serial-nbr   DATA-ELEMENT ::= {
DESCRIPTIVE-NAME    "DATEX.PUBLISH_Serial_number-ulong"
CONTEXT             "ITS"
DEFINITION          "Indicates how many responses have been published to the associated
                    request, including this publication (reply). The first publication to
                    a given response shall receive a Serial of 1 and each subsequent
                    publication to the same subscription (request) shall have a Serial
                    number which is incremented by one."
CLASS               "Communication Networks"
CLASS SCHEME        "ITS Classification Scheme"
SCHEME VERSION      "980201"
```

```
VALUE DOMAIN        "Number-ULONG"
DATA TYPE           INTEGER (0..4294967295)
VALUE RULE          "INTEGER (0..4294967295)"
CONSTRAINTS         "The value 0 is reserved."
}
```

B.34 DATEX.PUBLISH_SubscribeSerial_number-ulong

```
datexPublish-SubscribeSerial-nbr DATA-ELEMENT ::= {
DESCRIPTIVE-NAME    "DATEX.PUBLISH_SubscribeSerial_number-ulong"
CONTEXT             "ITS"
DEFINITION          "The serial number of the subscription (request) which caused the
                    generation of the publication (reply)."
CLASS               "Communication Networks"
CLASS SCHEME        "ITS Classification Scheme"
SCHEME VERSION      "980201"
VALUE DOMAIN        "Number-ULONG"
DATA TYPE           INTEGER (0..4294967295)
VALUE RULE          "INTEGER (0..4294967295)"
CONSTRAINTS         "The value of zero (0) shall be reserved for unrequested emergency
                    publications as agreed to by the centres."
}
```

B.35 DATEX.REGISTERED_DaysOfWeek_code-DaysOfWeek

```
datexRegistered-DaysOfWeek-cd DATA-ELEMENT ::= {
DESCRIPTIVE-NAME    "DATEX.REGISTERED_DaysOfWeek_code-DaysOfWeek"
CONTEXT             "ITS"
DEFINITION          "The days of week on which the subscription (request) is activated on
                    the server. "
CLASS               "Communication Networks"
CLASS SCHEME        "ITS Classification Scheme"
SCHEME VERSION      "980201"
VALUE DOMAIN        "Code-Days Of Week"
DATA TYPE           BIT STRING {other, sunday, monday, tuesday, wednesday, thursday,
                    friday, saturday} (SIZE (8))
VALUE RULE          "BIT STRING {other, sunday, monday, tuesday, wednesday, thursday,
                    friday, saturday} (SIZE (8))"
}
```

B.36 DATEX.REGISTERED_Duration_quantity-ushort

```
datexRegistered-Duration-qty  DATA-ELEMENT ::= {
DESCRIPTIVE-NAME    "DATEX.REGISTERED_Duration_quantity-ushort"
CONTEXT             "ITS"
DEFINITION          "The duration, in minutes, for which the subscription (request)
                    remains active after the Start Time. "
CLASS               "Communication Networks"
CLASS SCHEME        "ITS Classification Scheme"
SCHEME VERSION      "980201"
VALUE DOMAIN        "Qty-USHORT"
DATA TYPE           INTEGER (0..65535)
VALUE RULE          "INTEGER (0..65535)"

}
```

B.37 DATEX.REGISTERED_EndDate_frame

```
datexRegistered-EndDate   DATA-ELEMENT ::= {
DESCRIPTIVE-NAME    "DATEX.REGISTERED_EndDate_frame"
```

```
CONTEXT             "ITS"
DEFINITION          "The last date on which the subscription (request) may be activated."
CLASS               "Communication Networks"
CLASS SCHEME        "ITS Classification Scheme"
SCHEME VERSION      "980201"
VALUE DOMAIN        "Time data structure per A.8"
DATA TYPE           Time
VALUE RULE          "Time data structure per A.8"

}
```

B.38 DATEX.REGISTERED_EndTime_frame

```
datexRegistered-EndTime   DATA-ELEMENT ::= {
DESCRIPTIVE-NAME    "DATEX.REGISTERED_EndTime_frame"
CONTEXT             "ITS"
DEFINITION          "The time of day at which the subscription (request) is deactivated. "
CLASS               "Communication Networks"
CLASS SCHEME        "ITS Classification Scheme"
SCHEME VERSION      "980201"
VALUE DOMAIN        "Time data structure per A.8"
DATA TYPE           Time
VALUE RULE          "Time data structure per A.8"

}
```

B.39 DATEX.REGISTERED_StartDate_frame

```
datexRegistered-StartDate DATA-ELEMENT ::= {
DESCRIPTIVE-NAME    "DATEX.REGISTERED_StartDate_frame"
CONTEXT             "ITS"
DEFINITION          "The first date on which the subscription (request) may be activated."
CLASS               "Communication Networks"
CLASS SCHEME        "ITS Classification Scheme"
SCHEME VERSION      "980201"
VALUE DOMAIN        "Time data structure per A.8"
DATA TYPE           Time
VALUE RULE          "Time data structure per A.8"

}
```

B.40 DATEX.REGISTERED_StartTime_frame

```
datexRegistered-StartTime DATA-ELEMENT ::= {
DESCRIPTIVE-NAME    "DATEX.REGISTERED_StartTime_frame"
CONTEXT             "ITS"
DEFINITION          "The time of day at which the subscription (request) is activated, on
                    those days which are valid according to the StartDate, EndDate and
                    DaysOfWeek fields. If the date on which this subscription is received
                    is a valid day and the start time has already passed but the stop time
                    has not passed, the subscription shall immediately be activated. The
                    value 00:00 shall mean the start of the day."
CLASS               "Communication Networks"
CLASS SCHEME        "ITS Classification Scheme"
SCHEME VERSION      "980201"
VALUE DOMAIN        "Time data structure per A.8"
DATA TYPE           Time
VALUE RULE          "Time data structure per A.8"

}
```

B.41 DATEX.REGISTERED_UpdateDelay_quantity-ulong

```
datexRegistered-UpdateDelay-qty  DATA-ELEMENT ::= {
DESCRIPTIVE-NAME     "DATEX.REGISTERED_UpdateDelay_quantity-ulong"
CONTEXT              "ITS"
DEFINITION           "The update interval for the subscription (request). If the mode field
                     of the SubscriptionData structure is 'periodic', a new publication
                     (reply) shall be produced every DATEX.REGISTERED_UpdateDelay_quantity-
                     ulong seconds. If the mode field of the SubscriptionData structure is
                     'event-driven', a new publication data packet shall be generated
                     within DATEX.REGISTERED_UpdateDelay_quantity seconds from the time
                     that a new record or record update occurs, respectively, as defined by
                     the message referenced in the message field of the same
                     SubscriptionData structure; in this case, the value denotes a maximum
                     latency."
CLASS                "Communication Networks"
CLASS SCHEME         "ITS Classification Scheme"
SCHEME VERSION       "980201"
VALUE DOMAIN         "Qty-ULONG"
DATA TYPE            INTEGER (0..4294967295)
VALUE RULE           "INTEGER (0..4294967295)"
}
```

B.42 DATEX.REJECT_Login_code-datex6

```
datexReject-Login-cd  DATA-ELEMENT ::= {
DESCRIPTIVE-NAME     "DATEX.REJECT_Login_code-datex6 "
CONTEXT              "ITS"
DEFINITION           "The reason that the Login packet is being rejected."
CLASS                "Communication Networks"
CLASS SCHEME         "ITS Classification Scheme"
SCHEME VERSION       "980201"
VALUE DOMAIN         "Code-DATEX Reject Login"
DATA TYPE            ENUMERATED {other, unknownDomainName, accessDenied,
                     invalidNamePassword, timeoutTooSmall, timeoutTooLarge,
                     heartbeatTooSmall, heartbeatTooLarge, sessionExists,
                     maxSessionsReached, …}
VALUE RULE           "other - used for any other reason
                     unknownDomainName - the Client or Server domain name in the Login was
                     unknown or invalid
                     accessDenied - the Server is denying access for some reason
                     invalidNamePassword – the Server is denying access due to an invalid
                     name password pair.
                     timeoutTooSmall (TooLarge) - the timeout value in the Login was not
                     within a range that the Server supports
                     heartbeatTooSmall (TooLarge) - the heartbeat in the Login was not
                     within a range that the Server supports
                     sessionExists - a session already exists between the indicated Client
                     domain name and Server domain name over the specified transport
                     profile; only one session is allowed between the pair over the same
                     profile.
                     maxSessionsReached - the Server can not support any more sessions."
}
```

B.43 DATEX.REJECT_Packet_number-ulong

```
datexReject-Packet-nbrDATA-ELEMENT ::= {
DESCRIPTIVE-NAME     "DATEX.REJECT_Packet_number-ulong"
CONTEXT              "ITS"
DEFINITION           "The DATEX.MESSAGE_DataPacket_number-ulong, as defined in A.1, of the
                     request which is being rejected. For the rejection of specific
                     PublicationData structures within a Publication data packet, this
                     number shall be set to zero. "
```

```
CLASS               "Communication Networks"
CLASS SCHEME        "ITS Classification Scheme"
SCHEME VERSION      "980201"
VALUE DOMAIN        "Number-ULONG"
DATA TYPE           INTEGER (0..4294967295)
VALUE RULE          "INTEGER (0..4294967295)"
}
```

B.44 DATEX.REJECT_Publication_code-datex7

```
datexReject-Publication-cd    DATA-ELEMENT ::= {
DESCRIPTIVE-NAME    "DATEX.REJECT_Publication_code-datex7"
CONTEXT             "ITS"
DEFINITION          "A reason why the Publication (reply) data packet was rejected."
CLASS               "Communication Networks"
CLASS SCHEME        "ITS Classification Scheme"
SCHEME VERSION      "980201"
VALUE DOMAIN        "Code-DATEX Reject Publication"
DATA TYPE           ENUMERATED {other, invalidPublishFormat, …}
VALUE RULE          " other - used for any other error
                    invalidPublishFormat – used if the selected publication format is
                    invalid "
}
```

B.45 DATEX.REJECT_PublicationData_code-datex16

```
datexReject-PublicationData-cd    DATA-ELEMENT ::= {
DESCRIPTIVE-NAME    "DATEX.REJECT_PublicationData_code-datex16"
CONTEXT             "ITS"
DEFINITION          "A reason why the PublicationData structure was rejected."
CLASS               "Communication Networks"
CLASS SCHEME        "ITS Classification Scheme"
SCHEME VERSION      "980201"
VALUE DOMAIN        "Code-DATEX Reject Publication Data"
DATA TYPE           ENUMERATED {other, unknownSubscription, unknownPublicationNbr,
                    unknownPublicationMsgId, invalidPublicationMsgId,
                    invalidPublicationMsgContent, repeatedPublicationNbr, …}
                    VALUE RULE    " other - used for any other error
                    unknownSubscription - used if the subscription (request) is not
                    recognized
                    unknownPublicationNbr - used if the publication is not recognized
                    unknownPublicationMsgId – used if the publication message
                             identification number is
                    not recognized
                    invalidPublicationMsgId - used if the publication message
                             identification number is
                    recognized, but is an invalid message identification number
                    invalidPublicationMsgContent - the content of the publication message
                    is recognized,
                    but the content is invalid
                    repeatedPublicationNbr – a publication with this publication number
                    has already been
                    received for this subscription."
}
```

B.46 DATEX.REJECT_PublicationSerial_number-ulong

```
datexReject-PublicationSerial-nbr DATA-ELEMENT ::= {
DESCRIPTIVE-NAME    "DATEX.REJECT_PublicationSerial_number-ulong"
CONTEXT             "ITS"
DEFINITION          "The publication serial number of the PublicationData structure that
                    contained invalid data and is being rejected."
```

```
CLASS                "Communication Networks"
CLASS SCHEME         "ITS Classification Scheme"
SCHEME VERSION       "980201"
VALUE DOMAIN         "Number-ULONG"
DATA TYPE            INTEGER (0..4294967295)
VALUE RULE           " INTEGER (0..4294967295)"
}
```

B.47 DATEX.REJECT_Subscription_code-datex8

```
datexReject-Subscription-cd   DATA-ELEMENT ::= {
DESCRIPTIVE-NAME     "DATEX.REJECT_Subscription_code-datex8"
CONTEXT              "ITS"
DEFINITION           "A reason why the Subscription (request) was rejected."
CLASS                "Communication Networks"
CLASS SCHEME         "ITS Classification Scheme"
SCHEME VERSION       "980201"
VALUE DOMAIN         "Code-DATEX Reject Subscription"
DATA TYPE            ENUMERATED {other, unknownSubscriptionNbr, invalidTimes,
                     frequencyTooSmall, frequencyTooLarge, invalidMode,
                     publishFormatNotSupported, unknownSubscriptionMsgId,
                     invalidSubscriptionMsgId, invalidSubscriptionContent, …}
VALUE RULE           "other - used for any other error
                     unknownSubscriptionNbr - used if the subscription (request) is known
                              to be too big to
                     fit in a requested datagram
                     invalidTimes - used if the time of either the Request or event message
                     is not
                     recognized by the Server
                     frequencyTooSmall - used if the frequency is too small for one of the
                     fields in the
                     request or event message within the subscription
                     frequencyTooLarge - - used if the frequency is too large for one of
                     the fields in the
                     request or event message within the subscription
                     invalidMode – the subscrption contained an invalid mode.
                     publishFormatNotSupported – the publish format requested within the
                     subscription
                     is not supported
                     unknownSubscriptionMsgId – the subscription message identification is
                     unknown
                     invalidSubscriptionMsgId – the subscription message identification is
                     invalid
                     invalidSubscriptionContent – the content of the subscription message
                              is
                     invalid"
}
```

B.48 DATEX.REJECT_SubscriptionSerial_number-ulong

```
datexReject-SubscriptionSerial-nbr   DATA-ELEMENT ::= {
DESCRIPTIVE-NAME     "DATEX.REJECT_SubscriptionSerial_number-ulong"
CONTEXT              "ITS"
DEFINITION           "The subscription serial number of the PublicationData structure that
                     contained invalid data and is being rejected."
CLASS                "Communication Networks"
CLASS SCHEME         "ITS Classification Scheme"
SCHEME VERSION       "980201"
VALUE DOMAIN         "Number-ULONG"
DATA TYPE            INTEGER (0..4294967295)
VALUE RULE           " INTEGER (0..4294967295)"
}
```

B.49 DATEX.SUBSCRIBE_CancelReason_code-datex5

```
datexSubscribe-CancelReason-cd   DATA-ELEMENT ::= {
DESCRIPTIVE-NAME     "DATEX.SUBSCRIBE_CancelReason_code-datex5"
CONTEXT              "ITS"
DEFINITION           "The reason that the subscription (request) is being cancelled. "
CLASS                "Communication Networks"
CLASS SCHEME         "ITS Classification Scheme"
SCHEME VERSION       "980201"
VALUE DOMAIN         "Code-DATEX Cancel Subscription"
DATA TYPE            ENUMERATED {other, dataNotNeeded, errorsInPublication, pendingLogout,
                     processingMgmt, bandwidthMgmt, ...}
VALUE RULE           "other - not one of the standard reasons listed in this standard
                     dataNotNeeded - indicates that the Client no longer requires the data
                     errorsInPublication - indicates that the Client is canceling the
                     subscription (request) due to
                     excessive publications that could not be decoded or contained invalid
                     data.
                     pendingLogout - indicates that the Client is in the process of
                     gracefully terminating the
                     session
                     processingMgmt - indicates that the Client is cancelling the
                     subscription so that it
                     may concentrate its processing resources on other tasks
                     bandwidthMgmt - indicates that the Client is cancelling the
                     subscription so that it
                     may use the limited bandwidth for higher priority needs"
}
```

B.50 DATEX.SUBSCRIBE_Guarantee_boolean

```
datexSubscribe-Guarantee-bool DATA-ELEMENT ::= {
DESCRIPTIVE-NAME     "DATEX.SUBSCRIBE_Guarantee_boolean"
CONTEXT              "ITS"
DEFINITION           "Indicates whether or not the publication (reply) should be confirmed.
                     If the value is 'true', the server shall set
                     DATEX.PUBLISH_Guaranteed_boolean of the publication data packet(s) to
                     true; otherwise the indicated field shall be set to false."
CLASS                "Communication Networks"
CLASS SCHEME         "ITS Classification Scheme"
SCHEME VERSION       "980201"
VALUE DOMAIN         "Boolean"
DATA TYPE            BOOLEAN
VALUE RULE           "BOOLEAN"
}
```

B.51 DATEX.SUBSCRIBE_Persistent_boolean

```
datexSubscribe-Persistent-bool   DATA-ELEMENT ::= {
DESCRIPTIVE-NAME     "DATEX.SUBSCRIBE_Persistent_boolean"
CONTEXT              "ITS"
DEFINITION           " Indicates whether or not the subscription (request) will survive
                     session terminations."
CLASS                "Communication Networks"
CLASS SCHEME         "ITS Classification Scheme"
SCHEME VERSION       "980201"
VALUE DOMAIN         "Boolean"
DATA TYPE            BOOLEAN
VALUE RULE           "BOOLEAN"
}
```

B.52 DATEX.SUBSCRIBE_Priority_code-datex11

```
datexSubscribe-Priority-cd    DATA-ELEMENT ::= {
DESCRIPTIVE-NAME     "DATEX.SUBSCRIBE_Priority_code-datex11"
CONTEXT              "ITS"
DEFINITION           " Indicates the relative priority of the subscription. A priority of 1
                     is the highest priority and a priority of 10 is the lowest priority. "
CLASS                "Communication Networks"
CLASS SCHEME         "ITS Classification Scheme"
SCHEME VERSION       "980201"
VALUE DOMAIN         "Code-DATEX Priority"
DATA TYPE            INTEGER (1..10)
VALUE RULE           "INTEGER (1..10)"
}
```

B.53 DATEX.SUBSCRIBE_PublishFormat_code-datex4

```
datexSubscribe-PublishFormat-cd  DATA-ELEMENT ::= {
DESCRIPTIVE-NAME     "DATEX.SUBSCRIBE_PublishFormat_code-datex4"
CONTEXT              "ITS"
DEFINITION           "The format in which the corresponding publication(s) should be
                     exchanged. If the value is 'ftp' or 'tftp', the server shall store the
                     publication (reply) in a file and transmit the file name in the
                     'publication' data packet. If the value is 'dataPacket' the server
                     shall attempt to transmit the publication data within the
                     'publication' data packet."
CLASS                "Communication Networks"
CLASS SCHEME         "ITS Classification Scheme"
SCHEME VERSION       "980201"
VALUE DOMAIN         "Code-DATEX Publish Format"
DATA TYPE            ENUMERATED {other, ftp, tftp, dataPacket, …}
VALUE RULE           "ENUMERATED {other, ftp, tftp, dataPacket, …}"
}
```

B.54 DATEX.SUBSCRIBE_Serial_number-ulong

```
datexSubscribe-Serial-nbr DATA-ELEMENT ::= {
DESCRIPTIVE-NAME     "DATEX.SUBSCRIBE_Serial_number-ulong"
CONTEXT              "ITS"
DEFINITION           "The serial number for the subscription (request) being submitted.
                     Each subscription shall have an associated serial number; the number
                     may be used to update or cancel previously sent subscriptions . New
                     subscriptions shall not use a serial number which is already in use by
                     the Client/Server pair."
CLASS                "Communication Networks"
CLASS SCHEME         "ITS Classification Scheme"
SCHEME VERSION       "980201"
VALUE DOMAIN         "Number-ULONG"
DATA TYPE            INTEGER (0..4294967295)
VALUE RULE           "INTEGER (0..4294967295)"
CONSTRAINTS          "The value of zero shall not be used"
}
```

B.55 DATEX.SUBSCRIBE_Status_code-datex12

```
datexSubscribe-Status-cd  DATA-ELEMENT ::= {
DESCRIPTIVE-NAME     "DATEX.SUBSCRIBE_Status_code-datex12"
CONTEXT              "ITS"
DEFINITION           "An indication of the status of this subscription (request) message.
                     If the value is 'new', the request number shall be unique for the
                     client-server pair and shall cause a new subscription to be handled by
                     the Server. If the value is 'update', the server shall update the
```

```
                    previously stored request, i.e., the existing request with the same
                    serial number, according to the new information. The update feature
                    shall not be used to change the message id of the message field of the
                    SubscriptionData structure. If such a change is necessary, the
                    existing subscription shall be cancelled and a new subscription
                    submitted."
CLASS               "Communication Networks"
CLASS SCHEME        "ITS Classification Scheme"
SCHEME VERSION      "980201"
VALUE DOMAIN        "Code-DATEX Status"
DATA TYPE           ENUMERATED { new, update }
VALUE RULE          "ENUMERATED { new, update}"
}
```

B.56 DATEX.TERMINATE_Reason_code-datex14

```
datexTerminate-Reason-cd  DATA-ELEMENT ::= {
DESCRIPTIVE-NAME    "DATEX.TERMINATE_Reason_code-datex14"
CONTEXT             "ITS"
DEFINITION          "Indicates who initiated the termination of the session and a reason."
CLASS               "Communication Networks"
CLASS SCHEME        "ITS Classification Scheme"
SCHEME VERSION      "980201"
VALUE DOMAIN        "Code-DATEX Terminate"
DATA TYPE           ENUMERATED {other, serverRequested, clientRequested, serverShutdown,
                    clientShutdown, serverCommProblems, clientCommProblems, …}
VALUE RULE          "ENUMERATED {other, serverRequested, clientRequested, serverShutdown,
                    clientShutdown, serverCommProblems, clientCommProblems, …}"
}
```

B.57 DATEX.TRANSFER.DONE_FileName_text-memo

```
datexTransferDone-FileName-txt   DATA-ELEMENT ::= {
DESCRIPTIVE-NAME    "DATEX.TRANSFER.DONE_FileName_text-memo"
CONTEXT             "ITS"
DEFINITION          "The name of the file which the Client is no longer attempting to
                    retrieve."
CLASS               "Communication Networks"
CLASS SCHEME        "ITS Classification Scheme"
SCHEME VERSION      "980201"
VALUE DOMAIN        "Text-Memo"
DATA TYPE           UTF8String (SIZE (0..2000))
VALUE RULE          "UTF8String (SIZE (0..2000))"
}
```

B.58 DATEX.TRANSFER.DONE_Success_boolean

```
datexTransferDone-Success-bool   DATA-ELEMENT ::= {
DESCRIPTIVE-NAME    "DATEX.TRANSFER.DONE_Success_boolean"
CONTEXT             "ITS"
DEFINITION          "Indicates whether or not the file transfer was successful."
CLASS               "Communication Networks"
CLASS SCHEME        "ITS Classification Scheme"
SCHEME VERSION      "980201"
VALUE DOMAIN        "Boolean"
DATA TYPE           BOOLEAN
VALUE RULE          "BOOLEAN"
}
```

B.59 END.APPLICATION_Message_id

```
endApplication-Message-id DATA-ELEMENT ::= {
DESCRIPTIVE-NAME    "END.APPLICATION_Message_id"
CONTEXT             "ITS"
DEFINITION          "The object identifier associated with the end application message
                    that is contained within this data packet. The definition of end
                    application messages is outside the scope of this standard; however
                    ISO 14827-1 defines the generic structure within which end application
                    messages are defined."
CLASS               "Communication Networks"
CLASS SCHEME        "ITS Classification Scheme"
SCHEME VERSION      "980201"
VALUE DOMAIN        "As defined by the End Application message"
DATA TYPE           ISO14827-MESSAGE.&id
VALUE RULE          "As defined by the End Application message"
}
```

B.60 END.APPLICATION_Message_msg

```
endApplication-Message-msg   DATA-ELEMENT ::= {
DESCRIPTIVE-NAME    "END.APPLICATION_Message_msg"
CONTEXT             "ITS"
DEFINITION          "The encoded content of the end application message. The definition of
                    end application messages is outside the scope of this standard;
                    however ISO 14827-1 defines the generic structure within which end
                    application messages are defined."
CLASS               "Communication Networks"
CLASS SCHEME        "ITS Classification Scheme"
SCHEME VERSION      "980201"
VALUE DOMAIN        "As defined by the End Application message"
DATA TYPE           ISO14827-MESSAGE.&MessageBody
VALUE RULE          "As defined by the End Application message"
}
```

B.61 TIME_Centiseconds_quantity

```
time-Centiseconds-qty DATA-ELEMENT ::= {
DESCRIPTIVE-NAME    "TIME_Centiseconds_quantity"
CONTEXT             "ITS"
DEFINITION          "The hundredths of seconds of the time."
CLASS               "Communication Networks"
CLASS SCHEME        "ITS Classification Scheme"
SCHEME VERSION      "980201"
VALUE DOMAIN        "INTEGER (0..99)"
DATA TYPE           INTEGER (0..99)
VALUE RULE          "INTEGER (0..99)"
}
```

B.62 TIME_Day_quantity

```
time-Day-qty        DATA-ELEMENT ::= {
DESCRIPTIVE-NAME    "TIME_Day_quantity"
CONTEXT             "ITS"
DEFINITION          "The day of the month."
CLASS               "Communication Networks"
CLASS SCHEME        "ITS Classification Scheme"
SCHEME VERSION      "980201"
VALUE DOMAIN        "INTEGER (1..31)"
DATA TYPE           INTEGER (1..31)
VALUE RULE          "INTEGER (1..31)"
}
```

B.63 TIME_Deciseconds_quantity

```
time-Deciseconds-qty  DATA-ELEMENT ::= {
DESCRIPTIVE-NAME    "TIME_Deciseconds_quantity"
CONTEXT             "ITS"
DEFINITION          "The tenths of seconds of the time."
CLASS               "Communication Networks"
CLASS SCHEME        "ITS Classification Scheme"
SCHEME VERSION      "980201"
VALUE DOMAIN        "INTEGER (0..9)"
DATA TYPE           INTEGER (0..9)
VALUE RULE          "INTEGER (0..9)"
}
```

B.64 TIME_Hour_quantity

```
time-Hour-qty       DATA-ELEMENT ::= {
DESCRIPTIVE-NAME    "TIME_Hour_quantity"
CONTEXT             "ITS"
DEFINITION          "The hour units of the time."
CLASS               "Communication Networks"
CLASS SCHEME        "ITS Classification Scheme"
SCHEME VERSION      "980201"
VALUE DOMAIN        "INTEGER (0..23)"
DATA TYPE           INTEGER (0..23)
VALUE RULE          "INTEGER (0..23)"
}
```

B.65 TIME_Milliseconds_quantity

```
time-Milliseconds-qty DATA-ELEMENT ::= {
DESCRIPTIVE-NAME    "TIME_Milliseconds_quantity"
CONTEXT             "ITS"
DEFINITION          "The thousandths of seconds of the time."
CLASS               "Communication Networks"
CLASS SCHEME        "ITS Classification Scheme"
SCHEME VERSION      "980201"
VALUE DOMAIN        "INTEGER (0..999)"
DATA TYPE           INTEGER (0..999)
VALUE RULE          "INTEGER (0..999)"
}
```

B.66 TIME_Minute_quantity

```
time-Minute-qty     DATA-ELEMENT ::= {
DESCRIPTIVE-NAME    "TIME_Minute_quantity"
CONTEXT             "ITS"
DEFINITION          "The minute units of the time."
CLASS               "Communication Networks"
CLASS SCHEME        "ITS Classification Scheme"
SCHEME VERSION      "980201"
VALUE DOMAIN        "INTEGER (0..59)"
DATA TYPE           INTEGER (0..59)
VALUE RULE          "INTEGER (0..59)"
}
```

B.67 TIME_Month_quantity

```
time-Month-qty      DATA-ELEMENT ::= {
DESCRIPTIVE-NAME    "TIME_Month_quantity"
CONTEXT             "ITS"
DEFINITION          "The Month of the year."
CLASS               "Communication Networks"
CLASS SCHEME        "ITS Classification Scheme"
SCHEME VERSION      "980201"
VALUE DOMAIN        "INTEGER (1..12)"
DATA TYPE           INTEGER (1..12)
VALUE RULE          "INTEGER (1..12)"
}
```

B.68 TIME_Second_quantity

```
time-Second-qty     DATA-ELEMENT ::= {
DESCRIPTIVE-NAME    "TIME_Second_quantity"
CONTEXT             "ITS"
DEFINITION          "The full seconds of the time."
CLASS               "Communication Networks"
CLASS SCHEME        "ITS Classification Scheme"
SCHEME VERSION      "980201"
VALUE DOMAIN        "INTEGER (0..60)"
DATA TYPE           INTEGER (0..60)
VALUE RULE          "INTEGER (0..60)"
}
```

B.69 TIME_TimeZoneHour_quantity

```
time-TimeZoneHour-qty DATA-ELEMENT ::= {
DESCRIPTIVE-NAME    "TIME_TimeZoneHour_quantity"
CONTEXT             "ITS"
DEFINITION          "The number of hours offset from UTC for the local time."
CLASS               "Communication Networks"
CLASS SCHEME        "ITS Classification Scheme"
SCHEME VERSION      "980201"
VALUE DOMAIN        "INTEGER (-13..13)"
DATA TYPE           INTEGER (-13..13)
VALUE RULE          "INTEGER (-13..13)"
}
```

B.70 TIME_TimeZoneMinute_quantity

```
time-TimeZoneMinute-qty       DATA-ELEMENT ::= {
DESCRIPTIVE-NAME    "TIME_TimeZoneMinute_quantity"
CONTEXT             "ITS"
DEFINITION          "The minutes unit of the offset from UTC."
CLASS               "Communication Networks"
CLASS SCHEME        "ITS Classification Scheme"
SCHEME VERSION      "980201"
VALUE DOMAIN        "INTEGER (0..59)"
DATA TYPE           INTEGER (0..59)
VALUE RULE          "INTEGER (0..59)"
}
```

B.71 TIME_Year_quantity

```
time-Year-qty       DATA-ELEMENT ::= {
DESCRIPTIVE-NAME    "TIME_Year_quantity"
CONTEXT             "ITS"
DEFINITION          "The Year."
CLASS               "Communication Networks"
CLASS SCHEME        "ITS Classification Scheme"
SCHEME VERSION      "980201"
VALUE DOMAIN        "INTEGER (-32768..32767)"
DATA TYPE           INTEGER (-32768..32767)
VALUE RULE          "INTEGER (-32768..32767)"
}


END
```

Annex C
(normative)

Value domains

The value domains defined within this annex are defined using the following ASN.1 Information Object Specification:

```
VALUE-DOMAIN ::= CLASS {
&name           PrintableString (SIZE (0..255))
&namecontext       PrintableString (SIZE (0..40))
&definition     PrintableString (SIZE (0..65535))
&formula        PrintableString (SIZE (0..255))      OPTIONAL
&source     PrintableString (SIZE (0..255))      OPTIONAL
&keyword        PrintableString (SIZE (0..255))      OPTIONAL
&remarks        PrintableString (SIZE (0..2000))     OPTIONAL
&Type
&rule           PrintableString (SIZE (0..65535)) OPTIONAL
}
WITH SYNTAX {
NAME            &name
CONTEXT     &namecontext
DEFINITION      &definition
[FORMULA        &formula]
[SOURCE     &source]
[KEYWORDS       &keyword]
[REMARKS        &remarks]
DATA TYPE       &Type
[VALID VALUE RULE  &rule]
}
```

The fields are defined to conform with those specified in IEEE P1489-1999.

C.1 Boolean

```
boolean                 VALUE-DOMAIN ::= {
  NAME                  "Boolean"
  CONTEXT               "ITS"
  DEFINITION            "Indicates either 'true' or 'false'."
  FORMULA
  SOURCE                "ISO 8824-1"
  KEYWORDS
  REMARKS               "In PER (ISO 8825-2), this is single bit where False is zero
                        and True is one. Other encoding schemes may have different
                        representations, such as any non-zero value
                        being True."
  TYPE                  BOOLEAN
  VALID RULE            "BOOLEAN"
  }
```

C.2 Code-Currency

```
code-datex1             VALUE-DOMAIN ::= {
  NAME                  "Code-Currency"
```

```
CONTEXT             "ITS"
DEFINITION          "Indicates the three letter currency code"
FORMULA
SOURCE
KEYWORDS
REMARKS
TYPE                OCTET STRING (SIZE (3))
VALID RULE          "ISO 4217"
}
```

C.3 Code-DATEX initiator

```
code-datex2         VALUE-DOMAIN ::= {
NAME                "Code-DATEX Initiator"
CONTEXT             "ITS"
DEFINITION          "Indicates the initiator of an action"
FORMULA
SOURCE
KEYWORDS
REMARKS
TYPE                ENUMERATED {serverInitiated, clientInitiated, …}
VALID RULE          "ENUMERATED {serverInitiated, clientInitiated, …}"
}
```

C.4 Code-DATEX publication type

```
code-datex3         VALUE-DOMAIN ::= {
NAME                "Code-DATEX Publication Type"
CONTEXT             "ITS"
DEFINITION          "Indicates the type of response being sent."
FORMULA
SOURCE
KEYWORDS
REMARKS
TYPE                ENUMERATED { temporarilySuspended, resume, terminate-other,
                          terminate-dataNoLongerAvailable, terminate-
                     publicationsBeingRejected,
                      terminate-PendingShutdown, terminate-processingMgmt,
                      terminate-bandwidthMgmt, terminate-accessDenied,
                      unknownRequest, …}
VALID RULE          "ENUMERATED { temporarilySuspended, resume, terminate-other,
                      terminate-dataNoLongerAvailable,
                      terminatepublicationsBeingRejected,
                       terminate-PendingShutdown, terminate-processingMgmt,
                      terminate-bandwidthMgmt, terminate-accessDenied,
                      unknownRequest, …}"
}
```

C.5 Code-DATEX publish format

```
code-datex4         VALUE-DOMAIN ::= {
NAME                "Code-DATEX Publish Format"
CONTEXT             "ITS"
DEFINITION          "Indicates the format in which to publish information"
FORMULA
SOURCE
KEYWORDS
REMARKS
TYPE                ENUMERATED { other, ftp, tftp, dataPacket, …}
VALID RULE          "ENUMERATED { other, ftp, tftp, dataPacket, …}"
}
```

C.6 Code-DATEX cancel subscription

```
code-datex5          VALUE-DOMAIN ::= {
NAME                 "Code-DATEX Cancel Subscription"
CONTEXT              "ITS"
DEFINITION           "Indicates a reason why the subscription was cancelled."
FORMULA
SOURCE
KEYWORDS
REMARKS               "This enumeration is an enhancement of those errors known in
                     SNMP systems."
TYPE                 ENUMERATED {other, dataNotNeeded, errorsInPublication
                     pendingLogout, processingMgmt, bandwidthMgmt, …}
VALID RULE           "other - not one of the standard reasons listed in this
                     standard
                     dataNotNeeded - indicates that the Client no longer requires
                     the data
                     errorsInPublication - indicates that the Client is canceling
                     the subscription (request) due to
                     excessive publications that could not be decoded or contained
                     invalid data.
                     pendingLogout - indicates that the Client is in the process
                     of gracefully terminating the
                     session
                     processingMgmt - indicates that the Client is cancelling the
                     subscription so that it
                     may concentrate its processing resources on other tasks
                     bandwidthMgmt - indicates that the Client is cancelling the
                     subscription so that it
                     may use the limited bandwidth for higher priority needs"
}
```

C.7 Code-DATEX reject login

```
code-datex6          VALUE-DOMAIN ::= {
NAME                 "Code-DATEX Reject Login"
CONTEXT              "ITS"
DEFINITION           "Indicates a reason why the Login data packet was rejected."
FORMULA
SOURCE
KEYWORDS
REMARKS              "This enumeration is an enhancement of those errors known in
                     SNMP systems."
TYPE                 ENUMERATED {other, unknownDomainName, accessDenied,
                     invalidNamePassword, timeoutTooSmall, timeoutTooLarge,
                     heartbeatTooSmall,
                     heartbeatTooLarge, sessionExists, maxSessionsReached, …}
VALID RULE           "other - used for any other error
                     unknownDomainName - the Client or Server domain name in the
                     Login was unknown or
                     invalid
                     accessDenied - the Server is denying access for some reason
                     invalidNamePassword – the Server is denying access due to an
                     invalid name password pair.
                     timeoutTooSmall (TooLarge) - the timeout value in the Login
                     was not within a range that
                     the Server supports
                     heartbeatTooSmall (TooLarge) - the heartbeat in the Login was
                     not within a range that the
                     Server supports
                     sessionExists - a session already exists between the
                     indicated Client domain name and
                     Server domain name over the specified transport profile;
                     only one session is allowed
                     between the pair over the same profile.
```

```
                    maxSessionsReached – the Server can not support any more
                    sessions.”
}
```

C.8 Code-DATEX reject publication

```
code-datex7         VALUE-DOMAIN ::= {
NAME                “Code-DATEX Reject Publication”
CONTEXT             “ITS”
DEFINITION          “Indicates a reason why the publication data packet was
                    rejected.”
FORMULA
SOURCE
KEYWORDS
REMARKS             “This enumeration is an enhancement of those errors known in
                    SNMP systems.”
TYPE                ENUMERATED {other, invalidPublishFormat, …}
VALID RULE          “other - used for any other error
                    invalidPublishFormat – used if the selected publication
                    format is invalid”
                    }
```

C.9 Code-DATEX reject publication data

```
code-datex16        VALUE-DOMAIN ::= {
NAME                “Code-DATEX Reject PublicationData”
CONTEXT             “ITS”
DEFINITION          “Indicates a reason why the publication data packet was
                    rejected.”
FORMULA
SOURCE
KEYWORDS
REMARKS             “This enumeration is an enhancement of those errors known in
                    SNMP systems.”
TYPE                ENUMERATED {other, unknownSubscription,
                    unknownPublicationNbr,
                    unknownPublicationMsgId, invalidPublicationMsgId,
                    invalidPublicationMsgContent, repeatedPublicationNbr, …}
VALID RULE          “other - used for any other error
                    unknownSubscription - used if the subscription (request) is
                    not recognized
                    unknownPublicationNbr - used if the publication is not
                    recognized
                    unknownPublicationMsgId – used if the publication message
                    identification number is
                    not recognized
                    invalidPublicationMsgId - used if the publication message
                    identification number is
                    recognized, but is an invalid message identification number
                    invalidPublicationMsgContent - the content of the publication
                    message is recognized,
                    but the content is invalid
                    repeatedPublicationNbr – a publication with this publication
                    number has already been
                    received for this subscription.”
}
```

C.10 Code-DATEX reject subscription

```
code-datex8         VALUE-DOMAIN ::= {
NAME                “Code-DATEX Reject Subscription”
CONTEXT             “ITS”
```

```
DEFINITION          "Indicates a reason why the subscription data packet was
                    rejected."
FORMULA
SOURCE
KEYWORDS
REMARKS             "This enumeration is an enhancement of those errors known in
                    SNMP systems."
TYPE                ENUMERATED {other, unknownSubscriptionNbr, invalidTimes,
                    frequencyTooSmall, frequencyTooLarge, invalidMode,
                    publishFormatNotSupported, unknownSubscriptionMsgId,
                    invalidSubscriptionMsgId, invalidSubscriptionContent, …}
VALID RULE          "other - used for any other error
                    unknownSubscriptionNbr - used if the subscription (request)
                    is known to be too big to
                    fit in a requested datagram
                    invalidTimes - used if the time of either the Request or
                    event message is not
                    recognized by the Server
                    frequencyTooSmall - used if the frequency is too small for
                    one of the fields in the
                    request or event message within the subscription
                    frequencyTooLarge - - used if the frequency is too large for
                    one of the fields in the
                    request or event message within the subscription
                    invalidMode – the subscrption contained an invalid mode.
                    publishFormatNotSupported – the publish format requested
                    within the subscription
                    is not supported
                    unknownSubscriptionMsgId – the subscription message
                    identification is unknown
                    invalidSubscriptionMsgId – the subscription message
                    identification is invalid
                    invalidSubscriptionContent – the content of the subscription
                    message is
                    invalid"
}
```

C.11 Code-DATEX priority

```
code-datex11        VALUE-DOMAIN ::= {
NAME                "Code-DATEX Priority"
CONTEXT             "ITS"
DEFINITION          "Indicates the priority of the message. A value of 1 is
                    considered the highest priority and a
                    value of 10 is considered to be the lowest priority."
FORMULA
SOURCE
KEYWORDS
REMARKS
TYPE                INTEGER (1..10)
VALID RULE          "INTEGER (1..10)"
}
```

C.12 Code-DATEX status

```
code-datex12        VALUE-DOMAIN ::= {
NAME                "Code-DATEX Status"
CONTEXT             "ITS"
DEFINITION          "Indicates the DATEX Operation being performed."
FORMULA
SOURCE
KEYWORDS
REMARKS
TYPE                ENUMERATED {new, update}
```

```
VALID RULE          "ENUMERATED {new, update}"
}
```

C.13 Code-DATEX terminate

```
code-datex14        VALUE-DOMAIN ::= {
NAME                "Code-DATEX Terminate"
CONTEXT             "ITS"
DEFINITION          "Indicates who is requesting the session to be terminated."
FORMULA
SOURCE
KEYWORDS
REMARKS
TYPE                ENUMERATED {other, serverRequested, clientRequested,
                    serverShutdown,
                    clientShutdown, serverCommProblems, clientCommProblems, …}
VALID RULE          "ENUMERATED {other, serverRequested, clientRequested,
                    serverShutdown,
                    clientShutdown, serverCommProblems, clientCommProblems, …}"
}
```

C.14 Code-DATEX version

```
code-datex15        VALUE-DOMAIN ::= {
NAME                "Code-Datex Version"
CONTEXT             "ITS"
DEFINITION          "Indicates to which DATEX-ASN version the message conforms"
FORMULA
SOURCE
KEYWORDS
REMARKS
TYPE                ENUMERATED {experimental, version-1, …}
VALID RULE          "ENUMERATED {experimental, version-1, …}"
}
```

C.15 Code-Days of week

```
code-DaysOfWeek     VALUE-DOMAIN ::= {
NAME                "Code-Days Of Week"
CONTEXT             "ITS"
DEFINITION          "A bit map indicating which days of the week are being
                    selected."
FORMULA
SOURCE
KEYWORDS
REMARKS             "This is consistent with the existing NTCIP TS 3.4 and TCIP
                    definitions"
TYPE                BIT STRING {other, sunday, monday, tuesday, wednesday,
                    thursday, friday, saturday} (SIZE (8))
VALID RULE          "BIT STRING {other, sunday, monday, tuesday, wednesday,
                    thursday, friday, saturday} (SIZE (8))"
}
```

C.16 Id-Crc16

```
id-crc16            VALUE-DOMAIN ::= {
NAME                "Id-Crc16"
CONTEXT             "ITS"
DEFINITION          "A 16-bit Cyclical Redundancy Check (CRC) value."
FORMULA
```

```
SOURCE              "ISO 3309"
KEYWORDS
REMARKS
TYPE                OCTET STRING (SIZE(2))
VALID RULE          "OCTET STRING(SIZE(2))"
}
```

C.17 Id-Object Identifier

```
id-oid              VALUE-DOMAIN ::= {
NAME                "Id-Object Identifier"
CONTEXT             "ITS"
DEFINITION          "Indicates the object identifier on the ISO global naming
                    tree."
FORMULA
SOURCE              "ISO 8824-1"
KEYWORDS
REMARKS             "The root of the ISO tree is defined in ISO 8824-1:1998.
                    NTCIP TS 3.2-1996 also
                    contains relevant tree information."
TYPE                OBJECT IDENTIFIER
VALID RULE          "OBJECT IDENTIFIER"
}
```

C.18 Id-Object identifiers

```
id-oids             VALUE-DOMAIN ::= {
NAME                "Id-Object Identifiers"
CONTEXT             "ITS"
DEFINITION          "Indicates a list of object identifiers on the ISO global
                    naming tree."
FORMULA
SOURCE              "ISO 8824-1"
KEYWORDS
REMARKS             "The root of the ISO tree is defined in ISO 8824-1:1995.
                    NTCIP TS 3.2-1996 also
                    contains relavant tree information."
TYPE                SEQUENCE OF OBJECT IDENTIFIER
VALID RULE          "SEQUENCE OF OBJECT IDENTIFIER"
}
```

C.19 Location-Address

```
location-address    VALUE-DOMAIN ::= {
NAME                "Location-Address"
CONTEXT             "ITS"
DEFINITION          "Indicates the address."
FORMULA
SOURCE
KEYWORDS
REMARKS
TYPE                OCTET STRING
VALID RULE          "AddressCode ::= CHOICE {
                     gis          IMPLICIT [1] NumericString,
                     mhORName         IMPLICIT [2] MhORName, --X.400
                     dn           IMPLICIT [3] DistinguishedName, -- X.500
                     isdnOrPhonenumber  IMPLICIT [4] E164Form,
                     rfc822Address      IMPLICIT [5] PrintableString,
                     pstnAddress    IMPLICIT [6] NumericString
                    }"
}
```

C.20 Number-ULONG

```
number-ulong        VALUE-DOMAIN ::= {
NAME                "Number-ULONG"
CONTEXT             "ITS"
DEFINITION          "A four-byte unsigned integral number (not used for
                    quantification)."
FORMULA
SOURCE
KEYWORDS
REMARKS
TYPE                INTEGER (0..4294967295)
VALID RULE          "INTEGER (0..4294967295)"
}
```

C.21 Qty-UBYTE

```
quantity-ubyte      VALUE-DOMAIN ::= {
NAME                "Qty-UBYTE"
CONTEXT             "ITS"
DEFINITION          "A one byte unsigned integer indicating a quantity."
FORMULA
SOURCE
KEYWORDS
REMARKS
TYPE                INTEGER (0..255)
VALID RULE          "INTEGER (0..255)"
}
```

C.22 Qty-ULONG

```
quantity-ulong      VALUE-DOMAIN ::= {
NAME                "Qty-ULONG"
CONTEXT             "ITS"
DEFINITION          "A four-byte unsigned integer indicating a quantity."
FORMULA
SOURCE
KEYWORDS
REMARKS
TYPE                INTEGER (0..4294967295)
VALID RULE          "INTEGER (0..4294967295)"
}
```

C.23 Qty-unlimited

```
quantity            VALUE-DOMAIN ::= {
NAME                "Qty-unlimited"
CONTEXT             "ITS"
DEFINITION          "An unsigned integer indicating a quantity."
FORMULA
SOURCE
KEYWORDS
REMARKS
TYPE                INTEGER
VALID RULE          "INTEGER"
}
```

C.24 Qty-USHORT

```
quantity-ushort     VALUE-DOMAIN ::= {
NAME                "Qty-USHORT"
CONTEXT             "ITS"
DEFINITION          "A two-byte unsigned integer indicating a quantity."
FORMULA
SOURCE
KEYWORDS
REMARKS
TYPE                INTEGER (0..65535)
VALID RULE          "INTEGER (0..65535)"
}
```

C.25 Text-Memo

```
text-memo           VALUE-DOMAIN ::= {
NAME                "Text-Memo"
CONTEXT             "ITS"
DEFINITION          "A text field of up to 2000 characters in length."
FORMULA
SOURCE
KEYWORDS
REMARKS
TYPE                UTF8String (SIZE (0..2000))
VALID RULE          "UTF8String (SIZE (0..2000))"
}
```

C.26 Text-Name

```
text-name           VALUE-DOMAIN ::= {
NAME                "Text-Name"
CONTEXT             "ITS"
DEFINITION          "A text field of up to 40 characters in length."
FORMULA
SOURCE
KEYWORDS
REMARKS
TYPE                UTF8String (SIZE (0..40))
VALID RULE          "UTF8String (SIZE (0..40))"
}
```

C.27 Text-OctetString 255

```
text-general255     VALUE-DOMAIN ::= {
NAME                "Text-OctetString 255"
CONTEXT             "ITS"
DEFINITION          "A text field of up to 255 characters in length."
FORMULA
SOURCE
KEYWORDS
REMARKS
TYPE                OCTET STRING (SIZE (0..255))
VALID RULE          "OCTET STRING (SIZE (0..255))"
}
```

C.28 Text-OctetString unlimited

```
text-general        VALUE-DOMAIN ::= {
NAME                "Text-OctetString Unlimited"
CONTEXT             "ITS"
DEFINITION          "A text field of unlimited characters in length."
FORMULA
SOURCE
KEYWORDS
REMARKS
TYPE                OCTET STRING
VALID RULE          "OCTET STRING"
}
```

Annex D
(normative)

DATEX-ASN over internet protocols

D.1 Internet implementation issues

The requirements in this annex only apply to implementations using either UDP or TCP for transport services and using port 355. The maximum datagram size defaults to 576 bytes, but may be changed via the login request. The encoding rules for the data packets used to establish a session (i.e. all data packets included in the procedure described by 6.3.1, are BER. All data packets exchanged after the establishment of a session shall use the encoding rules as negotiated during the session establishment procedure.

This part of ISO 14827 exchanges data over an application layer session. The session is maintained over either a connection, in the case of TCP, or a pseudo-connection, in the case of UDP. The methodology of using well-known port numbers to establish TCP connections is well documented in the Internet Standards and will not be discussed here. The methodology of using well-known port numbers during a UDP pseudo-connection is provided below.

In order to create a pseudo-connection over UDP, the station attempting to begin a session (i.e. a server sending an Initiate data packet or a client sending an unrequested login data packet) shall choose a random port number (PN) for itself. It then sends the initial data packet to the well-known port 355 from the PN port. The receiver of this message shall then select its own PN and issue the appropriate response; the destination port of this response shall indicate the PN selected by the initiator and the source port shall indicate the PN selected by the responder. This pair of PNs shall then be used for the remaining duration of the pseudo-connection.

As an example, the following shows the steps used for a server to establish a UDP session:

a) Server sends the client an “initiate” data packet with source port set to the server-selected PN and the destination port set to the well-known port 355.

b) Client responds with a “login” data packet with source port set to the client-selected PN and the destination port set to the server-selected PN.

c) Server responds with an “accept” data packet with source port set to the server-selected PN and the destination port set to the client-selected PN.

Annex E
(normative)

Protocol requirements list

E.1 General notations

This annex provides a Prototcol Requirements List (PRL) for implementation of this part of ISO 14827.

The PRL uses the following symbols to indicate the status of various features.

m	mandatory
m.<n>	support of every item of the group labeled by the same numeral <n> required, but only one is active at a time
o	optional
o.<n>	optional, but support of at least one of the group of options labeled by the same numeral <n> is required
c	conditional
-	non-applicable (i.e. logically impossible in the scope of the profile)
x	excluded or prohibited
I	out of scope of profile (left as an implementation choice)
d	deprecated (listed for compatibility with older systems)

The o.<n> notation is used to show a set of selectable options (i.e. one or more of the set must be implemented) with the same identifier <n>.

Two character combinations may be used for dynamic conformance requirements. In this case, the first character refers to the static (implementation) status, and the second refers to the dynamic (use); thus “mo” means “mandatory to be implemented, optional to be used.”

E.2 Conditional status notation

The following notation is used to indicate conditional requirements:

<index>:	This predicate symbol means that the status following it applies only when the PRL or PICS states that the features identified by the index are supported.

E.3 Basic requirements

Table E.1 indicates the basic requirements necessary to be in compliance with this part of ISO 14827.

Table E.1 — Basic requirements

Index	Topic	Clause	Status
client	Does the implementation claim to be a client?	6.3	o.1
server	Does the implementation claim to be a server?	6.3	o.1
1	Use of a single transport profile for all data packets within a single session?	6.1.2	m
port 355	Use UDP or TCP with port 355?	D.1	o
2	Use Port Number 355 and BER to intiate sessions?	D.1	port 355:m
3	Does the system properly handle time-out negotiation?	6.1.3	m
4	Does the system retransmit identical messages after a time-out occurs?	6.1.4	m
5	Does the system respond to a duplicate data packet with a newly created response data packet?	6.1.5	m
files	Does the system support file transfer?	6.2	o
ftp	Does the system support the FTP protocol?	6.2	files:o.2
tftp	Does the system support the TFTP protocol?	6.2	files:o.2
6	Does the file contain the “publication data” structure and nothing else?	6.2	files:m
dial-up	Does the implementation claim support for switched connections?	6.4	o

E.3.1 Server requirements

Table E.2 identifies the status of the various features for servers.

Table E.2 — Server requirements

Index	Topic	Clause	Status
7	Does the server reject a login request from client domain name with an existing session on the same transport profile?	6.3	m
8	Is the server able to generate an “initiate” request?	6.3.1	dial-up:m
9	Does the server accept a valid login request from a new client?	6.3.1	m
10	Does the server respond to a valid login with an “accept” message?	6.3.1	m
11	Does the server reject an invalid login request?	6.3.1	m
11.1	Does the server reject a login by transmitting a “reject” data packet?	6.3.1	o.3
11.2	Does the server silently reject an invalid login?	6.3.1	o.3
12	Does the server transmit a FrED response to a FrED with the “confirm message” number set to zero (0)?	6.3.2	m
13	Does the server terminate a session when no messages are received for a period exceeding the maximum heartbeat duration?	6.3.2	m
14	Can the server transmit a “terminate” data packet?	6.3.3	m

Table E.2 (*continued*)

Index	Topic	Clause	Status
15	Does the server properly support the reception of a “logout” data packet?	6.3.3	m
15.1	Does the server issue a FrED response to a valid “logout” data packet?	6.3.3	m
15.2	Does the server terminate the session after receiving a valid logout?	6.3.3	m
16	Does the server provide for off-line subscriptions for all subscriptions ?	6.4.1	o.4
server-on-line	Does the server support on-line subscriptions ?	6.4.2	o.4
17	Does the server support on-line subscriptions for all subscriptions it supports?	6.4.2	server-on-line:m
18	Does the server accept “subscription” data packets?	6.4.2	server-on-line:m
19	Does the server properly respond to “subscription” data packets?	6.4.2	server-on-line:m
19.1	Does the server transmit an “accept” data packet for valid subscriptions ?	6.4.2	server-on-line:m
19.2	Does the server transmit a “reject” data packet for an invalid subscription?	6.4.2	server-on-line:m
19.3	Does the server allow for a cancel subscription?	6.4.2	server-on-line:m
19.4	Does the server support single subscriptions ?	6.5.2	m
19.5	Does the server support registered periodic subscriptions ?	6.5.3	m
19.6	Does the server support registered on-occurrence subscriptions ?	6.5.3	m
20	Can the server publish publications for all subscriptions for which it claims support?	6.5	m
21	Does the server support any subscriptions which require file publications?	6.5	o
22	Does the server support the transmission of multiple publications within a single file/data packet?	6.5	o
23	Does the server retransmit the publication if an “accept” or “reject” data packet is not received within the time-out period when the guarantee flag is set to true?	6.1.4	m
24	Does the server accept the “transfer-done” data packet as a notice that it may delete the subject file?	6.5.1.5	m
25	Does the server respond to a “transfer-done” data packet with FrED data packet?	6.5.1.5	m
26	Can the server terminate a subscription through a publication?	6.5.1.5	m

E.3.2 Client requirements

Table E.3 identifies the status of the various features for clients.

Table E.3 — Client requirements

Index	Topic	Clause	Status
27	Does the client transmit a login in response to a valid initiate request?	6.3.1	dial-up:m
28	Does the client ignore an invalid “initiate” request from a server?	6.3.1	m
29	Is the client capable of sending a login request without the need for an “initiate” request?	6.3.1	m
30	Does the client retransmit a login if it does not receive a response within the time-out?	6.1.4	m
31	Does the client recognize the absense of a session if login is unsuccessful?	6.3.1	m
32	Does the client transmit FrED heartbeat messages as necessary to keep the session alive?	6.3.2	m
32.1	Does the client terminate a session if no messages are received from the server for a period exceeding the Maximum Heartbeat Duration?	6.3.2	m
32.2	Does the client transmit a logout request (after any subscription cancellations) in response to a valid “terminate” request?	6.3.3	m
33	Does the client ignore an invalid “terminate” request?	6.3.3	m
34	Can the client transmit a logout without the need for a “terminate” request?	6.3.3	m
35	Does the client retransmit a logout if a FrED is not received within the time-out period?	6.1.4	m
client-off-line	Does the client support off-line subscriptions ?	6.4.1	o.5
client-on-line	Does the client support on-line subscriptions ?	6.4.2	o.5
36.1	Does the client support “subscription” data packets for all supported subscriptions	6.4.2	client-on-line:m
36.2	Does the client retransmit a subscription if a response is not received within the time-out period?	6.1.4	client-on-line:m
37	Does the client recognize the failure of a subscription when a “reject” message is received?	6.4.2	m
38	Does the client support the persistant flag?	6.3.3	dial-up:m
39	Can the client cancel a subscription?	6.4.2	m
40	Does the client reject an invalid “publication” data packet?	6.5.1.4	m
41	Does the client transmit an “accept” data packet in response to a valid publication with the guaranteed flag set to true?	6.5.1.4	m
41.1	Does the client silently accept a valid publication with the guaranteed flag set to false?	6.5.1.4	m
41.2	Does the client initiate a file transfer if the publication indicated a file?	6.5.1.5	files:m
41.3	Does the client transmit a transfer “done-data” packet once the file transfer is completed?	6.5.1.5	files:m
41.4	Does the client retransmit a transfer “done-data” packet if it does not receive a FrED within the time-out period?	6.5	files:m

Table E.3 (*continued*)

Index	Topic	Clause	Status
41.5	Does the client accept multiple “PublicationData” structures in a single publication ?	6.5	m
41.6	Does the client reject any and all invalid “PublicationData” structures contained in the publication?	6.5.1.6	m
41.7	Does the client support single subscriptions ?	6.5.2	m
41.8	Does the client support registered periodic subscriptions ?	6.5.3	m
41.9	Does the client support registered on-occurrence subscriptions ?	6.5.3	m
42	Does the client recognize the terminatation of a subscription when such a publication is received?	6.5.1.5	m

Annex F
(informative)

Implementation guidance

The DATEX-ASN protocol is designed as an application layer protocol. Annex D specifies a common way to implement this protocol over an Internet stack and specifies a registered port number for such a design, but this does not prevent alternative implementations. The DATEX-ASN protocol may be implemented over any communications stack and may use any set of encoding rules. The requirement for BER is limited to those implementations using the registered port 355.

Annex G
(informative)

Advantages of DATEX-ASN

In 1996, the United States undertook an effort to standardize the protocols used for center-to-center data exchanges. This effort started with an analysis of the different requirements of various center-to-center links. The initial intent was to use these requirements in order to select one existing protocol to use as the industry standard. However, once the requirements were listed, it soon became clear that one protocol would not meet all of the stated requirements.

A summary of the requirements identified in this analysis is provided below:

— The protocol needs to support one-way and two-way data exchanges.

— The protocol needs to support event-driven, request-reply, and periodic responses.

— The protocol needs to support both permanent connections and dial-up connections, even for requests with event-driven responses.

— Some implementations will have limited bandwidth, thus the protocol should be bandwidth-efficient.

— Some implementations will be connected to an extensive network of centers, and there should be tools available to facilitate management of these connections.

— Some data will need to be protected through security schemes.

— Some centers may have a requirement to frequently transmit real-time (e.g. once-per-second) data exchanges of considerable size (e.g. 100 Kb of 1 byte data elements) over relatively low bandwidth (e.g. 28,8 kbps) links.

— Some centers will have limited resources and the protocol should be simple and inexpensive enough for small centers exchanging minimal data.

— The protocol should be scalable to also support the needs of large, complex, multi-agency centers.

Once the requirements were identified, the effort focused on what existing protocols might satisfy them. This search resulted in the evaluation of a number of protocols and languages. All protocols have limitations; examples of these limitations are explained below.

— SNMP: The MIB structure used by SNMP systems is not readily scalable; the protocol is also very bandwidth intensive.

— CMIP: This is similar to SNMP, but even more bandwidth-intensive.

— Raw Sockets: This has a very well-defined interface, but does not provide any real services above the transport layer. It has, however, been selected for use as an underlying mechanism of both CORBA and DATEX-ASN.

— SQL: This language has a great deal of software support within the industry, but to use many of these tools would require a tight coupling of system designs; further, it does not provide some of the dial-up features requested by various systems.

— FTP: This protocol provides for the transfer of files, but does not provide the rules on when files should be exchanged, or what they should contain.

— TFTP: Similar to FTP, this protocol provides an adequate file transfer scheme, but does not define the rules on when to exchange data or how that data should be formatted.

— HTTP/HTML: While this protocol/language provides an ideal solution for ad hoc human-driven systems, it does not provide the necessary features for automated systems such as a standard format of data and rules on what to send or when to send data.

— Self-Describing Data: This scheme was also determined to be too bandwidth-intensive, with minimal functionality.

— CORBA: This protocol requires significant resources for implementations, including programmer knowledge, memory, bandwidth and processing power. While this was a good fit for the high-end systems, it would be difficult to implement on smaller systems.

— D-COM: This protocol had many of the same limitations as the CORBA solution.

— DATEX-Net: This protocol was seen to have many advantages, but the structure of the protocol was not layered well enough to readily expand its implementation to new data exchanges.

As a result of this investigation, the committee decided that no single protocol by itself met all the needs. Instead, there was a need for a high-end protocol (CORBA) and a low-end protocol. In order to meet the maximum number of needs, it was determined that the best approach to meeting the low-end needs was to develop a simple protocol based on the DATEX-Net protocol used in Europe, which was the foundation of this part of ISO 14827. This protocol was then enhanced within this part of ISO 14827 to produce a more layered design and to ensure that all of the base requirements were met.

Future efforts might define how to use CORBA within the transportation environment.

ICS 03.220.01; 35.240.60

Price based on 63 pages