



Industrial automation systems — Manufacturing Message Specification —

Part 2: Protocol specification

TECHNICAL CORRIGENDUM 1

Systèmes d'automatisation industrielle — Spécification de messagerie industrielle —

Partie 2: Spécification de protocole

RECTIFICATIF TECHNIQUE 1

Technical corrigendum 1 to International Standard ISO/IEC 9506-2:1990 was prepared by Technical Committee ISO/TC 184, *Industrial automation systems and integration*, Subcommittee SC 5, *Architecture and communications*.

Page 12

Subclause 5.14

In the note, lines 1 and 2, change

"this part of this part of"

to

"this part of"

Page 16

Subclause 6.3.1.2

In item a) in the third line change "error class SERVICE" to "error class SERVICE-PREEMPT".

In item b) in the second line change

"error class SERVICE" to "error class SERVICE-PREEMPT".

Page 19

Clause 7

Change

```
{ iso standard 8650 abstract-syntax(2) acse-pdi(1) }
```

to

```
{ joint-iso-ccitt association-control(2) abstract(1) }
```

Page 19

Subclause 7.1

Replace the comment line in the Confirmed-RequestPDU production. The protocol in this subclause should now read:

```
Confirmed-RequestPDU ::= SEQUENCE {
  invokeID          Unsigned32,
  listOfModifier    SEQUENCE OF Modifier OPTIONAL,
  ConfirmedServiceRequest,
  [79] CS-Request-Detail OPTIONAL,
  -- shall not be transmitted if value is the value
  -- of a tagged type derived from NULL
}
```

Page 20

Subclause 7.1

At the end of the subclause, that is, immediately preceding 7.2, add the following note:

NOTE — The intent of the comment in the Confirmed-RequestPDU production is to preclude the transmission of the CS-Request-Detail field for all occurrences in the abstract syntax defined in this part of ISO/IEC 9506. This is done by assigning the NULL type to all the types referenced in the CS-Request-Detail type. (See the type definitions in 7.5.2 and in clause 19.) MMS Companion Standards may also preclude the transmission of this field either by defining the appropriate type to be NULL, similarly to the definitions in clause 19, or by selecting a NULL choice from a CHOICE type.

*Page 20***Subclause 7.2**

Replace the comment line in the Unconfirmed-PDU production.
The protocol in this subclause should now read:

```
Unconfirmed-PDU ::= SEQUENCE {
    UnconfirmedService,
    [79] CS-Unconfirmed-Detail OPTIONAL,
    -- shall not be transmitted if value is the value
    -- of a tagged type derived from NULL
}
```

*Page 20***Subclause 7.2**

At the end of the subclause, that is immediately preceding 7.3, add the following note:

NOTE — The intent of the comment in the Unconfirmed-PDU production is to preclude the transmission of the CS-Unconfirmed-Detail field for all occurrences in the abstract syntax defined in this part of ISO/IEC 9506. This is done by assigning the NULL type to all the types referenced in the CS-Unconfirmed-Detail type. (See the type definitions in 7.5.3 and in clause 19.) MMS Companion Standards may also preclude the transmission of this field either by defining the appropriate type to be NULL, similarly to the definitions in clause 19, or by selecting a NULL choice from a CHOICE type.

*Page 20***Subclause 7.3**

Replace the comment line in the Confirmed-ResponsePDU production.
The protocol in this subclause should now read:

```
Confirmed-ResponsePDU ::= SEQUENCE {
    invokeID Unsigned32,
    ConfirmedServiceResponse,
    [79] CS-Response-Detail OPTIONAL,
    -- shall not be transmitted if value is the value
    -- of a tagged type derived from NULL
}
```

*Page 20***Subclause 7.3**

At the end of the subclause, that is immediately preceding 7.4, add the following note:

NOTE — The intent of the comment in the Confirmed-ResponsePDU production is to preclude the transmission

of the CS-Response-Detail field for all occurrences in the abstract syntax defined in this part of ISO/IEC 9506. This is done by assigning the NULL type to all the types referenced in the CS-Response-Detail type. (See the type definitions in 7.5.4 and in clause 19.) MMS Companion Standards may also preclude the transmission of this field either by defining the appropriate type to be NULL, similarly to the definitions in clause 19, or by selecting a NULL choice from a CHOICE type.

Page 32

Subclause 7.6.2

In the definition of Identifier, change "___" to "_" (only a single underscore)

Page 36

Subclause 8.2

Remove 'akec' named bit from the ParameterSupportOptions production.
This production should now read:

```
ParameterSupportOptions ::= BIT STRING {  
  str1    (0),  
  str2    (1),  
  vnam    (2),  
  valt    (3),  
  vadr    (4),  
  vsca    (5),  
  tpy     (6),  
  vlis    (7),  
  real    (8),  
  --      bit 9 reserved for future definition  
  cei     (10)  
}
```

Page 47

Subclause 10.8

Replace the line

```
listOfCapabilities [1] IMPLICIT SEQUENCE OF VisibleString,
```

with

```
listOfCapabilities [1] IMPLICIT SEQUENCE OF VisibleString OPTIONAL,
```