

DIN EN IEC 61158-5-26

ICS 25.040.40; 35.100.70; 35.110

**Industrielle Kommunikationsnetze –
Feldbusse –
Teil 5-26: Dienstfestlegungen des Application Layer
(Anwendungsschicht) – Typ 26-Elemente
(IEC 61158-5-26:2019);
Englische Fassung EN IEC 61158-5-26:2019, nur auf CD-ROM**

Industrial communication networks –
Fieldbus specifications –
Part 5-26: Application layer service definition – Type 26 elements
(IEC 61158-5-26:2019);
English version EN IEC 61158-5-26:2019, only on CD-ROM

Réseaux de communication industriels –
Spécifications des bus de terrain –
Partie 5-26: Spécification du protocole de la couche application – Éléments de type 26
(IEC 61158-5-26:2019);
Version anglaise EN IEC 61158-5-26:2019, seulement en CD-ROM

Gesamtumfang 116 Seiten

DKE Deutsche Kommission Elektrotechnik Elektronik Informationstechnik in DIN und VDE

Anwendungsbeginn

Anwendungsbeginn für die von CENELEC am 2019-05-23 angenommene Europäische Norm als DIN-Norm ist 2020-07-01.

Nationales Vorwort

Vorausgegangener Norm-Entwurf: E DIN EN 61158-500:2017-12.

Dieser Entwurf betrifft auch die Normen DIN EN IEC 61158-5-2:2020-07, DIN EN IEC 61158-5-4:2020-07, DIN EN IEC 61158-5-10:2020-07, DIN EN IEC 61158-5-12:2020-07, DIN EN IEC 61158-5-19:2020-07, DIN EN IEC 61158-5-21:2020-07, DIN EN IEC 61158-5-23:2020-07 und DIN EN IEC 61158-5-25:2020-07.

Für dieses Dokument ist das nationale Arbeitsgremium K 956 „Industrielle Kommunikation“ der DKE Deutsche Kommission Elektrotechnik Elektronik Informationstechnik in DIN und VDE (www.dke.de) zuständig.

Die enthaltene IEC-Publikation wurde vom SC 65C „Industrial networks“ erarbeitet.

Das IEC-Komitee hat entschieden, dass der Inhalt dieses Dokuments bis zu dem Datum (stability date) unverändert bleiben soll, das auf der IEC-Website unter „<http://webstore.iec.ch>“ zu diesem Dokument angegeben ist. Zu diesem Zeitpunkt wird entsprechend der Entscheidung des Komitees das Dokument

- bestätigt,
- zurückgezogen,
- durch eine Folgeausgabe ersetzt oder
- geändert.

Für den Fall einer undatierten Verweisung im normativen Text (Verweisung auf ein Dokument ohne Angabe des Ausgabedatums und ohne Hinweis auf eine Abschnittsnummer, eine Tabelle, ein Bild usw.) bezieht sich die Verweisung auf die jeweils aktuellste Ausgabe des in Bezug genommenen Dokuments.

Für den Fall einer datierten Verweisung im normativen Text bezieht sich die Verweisung immer auf die in Bezug genommene Ausgabe des Dokuments.

Der Zusammenhang der zitierten Dokumente mit den entsprechenden deutschen Dokumenten ergibt sich, soweit ein Zusammenhang besteht, grundsätzlich über die Nummer der entsprechenden IEC-Publikation. Beispiel: IEC 60068 ist als EN 60068 als Europäische Norm durch CENELEC übernommen und als DIN EN 60068 ins Deutsche Normenwerk aufgenommen.

Das Präsidium des DIN hat mit Präsidialbeschluss 1/2004 festgelegt, dass DIN-Normen, deren Inhalt sich auf internationale Arbeitsergebnisse der Informationsverarbeitung gründet, unter bestimmten Bedingungen allein in englischer Sprache veröffentlicht werden dürfen. Diese Bedingungen sind für die vorliegende Norm erfüllt.

Da sich die Benutzer der vorliegenden Norm der englischen Sprache als Fachsprache bedienen, wird die Englische Fassung der EN IEC 61158-5-26 veröffentlicht. Zu deren Abschnitt 1, der den Anwendungsbereich festlegt und Unterabschnitt 3.2, der die Begriffe festlegt, wurde eine Übersetzung angefertigt und als informativer Nationaler Anhang NA der vorliegenden Norm hinzugefügt. Für die meisten der verwendeten Begriffe existieren keine gebräuchlichen deutschen Benennungen, da sich die deutschen Anwender in der Regel ebenfalls der englischen Benennungen bedienen. Diese Norm steht nicht in unmittelbarem Zusammenhang mit Rechtsvorschriften und ist nicht als Sicherheitsnorm anzusehen.

Ein Feldbus ist ein digitaler serieller Datenbus für die Kommunikation zwischen Geräten der industriellen Automatisierungstechnik wie beispielsweise Messeinrichtungen, Stellgliedern, Reglern und speicherprogrammierbaren Steuerungen. Die Normen der Reihe DIN EN IEC 61158 legen eine Anzahl von Feldbus-Protokolltypen fest. Die vorliegende Norm legt die Dienste fest, die von einem Application Layer der einzelnen Typen zur Verfügung gestellt werden.

Nationaler Anhang NA (informativ)

Auszugsweise Übersetzung

Reihenfolge und Inhalt der folgenden Abschnitte sind identisch mit denen in den jeweiligen Abschnitten der Englischen Fassung.

EINLEITUNG

Dieses Dokument ist Teil einer Reihe, die entwickelt wurde, um die Zusammenschaltung von Komponenten eines Automatisierungssystems zu erleichtern. Es steht im Zusammenhang mit den anderen Normen der Reihe, wie sie durch das in IEC 61158-1 beschriebene „Drei-Layer“ Feldbus-Referenzmodell definiert sind.

Der Anwendungsdienst wird durch das Anwendungsprotokoll bereitgestellt, welches die Dienste nutzt, die vom Data-Link Layer oder einer anderen unmittelbar darunter liegenden Schicht zur Verfügung stehen. Dieses Dokument definiert die Eigenschaften der Anwendungsdienste, die von Feldbusanwendungen und/oder dem Systemmanagement genutzt werden können.

In der gesamten Reihe von Feldbusnormen bezieht sich der Begriff „Service“ auf die abstrakte Fähigkeit, die eine Schicht des OSI-Basisreferenzmodells der unmittelbar darüber liegenden Schicht bietet. Demzufolge ist der in diesem Dokument definierte Dienst des Data-Link Layer ein konzeptioneller Architekturdienst, unabhängig vom Administrativen- und Implementierungsbereich.

1 Anwendungsbereich

1.1 Überblick

Der Feldbus Application Layer (FAL, en: Fieldbus Application Layer) stellt Anwendungsprogramme mit Mitteln zum Zugriff auf die Feldbuskommunikationsumgebung bereit. Insofern kann der FAL als ein „Fenster zwischen entsprechenden Anwendungsprogrammen“ angesehen werden.

Dieser Teil von IEC 61158 legt allgemeine Elemente für grundlegende zeitkritische und nicht zeitkritische Messaging-Kommunikationen zwischen Anwendungsprogrammen in einem Automatisierungsumfeld und für Typ-26-Feldbusse spezifisches Material fest. Der Begriff „zeitkritisch“ wird zum Angeben des Vorhandenseins eines Zeitfensters, in dem eine oder mehrere festgelegte Aktionen mit einem gewissen definierten Sicherheitsgrad abzuschließen sind, verwendet. Das Nichtabschließen festgelegter Aktionen in dem Zeitfenster riskiert den Ausfall der Anwendungen, die die Aktionen anfordern, zusammen mit einem begleitenden Risiko für Betriebsmittel, die Anlage und möglicherweise Menschenleben.

Diese internationale Norm definiert auf abstrakte Art und Weise den nach außen erkennbaren Dienst, der durch die verschiedenen Feldbus Application Layer-Typen bereitgestellt wird, hinsichtlich

- a) eines abstrakten Modells zur Definition von Anwendungsressourcen (Objekten), die durch Benutzer mittels Verwendung des FAL-Dienstes manipuliert werden können,
- b) der Aktivitäten und Ereignisse des Dienstprimitivs;
- c) der Parameter, die jeder Aktivität und jedem Ereignis der Dienstprimitive zugeordnet sind, und der Form, die sie annehmen und
- d) des Zusammenhangs zwischen diesen Aktionen und Ereignissen und ihrer gültigen Sequenzen.

Der Zweck dieses Dokuments ist es, die Dienste zu definieren, die bereitgestellt werden an

- a) den FAL-Benutzer an der Grenze zwischen dem Benutzer und dem Application Layer des Feldbus-Referenzmodells und
- b) das Systemmanagement an der Grenze zwischen dem Application Layer und Systemmanagement des Feldbus-Referenzmodells.

Dieses Dokument legt die Struktur und die Dienste des IEC-Feldbus Application Layer in Übereinstimmung mit dem OSI-Basisreferenzmodell (ISO/IEC 7498-1) und der OSI-Anwendungsschichtstruktur (ISO/IEC 9545) fest.

FAL-Dienste und Protokolle werden durch FAL-Anwendungsinstanzen (AE, en: application entities), die in den Anwendungsprozessen enthalten sind, bereitgestellt. Die FAL-AE besteht aus einem Satz objektorientierter Anwendungsdienstelemente (ASE, en: Application Service Elements) und einer Einheit für Schichtenmanagement (LME, en: Layer Management Entity), die die AE verwaltet. Die ASE stellen Kommunikationsdienste bereit, die auf einem Satz zugehöriger Anwendungsprozessobjekt-(APO)-Klassen operieren. Eines der FAL-ASE ist ein Management-ASE, das einen gemeinsamen Satz von Diensten für die Verwaltung der Instanzen von FAL-Klassen bereitstellt.

Obwohl diese Dienste, aus dem Blickwinkel der Anwendung, beschreiben, wie Anfrage und Antwort ausgegeben und abgegeben werden, beinhalten sie keine Festlegung, was die anfragenden und antwortenden Anwendungen mit diesen vorzunehmen haben. Die Verhaltensaspekte der Anwendungen sind nicht festgelegt, nur eine Definition, welche Anfragen und Antworten diese senden/empfangen können, ist beschrieben. Dies erlaubt den FAL-Benutzern eine größere Flexibilität, um solch ein Objektverhalten zu standardisieren. Zusätzlich zu diesen Diensten sind in dieser Norm ebenfalls einige unterstützende Dienste festgelegt, um den Zugriff zum FAL bereitzustellen und somit bestimmte Aspekte des Betriebs zu steuern.

1.2 Spezifikationen

Das wesentliche Ziel dieses Dokuments ist es, die Merkmale konzeptioneller Application Layer-Dienste, die für zeitkritische Kommunikation geeignet sind, festzulegen und somit das OSI-Basisreferenzmodell beim Lenken der Entwicklung von Application Layer-Protokollen für zeitkritische Kommunikation zu ergänzen.

Ein Nebenziel ist es, Migrationspfade von zuvor bestehenden industriellen Kommunikationsprotokollen bereitzustellen. Es ist dieses letztgenannte Ziel, das zu der Vielfalt von Diensten, die als die verschiedenen Typen nach IEC 61158 genormt sind, und den entsprechenden Protokollen, die in Unterteilen von IEC 61158-6 genormt sind, führt.

Diese Spezifikation darf als Basis für formale Anwendungsprogrammierschnittstellen verwendet werden. Nichtsdestotrotz handelt es sich nicht um eine formale Programmierschnittstelle und eine solche Schnittstelle muss Fragen hinsichtlich der Implementierung, die nicht durch diese Spezifikation abgedeckt sind, behandeln, einschließlich

- a) der Größen und der Oktett-Anordnung verschiedener Multi-Oktett-Dienstparameter und
- b) der Korrelation gepaarter Anfrage- und Bestätigungsprimitive oder Anzeige- und Antwortprimitive.

1.3 Konformität

Dieses Dokument legt keine individuellen Implementierungen oder Produkte fest, noch beschränkt es die Implementierungen von Application Layer-Instanzen in industriellen Automatisierungssystemen.

Es gibt keine Betriebsmittelkonformität betreffend dieser Application Layer-Dienstdefinitionsnorm. Stattdessen wird Konformität durch Implementierung entsprechender Application Layer-Protokolle, die jeglichen in diesem Dokument definierten Typ von Application Layer-Diensten erfüllen, erreicht.

3.2 Spezifische Begriffe für Typ-26-Feldbus Application Layer

Für die Anwendung dieses Dokuments gelten die folgenden Begriffe.

3.2.1

Anwendung

(en: application)

Funktion oder Datenstruktur, für die Daten konsumiert oder produziert werden

3.2.2**Anwendungsobjekte**

(en: application objects)

mehrere Objektklassen, die zur Laufzeit für einen Nachrichtenaustausch im Netzwerk und innerhalb des Netzwerkgerätes sorgen und diesen verwalten

3.2.3**Anwendungsprozessobjekt**

(en: application process object)

Komponente eines Anwendungsprozesses, die über eine FAL-Anwendungsbeziehung identifizierbar und zugänglich ist

Anmerkung 1 zum Begriff: Definitionen von Anwendungsprozessobjekten setzen sich aus einem Satz von Werten für die Attribute ihrer Klasse zusammen (siehe Definition von „Anwendungsprozessobjektklasse“ (en: Application Process Object Class)). Auf Anwendungsprozessobjekte kann mithilfe der Dienste des Objektverwaltungs-ASE der FAL von einem entfernten Standort aus zugegriffen werden. Objektverwaltungsdienste des FAL können zum Laden oder Aktualisieren von Objektdefinitionen, zum Lesen von Objektdefinitionen und zum dynamischen Erstellen und Löschen von Anwendungsobjekten und deren zugehörigen Definitionen genutzt werden.

3.2.4**Anwendungsprozessobjektklasse**

(en: application process object class)

Klasse von Anwendungsprozessobjekten, die in Bezug auf den Satz der über das Netzwerk zugänglichen Attribute und Dienste der Anwendungsprozessobjekte definiert ist

3.2.5**Anwendungsbeziehung**

(en: application relationship)

kooperative Verbindung zwischen zwei oder mehr Aufrufen von Anwendungsinstanzen zum Zweck des Informationsaustausches und der Koordination des gemeinsamen Betriebs

Anmerkung 1 zum Begriff: Diese Beziehung wird entweder durch den Austausch von Protokolldateneinheiten der Anwendung oder als Ergebnis von Vorkonfigurationsaktivitäten aktiviert.

3.2.6**Anwendungsbeziehung ASE**

(en: application relationship ASE)

Anwendungsdienstelement (ASE) für den Aufbau einer beendenden Anwendungsbeziehung

3.2.7**Endpunkt der Anwendungsbeziehung**

(en: application relationship endpoint)

Kontext und Verhalten einer Anwendungsbeziehung entsprechend der Sicht eines und entsprechend der Verwaltung durch einen der in die Anwendungsbeziehung eingebundenen Anwendungsprozesse

Anmerkung 1 zum Begriff: Jeder Anwendungsprozess, der in die Anwendungsbeziehung eingebunden ist, verwaltet seinen eigenen Endpunkt der Anwendungsbeziehung.

3.2.8**Attribut**

(en: attribute)

Beschreibung eines von außen sichtbaren Merkmals oder einer von außen sichtbaren Eigenschaft eines Objekts

Anmerkung 1 zum Begriff: Die Attribute eines Objekts enthalten Informationen zu veränderlichen Teilen eines Objekts. Üblicherweise stellen sie Statusinformationen bereit oder steuern den Betrieb eines Objekts. Attribute können auch das Verhalten eines Objekts beeinflussen. Attribute werden in Klassenattribute und Instanzattribute unterteilt.

3.2.9

Verhalten

(en: behavior)

Angabe, wie ein Objekt auf bestimmte Ereignisse reagiert

3.2.10

Kanal

(en: channel)

einzelner physischer oder logischer Verbindungszweig eines Eingabe- oder Ausgabe-Anwendungsobjekts eines Servers zu dem Prozess

3.2.11

Klasse

(en: class)

Menge von Objekten, die allesamt die gleiche Art Systemkomponente repräsentieren

Anmerkung 1 zum Begriff: Eine Klasse ist eine Verallgemeinerung eines Objekts; eine Vorlage für die Festlegung von Variablen und Methoden. Alle Objekte in einer Klasse sind in ihrer Form und in ihrem Verhalten identisch, enthalten aber üblicherweise unterschiedliche Daten in ihren Attributen.

3.2.12

Klassenattribute

(en: class attributes)

Attribut, das allen Objekten innerhalb derselben Klasse gemeinsam ist

3.2.13

Client

(en: client)

<aus Sicht des Objekts> Objekt, das die Dienste eines anderen (Server-)Objekts zur Ausführung einer Aufgabe nutzt

3.2.14

Client

(en: client)

<aus Sicht der Kommunikation> Initiator einer Nachricht, auf die ein Server reagiert

3.2.15

allgemeiner Speicher

(en: common-memory)

virtueller Speicher, auf welchen mit einer logisch eindeutigen Adresse zugegriffen werden kann und der für die zyklische Übertragung verwendet wird

Anmerkung 1 zum Begriff: Wobei er aus dem Speicherbereich-1 und -2 der Speichergröße 512 Wörter beziehungsweise 8 192 Wörter besteht und mit den Typ-26-Knoten in einem Typ-26-Feldbus-Netz gemeinsam verwendet wird.

3.2.16

Verbindung

(en: connection)

logische Verknüpfung zwischen Anwendungsobjekten, die sich in demselben Gerät oder in unterschiedlichen Geräten befinden dürfen

Anmerkung 1 zum Begriff: Verbindungen können entweder Punkt-zu-Punkt-Verbindungen oder Mehrpunktverbindungen sein.

3.2.17

Konsument

(en: consumer)

Knoten oder Senke, der/die Daten von einem Produzenten empfängt

3.2.18**Steuerbefehle**

(en: control commands)

vom Client zum Server übertragene Aktionsaufrufe, um Ausgänge freizugeben, Eingänge zu blockieren (Freeze) und/oder Ausgänge zu synchronisieren

3.2.19**zyklisch**

(en: cyclic)

sich regelmäßig wiederholend

3.2.20**Rahmen zyklischer Daten**

(en: cyclic-data frame)

Rahmen, der für die zyklische Übertragung vom Typ 26 verwendet wird

3.2.21**Übertragung zyklischer Daten**

(en: cyclic-data transmission)

Datenübertragung, mit der ein Typ-26-Knoten Daten an einen vorbestimmten Bereich für den Knoten des allgemeinen Speichers aussendet, wobei der Knoten bei Rundsendungen (en: Broadcasting) an alle anderen Typ-26-Knoten jedes Mal ein Token bezieht

3.2.22**zugeordnete AR**

(en: dedicated AR)

AR, die direkt vom FAL-Benutzer verwendet wird

Anmerkung 1 zum Begriff: Bei zugeordneten AR werden nur der FAL-Header und die Benutzerdaten übertragen.

3.2.23**Gerät**

(en: device)

physische Hardware, die mit dem Verbindungszweig verbunden ist

Anmerkung 1 zum Begriff: Ein Gerät darf mehr als einen Knoten umfassen.

3.2.24**Geräteprofil**

(en: device profile)

Zusammenstellung geräteabhängiger Informationen und Funktionalitäten zur Bereitstellung von Konsistenz zwischen ähnlichen Geräten desselben Gerätetyps

3.2.25**Endknoten**

(en: end node)

erzeugender oder konsumierender Knoten

3.2.26**Endpunkt**

(en: endpoint)

eine der Kommunikationsinstanzen, die zu einer Verbindung gehören

3.2.27**Fehler**

(en: error)

Nichtübereinstimmung zwischen einem berechneten, beobachteten oder gemessenen Wert oder Zustand und dem festgelegten oder theoretisch richtigen Wert oder Zustand

3.2.28

Ereignis

(en: event)

Vorgang einer Änderung von Zuständen

3.2.29

Protokoll einer FA-Verbindung

(en: FA-link protocol)

Kommunikationsprotokoll für einen Typ-26-Feldbus

Anmerkung 1 zum Begriff: Legt die Verwaltung von Token, die zyklische Datenübertragung und die Nachrichtendatenübertragung fest, mit welchen jeder Typ-26-Knoten eine Datenübertragung auf gleichberechtigter Basis durchführen kann.

3.2.30

FAL-Subnetz

(en: FAL subnet)

Subnetzwerke, die aus einem oder mehreren Data-Link-Segmenten bestehen und durch eine Untermenge der Netzwerkadresse identifiziert werden

3.2.31

Rahmen

(en: frame)

umgangssprachliches Synonym für DLPDU

3.2.32

Aufruf

(en: invocation)

Vorgang der Nutzung eines Dienstes oder einer anderen Ressource eines Anwendungsprozesses

Anmerkung 1 zum Begriff: Jeder Aufruf bildet einen einzelnen Steuerungsstrang, der durch seinen Kontext beschrieben werden kann. Sobald der Dienst ausgeführt oder die Nutzung der Ressource beendet ist, erlischt der Aufruf. Bei Dienstaufrufen wird ein Dienst, der initiiert, aber noch nicht ausgeführt wurde, als ausstehender Dienstaufruf bezeichnet.

3.2.33

Index

(en: index)

Adresse eines Objekts innerhalb eines Anwendungsprozesses

3.2.34

Instanz

(en: instance)

tatsächliches physisches Vorkommen eines Objekts innerhalb einer Klasse, das eines von mehreren Objekten innerhalb derselben Objektklasse identifiziert

BEISPIEL „Kalifornien“ ist eine Instanz der Objektklasse „Staat“.

Anmerkung 1 zum Begriff: Die Benennungen „Objekt“, „Instanz“ und „Objektinstanz“ beziehen sich auf eine spezifische Instanz.

3.2.35

Instanzattribute

(en: instance attributes)

Attribut, das für eine Objektinstanz eindeutig ist und von der Objektklasse nicht gemeinsam genutzt wird

3.2.36**instanziiert**

(en: instantiated)

Objekt, das in einem Gerät erzeugt wurde

3.2.37**logisches Gerät**

(en: logical device)

bestimmte FAL-Klasse, die eine Softwarekomponente oder eine Firmwarekomponente als eigenständige Einrichtung eines Automatisierungsgerätes abstrahiert

3.2.38**Verwaltungsinformationen**

(en: management information)

über das Netzwerk zugängliche Informationen, die die Verwaltung des Betriebs des Feldbussystems, einschließlich des Application Layer, unterstützen

Anmerkung 1 zum Begriff: Zur Verwaltung gehören Funktionen wie die Steuerung, die Überwachung und die Diagnose.

3.2.39**Rahmen für Nachrichtendaten**

(en: message-data frame)

Rahmen, der für die Übertragung von Nachrichtendaten vom Typ 26 verwendet wird

3.2.40**Übertragung von Nachrichtendaten**

(en: message-data transmission)

unregelmäßige Datenübertragung, mit der ein Typ-26-Knoten die Nachrichtendaten einer Anfrage, einer Antwort, eines Informationsberichts, einer Benachrichtigung zum Auftreten eines Ereignisses am Knoten usw. an andere Typ-26-Knoten aussendet

3.2.41**Methode**

(en: method)

Synonym für einen funktionsfähigen Dienst, der von dem Server-ASE bereitgestellt und von einem Client aufgerufen wird

3.2.42**Modul**

(en: module)

Hardware oder logische Komponente eines physikalischen Gerätes

3.2.43**Mehrpunktverbindung**

(en: multipoint connection)

Verbindung von einem Knoten zu vielen Knoten

Anmerkung 1 zum Begriff: Durch Mehrpunktverbindungen können Nachrichten von einem einzelnen Produzenten von vielen Konsumentenknuten empfangen werden.

3.2.44**Netzwerk**

(en: network)

Menge von Knoten, die über ein bestimmtes Kommunikationsmedium miteinander verbunden sind, einschließlich jeglicher zwischengeschalteter Repeater, Bridges, Router und Gateways unterer Schichten

3.2.45

Netzwerklast-Messfunktion

(en: network load measurement function)

Bereitstellung zur Überwachung und Messung der Kommunikationslast über einen Typ-26-Feldbus mit einem Typ-26-Knoten

3.2.46

Objekt

(en: object)

abstrakte Darstellung einer bestimmten Komponente innerhalb eines Gerätes, üblicherweise eine Zusammenstellung zugehöriger Daten (in Form von Variablen) und Methoden (Prozeduren) für den Betrieb mit diesen Daten, die eine klar definierte Schnittstelle und ein klar definiertes Verhalten aufweisen

3.2.47

Teilnahme-Anfrage-Rahmen

(en: participation-request-frame)

Rahmen, der für die Benachrichtigung eines Typ-26-Knotens über seine Teilnahme-Anfrage in einem Typ-26-Feldbus-Netz verwendet wird

3.2.48

Peer

(en: peer)

Rolle eines AR-Endpunktes, in der dieser in der Lage ist, sowohl als Client als auch als Server zu agieren

3.2.49

physikalisches Gerät

(en: physical device)

Automatisierungsgerät oder anderes Netzwerkgerät

3.2.50

Punkt-zu-Punkt-Verbindung

(en: point-to-point connection)

Verbindung, die zwischen genau zwei Anwendungsobjekten besteht

3.2.51

Eigenschaft

(en: property)

allgemeine Bezeichnung für beschreibende Informationen zu einem Objekt

3.2.52

Protokolltyp

(en: protocol type)

Kennung des verwendeten Kommunikationsprotokolls

3.2.53

Publisher

(en: publisher)

Rolle eines AR-Endpunktes, in der dieser APDU auf den Feldbus für den Konsum durch einen oder mehrere Subscriber überträgt

Anmerkung 1 zum Begriff: Ein Publisher braucht die Identität oder die Anzahl der Subscriber nicht zu kennen und darf seine APDU mithilfe einer zugeordneten AR veröffentlichen.

3.2.54

Pull-Publisher

(en: pull publisher)

Typ eines Publishers, der ein Objekt als Antwort auf eine von dem entsprechenden veröffentlichenden Pull-Manager empfangene Anforderung veröffentlicht

3.2.55**veröffentlichender Pull-Manager**

(en: pull publishing manager)

Typ eines veröffentlichenden Managers, der die Veröffentlichung eines bestimmten Objekts in einer entsprechenden Antwort-APDU anfordert

3.2.56**Push-Publisher**

(en: push publisher)

Typ eines Publishers, der ein Objekt in einer unbestätigten Dienstanforderungs-APDU veröffentlicht

3.2.57**veröffentlichender Push-Manager**

(en: push publishing manager)

Typ eines veröffentlichenden Managers, der die Veröffentlichung eines bestimmten Objekts mithilfe eines unbestätigten Dienstes anfordert

3.2.58**Push-Subscriber**

(en: push subscriber)

Typ eines Subscribers, der empfangene unbestätigte Dienstanforderungs-APDU als veröffentlichte Objektdaten erkennt

3.2.59**Ressource**

(en: resource)

Verarbeitungs- oder Informationsfähigkeit eines Subsystems

3.2.60**Sequenznummer**

(en: sequence number)

Wert für andere Knoten zur Bestimmung, ob empfangende Nachrichtendaten von einem Knoten bereits vorher empfangen wurden und somit redundant sind

Anmerkung 1 zum Begriff: Wird zum Zeitpunkt der Inbetriebnahme des Netzwerkes auf 0 (null) gesetzt und wird jedes Mal, wenn eine Übertragung von Nachrichtendaten abgeschlossen ist, um 1 (eins) erhöht.

3.2.61**Server**

(en: server)

<aus Sicht der Kommunikation> Rolle eines AREP, in der dieser eine bestätigte Dienstantwort-APDU an den Client, der die Anforderung initiiert hat, zurückgibt

3.2.62**Server**

(en: server)

<aus Sicht des Objekts> Objekt, das einem anderen (Client-)Objekt Dienste bereitstellt

3.2.63**Dienst**

(en: service)

Operation oder Funktion, die ein Objekt und/oder eine Objektklasse auf Anforderung eines anderen Objekts und/oder einer anderen Objektklasse hin ausführt

3.2.64**Subscriber**

(en: subscriber)

Rolle eines AREP, in der dieser von einem Publisher produzierte APDU empfängt

3.2.65

Token

(en: token)

Recht zum Senden von Daten über das Übertragungsmedium, und wird in einem Typ-26-Feldbus-Netzwerk nacheinander von Knoten zu Knoten weitergegeben

3.2.66

Token-Frame

(en: token-frame)

Rahmen, der für die Übergabe von Token zwischen Typ-26-Knoten verwendet wird

3.2.67

Transaktionscode

(en: transaction code)

wird verwendet, um jeden Rahmen der Datenübertragung zu unterscheiden und zu kennzeichnen, und repräsentiert die Transaktion eines empfangenden Rahmens

3.2.68

Trigger-Frame

(en: trigger-frame)

Rahmen, der im Zustand der Netzinbetriebnahme für einen Typ-26-Knoten verwendet wird, der über seinen Eintritt in eine neue Teilnehmeranfrage über ein Typ-26-Feldbus-Netzwerk benachrichtigt

3.2.69

Zustand gültiger Verbindung

(en: valid-linking state)

Zustand, in welchem ein Token nacheinander zwischen Knoten in einem Typ-26-Feldbus-Netzwerk verbreitet wird

3.2.70

Version einer Sequenznummer

(en: version of sequence number)

Versionsnummer der Sequenznummer in einer empfangenden Daten-Nachricht von anderen Knoten, die verwendet wird, um festzustellen, ob die empfangende Daten-Nachricht gültig und nicht überfällig ist

Anmerkung 1 zum Begriff: Der Wert wird zum Zeitpunkt der Beendigung der Teilnahme eines Typ-26-Knotens an einem Typ-26-Feldbus-Netzwerk festgelegt und bleibt bis zum Ausscheiden des Knotens aus dem Typ-26-Feldbus-Netzwerk erhalten.

3.2.71

virtueller Adressraum

(en: virtual-address-space)

virtueller Speicherplatz mit einer pseudologischen Adresse, verwendet für die Nachrichtenübertragung

English Version

**Industrial communication networks - Fieldbus specifications -
Part 5-26: Application layer service definition - Type 26 elements
(IEC 61158-5-26:2019)**

Réseaux de communication industriels - Spécifications des
bus de terrain - Partie 5-26: Spécification du protocole de la
couche application - Éléments de type 26
(IEC 61158-5-26:2019)

Industrielle Kommunikationsnetze - Feldbusse - Teil 5-26:
Dienstfestlegungen des Application Layer
(Anwendungsschicht) - Typ 26-Elemente
(IEC 61158-5-26:2019)

This European Standard was approved by CENELEC on 2019-05-23. CENELEC members are bound to comply with the CEN/CENELEC Internal Regulations which stipulate the conditions for giving this European Standard the status of a national standard without any alteration.

Up-to-date lists and bibliographical references concerning such national standards may be obtained on application to the CEN-CENELEC Management Centre or to any CENELEC member.

This European Standard exists in three official versions (English, French, German). A version in any other language made by translation under the responsibility of a CENELEC member into its own language and notified to the CEN-CENELEC Management Centre has the same status as the official versions.

CENELEC members are the national electrotechnical committees of Austria, Belgium, Bulgaria, Croatia, Cyprus, the Czech Republic, Denmark, Estonia, Finland, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Lithuania, Luxembourg, Malta, the Netherlands, Norway, Poland, Portugal, Republic of North Macedonia, Romania, Serbia, Slovakia, Slovenia, Spain, Sweden, Switzerland, Turkey and the United Kingdom.



European Committee for Electrotechnical Standardization
Comité Européen de Normalisation Electrotechnique
Europäisches Komitee für Elektrotechnische Normung

CEN-CENELEC Management Centre: Rue de la Science 23, B-1040 Brussels

European foreword

The text of document 65C/947/FDIS, future edition 1 of IEC 61158-5-26, prepared by SC 65C "Industrial networks" of IEC/TC 65 "Industrial-process measurement, control and automation" was submitted to the IEC-CENELEC parallel vote and approved by CENELEC as EN IEC 61158-5-26:2019.

The following dates are fixed:

- latest date by which the document has to be implemented at national level by (dop) 2020-02-23
publication of an identical national standard or by endorsement
- latest date by which the national standards conflicting with the (dow) 2022-05-23
document have to be withdrawn

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

Endorsement notice

The text of the International Standard IEC 61158-5-26:2019 was approved by CENELEC as a European Standard without any modification.

Annex ZA (normative)

Normative references to international publications with their corresponding European publications

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements 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.

NOTE 1 Where an International Publication has been modified by common modifications, indicated by (mod), the relevant EN/HD applies.

NOTE 2 Up-to-date information on the latest versions of the European Standards listed in this annex is available here: www.cenelec.eu.

<u>Publication</u>	<u>Year</u>	<u>Title</u>	<u>EN/HD</u>	<u>Year</u>
IEC 61131-3	-	Programmable controllers - Part 3: Programming languages	EN 61131-3	-
IEC 61158-1	2019	Industrial communication networks - Fieldbus specifications - Part 1: Overview and guidance for the IEC 61158 and IEC 61784 series	-	-
IEC 61158-6-26	2019	Industrial communication networks – Fieldbus specifications – Part 6-26: Application layer protocol specification – Type 26 elements	EN 61158-6-26	2019
IEC 61784-2	2019	Industrial communication networks - Profiles - Part 2: Additional fieldbus profiles for real-time networks based on ISO/IEC/IEEE 8802-3	-	-
ISO/IEC 646	-	Information technology - ISO 7-bit coded character set for information interchange	-	-
ISO/IEC 7498-1	-	Information technology - Open Systems Interconnection - Basic Reference Model: The Basic Model	-	-
ISO/IEC/IEEE 8802-3-	-	Standard for Ethernet	-	-
ISO/IEC 8822	-	Information technology - Open Systems Interconnection - Presentation service definition	-	-
ISO/IEC 8824-1	-	Information technology - Abstract Syntax Notation One (ASN.1): Specification of basic notation	-	-
ISO/IEC 9545	-	Information technology - Open Systems Interconnection - Application layer structure	-	-
ISO/IEC 10646	-	Information technology - Universal Coded Character Set (UCS)	-	-
ISO/IEC 10731	-	Information technology - Open Systems Interconnection - Basic Reference Model - Conventions for the definition of OSI services	-	-
ISO/IEC/IEEE 60559 -	-	Information technology - Microprocessor Systems - Floating-Point arithmetic	-	-

<u>Publication</u>	<u>Year</u>	<u>Title</u>	<u>EN/HD</u>	<u>Year</u>
IETF RFC 768	-	User Datagram Protocol	-	-
IETF RFC 791	-	Internet Protocol	-	-
IETF RFC 792	-	Internet Control Message Protocol	-	-
IETF RFC 793	-	Transmission Control Protocol,	-	-
IETF RFC 796	-	Address mappings	-	-
IETF RFC 826	-	Ethernet Address Resolution Protocol: Or Converting Network Protocol Addresses to 48.bit Ethernet Address for Transmission on Ethernet Hardware	-	-
IETF RFC 894	-	Standard for the Transmission of IP Datagrams over Ethernet Networks	-	-
IETF RFC 919	-	Broadcasting Internet Datagrams	-	-
IETF RFC 922	-	Broadcasting Internet datagrams in the presence of subnets	-	-
IETF RFC 950	-	Internet Standard Subnetting Procedure	-	-

CONTENTS

European foreword	2
Annex ZA (normative) Normative references to international publications with their corresponding European publications.....	3
INTRODUCTION	8
1 Scope	9
1.1 Overview.....	9
1.2 Specifications	10
1.3 Conformance	10
2 Normative references	10
3 Terms, definitions, symbols, abbreviations and conventions.....	11
3.1 Terms and definitions from other ISO/IEC standards	11
3.1.1 Terms and definitions from ISO/IEC 7498-1	11
3.1.2 Terms and definitions from ISO/IEC 8822	12
3.1.3 Terms and definitions from ISO/IEC 9545	12
3.1.4 Terms and definitions from ISO/IEC 8824-1	12
3.2 Fieldbus application layer Type 26-specific definitions	12
3.3 Abbreviations and symbols.....	19
3.4 Conventions.....	21
3.4.1 Overview.....	21
3.4.2 General conventions	21
3.4.3 Conventions for class definitions.....	21
3.4.4 Conventions for service definitions	22
4 Concepts	23
5 Data type ASE	23
5.1 Overview.....	23
5.2 Formal definition of data type objects.....	24
5.2.1 Data type class definitions.....	24
5.2.2 Attributes	24
5.3 FAL defined data types.....	26
5.3.1 Fixed length types	26
5.3.2 String types	30
5.4 Data type ASE service specification.....	30
6 Communication model specification.....	30
6.1 General.....	30
6.2 Protocol stack for Type 26 fieldbus	31
6.3 Overview of Type 26 communication model	32
6.4 Cyclic data communication service with Common-memory	32
6.4.1 Overview.....	32
6.4.2 Common-memory: allocation to each node.....	33
6.4.3 Data sharing among nodes with the CM.....	34
6.4.4 CM data type	35
6.5 ASEs.....	35
6.5.1 Overview of Type 26 ASEs	35
6.5.2 Type 26 specific conventions for FAL service common parameters	36
6.5.3 Cyclic-data ASE	37

6.5.4	Message data ASE	42
6.5.5	Communication load measurement ASE	69
6.5.6	Network management ASE	74
6.5.7	General purpose command server ASE	87
6.5.8	AR ASE	90
6.5.9	FAL ASE summary	101
Bibliography		104
Figure 1 – Protocol stack for Type 26 fieldbus		31
Figure 2 – Unconfirmed Push-Publisher/Subscriber type interaction		32
Figure 3 – Unconfirmed/Confirmed Client/Server type interaction		32
Figure 4 – Common memory allocation		33
Figure 5 – Data sharing with the CM		34
Figure 6 – Node #01 for reception only		34
Figure 7 – Node #01 without the CM		35
Figure 8 – The structure of ASEs for Type 26 FAL		36
Figure 9 – Virtual-address-space for Byte block		45
Figure 10 – Virtual-address-space for Word block		48
Figure 11 – AR ASE internal architecture		91
Figure 12 – Structure of IP address		99
Table 1 – Write service parameters		39
Table 2 – Send-CM service parameters		40
Table 3 – Read service parameters		40
Table 4 – Update memory service parameters		41
Table 5 – Get- buffer service parameters		42
Table 6 – Byte block read service parameters		46
Table 7 – Byte block write service parameters		47
Table 8 – Word block read service parameters		48
Table 9 – Word block write service parameters		49
Table 10 – Network parameter read service parameters		50
Table 11 – Network parameter write service parameters		51
Table 12 – Stop command service parameters		52
Table 13 – Operation command service parameters		53
Table 14 – profile read service parameters		53
Table 15 – Transparent message service parameters		56
Table 16 – Log data read service parameters		57
Table 17 – Log data items		57
Table 18 – Log data clear service parameters		61
Table 19 – Message return service parameters		62
Table 20 – Vendor specific message service parameters		63
Table 21 – Set remote node configuration parameter service parameters		64
Table 22 – Data elements and Node configuration parameters		64

Table 23 – Service parameters of Read remote participating node management information parameter service	65
Table 24 – Participating node management information parameters	66
Table 25 – Read remote node management information parameter service parameters	67
Table 26 – Node management information parameters.....	67
Table 27 – Read remote node setting information parameter service parameters.....	68
Table 28 –Node setting information parameters	69
Table 29 – Start TK-holding-time measurement service parameters	71
Table 30 – Terminate TK-holding-time measurement service parameters.....	71
Table 31 – Token-holding-time measurement result.....	72
Table 32 – Start GP_Comm sender log service parameters	73
Table 33 – Terminate GP_Comm sender log service parameters.....	73
Table 34 – GP_Comm sender log measurement result.....	74
Table 35 – Service parameters for Set configuration parameter	77
Table 36 – Configuration parameters.....	78
Table 37 – Read node management information parameter service parameters.....	78
Table 38 – Node management information parameters.....	79
Table 39 – Service parameters for Read participating node mgt. information parameter	80
Table 40 – Participating node management information parameters	80
Table 41 – Service parameters for Read network management information parameter	81
Table 42 – Network management information parameters	81
Table 43 – Service parameters for Read message sequence number management information.....	82
Table 44 – Read message sequence number management information parameters.....	82
Table 45 –Read node status service parameters	83
Table 46 – Read node status parameters	83
Table 47 – Upper layer operating condition matrix	84
Table 48 –Reset node service parameters	84
Table 49 – Set network address service parameters.....	85
Table 50 – Register service parameters	85
Table 51 – Event service parameters.....	86
Table 52 – Activate/Deactivate measurement service parameters.....	86
Table 53 – Get log data service parameters	87
Table 54 – Send command service parameters.....	90
Table 55 – CT send service parameters	93
Table 56 – MT send service parameters.....	95
Table 57 – CS send service parameters.....	96
Table 58 – Notify state change service parameters.....	97
Table 59 – Control measurement service parameters	97
Table 60 – DLSAP assignments	99
Table 61 – DLS Primitives and parameters.....	100
Table 62 – Lower layer T-profile and the required standards	101
Table 63 – Summary of FAL ASEs	101

INTRODUCTION

This document is one of a series produced to facilitate the interconnection of automation system components. It is related to other standards in the set as defined by the “three-layer” fieldbus reference model described in IEC 61158-1.

The application service is provided by the application protocol making use of the services available from the data-link or other immediately lower layer. This document defines the application service characteristics that fieldbus applications and/or system management may exploit.

Throughout the set of fieldbus standards, the term “service” refers to the abstract capability provided by one layer of the OSI Basic Reference Model to the layer immediately above. Thus, the application layer service defined in this document is a conceptual architectural service, independent of administrative and implementation divisions.

INDUSTRIAL COMMUNICATION NETWORKS – FIELDBUS SPECIFICATIONS –

Part 5-26: Application layer service definition – Type 26 elements

1 Scope

1.1 Overview

The Fieldbus Application Layer (FAL) provides user programs with a means to access the Fieldbus communication environment. In this respect, the FAL can be viewed as a “window between corresponding application programs.”

This part of IEC 61158 provides common elements for basic time-critical and non-time-critical messaging communications between application programs in an automation environment and material specific to Type 26 fieldbus. The term “time-critical” is used to represent the presence of a time-window, within which one or more specified actions are required to be completed with some defined level of certainty. Failure to complete specified actions within the time window risks failure of the applications requesting the actions, with attendant risk to equipment, plant and possibly human life.

This International Standard defines in an abstract way the externally visible service provided by the different Types of fieldbus Application Layer in terms of

- a) an abstract model for defining application resources (objects) capable of being manipulated by users via the use of the FAL service;
- b) the primitive actions and events of the service;
- c) the parameters associated with each primitive action and event, and the form which they take; and
- d) the interrelationship between these actions and events, and their valid sequences.

The purpose of this document is to define the services provided to

- a) the FAL user at the boundary between the user and the Application Layer of the Fieldbus Reference Model, and
- b) Systems Management at the boundary between the Application Layer and Systems Management of the Fieldbus Reference Model.

This document specifies the structure and services of the IEC Fieldbus Application Layer, in conformance with the OSI Basic Reference Model (see ISO/IEC 7498-1) and the OSI Application Layer Structure (see ISO/IEC 9545).

FAL services and protocols are provided by FAL application-entities (AE) contained within the application processes. The FAL AE is composed of a set of object-oriented Application Service Elements (ASEs) and a Layer Management Entity (LME) that manages the AE. The ASEs provide communication services that operate on a set of related application process object (APO) classes. One of the FAL ASEs is a management ASE that provides a common set of services for the management of the instances of FAL classes.

Although these services specify, from the perspective of applications, how request and responses are issued and delivered, they do not include a specification of what the requesting and responding applications are to do with them. That is, the behavioral aspects of the applications are not specified; only a definition of what requests and responses they can send/receive is specified. This permits greater flexibility to the FAL users in

standardizing such object behavior. In addition to these services, some supporting services are also defined in this document to provide access to the FAL to control certain aspects of its operation.

1.2 Specifications

The principal objective of this document is to specify the characteristics of conceptual application layer services suitable for time-critical communications, and thus supplement the OSI Basic Reference Model in guiding the development of application layer protocols for time-critical communications.

A secondary objective is to provide migration paths from previously-existing industrial communications protocols. It is this latter objective which gives rise to the diversity of services standardized as the various Types of IEC 61158, and the corresponding protocols standardized in subparts of IEC 61158-6.

This specification may be used as the basis for formal Application Programming-Interfaces. Nevertheless, it is not a formal programming interface, and any such interface will need to address implementation issues not covered by this specification, including

- a) the sizes and octet ordering of various multi-octet service parameters, and
- b) the correlation of paired request and confirm, or indication and response, primitives.

1.3 Conformance

This document does not specify individual implementations or products, nor does it constrain the implementations of application layer entities within industrial automation systems.

There is no conformance of equipment to this application layer service definition standard. Instead, conformance is achieved through implementation of conforming application layer protocols that fulfill any given Type of application layer services as defined in this document.

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements 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.

NOTE All parts of the IEC 61158 series, as well as IEC 61784-1 and IEC 61784-2, are maintained simultaneously. Cross-references to these documents within the text therefore refer to the editions as dated in this list of normative references.

IEC 61131-3, *Programmable controllers – Part 3: Programming languages*

IEC 61158-1:2019, *Industrial communication networks – Fieldbus specifications – Part 1: Overview and guidance for the IEC 61158 and IEC 61784 series*

IEC 61158-6-26:2019, *Industrial communication networks – Fieldbus specifications – Part 6-26: Application layer protocol specification – Type 26 elements*

IEC 61784-2:2019, *Industrial communication networks – Profiles – Part 2: Additional fieldbus profiles for real-time networks based on ISO/IEC/IEEE 8802-3*

ISO/IEC 646, *Information technology – ISO 7-bit coded character set for information interchange*

ISO/IEC 7498-1, *Information technology – Open Systems Interconnection – Basic Reference Model – Part 1: The Basic Model*

ISO/IEC/IEEE 8802-3, *Information technology – Telecommunications and information exchange between systems – Local and metropolitan area networks – Specific requirements – Part 3: Standard for Ethernet*

ISO/IEC 8822, *Information technology – Open Systems Interconnection – Presentation service definition*

ISO/IEC 8824-1, *Information Technology – Abstract Syntax notation One (ASN-1): Specification of basic notation*

ISO/IEC 9545, *Information technology – Open Systems Interconnection – Application Layer structure*

ISO/IEC 10646, *Information technology – Universal Coded Character Set (UCS)*

ISO/IEC 10731, *Information technology – Open Systems Interconnection – Basic Reference Model – Conventions for the definition of OSI services*

ISO/IEC 60559, *Information technology – Microprocessor Systems – Floating-Point arithmetic*

IETF RFC 768, *User Datagram Protocol*, available at <http://www.ietf.org> [viewed 2018-09-20]

IETF RFC 791, *Internet Protocol*, available at <http://www.ietf.org> [viewed 2018-09-20]

IETF RFC 792, *Internet Control Message Protocol*, available at <http://www.ietf.org> [viewed 2018-09-20]

IETF RFC 793, *Transmission Control Protocol*, available at <http://www.ietf.org> [viewed 2018-09-20]

IETF RFC 796, *Address mappings*, available at <http://www.ietf.org> [viewed 2018-09-20]

IETF RFC 826, *An Ethernet Address Resolution Protocol*, available at <http://www.ietf.org> [viewed 2018-09-20]

IETF RFC 894, *A Standard for the Transmission of IP Datagrams over Ethernet Networks*, available at <http://www.ietf.org> [viewed 2018-09-20]

IETF RFC 919, *Broadcasting Internet Datagrams*, available at <http://www.ietf.org> [viewed 2018-09-20]

IETF RFC 922, *Broadcasting Internet Datagrams in the presence of subnets*, available at <http://www.ietf.org> [viewed 2018-09-20]

IETF RFC 950, *Internet Standard Subnetting Procedure*, available at <http://www.ietf.org> [viewed 2018-09-20]

3 Terms, definitions, symbols, abbreviations and conventions

For the purposes of this document, the following terms, definitions, symbols, abbreviations and conventions apply.

ISO and IEC maintain terminological databases for use in standardization at the following addresses:

- IEC Electropedia: available at <http://www.electropedia.org/>
- ISO Online browsing platform: available at <http://www.iso.org/obp>

3.1 Terms and definitions from other ISO/IEC standards

3.1.1 Terms and definitions from ISO/IEC 7498-1

For the purposes of this document, the following terms and definitions given in ISO/IEC 7498-1 apply.

- a) application entity
- b) application process

- c) application protocol data unit
- d) application service element
- e) application entity invocation
- f) application process invocation
- g) application transaction
- h) real open system
- i) transfer syntax

3.1.2 Terms and definitions from ISO/IEC 8822

For the purposes of this document, the following terms and definitions given in ISO/IEC 8822 apply.

- a) abstract syntax
- b) presentation context

3.1.3 Terms and definitions from ISO/IEC 9545

For the purposes of this document, the following terms and definitions given in ISO/IEC 9545 apply.

- a) application-association
- b) application-context
- c) application context name
- d) application-entity-invocation
- e) application-entity-type
- f) application-process-invocation
- g) application-process-type
- h) application-service-element
- i) application control service element

3.1.4 Terms and definitions from ISO/IEC 8824-1

For the purposes of this document, the following terms and definitions given in ISO/IEC 8824-1 apply.

- a) object identifier
- b) type
- c) value
- d) simple type
- e) structured type
- f) tag
- g) Boolean type
- h) true
- i) false
- j) integer type
- k) bitstring type
- l) octetstring type

3.2 Fieldbus application layer Type 26-specific definitions

For the purposes of this document, the following terms and definitions apply.

3.2.1

application

function or data structure for which data is consumed or produced

3.2.2

application objects

multiple object classes that manage and provide a run time exchange of messages across the network and within the network device

3.2.3

application process object

component of an application process that is identifiable and accessible through an FAL application relationship

Note 1 to entry: Application process object definitions are composed of a set of values for the attributes of their class (see the definition for Application Process Object Class Definition). Application process objects can be accessed remotely using the services of the FAL Object Management ASE. FAL Object Management services can be used to load or update object definitions, to read object definitions, and to dynamically create and delete application objects and their corresponding definitions.

3.2.4

application process object class

class of application process objects defined in terms of the set of their network-accessible attributes and services

3.2.5

application relationship

cooperative association between two or more application-entity-invocations for the purpose of exchange of information and coordination of their joint operation

Note 1 to entry: This relationship is activated either by the exchange of application-protocol-data-units or as a result of preconfiguration activities.

3.2.6

application relationship ASE

ASE used for establishing and terminating application relationship

3.2.7

application relationship endpoint

context and behavior of an application relationship as seen and maintained by one of the application processes involved in the application relationship

Note 1 to entry: Each application process involved in the application relationship maintains its own application relationship endpoint.

3.2.8

attribute

description of an externally visible characteristic or feature of an object

Note 1 to entry: The attributes of an object contain information about variable portions of an object. Typically, they provide status information or govern the operation of an object. Attributes may also affect the behavior of an object. Attributes are divided into class attributes and instance attributes.

3.2.9

behavior

indication of how an object responds to particular events

3.2.10

channel

single physical or logical link of an input or output application object of a server to the process

3.2.11

class

set of objects, all of which represent the same kind of system component

Note 1 to entry: A class is a generalization of an object; a template for defining variables and methods. All objects in a class are identical in form and behavior, but usually contain different data in their attributes.

3.2.12

class attributes

attribute that is shared by all objects within the same class

3.2.13

client

<object view> object which uses the services of another (server) object to perform a task

3.2.14

client

<communication view> initiator of a message to which a server reacts

3.2.15

common-memory

virtual memory accessible with logically unique address used for the cyclic-data transmission

Note 1 to entry: It is composed of the memory-area-1 and -2 of the memory size 512 words and 8 192 words respectively and is shared with the Type 26 nodes in a Type 26 fieldbus network.

3.2.16

connection

logical binding between application objects that are within the same or different devices

Note 1 to entry: Connections can be either point-to-point or multipoint.

3.2.17

consumer

node or sink that is receiving data from a producer

3.2.18

control commands

action invocations transferred from client to server to clear outputs, freeze inputs and/or synchronize outputs

3.2.19

cyclic

repetitive in a regular manner

3.2.20

cyclic-data frame

frame used for the Type 26 cyclic-data transmission

3.2.21

cyclic-data transmission

data transmission with which a Type 26 node sends out data on a pre-assigned area for the node on the common-memory, in broadcasting to all other Type 26 nodes every time the node obtains a token

3.2.22

dedicated AR

AR used directly by the FAL User

Note 1 to entry: On dedicated ARs, only the FAL Header and the user data are transferred.

3.2.23

device

physical hardware connected to the link

Note 1 to entry: A device can contain one or more than one node.

3.2.24

device profile

collection of device dependent information and functionality providing consistency between similar devices of the same device type

3.2.25

end node

producing or consuming node

3.2.26

endpoint

one of the communicating entities involved in a connection

3.2.27

error

discrepancy between a computed, observed or measured value or condition and the specified or theoretically correct value or condition

3.2.28

event

instance of a change of conditions

3.2.29

FA-link protocol

communication protocol for Type 26 fieldbus

Note 1 to entry: It defines the token management, the cyclic data transmission and the message data transmission, with which each Type 26 node performs data transmission on an equal basis.

3.2.30

FAL subnet

subnetworks composed of one or more data link segments, identified by a subset of the network address

3.2.31

frame

synonym for DLPDU

3.2.32

invocation

act of using a service or other resource of an application process

Note 1 to entry: Each invocation represents a separate thread of control that can be described by its context. Once the service completes, or use of the resource is released, the invocation ceases to exist. For service invocations, a service that has been initiated but not yet completed is referred to as an outstanding service invocation.

3.2.33

index

address of an object within an application process

3.2.34

instance

actual physical occurrence of an object within a class that identifies one of many objects within the same object class

EXAMPLE California is an instance of the object class state.

Note 1 to entry: The terms object, instance, and object instance are used to refer to a specific instance.

3.2.35

instance attributes

attribute that is unique to an object instance and not shared by the object class

3.2.36

instantiated

object that has been created in a device

3.2.37

logical device

FAL class that abstracts a software component or a firmware component as an autonomous self-contained facility of an automation device

3.2.38

management information

network-accessible information that supports managing the operation of the fieldbus system, including the application layer

Note 1 to entry: Managing includes functions such as controlling, monitoring, and diagnosing.

3.2.39

message-data frame

frame used for the Type 26 message-data transmission

3.2.40

message-data transmission

sporadic data transmission with which a Type 26 node sends out the message data of request, response, information report, notification upon event happened at the node and so forth, to other Type 26 nodes

3.2.41

method

synonym for an operational service which is provided by the server ASE and invoked by a client

3.2.42

module

hardware or logical component of a physical device

3.2.43

multipoint connection

connection from one node to many

Note 1 to entry: Multipoint connections allow messages from a single producer to be received by many consumer nodes.

3.2.44

network

set of nodes connected by some type of communication medium, including any intervening repeaters, bridges, routers and lower-layer gateways

3.2.45

network load measurement function

provision for monitoring and measurement of the communication load over a Type 26 fieldbus with a Type 26 node

3.2.46

object

abstract representation of a particular component within a device, usually a collection of related data (in the form of variables) and methods (procedures) for operating on that data that have clearly defined interface and behavior

3.2.47

participation-request-frame

frame used for a Type 26 node notifying of its participation-request in a Type 26 fieldbus network

3.2.48

peer

role of an AR endpoint in which it is capable of acting as both client and server

3.2.49

physical device

automation or other network device

3.2.50

point-to-point connection

connection that exists between exactly two application objects

3.2.51

property

general term for descriptive information about an object

3.2.52

protocol type

identification of communication protocol

3.2.53

publisher

role of an AR endpoint that transmits APDUs onto the fieldbus for consumption by one or more subscribers

Note 1 to entry: A publisher may not be aware of the identity or the number of subscribers and it may publish its APDUs using a dedicated AR.

3.2.54

pull publisher

type of publisher that publishes an object in response to a request received from its pull publishing manager

3.2.55

pull publishing manager

type of publishing manager that requests that a specified object be published in a corresponding response APDU

3.2.56

push publisher

type of publisher that publishes an object in an unconfirmed service request APDU

3.2.57

push publishing manager

type of publishing manager that requests that a specified object be published using an unconfirmed service

3.2.58

push subscriber

type of subscriber that recognizes received unconfirmed service request APDUs as published object data

3.2.59

resource

processing or information capability of a subsystem

3.2.60

sequence number

consecutive number used to determine whether a receiving message-data from other nodes has been previously received and is redundant

Note 1 to entry: The sequence number of a node is set to 0 (zero) at the time of the network-startup, and it is incremented by 1 (one) every time a message-data transmission is completed. The sequence number of the node is transmitted to other nodes by the message-data transmission.

3.2.61

server

<communication view> role of an AREP in which it returns a confirmed service response APDU to the client that initiated the request

3.2.62

server

<object view> object which provides services to another (client) object

3.2.63

service

operation or function than an object and/or object class performs upon request from another object and/or object class

3.2.64

subscriber

role of an AREP in which it receives APDUs produced by a publisher

3.2.65

token

right to send out data on the transmission medium, which is passed successively from node to node for Type 26 fieldbus network

3.2.66

token-frame

frame used for hand-over the token between Type 26 nodes

3.2.67

transaction code

code used to discriminate and identify each data transmission frame, and which represents the transaction of a receiving frame

3.2.68

trigger-frame

frame used in the network-startup state for a Type 26 node notifying of its entrance into a new participation-request over a Type 26 fieldbus network

3.2.69

valid-linking state

state in which a token is consecutively circulated among nodes over Type 26 fieldbus network

3.2.70

version of sequence number

version number of the sequence number in a receiving message-data from other nodes, used to determine whether the receiving message-data is valid and is not overdue

Note 1 to entry: The value is set at the time when a Type 26 node completes to participate in a Type 26 fieldbus network, and is maintained until the node drops out of the Type 26 fieldbus network.

3.2.71

virtual-address-space

virtual memory space with pseudo logical address, used for the message-transmission

3.3 Abbreviations and symbols

AE	Application entity
AL	Application layer
ALME	Application layer management entity
ALP	Application layer protocol
ALS	Application layer service
AP	Application process
APO	Application process object
APDU	Application protocol data unit
AR	Application relationship
AREP	Application relationship endpoint
ASCII	American Standard Code for Information Interchange
ASE	Application service element
BCT	Broadcast transmission
C_AD1	Data-head-address on common-memory-area-1
C_AD2	Data-head-address on common-memory-area-2
CIM	Computer integrated manufacturing
CM	Common-memory
Cnf	Confirmation
C_SZ1	Data-size on common-memory-area-1
C_SZ2	Data-size on common-memory-area-2
DA	Destination-address
DL-	(as a prefix) data-link-
DLC	Data-link connection
DLCEP	Data-link connection endpoint
DLL	Data-link layer
DLS	Data-link service
DLSAP	Data-link service access point
DLSDU	DL-service-data-unit
DNA	Destination-node-number
DNS	Domain name service
FAL	Fieldbus application layer
HMI	Human-machine interface

H_TYPE	Header-type
ID	Identifier
IEC	International Electrotechnical Commission
Ind	Indication
IP	Internet protocol
ISO	International Organization for Standardization
LME	Layer management entity
LSB	Least Significant Bit
M_ADD	Message-offset-address in the virtual-address-space
MAC	Media Access Control
MAJ_VER	FA-link protocol version: major-version
M_CTL	Message-control
MFT	Allowable-minimum-frame-interval-time
MIN_VER	FA-link protocol version: minor-version
MSB	Most Significant Bit
MSN	Manufacturer's model name of a Type 26 node
M_SZ	Message-data-size in the virtual-address-space
M_RLT	Message result
NDN	Node name of Type 26 node
OSI	Open Systems Interconnect
PDU	Protocol data unit
PhL	Ph-layer
PTP	Point-to-Point
P_TYPE	Protocol-type
RCT	Allowable-refresh-cycle-time
Req	Request
RMT	Refresh-cycle-measurement-time
Rsp	Response
RT	Real Time
SA	Source-address
SAP	Service access point
SDU	Service data unit
SEQ	Sequence-number
SNA	Source-node-number
TCD	Transaction code
TCP	Transmission Control Protocol
TW	Token-watchdog
UDP	User Datagram protocol
ULS	Upper layer status
VDN	Vendor code of the node
V_SEQ	Version of sequence number

3.4 Conventions

3.4.1 Overview

The FAL is defined as a set of object-oriented ASEs. Each ASE is specified in a separate subclause. Each ASE specification is composed of two parts, its class specification, and its service specification.

The class specification defines the attributes of the class. The attributes are accessible from instances of the class using the ASE services specified in 0 of this document. The service specification defines the services that are provided by the ASE.

3.4.2 General conventions

This document uses the descriptive conventions given in ISO/IEC 10731.

3.4.3 Conventions for class definitions

Class definitions are described using templates. Each template consists of a list of attributes and services for the class. The general form of the template is shown below:

FAL ASE:		ASE Name
CLASS:		Class Name
CLASS ID:		#
PARENT CLASS:		Parent Class Name
ATTRIBUTES:		
1	(o)	Key Attribute: numeric identifier
2	(o)	Key Attribute: name
3	(m)	Attribute: attribute name (values)
4	(m)	Attribute: attribute name (values)
4.1	(s)	Attribute: attribute name (values)
4.2	(s)	Attribute: attribute name (values)
4.3	(s)	Attribute: attribute name (values)
5	(c)	Constraint: constraint expression
5.1	(m)	Attribute: attribute name (values)
5.2	(o)	Attribute: attribute name (values)
6	(m)	Attribute: attribute name (values)
6.1	(s)	Attribute: attribute name (values)
6.2	(s)	Attribute: attribute name (values)
SERVICES:		
1	(o)	OpsService: service name
2	(c)	Constraint: constraint expression
2.1	(o)	OpsService: service name
3	(m)	MgtService: service name

- The "FAL ASE:" entry is the name of the FAL ASE that provides the services for the class being specified.
- The "CLASS:" entry is the name of the class being specified. All objects defined using this template will be an instance of this class. The class may be specified by this document, or by a user of this document.
- The "CLASS ID:" entry is a number that identifies the class being specified. This number is unique within the FAL ASE that will provide the services for this class. When qualified by the identity of its FAL ASE, it unambiguously identifies the class within the scope of the FAL. The value "NULL" indicates that the class cannot be instantiated. Class IDs between 1 and 255 are reserved by this document to identify standardized classes. They have been assigned to maintain compatibility with existing national standards. CLASS IDs between 256 and 2 048 are allocated for identifying user defined classes.
- The "PARENT CLASS:" entry is the name of the parent class for the class being specified. All attributes defined for the parent class and inherited by it are inherited for the class being defined, and therefore do not have to be redefined in the template for this class.

NOTE The parent-class "TOP" indicates that the class being defined is an initial class definition. The parent class TOP is used as a starting point from which all other classes are defined. The use of TOP is reserved for classes defined by this document.

- The "ATTRIBUTES" label indicate that the following entries are attributes defined for the class.

- 1) Each of the attribute entries contains a line number in column 1, a mandatory (m) / optional (o) / conditional (c) / selector (s) indicator in column 2, an attribute type label in column 3, a name or a conditional expression in column 4, and optionally a list of enumerated values in column 5. In the column following the list of values, the default value for the attribute may be specified.
- 2) Objects are normally identified by a numeric identifier or by an object name, or by both. In the class templates, these key attributes are defined under the key attribute.
- 3) The line number defines the sequence and the level of nesting of the line. Each nesting level is identified by period. Nesting is used to specify
 - i) fields of a structured attribute (4.1, 4.2, 4.3),
 - ii) attributes conditional on a constraint statement (5). Attributes may be mandatory (5.1) or optional (5.2) if the constraint is true. Not all optional attributes require constraint statements as does the attribute defined in (5.2),
 - iii) the selection fields of a choice type attribute (6.1 and 6.2).
- f) The "SERVICES" label indicates that the following entries are services defined for the class.
 - 1) An (m) in column 2 indicates that the service is mandatory for the class, while an (o) indicates that it is optional. A (c) in this column indicates that the service is conditional. When all services defined for a class are defined as optional, at least one has to be selected when an instance of the class is defined.
 - 2) The label "OpsService" designates an operational service (1).
 - 3) The label "MgtService" designates a management service (2).
 - 4) The line number defines the sequence and the level of nesting of the line. Each nesting level is identified by period. Nesting within the list of services is used to specify services conditional on a constraint statement.

3.4.4 Conventions for service definitions

3.4.4.1 General

The service model, service primitives, and time-sequence diagrams used are entirely abstract descriptions. Therefore, they do not represent a specification for implementation.

3.4.4.2 Service parameters

Service primitives, used to represent service user/service provider interactions (see ISO/IEC 10731), convey parameters that indicate information available in the user/provider interaction. In any particular interface, not all parameters need be explicitly stated.

This document uses a tabular format to describe the component parameters of the service primitives. The parameters that apply to each group of service primitives are set out in tables throughout the remainder of this document. Each table consists of up to five columns, containing the name of the service parameter, and a column each for those primitives and parameter-transfer directions used by the service:

- a) parameter name,
- b) request primitive (transmitted from the sender),
- c) indication primitive (transmitted to the receiver),
- d) response primitive (transmitted from the receiver), and
- e) confirm primitive (transmitted to the sender).

One parameter (or part of it) is listed in each row of each table. Under the appropriate service primitive columns, a code is used to specify the type of usage of the parameter on the primitive and parameter direction specified in the column:

M parameter is mandatory for the primitive

- U parameter is a User option, and may or may not be provided depending on the dynamic usage of the service-user. When not provided, a default value for the parameter is assumed.
- C parameter is conditional upon other parameters or upon the environment of the service-user.
- (blank) parameter is never present.
- S parameter is a selected item.

Some entries are further qualified by items in brackets. These may be

- a parameter-specific constraint:
“(=)” indicates that the parameter is semantically equivalent to the parameter in the service primitive to its immediate left in the table, and
- an indication that some note applies to the entry:
“(n)” indicates that the following note "n" contains additional information pertaining to the parameter and its use.

3.4.4.3 Service procedures

The procedures are defined in terms of:

- the interactions between application entities through the exchange of fieldbus Application Protocol Data Units, and
- the interactions between an application layer service provider and an application layer service user in the same system through the invocation of application layer service primitives.

These procedures are applicable to instances of communication between systems which support time-constrained communications services within the FAL.

4 Concepts

The FAL is defined as a set of object-oriented ASEs. Each ASE is specified in a separate subclause. Each ASE specification is composed of three parts: the class definitions, the services and the protocol specification. The first two are contained in this document. The protocol specification for each of the ASEs is defined in IEC 61158-6-26.

The fundamental concepts and templates used to describe the application layer service in this document follow Clause 9 of IEC 61158-1.

The FAL defined in this document identifies a peer based controller communications network supporting a Common-memory model in 0 as well as client/server models.

5 Data type ASE

5.1 Overview

Fieldbus data types specify the machine independent syntax for application data conveyed by FAL services. The FAL supports the definition and transfer of both basic and constructed data types. Basic types are atomic types. Constructed types are types composed of basic types and other constructed types.

IEC 61158-1, 10.1 provides an overview of the data type ASE and the relationship between data types, and is incorporated herein by reference.

Most basic types are defined from a set of ISO/IEC 8824 types (simple types):

- Basic types have a constant length;