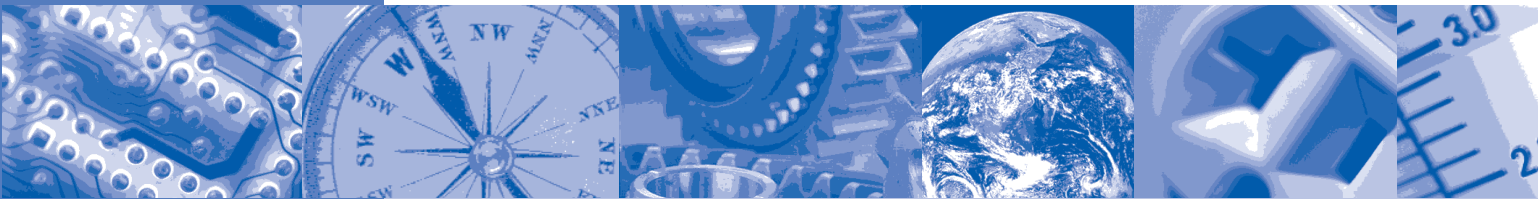


# **ANSI/ISA-72.02-1993**

Formerly ANSI/ISA-S72.02-1993



## **Manufacturing Message Specification: Companion Standard for Process Control**

**Identical to ISO/IEC 9506-6**



**Approved 24 March 1994**

ANSI/ISA-72.02-1993, Manufacturing Message Specification: Companion Standard for Process Control

ISBN: 1-55617-519-1

Copyright © 1993 by the Instrument Society of America. All rights reserved. Printed in the United States of America. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), without the prior written permission of the publisher.

ISA  
67 Alexander Drive  
P.O. Box 12277  
Research Triangle Park, North Carolina 27709

---

## Preface

---

This preface is included for information purposes and is not part of ANSI/ISA-72.02-1993.

This standard has been prepared as part of the service of ISA, the international society for measurement and control, toward a goal of uniformity in the field of instrumentation. To be of real value, this document should not be static, but should be subject to periodic review. Toward this end, the Society welcomes all comments and criticisms, and asks that they be addressed to the Secretary, Standards and Practices Board, ISA, 67 Alexander Drive, P. O. Box 12277, Research Triangle Park, NC 27709, Telephone (919) 990 9227, Fax (919) 549 8288, e-mail: standards@isa.org.

The ISA Standards and Practices Department is aware of the growing need for attention to the metric system of units in general, and the International System of Units (SI) in particular, in the preparation of instrumentation standards. The Department is further aware of the benefits to USA users of ISA Standards of incorporating suitable references to the SI (and the metric system) in their business and professional dealings with other countries. Toward this end, this Department will endeavor to introduce SI-acceptable metric units in all new and revised standards to the greatest extent possible. *The Metric Practice Guide*, which has been published by the Institute of Electrical and Electronic Engineers as ANSI/IEEE Std. 268 1982, and future revisions will be the reference guide for definitions, symbols, abbreviations, and conversion factors.

It is the policy of ISA to encourage and welcome the participation of all concerned individuals and interests in the development of ISA standards. Participation in the ISA standards making process by an individual in no way constitutes endorsement by the employer of that individual, of the Instrument Society of America, or of any of the standards that ISA develops.

This standard has been developed in cooperation with IEC SC65C/WG1, "Message data format for information transferred on process and control data highways." The ISA committee and IEC working group held concurrent meetings and have harmonized the developing drafts throughout the standards process.

The following people served as members of ISA Subcommittee SP72.02:

| NAME                        | COMPANY               |
|-----------------------------|-----------------------|
| C. Williams, Chairman       | Eastman Kodak Company |
| C. Gross, Managing Director | Dow Chemical Company  |

The following people served as members of ISA Committee SP72:

| NAME                        | COMPANY                |
|-----------------------------|------------------------|
| C. Williams, Chairman       | Eastman Kodak Company  |
| C. Gross, Managing Director | Dow Chemical Company   |
| F. Allegrezza               | Moore Products Company |

| NAME            | COMPANY                                |
|-----------------|----------------------------------------|
| *J. Barat       | Digital Equipment Company              |
| D. Barber       | Rust International Corporation         |
| P. Brett        | Honeywell, Inc.                        |
| *H. Burns       | Fisher Controls International, Inc.    |
| A. Capel        | Comgate Engineering, Ltd.              |
| R. Caro         | Arthur D. Little, Inc.                 |
| R. Crowder      | SHIP STAR Associates, Inc.             |
| U. Dobrich      | Siemens AG                             |
| H. Falk         | Systems Integ. Specialist Company      |
| B. Featherstone | ABB Industrial Automation              |
| W. Genter       | PCMT ATC B                             |
| *P. Gorla       | Honeywell SSDC                         |
| M. Hagar        | Texas Instruments, Inc.                |
| R. Harold       | Samborn Steketee Otis & Evan           |
| R. Hatcher      | Amoco Oil Company                      |
| S. Heatley      | NIST                                   |
| P. Heyliger     | Rhone Poulenc, Inc.                    |
| W. Hodson       | Leeds & Northrup                       |
| W. Hullsiek     | EMA, Inc.                              |
| M. Huras        | IBM Corporation                        |
| *R. Knipp       | E I du Pont                            |
| *K. Krivoshein  | Fisher Controls International, Inc.    |
| *Y. Kumeda      | Yamatake Honeywell Company, Ltd.       |
| *C. Langford    | E I du Pont                            |
| N. Laurance     | Ford Motor Company                     |
| T. Lehner       | AEG                                    |
| *R. Lyskowski   | Digital Equipment Corporation          |
| W. Maxwell      | Lower Colorado River Authority         |
| R. Mergen       | Lubrizol Corporation                   |
| *N. J. Miller   | Digital Equipment Corporation          |
| N. T. Miller    | Rosemount, Inc.                        |
| D. Modell       | MPI                                    |
| A. Mukherji     | EPRI M&D Center                        |
| L. Neitzel      | Computer Technology Associates         |
| J. Nelson       | Measure                                |
| M. Newman       | Cornell University                     |
| T. Ogowa        | Yokogawa Electric Company              |
| K. Oress        | Lubrizol Corporation                   |
| J. Pattison     | British Coal                           |
| *M. Patz        | Softing GmbH                           |
| M. Pelley       | Allen Bradley Company                  |
| R. Rammler      | Powell Process System                  |
| R. Rupp         | BRL                                    |
| R. Sawyer       | Foxboro Company                        |
| K. Schug        | CTA, Inc.                              |
| E. Skabowski    | Chevron USA                            |
| J. Slavinsky    | Federal Office for Standards & Measure |
| G. Stevens      | Modicon, Inc.                          |

---

\*One vote per company

| NAME                 | COMPANY                    |
|----------------------|----------------------------|
| D. Sweeton           | Commsoft                   |
| C. Terrell Tennessee | Eastman Company            |
| N. Tobol             | Ronan Engineering          |
| *V. Von Rein         | Softing GmbH               |
| T. Williams          | Purdue University          |
| G. Workman           | General Motors Corporation |

This standard was approved for publication by the ISA Standards and Practices Board in December, 1993.

| NAME                       | COMPANY                                        |
|----------------------------|------------------------------------------------|
| W. Weidman, Vice President | Gilbert Commonwealth, Inc.                     |
| H. Baumann                 | H.D. Baumann & Associates, Ltd.                |
| C. Gross                   | Dow Chemical Company                           |
| H. Hopkins                 | Utility Products of Arizona                    |
| A. Iverson                 | Lyondell Petrochemical Company                 |
| K. Lindner                 | Endress + Hauser GmbH + Company                |
| G. McFarland               | ABB Power Plant Controls                       |
| E. Montgomery              | Fluor Daniel, Inc.                             |
| E. Nesvig                  | ERDCO Engineering Corporation                  |
| R. Prescott                | Moore Products Company                         |
| D. Rapley                  | Rapley Engineering Services                    |
| R. Reimer                  | Allen Bradley Company                          |
| J. Rennie                  | Factory Mutual Research Corporation            |
| R. Webb                    | Pacific Gas & Electric Company                 |
| J. Weiss                   | Electric Power Research Institute              |
| J. Whetstone               | National Institute of Standards and Technology |
| M. Widmeyer                | The Supply System                              |
| C. Williams                | Eastman Kodak Company                          |
| M. Zielinski               | Rosemount, Inc.                                |
| **D. Bishop                | Chevron USA Production Company                 |
| **P. Bliss                 | Consultant                                     |
| **W. Calder, III           | Consultant                                     |
| **B. Christensen           | Consultant                                     |
| **L. Combs                 | Retired/Consultant                             |
| **N. Conger                | Consultant                                     |
| **T. Harrison              | FAMU/FSU College of Engineering                |
| **R. Jones                 | Consultant                                     |
| **R. Keller                | Consultant                                     |
| **O. Lovett, Jr.           | Consultant                                     |
| **E. Magison               | Honeywell, Inc.                                |
| **R. Marvin                | Roy G. Marvin Company                          |
| **A. McCauley, Jr.         | Chagrin Valley Controls, Inc.                  |
| **W. Miller                | Retired/Consultant                             |
| **J. Mock                  | Bechtel                                        |

---

\*One vote per company

\*\*Directors Emeriti

**\*\*G. Platt**

Retired/Consultant

**NAME**

**COMPANY**

**\*\*C. Reimann**

National Institute of Standards & Technology

**\*\*K. Whitman**

ABB Combustion Engineering

**\*\*J. Williams**

Consultant

---

**\*\*Directors Emeriti**

---

# Contents

---

|                                                                           |           |
|---------------------------------------------------------------------------|-----------|
| <b>Foreword</b>                                                           | <b>9</b>  |
| <b>Introduction</b>                                                       | <b>11</b> |
| <b>1 Scope</b>                                                            | <b>13</b> |
| <b>2 Normative References</b>                                             | <b>14</b> |
| <b>3 Definitions</b>                                                      | <b>15</b> |
| 3.1 Reference model definitions                                           | 15        |
| 3.2 Definitions unique to this part of ISO/IEC 9506                       | 15        |
| <b>4 Abbreviations</b>                                                    | <b>16</b> |
| <b>5 Application description</b>                                          | <b>17</b> |
| 5.1 Process control models                                                | 17        |
| 5.2 Process control functions                                             | 19        |
| 5.3 Application Models                                                    | 21        |
| <b>6 Process control context mapping</b>                                  | <b>23</b> |
| 6.1 Mapping the process control model to the VMD                          | 23        |
| 6.2 Definition of process control objects that map to Domains             | 23        |
| 6.3 Definition of process control objects that map to Program Invocations | 23        |
| 6.4 Process control requirements that affect other MMS objects            | 24        |
| 6.5 Definitions of new MMS abstract objects                               | 26        |
| 6.6 Parameter conformance                                                 | 28        |
| <b>7 Services</b>                                                         | <b>28</b> |
| 7.1 Use of ACSE services                                                  | 28        |
| 7.2 Use of MMS services                                                   | 29        |
| 7.3 Definition and use of process control specific services               | 44        |
| 7.4 The Initiate Service and Protocol                                     | 89        |
| 7.5 Generalized protocol extensions                                       | 93        |
| 7.6 End of module                                                         | 96        |
| <b>8 Standardized objects</b>                                             | <b>96</b> |
| <b>9 Conformance</b>                                                      | <b>96</b> |
| 9.1 Conformance classes                                                   | 97        |
| 9.2 PICS Part One: Implementation information                             | 103       |
| 9.3 PICS part Two: Service CBBs                                           | 103       |
| 9.4 PICS Part Three: Parameter CBBs                                       | 104       |
| 9.5 PICS Part Four: Local implementation values                           | 104       |

## Annexes

|                                                                                |     |
|--------------------------------------------------------------------------------|-----|
| A — Application Association model (Normative) .....                            | 105 |
| B — Block concepts (Informative) .....                                         | 107 |
| C — Use of this part of ISO/IEC 9506 for batch processing. (Informative) ..... | 113 |
| D — Block symbol definitions (Informative) .....                               | 115 |

## Figures

|                                                    |    |
|----------------------------------------------------|----|
| 1 — Classes of communications.....                 | 12 |
| 2 — Interaction in a peer to peer environment..... | 17 |
| 3 — An example hardware configuration.....         | 18 |

## Tables

|                                                                                               |     |
|-----------------------------------------------------------------------------------------------|-----|
| 1 — DefineEventCondition parameter extensions .....                                           | 31  |
| 2 — GetEventConditionAttributes parameter extensions.....                                     | 32  |
| 3 — AlterEventConditionMonitoring parameter extensions.....                                   | 34  |
| 4 — DefineEventEnrollment parameter extensions .....                                          | 36  |
| 5 — GetEventEnrollmentAttributes parameter extensions.....                                    | 37  |
| 6 — AlterEventEnrollment parameter extensions .....                                           | 38  |
| 7 — EventNotification parameter extension .....                                               | 39  |
| 8 — Control Element Parameter .....                                                           | 44  |
| 9 — Interaction of Unit Control primitives.....                                               | 47  |
| 10 — InitiateUnitControlLoad service.....                                                     | 47  |
| 11 — UnitControlLoadSegment service .....                                                     | 49  |
| 12 — UnitControlUpload service .....                                                          | 51  |
| 13 — StartUnitControl service.....                                                            | 55  |
| 14 — StopUnitControl service .....                                                            | 57  |
| 15 — CreateUnitControl service.....                                                           | 59  |
| 16 — AddToUnitControl service.....                                                            | 61  |
| 17 — RemoveFromUnitControl service.....                                                       | 63  |
| 18 — GetUnitControlAttributes service .....                                                   | 64  |
| 19 — LoadUnitControlFromFile service .....                                                    | 66  |
| 20 — StoreUnitControlToFile service.....                                                      | 68  |
| 21 — DeleteUnitControl service .....                                                          | 70  |
| 22 — DefineEventConditionList service .....                                                   | 72  |
| 23 — DeleteEventConditionList service .....                                                   | 75  |
| 24 — AddEventConditionListReference service .....                                             | 77  |
| 25 — RemoveEventConditionListReference service .....                                          | 80  |
| 26 — GetEventConditionListAttributes service .....                                            | 82  |
| 27 — ReportEventConditionListStatus service .....                                             | 84  |
| 28 — AlterEventConditionListMonitoring service .....                                          | 87  |
| 29 — Init Request Detail parameter .....                                                      | 89  |
| 30 — Init Response Detail parameter .....                                                     | 91  |
| 31 — Conformance classes .....                                                                | 97  |
| 32 — Service requirements for conformance classes.....                                        | 98  |
| 33 — Parameter requirements for conformance classes .....                                     | 102 |
| 34 — Additional service CBBs .....                                                            | 103 |
| 35 — Additional parameter CBBs .....                                                          | 104 |
| 36 — Application objects and MMS objects in Process Control and Programmable Controllers..... | 112 |

# **Industrial Automation Systems — Manufacturing Message Specification —**

## **Part 6: Comparison Standard for Process Control**

---

### **Foreword**

---

This part of ISO/IEC 9506 was developed by IEC/TC 65C and circulated to the IEC national bodies for approval. It presents new material not previously covered by other standards. Annex A is normative; Annexes B, C and D are informative. This is one of several standards intended as a companion to ISO/IEC 9506, Manufacturing Message Specification (MMS). This part of this Standard deals with control systems as found in the process industries. This part of this Standard should be used when process control systems are connected to a network employing the MMS services and protocol. Together with MMS and its other companion standards, this part of this Standard will enable the networking of different classes of programmable devices on the factory floor.



---

# Introduction

---

## General

This document is a part of a standard developed to facilitate the interconnection of information processing systems. It is positioned within the application layer of the Open Systems Interconnection environment as an application service element (ASE) with respect to other related standards by the basic reference model for open systems interconnection (ISO 7498)

The aim of open systems interconnection is to allow, with a minimum of technical agreement outside the interconnection standards, the interconnection of information processing systems:

- a) From different manufacturers
- b) Under different managements
- c) Of different levels of complexity
- d) Of different evolutionary implementations

This part of ISO/IEC 9506 is concerned with the communications and interworking of programmable devices of industrial process control systems utilized in the process industries.

## Purpose

The purpose of this part of ISO/IEC 9506 is to augment the use of the Manufacturing Message Specification, ISO/IEC 9506 1 and ISO/IEC 9506 2, for process control applications

This part of ISO/IEC 9506 is a companion standard to the Manufacturing Message Specification (MMS). It also uses and references the Association Control Service Element Definition (ISO 8649) whose provisions it assumes in order to accomplish the aims of the Manufacturing Message Specification. In the Process Control Environment, three forms of communication are recognized; these classes are depicted in [Figure 1](#) and described as follows:

Class A - Communications between a computer and a process control system (PCS), or communications between a computer and a distributed process control system (DCS). The computer performs "higher level" functions which are not part of the PCS or DCS functionality. These functions may include supervisory control, production control and management, remote diagnosis, expert system advice, or any combination of these or other functions not contained within the specific PCS or DCS employed to serve the application. Communications within a DCS may be proprietary to the manufacturer of the DCS, in which case a gateway is required at the DCS to provide communications services and protocols in conformance with this part of ISO/IEC 9506. The gateway functions may reside within a special device at the DCS or may share a DCS device with other functionality. Communications with a gateway may be used to connect PCS systems, or DCS systems, or both where necessary.

Class B - Communications between the component devices of the DCS. The DCS is composed of devices from multiple manufacturers, and communications be-

tween devices take place using the services and protocol specified by this part of ISO/IEC 9506.

Class C - Communications between a PCS or a DCS and field devices including sensors, actuators and field multiplexers, as well as communication between field devices.

The application description in Clause 5 of this part of ISO/IEC 9506 focuses primarily on Class A communications. The description may be partially applicable to Class B communications. Interworking of systems conforming to this part of ISO/IEC 9506 and systems supporting Class C communications may be the subject of other standardization efforts, e.g., the International Fieldbus work.

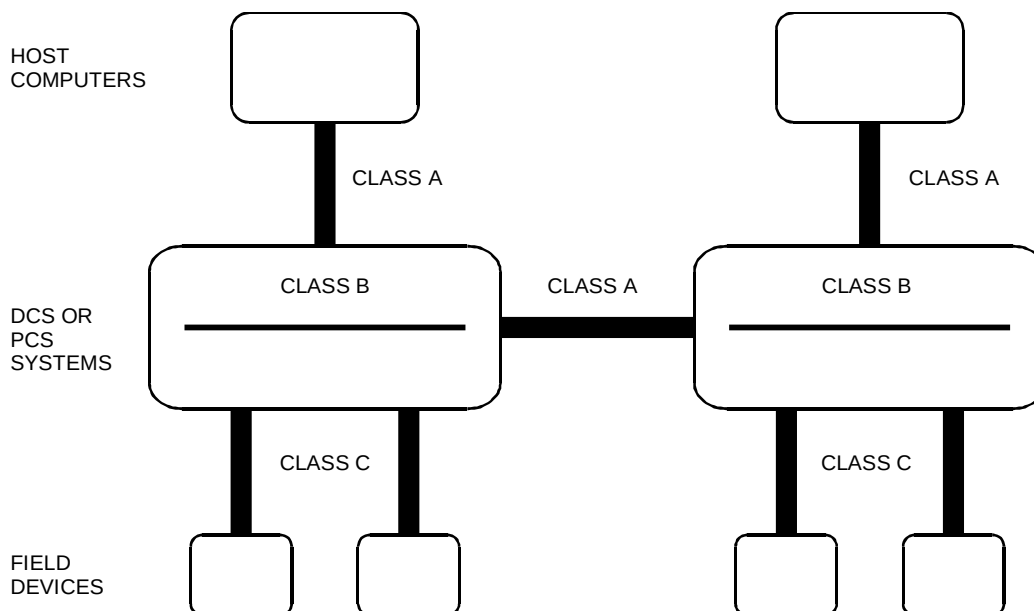


**Figure 1 — Classes of communications**

This part of ISO/IEC 9506 emphasizes communications in support of supervisory monitoring and control, or class A communications. Specific features of this communication include, but are not limited to:

- a) Peer to peer communications between computers of various implementations, used for production control and management, and various process control systems
- b) Access to control and monitoring Blocks and their attributes for the purpose of achieving improved control
- c) Peer to peer communication with other equipment used in the process control application, such as programmable controllers

---

## 1 Scope

---

This part of ISO/IEC 9506 describes the use of the Manufacturing Message Specification in the Process Control Environment in terms of:

- a) The interaction requirements of process control applications
- b) A set of abstract models defining the interaction between process control applications
- c) The externally visible functionality of implementations conforming to this part of ISO/IEC 9506 in the form of procedural requirements associated with the execution of service requests
- d) Objects required to be externally visible at implementations conforming to this part of ISO/IEC 9506, in order to support the interactions of process control applications
- e) Conformance requirements to this part of ISO/IEC 9506, which include minimum requirements or simple applications as described in Clause 5

This part of ISO/IEC 9506 specifies requirements for the server role, except where specifically noted otherwise.

This part of ISO/IEC 9506 is for use between a supervisory computer and any type of programmable equipment used in process control, including programmable controllers. In addition, it may be used between two instances of programmable equipment.

This part of ISO/IEC 9506 is a companion standard to the Manufacturing Message Specification, ISO/IEC 9506 1 and ISO/IEC 9506 2, designed to support messaging communications to and from programmable devices in an integrated process control application environment. This environment is referred to in this part of ISO/IEC 9506 as the Process Control Environment.

This part of ISO/IEC 9506 specifies a particular mapping of the Process Control Environment onto the Virtual Manufacturing Device (VMD) model and describes the use of the Manufacturing Message Specification services within this environment. This part of ISO/IEC 9506 specifies additions to the services of the Manufacturing Message Specification to support the requirements of the Process Control Environment and specifies additional protocol to support these services.

---

## 2 Normative References

---

The following standards contain provisions which, through reference in this text, constitute provisions of this part of ISO/IEC 9506. At the time of publication, the editions indicated were valid. All standards are subject to revision, and parties to agreements based on this part of ISO/IEC 9506 are encouraged to investigate the possibility of applying the most recent editions of the standards listed below. Members of IEC and ISO maintain registers of currently valid International Standards.

|                            |                                                                                                                                                                                                                      |
|----------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>ISO 646:1983</b>        | <b><i>Information Processing - ISO 7 bit Coded Character Set for Information Interchange</i></b>                                                                                                                     |
| <b>ISO 7498 1:1984</b>     | <b><i>Information processing systems - Open systems Interconnection Basic Reference Model</i></b>                                                                                                                    |
| <b>ISO TR/8509:1987</b>    | <b><i>Information processing systems - Open systems interconnection Service Conventions</i></b>                                                                                                                      |
| <b>ISO 8824:1987</b>       | <b><i>Information processing systems - Open systems interconnection Specification of Abstract Syntax Notation One (ASN.1)</i></b>                                                                                    |
| <b>ISO 8649:1988</b>       | <b><i>Information processing systems - Open systems interconnection Service definition for the Association Control Service Element</i></b>                                                                           |
| <b>ISO 8649/AM 1:1990</b>  | <b><i>Information processing systems - Open systems interconnection Service definition for the Association Control Service Element, Amendment 1 covering authentication during association establishment</i></b>     |
| <b>ISO 8650:1988</b>       | <b><i>Information processing systems - Open systems interconnection Protocol Specification for the Association Control Service Element</i></b>                                                                       |
| <b>ISO 8650/AM 1:1990</b>  | <b><i>Information processing systems - Open systems interconnection Protocol Specification for the Association Control Service Element, Amendment 1 covering authentication during association establishment</i></b> |
| <b>ISO/IEC 9506 1:1990</b> | <b><i>Industrial automation systems - Manufacturing message specification Service definition</i></b>                                                                                                                 |
| <b>ISO/IEC 9506 2:1990</b> | <b><i>Industrial automation systems - Manufacturing message specification Protocol specification</i></b>                                                                                                             |

---

## 3 Definitions

---

For the purposes of this part of ISO/IEC 9506, the definitions in the following subclauses apply.

### 3.1 Reference model definitions

Clause 3 of ISO/IEC 9506 1 and ISO/IEC 9506 2 list a number of terms defined in ISO 7498, in ISO TR 8509, and in ISO 8824 as well as its own definitions. These definitions are included in this part of ISO/IEC 9506 by reference.

### 3.2 Definitions unique to this part of ISO/IEC 9506

For the purpose of this part of ISO/IEC 9506, the following definitions also apply:

**3.2.1 batch control computer:** A computer used to monitor and control the execution of a product that is manufactured in discrete batches.

**3.2.2 cascade structure:** A control structure in which the output variable of one controller is the reference variable for one or more secondary control loops.

**3.2.3 configuration:** The result of customizing a general control system for a particular physical location, application, class of application, or any combination thereof.

**3.2.4 control, feedback:** Control in which a measured variable is compared to its desired value to produce an actuating error signal that is acted upon in such a way as to reduce the magnitude of the error.

**3.2.5 control loop:** The collection of I/O equipment, algorithms and other hardware and software modules used to implement feedback control.

**3.2.6 control, supervisory:** Control in which the control loops operate independently subject to intermittent corrective action, e.g., set point changes from an external source.

**3.2.7 control system:** A system in which deliberate guidance or manipulation is used to achieve a prescribed value of a variable.

**3.2.8 controller:** A device that operates automatically to regulate a controlled variable.

**3.2.9 controller, PID:** A controller that produces proportional plus integral (reset) plus derivative (rate) control.

**3.2.10 device:** An apparatus for performing a prescribed function.

**3.2.11 element:** A component of a device or system.

**3.2.12 hardware:** Physical equipment directly involved in performing industrial process measuring and controlling functions.

**3.2.13 historian:** A system whose function is to archive records of the actions taken within the process control system.

**3.2.14 linking device:** A device which provides a gateway function between two dissimilar networks, e.g., between the backbone and the control network, or between the backbone and the Fieldbus.

**3.2.15 loop controller:** A controller which operates a control loop.

**3.2.16 multiloop controller:** A controller which operates more than one control loop.

**3.2.17 operator's console:** A system for the monitoring or operation of a process or both. An operator's console includes the Operator Station object of ISO/IEC 9506 1. It is usually capable of colour graphic displays with live updating data.

**3.2.18 parameter:** A quantity or property treated as a constant but which may sometimes vary or be adjusted.

**3.2.19 process:** Physical or chemical change of matter or conversion of energy, e.g., change in pressure, temperature, speed, electrical potential, etc.

**3.2.20 process control:** The regulation or manipulation of variables influencing the conduct of a process in such a way as to obtain a product of desired quality and quantity in an efficient manner.

**3.2.21 process monitoring application:** An application that may provide any of the following functions: observe process trends, initiate actions for process optimization, involve the use of expert system technology.

**3.2.22 reflexive action:** See reflexive processing.

**3.2.23 reflexive processing:** Action taken without consultation with an operator or supervisory computer, usually conditioned on the occurrence of specified condition or event.

**3.2.24 regulatory control:** A synonym for feedback control. (See 3.2.4.)

**3.2.25 signal conditioning;** The act of operating on a signal to make it useful to the device which will utilize the signal. Examples include noise suppression and inversion.

**3.2.26 signal inversion:** A process in signal conditioning in which the sign of the signal (+, -) is "inverted" to the other sign.

---

## 4 Abbreviations

---

|      |                                               |
|------|-----------------------------------------------|
| CBB  | Conformance Building Block                    |
| DCS  | Distributed Process Control System            |
| MMS  | Manufacturing Message Specification           |
| PDU  | Protocol Data Unit                            |
| PICS | Proforma Implementation Conformance Statement |
| PID  | Proportional Integral Derivative              |
| PC   | Programmable Controller                       |
| PCS  | Process Control System                        |
| VMD  | Virtual Manufacturing Device                  |

---

## 5 Application description

---

The interactions between applications in the Process Control Environment take place between communicating peers in support of the exchange of control information or monitoring information or both. During the lifetime of any instance of an application association, a real system may adopt the client role or the server role or both. This part of ISO/IEC 9506 places no restrictions on the behaviour of a client, other than the implied requirement that the system acting in the client role be capable of issuing appropriate requests and receiving appropriate responses. In very simple environments, a single application association may suffice between two communicating peers. In more complex environments, there may be a necessity for more than one application association.

### 5.1 Process control models

#### 5.1.1 Process manufacturing communication models

##### 5.1.1.1 Peer to peer communication model

Figure 2 illustrates peer to peer communication which may be applied between a cooperating client and server in the Process Control Environment for the purpose of optimizing behaviour in a control system. This model may be applied to communications between a batch control application and a monitoring and control device or between a process monitoring application and a large, distributed process control system, where the distributed process control system is treated as a single device containing numerous monitoring and control elements. This model may be used between two distributed process control systems, each treated as a single device, and may also be applied to interactions between elements of a distributed process control system or between control elements distributed on a control communications highway.

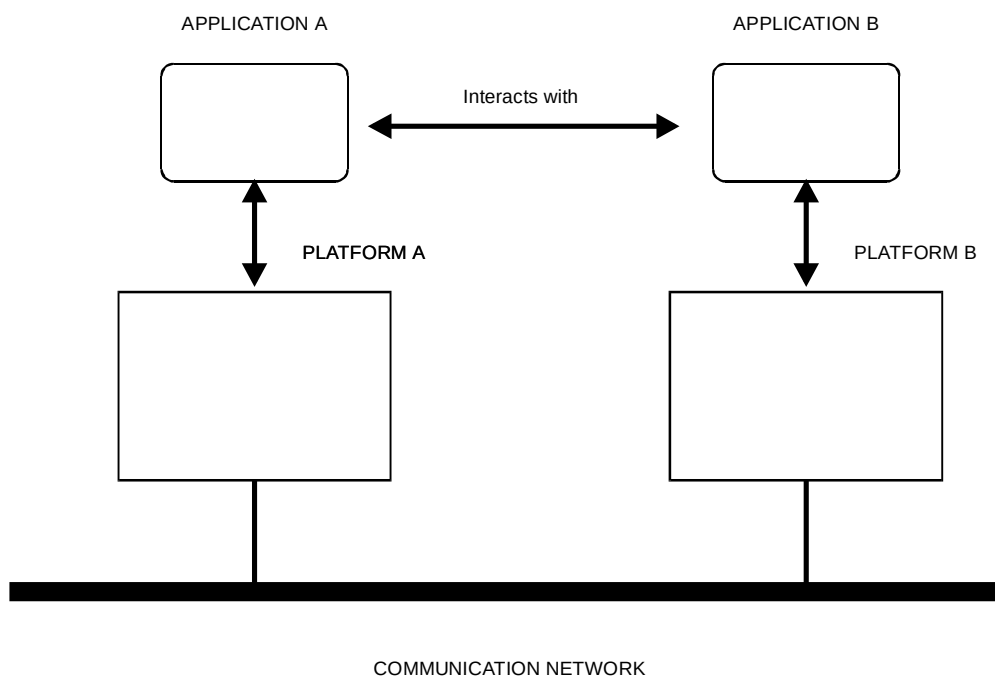


**Figure 2 — Interaction in a peer to peer environment**

### 5.1.1.2 Multi-tiered example:

Figure 3 shows an example of a hardware configuration which integrates a distributed process control system with one or more host computers. The distributed process control system's internal network is represented by the lower horizontal line, and its component elements are shown as boxes attached to the internal network. The elements shown are representative of those commonly found in distributed control systems, and include analog and digital input/output controllers, single and multiloop controllers, a programmable controller (PC), batch control computers, a historian, operator workstations and a device (linking device) which performs a linking (potentially a gateway) function between the distributed process control system's internal network and the supervisory network. The supervisory network is represented by the upper horizontal line, and the host computers are attached to the supervisory network.

In this example, a Production Control application operating on a host computer may be used to download batch recipes and retrieve batch product data. The host computer may also communicate with the operator's consoles to setup displays or give operator instructions. The host computer or operator's console may also recall the data from the historian. In this case, the historian may act as a client when collecting data from controllers and as a server when providing data for the host computer or the operator's console. Where application circumstances warrant, some equipment may be provided with connections to both the control network and the supervisory network. One example of such an architecture might include a historian and an operator's console, which may rely on the control network for data acquisition, and may additionally rely on the supervisory network for communications of a supervisory nature.

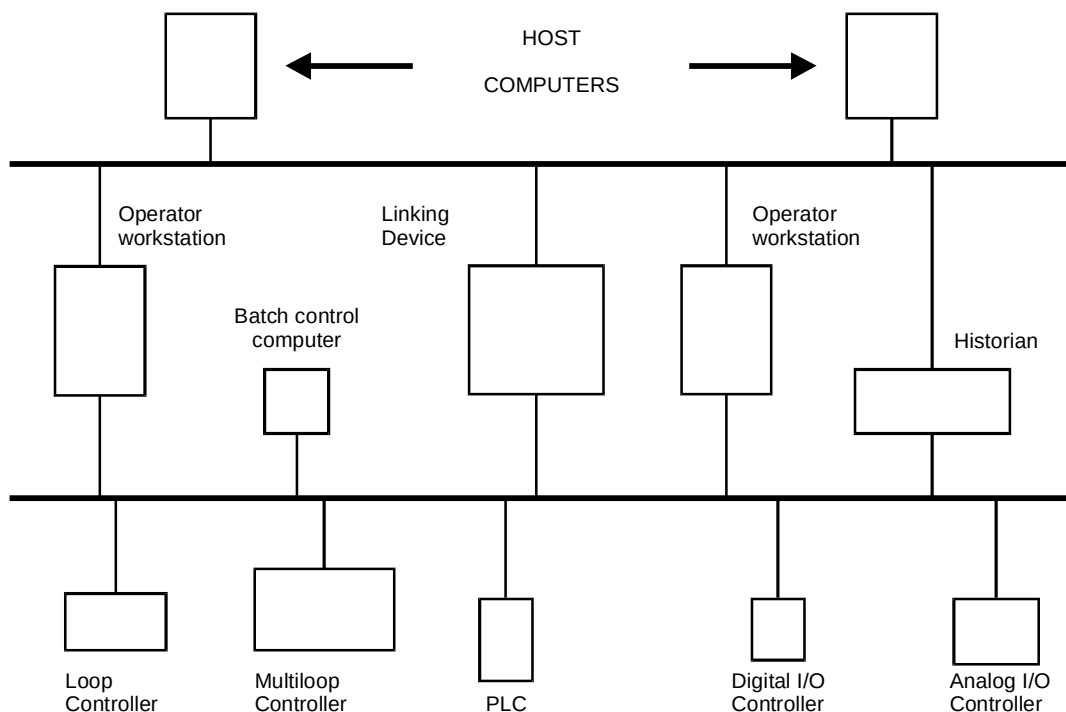


Figure 3 — An example hardware configuration

## 5.2 Process control functions

### 5.2.1 Support for regulatory control

#### 5.2.1.1 General

In the Process Control Environment, regulatory control is conceptually accomplished through the utilization of Blocks, such as signal conditioning, mathematical conversions, X Y characterizations and Proportional Integral Derivative (PID) control loops. Blocks represent real instances of functions in a real process control system.

Simple functions may be combined to form complex functions by connecting the output of one Block to the input of another Block. Each Block has associated with it a set of Block parameters which represent the inputs, outputs, and internal state variables required by the function.

#### 5.2.1.2 Protection

Blocks are subject to alteration during operation of the process control system. This alteration may take the form of tuning block parameters for optimal operation. It is a requirement that Blocks be protected against unauthorized modification; in order to accomplish this, it is necessary that the identity and privilege of the requester be determined before a Block is modified.

#### 5.2.1.3 Groups of Blocks

Since, in general, there will be a large number of Block instances employed in a typical process control system, it is a requirement that there be an efficient way of handling groups of Blocks. This includes facilities for downloading and uploading Blocks, starting and stopping groups of Blocks, and otherwise dealing with them in groups.

### 5.2.2 Management of events and alarms

#### 5.2.2.1 General

There is a functional requirement to continuously monitor processes under control and identify as "process events" the occurrence of certain conditions related to the state of the process under control. Management of a process event implies an ability to communicate the occurrence of the process event to a remote application. There is a functional requirement for the creation of a "notification request," either by or on behalf of a communicating peer, which will result in the creation and transmission of a notification message to the communicating peer containing information about the transition into the specified condition and process state.

When the occurrence of an event is communicated, there is a functional requirement to allow inclusion of textual strings, or identifying numbers, in the message communicating the occurrence. Depending on application considerations, the content of the string or identifying number may be identical for each reported occurrence, or different for each reported occurrence. This information is referred to as "display enhancements."

**NOTE** — Provision has been included for both numbers and text strings due to the requirement to be able to display the string directly on an operator's console or to use the number to index into an array containing multiple strings in different languages.

Under certain circumstances determined by an application or a configuration process, the response to a process event may include "reflexive actions" which are executed as autonomous actions, as a direct result of detection of the process event. A requirement exists to communicate the results of the reflexive action to communicating peers which have a desire to be aware of the results. As an efficiency move, the results of reflexive actions are communicated with the report

of the occurrence of a process event. The creation of a notification request will be used to specify a request for reflexive processing.

**Example:**

1 - A reflexive action may be a simple act, such as the starting of a Program Invocation, which may lead to several complex actions managed by the system.

**Example:**

2 - A reflexive action may be an action which must be taken in the interests of personal safety or safety to equipment. The dumping of the contents of a quench tank into an exothermic chemical reaction which is running away constitutes another example.

When process events require acknowledgement by a remote entity, whether a human operator or a supervisory application, the event is termed an "alarm" and there is a functional requirement for related alarm management techniques such as display, recording, and acknowledgement.

### **5.2.2.2 Event and alarm groupings**

In certain process control applications, there is a functional requirement for a capability to group process events into "Process Event Groups," for purposes of status inquiry and alarm management. In most cases, such groups are associated with a unit of equipment or plant, or in some cases represent a grouping of groups, or an instance of a hierarchical relationship of groups. It is a functional requirement that the hierarchical relationships between groups be preserved, while allowing changes of groups at lower levels in the hierarchy to be reflected in higher level groups.

**Example:**

The addition of a temperature sensor to a feedwater pump group should be automatically included when the points in the boiler support groups are requested.

An additional function requirement is a capability to dynamically override the priority of individual events in process event groups.

**NOTE** — This capability has utility during process start up and shutdown, as well as facility maintenance, where groups of event and alarm conditions are managed in correspondence to units of affected plant. Priority of individual events is described in ISO 9506 1.

### **5.2.2.3 Alarm suppression**

During maintenance periods, process start up, or abnormal shutdown, there is a functional requirement to suppress notifications of process events which are reporting conditions known to be abnormal. There is a functional requirement for a capability to perform event and alarm suppression on groups of process events, corresponding to affected units of plant, or equipment. When conditions are stable, an additional functional requirement exists to return event and alarm reporting to normal.

### **5.2.3 Support for batch manufacturing**

In many process industries applications, production occurs in "batches," i.e., aggregations of large quantities of material to be made into final products. Typically, one computer, the "batch control computer," will contain the master plan for this batch manufacture, and the process will be carried out under the control of one or more process control computers. In dealing with the interactions between the batch control computer and the process control computers, we can identify five types of requirements.

#### **5.2.3.1 Determination of the status of a batch process**

The batch control computer needs to be able to determine the status of the batch process or the status of the components of the batch process as sensed at the process control computer.

#### **5.2.3.2 Control of a batch process**

The batch control computer needs to be able to control the batch process by changing the state of the control applications operating in the process control computer.

#### **5.2.3.3 Alteration of the control of a batch process**

In some cases, it may be necessary for a human operator present at the process control computer to assume control of the process. There is a need for an orderly way for the batch control computer to relinquish control in such circumstances.

#### **5.2.3.4 Direct communication with a human operator**

In some cases, it may be necessary for the batch control computer to display information to a human operator present at the process control computer to solicit his input.

#### **5.2.3.5 Synchronization and coordination**

The batch control computer may need to coordinate the use of resources with other processes in the Process Control Environment. Such resources may be a common source of supply of raw materials or a shared device such as an error logging printer.

### **5.2.4 Support for creation and retrieval of historical records**

There is a functional requirement for the retention of information related to a particular process or processes. The information should be available in chronological order when retrieved. Information typically stored in "historical records" includes time stamped records of process events, time stamped records of process variables of interest, either recorded in conjunction with records of process events or independently, and comments entered by a human operator.

### **5.2.5 Support for process control system configuration and management**

A functional requirement exists for the transfer of configuration data from a host computer into an unconfigured as well as into an operational process control system; conversely, a functional requirement exists for a capability to retrieve configuration data for off system storage following debugging. Configuration data includes, but is not limited to, program code, data tables, and code patches.

## **5.3 Application Models**

### **5.3.1 Block**

The Block conceptually represents an installed instance of a control or monitoring function in a process control system. The basic functional aspects of a Block necessary for prescribing communication behaviour are described here. Further information about Blocks and their use in the Process Industry may be found in Annex B.

#### **Example:**

Blocks may be utilized to perform signal conditioning, mathematical or complex monitoring, and control functions.

Model: Block

Key Attribute: Block Tag

Attribute: InService (TRUE, FALSE)

Attribute: Algorithm reference

Attribute: Mode State

Attribute: List of Block Parameters

#### **5.3.1.1 Block Tag**

This attribute is the name of the Block. It is unique throughout the process control system. It is assigned by the end user or configuration manager of the process control system.

#### **5.3.1.2 InService**

This attribute indicates whether (true) or not (false) the Block is actively operating.

#### **5.3.1.3 Algorithm Reference**

This attribute contains a reference to the function algorithm. There is a reference for each Block in a real control system; however, use of a common algorithm for multiple Blocks is not precluded so long as individual references from each Block are maintained.

#### **5.3.1.4 Mode State**

This attribute governs the operation of the Block and the protection of its elements. Further information can be found in Annex B.

#### **5.3.1.5 List of Block Parameters**

This attribute is the list of Block parameters which represents the inputs, outputs, and internal state variables of the Block.

### **5.3.2 Algorithm**

The Algorithm object represents the algorithm used by a Block instance to transform its inputs into its outputs. It is a separate object to reflect the fact that many Block instances may share a common algorithm.

Object: Algorithm

Key Attribute: Algorithm Name

Attribute: Algorithm content

#### **5.3.2.1 Algorithm Name**

This attribute is the name associated with the algorithm.

#### **5.3.2.2 Algorithm content**

This attribute represents the substance of the algorithm, the program or rules by which the output is determined based on the input, and the internal state.

---

## 6 Process control context mapping

---

This clause prescribes how the facilities of MMS shall be used to satisfy the communication requirements of the Process Control Environment described in the previous clause. This clause also relates the models of process control systems to the abstract model of the Virtual Manufacturing Device (VMD) described in ISO/IEC 9506 1.

### 6.1 Mapping the process control model to the VMD

Each device of a process control system that offers Class A communication capabilities as a server shall be mapped onto at least one VMD.

#### Example:

A programmable controller engaged in a process control application may consist of a single VMD. A large distributed process control system, when viewed through a gateway from a production control computer, may also consist of a single VMD. A distributed process control system, when viewed from an internal element of the control system, such as an operator's console, may contain multiple VMDs with at least one VMD representing each station connected to the distributed process control system. A large distributed process control system with multiple connections to the supervisory network from the system gateway, the historian, and operator's console may appear, from the supervisory network, as multiple VMDs.

### 6.2 Definition of process control objects that map to Domains

#### 6.2.1 The Block object

The Block object shall map to a Domain and to a Program Invocation bound to this Domain. The Domain shall contain Domain specific Named Variable objects that shall represent the inputs, outputs, and internal state parameters of the Block. The Mode State attribute of the Block shall also be represented as a Named Variable object of the Domain. If the syntax of the Block Tag is compatible with the syntax of an Identifier (see 7.6.2 of ISO/IEC 9506 2), the name of the Domain shall be the same as the Block Tag. If the syntax of the Block Tag is not compatible with Identifier syntax, there are no requirements on the Domain name. However, Annex B.4 provides guidance on a recommended method for deriving a Domain name from the Block Tag.

#### 6.2.2 The Algorithm object

The Algorithm object shall map to a Domain. The Domain name shall be the name of the Algorithm.

### 6.3 Definition of process control objects that map to Program Invocations

#### 6.3.1 The Block object

The Block object shall map to a Domain and to a Program Invocation bound to this Domain. If the syntax of the Block Tag is compatible with the syntax of an Identifier (see 7.6.2 of ISO/IEC 9506 2), the name of the Program Invocation shall be the same as the Block Tag. If the syntax of the Block Tag is not compatible with Identifier syntax, there are no requirements on the Program

Invocation name. However, Annex B.4 provides guidance on a recommended method for deriving a Program Invocation name from the Block Tag.

The Program Invocation representing the Block will normally have two Domains bound to it, one whose name is the same as the Program Invocation supports the Named Variables that represent the inputs, outputs, internal state variables, and Mode State of the Block. The other Domain, which may be sharable, represents the algorithm for the Block execution. Depending on the nature of the Block, there may be additional Domains bound to the Program Invocation. In some cases, the entire Block may be represented by a single Domain bound to a Program Invocation.

The state of the Program Invocation is used in this model to represent the InService attribute of the Block. When the Program Invocation is in the IDLE or STOPPED state, the InService attribute is FALSE. When the Program Invocation is in the RUNNING state, the InService attribute is TRUE.

## **6.4 Process control requirements that affect other MMS objects**

### **6.4.1 Extensions to the Event Condition object**

The Event Condition object in Clause 15 of ISO/IEC 9506 1 is extended by the following attributes:

Object: Event Condition

All MMS defined attributes

Attribute: Display Enhancement Class (TEXT, NUMBER, UNDEFINED)

Constraint: Display Enhancement Class = TEXT

Attribute: Display Enhancement (String)

Constraint: Display Enhancement Class = NUMBER

Attribute: Display Enhancement (Integer)

Attribute: Group Priority Override

Attribute: List of referencing Event Condition List references

#### **6.4.1.1 Display Enhancement Class**

This attribute identifies the type of the Display Enhancement attribute of the Event Condition object. If the value of this attribute is TEXT, the Display Enhancement is of type character string. If the value of this attribute is NUMBER, the Display Enhancement is of type integer. If the value of this attribute is UNDEFINED, the Display Enhancement attribute is not present.

#### **6.4.1.2 Display Enhancement**

This attribute is used to contain information useful in preparing operator displays at the MMS user receiving notification of the event. The type of this attribute is either character string or integer, depending on the value of the Display Enhancement Class attribute.

#### 6.4.1.3 Group Priority Override

This attribute shall contain a value that is either UNDEFINED or an integer value between zero (0) and one hundred twenty-seven (127). Zero shall represent the highest priority and one hundred twenty-seven shall represent the lowest priority. If the value of this attribute is defined, it shall represent an override priority value that shall be utilized by the VMD in place of the value contained in the Priority attribute. If the value of the Group Priority Override attribute is UNDEFINED, the VMD shall utilize the value of the Priority attribute in determining the importance of the Event Condition object. When an Event Condition object is created using the Define Event Condition service, the value of this attribute shall be initialized to UNDEFINED.

##### Example:

Consider the case in which an Event Condition object's priority is modified using a service operating on an Event Condition List, and then individually altered using the AlterEventConditionMonitoring service, and then modified again at the group priority level using a service operating on an Event Condition List. Following the first change, the Group Priority Override attribute has changed and takes precedence over the Priority attribute. The Priority attribute has not changed, but is not considered since there is a group override in effect. The Priority attribute does change following the Priority change, but is still not considered until the group override is removed.

The Group Priority Override attribute imposes an absolute override, whether the value is higher or lower than the value of the Priority attribute.

This attribute need not be implemented if Event Condition List objects are not supported.

#### 6.4.1.4 List of referencing Event Condition List references

This attribute shall contain a list of references to Event Condition List objects that reference the specific Event Condition object. The Event Condition List object is defined in of this part of ISO/IEC 9506. When an Event Condition object is created using the DefineEventCondition service, the value of this attribute shall be initialized to an empty list.

This attribute need not be implemented if Event Condition List objects are not supported.

### 6.4.2 Extensions to Event Enrollment objects

The Event Enrollment object in described in Clause 15 of ISO/IEC 9506 1 is extended as follows:

Object: Event Enrollment

All MMS defined attributes

Attribute: Display Enhancement Class (TEXT, NUMBER, UNDEFINED)

Constraint: Display Enhancement Class = TEXT

Attribute: Display Enhancement (String)

Constraint: Display Enhancement Class = NUMBER

Attribute: Display Enhancement (Integer)

#### 6.4.2.1 Display Enhancement Class

This attribute identifies the type of the Display Enhancement attribute of the Event Enrollment object. If the value of this attribute is TEXT, the Display Enhancement is of type character string. If the value of this attribute is NUMBER, the Display Enhancement is of type integer. If the value of this attribute is UNDEFINED, the Display Enhancement attribute is not present.

### 6.4.2.2 Display Enhancement

This attribute is used to contain information useful in preparing operator displays at the MMS user receiving notification of the event. The type of this attribute is either character string or integer, depending on the value of the Display Enhancement Class attribute.

## 6.5 Definitions of new MMS abstract objects

### 6.5.1 Event Condition List object

The Event Condition List object shall be used to reference groups of Event Condition objects that are required to be operated on as groups. Support of the RECL (See 6.6.3) CBB indicates support for Event Condition Lists that may refer to other Event Condition Lists. If this CBB is not supported, the list may contain only references to Event Conditions.

Object: Event Condition List

Key Attribute: Event Condition List name

Attribute: MMS Deletable

Attribute: List of Event Condition references

Attribute: List of Event Condition List references

Attribute: List of referencing Event Condition List references

#### 6.5.1.1 Event Condition List name

This attribute, of type Object Name, shall be the name by which the Event Condition List is identified. An Event Condition List name may have VMD, Domain specific, or AA specific scope.

#### 6.5.1.2 MMS deletable

This attribute, of type Boolean, shall indicate whether (true) or not (false). This object may be deleted through the use of the service defined in 7.3.15 of this part of ISO/IEC 9506.

#### 6.5.1.3 List of Event Condition references

This attribute shall contain a list of references to Event Condition objects. The scope of services operating on the Event Condition List object shall include objects in both the List of Event Condition references attribute and the List of Event Condition List references attribute.

Because of visibility constraints, if the Event Condition List name has VMD specific or Domain specific scope, this attribute shall contain references only to Event Condition objects which have VMD specific or Domain specific scope. If the Event Condition List name attribute has AA specific scope, this attribute may reference Event Condition objects of any scope.

#### 6.5.1.4 List of Event Condition List references

This attribute shall contain a list of references to Event Condition List objects that are hierarchically subordinate to this Event Condition List object. It shall not be permissible for this attribute, nor for this attribute in subordinate Event Condition List objects, to contain circular references. The scope of services operating on the Event Condition List object shall include objects included in both the List of Event Condition references attribute and the List of Event Condition List references attribute. The List of Event Condition List references attribute shall be present only if the RECL CBB has been negotiated.

Because of visibility constraints, if the Event Condition List name has VMD specific or Domain specific scope, this attribute shall contain references only to Event Condition List objects which have VMD specific or Domain specific scope. If the Event Condition List name attribute has AA specific scope, this attribute may reference Event Condition List objects of any scope.

#### **6.5.1.5 List of referencing Event Condition List references**

This attribute shall contain a list of references to other Event Condition List objects that reference this specific Event Condition List object. If there are no references to this Event Condition List object, the value of this attribute shall be an empty list. This attribute shall be present only if the RECL CBB has been negotiated.

**NOTE** — This attribute is necessary to fully describe the service procedures that operate on the Event Condition List object. This attribute is not visible nor modifiable via the services provided in this part of ISO/IEC 9506.

### **6.5.2 Unit Control object**

The Unit Control object shall represent a collection of MMS objects representing groups of Domains and/or Program Invocations that may be loaded and managed as a collective unit. Downloads and uploads may be performed on a Unit Control object to efficiently transmit the information representing multiple Domains to and from a process control system in one or more download or upload sequences. While introduced to satisfy the requirement of manipulating groups of Blocks, the Unit Control object may be used to manipulate groups of Domains and/or Program Invocations. Services that operate on the Unit Control object are described in 7.3.1 through 7.3.13.

**NOTE** – The Unit Control object may be utilized to minimize the number of PDU's necessary to download large numbers of Domains or to create large numbers of Program Invocations. Rules concerning the applicability of grouping of these objects is outside the scope of this part of ISO/IEC 9506.

Object: Unit Control

Key Attribute: Unit Control Name

Attribute: MMS Deletable (TRUE, FALSE)

Attribute: List of Domain references

Attribute: List of Program Invocation references

#### **6.5.2.1 Unit Control Name**

This attribute shall uniquely identify the Unit Control object at the VMD. The name scope of the Unit Control object shall be VMD specific.

#### **6.5.2.2 MMS Deletable**

This attribute shall indicate whether (true) or not (false) the Unit Control object may be deleted using the DeleteUnitControl service.

#### **6.5.2.3 List of Domain references**

This attribute shall identify the Domain objects that are constituents of the Unit Control object and that may be affected by operations on the Unit Control object.

#### **6.5.2.4 List of Program Invocation references**

This attribute shall identify the Program Invocation objects that are constituents of the Unit Control object and may be affected by operations on the Unit Control object.

## 6.6 Parameter conformance

This part of ISO/IEC 9506 introduces three new parameter conformance building blocks (CBB's) in the system specification.

### 6.6.1 DES

The DES parameter conformance building block shall establish the validity of the text form of the display enhancement parameter, whenever it occurs in a request or indication in a service table.

If DES is supported, the parameter is valid. Otherwise, the parameter is invalid. If DES is not supported, a request specifying this parameter shall constitute a protocol error.

If DES is supported, DEI shall not be supported.

### 6.6.2 DEI

The DEI parameter conformance building block shall establish the validity of the numeric form of the display enhancement parameter, whenever it occurs in a request or indication in a service table.

If DEI is supported, the parameter is valid. Otherwise, the parameter is invalid. If DEI is not supported, a request specifying this parameter shall constitute a protocol error.

If DEI is supported, DES shall not be supported.

### 6.6.3 RECL

The RECL parameter conformance building block shall establish the validity of the List of Event Condition List parameter, whenever it occurs in a service table.

If RECL is supported, this parameter is valid. Otherwise, the parameter is invalid. A request or response specifying this parameter shall constitute a protocol error. If RECL is not supported, a request which, if honoured, would require a response specifying this parameter shall result in a service error specifying error class equal to ACCESS and error code equal to OBJECT ACCESS UNSUPPORTED.

---

## 7 Services

---

### 7.1 Use of ACSE services

In order to support certain MMS operations that affect the state of the process control system, the MMS server shall require that the MMS client has provided the AP title in the A ASSOCIATE service and further that the MMS client has specified the Authentication functional unit of ACSE and provided proper values of Privilege Classification and Privilege Identifier as part of the Authentication Value of the A ASSOCIATE service. These values shall become values of the corresponding attributes of the Application Association object (See Annex A). If the MMS client has not supplied such values, the corresponding attributes shall be set to a value of UNDEFINED or the MMS server may, at its option, refuse establishment of the association.

For the purpose of providing a syntax definition for the Authentication mechanism, this part of ISO/IEC 9506 assigns the ASN.1 object identifier value and the associated ASN.1 module for the external choice of Authentication value.

```
ISO 9506 Authentication { iso standard 9506 part(6)
```

```
authentication(3) }
```

```
DEFINITIONS ::= BEGIN
```

```
Authentication syntax ::= SEQUENCE {
```

```
    user                      VisibleString,
```

```
    password                  OCTET STRING OPTIONAL,
```

```
    privilege classification  INTEGER OPTIONAL
```

```
    }
```

```
END
```

## 7.2 Use of MMS services

This subclause specifies the use of services and protocol defined in ISO/IEC 9506 1 and ISO/IEC 9506 2, extensions to some of those services, and protocol to support the extensions.

### 7.2.1 Process application context name

For the purpose of being able to use an application that only contains the ACSE and MMS as ASEs, this part of ISO/IEC 9506 uses the object identifier value and the object descriptor value defined in 17.12 of ISO/IEC 9506 2.

### 7.2.2 Process control abstract syntax definition

This part of ISO/IEC 9506 assigns the ASN.1 object identifier value

```
{iso standard 9506 part (6) mms process syntax version(1)}
```

to the abstract syntax defined in this clause.

### 7.2.3 Process control specific ASN.1 module definition

The MMS services and protocol were developed to be used by a wide range of manufacturing devices. This section defines the process control services and protocol for those elements identified as requiring companion standard definition in MMS. These definitions should be used when the abstract syntax defined in this part of ISO/IEC 9506 is negotiated.

All ASN.1 definitions provided in this part of ISO/IEC 9506 are part of the ASN.1 Module "ISO 9506 MMS PROCESS 1." The beginning and closing statements indicating that each ASN.1 definition provided is a part of this module is omitted in order to make reading of the document easier. Each ASN.1 definition provided implicitly contains the statement:

```
ISO 9506 MMS PROCESS 1 {iso standard 9506 part(6) mms-process-module-  
version1(2)}
```

```
DEFINITIONS ::= BEGIN
```

at the beginning of the definition and contains the keyword "END" at the end of the definition.

**NOTE** — ISO 9506 MMS PROCESS 1 represents version number 1 of the MMS (ISO 9506) Companion Standard for Process Control.

Many of the terms and abbreviations used in this clause use the terminology of MMS service and protocol descriptions (See Clause 5 of ISO/IEC 9506 1 and Clause 5 of ISO/IEC 9506 2). In particular, refer to Clause 5 of ISO/IEC 9506 1 for general rules on how to interpret the service tables in this part of ISO/IEC 9506.

```
IMPORTS MMSpdu,
    ParameterSupportOptions,
    ServiceSupportOptions,
    Integer16,
    StatusResponse,
    Identifier,
    ObjectName,
    ProgramInvocationState,
    FileName,
    ApplicationReference,
    Priority,
    EventTime,
    EC State
FROM MMS General Module 1
    {iso standard 9506 part(2) mms-general-module-version1(2)};
```

## 7.2.4 VMD support services

### 7.2.4.1 GetNameList service extensions

The GetNameList service shall be extended to include the Event Condition List object and the Unit Control object.

#### 7.2.4.1.1 CS Object Class service parameter

This parameter, of type integer, shall identify the Event Condition List object and the Unit Control object. The value zero (0) shall represent the Event Condition List object, and the value one (1) shall represent the Unit Control object.

#### 7.2.4.1.2 Protocol extensions

The abstract syntax of the CsAdditionalObjectClasses parameter shall be specified as follows:

```
CsAdditionalObjectClasses ::= IMPLICIT INTEGER {
    eventConditionList      (0),
    unitControl              (1)
}
```

## 7.2.5 Event management services

This subclause specifies the extension of the Event Management services.

### 7.2.5.1 DefineEventCondition service

This subclause specifies the extension of the DefineEventCondition service.

#### 7.2.5.1.1 Parameter extensions

The DefineEventCondition service shall be extended to include specification of the value of the Display Enhancement attribute of the Event Condition object. The structure of the DefineEventCondition parameter extensions is specified in [Table 1](#).

**Table 1 — DefineEventCondition parameter extensions**

| Parameter name             | Req | Ind  | Rsp | Cnf | CBB |
|----------------------------|-----|------|-----|-----|-----|
| Display Enhancement        | M   | M(=) |     |     |     |
| Display Enhancement string | S   | S(=) |     |     | DES |
| Display Enhancement index  | S   | S(=) |     |     | DEI |
| No Enhancement             | S   | S(=) |     |     |     |

##### 7.2.5.1.1.1 Display Enhancement

Selection of this parameter indicates that the Display Enhancement attribute of the Event Condition shall be altered by this service. If this parameter is selected, one of the following parameters shall appear.

###### 7.2.5.1.1.1.1 Display Enhancement string

This parameter, of type character string, is the string form of the Display Enhancement parameter. This selection may be made only if the DES CBB has been negotiated.

###### 7.2.5.1.1.1.2 Display Enhancement index

This parameter, of type integer, is the numeric form of the Display Enhancement parameter. This selection may be made only if the DEI CBB has been negotiated.

###### 7.2.5.1.1.1.3 No Enhancement

This parameter, of type NULL, specifies that no Display Enhancement is present. This parameter shall be selected if neither DES nor DEI has been negotiated.

##### 7.2.5.1.1.2 Service procedure extensions

The service procedure of 15.2.2 of ISO/IEC 9506 1 shall be performed. If the Display Enhancement string has been selected, the value of the Display Enhancement Class attribute of the Event Condition shall be TEXT and the Display Enhancement attribute shall be set to the value of the Display Enhancement string parameter. If the Display Enhancement index is selected, the Display Enhancement Class attribute of the Event Condition shall be NUMBER and the Display Enhancement attribute shall be set to the value of the Display Enhancement index parameter. Otherwise the Display Enhancement Class attribute shall be set to UNDEFINED.

##### 7.2.5.1.1.3 Protocol extensions

The extensions to the DefineEventCondition service argument shall be the CS DefineEventCondition Request.

```

CS DefineEventCondition Request ::= [0] Choice {
    enhancementString      [0] IMPLICIT VisibleString,
    enhancementIndex       [1] IMPLICIT INTEGER,
    noEnhancement          NULL
}

```

**NOTE** — As a result of the way in which Confirmed RequestPDU is specified in ISO/IEC 9506 2, a NULL specified as a Companion Standard extension is not transmitted. The effect is as though the parameter were not included in the protocol. Specification of the NULL is required at the service interface, however.

### 7.2.5.2 DeleteEventCondition service extensions

The service procedure of the DeleteEventCondition service shall be extended to include a verification that the value of the List of referencing Event Condition List references attribute of the specified Event Condition object is equal to an empty list. The verification shall be performed prior to performing the procedure specified in ISO/IEC 9506 1. If the value of this attribute is not equal to an empty list, the specified Event Condition object shall be counted as a Candidate Not Deleted, the Event Condition shall not be deleted, and a Response(-) shall be returned.

### 7.2.5.3 GetEventConditionAttributes service

This subclause specifies the extension of the GetEventConditionAttributes service.

#### 7.2.5.3.1 Parameter extensions

The GetEventConditionAttributes service response shall be extended to include the Group Priority Override parameter, the List of referencing Event Condition Lists parameter and the Display Enhancement parameter. The structure of the GetEventConditionAttributes parameter extensions is specified in [Table 2](#).

**Table 2 — GetEventConditionAttributes parameter extensions**

| Parameter name                            | Req | Ind | Rsp | Cnf   | CBB |
|-------------------------------------------|-----|-----|-----|-------|-----|
| Group Priority Override                   |     |     | C   | C (=) |     |
| List of referencing Event Condition Lists |     |     | C   | C (=) |     |
| Display Enhancement                       |     |     | M   | M (=) |     |
| Display Enhancement string                |     |     | S   | S (=) |     |
| Display Enhancement index                 |     |     | S   | S (=) |     |
| No Enhancement                            |     |     | S   | S (=) |     |

#### 7.2.5.3.2 Group priority override

This parameter, of type integer, shall contain the value of the Group Priority Override attribute of the Event Condition object.

### 7.2.5.3.3 List of referencing Event Condition Lists

This parameter shall contain a list of names, derived from the contents of the List of referencing Event Condition List references attribute of the specified Event Condition object. Each name shall be equal to the value of the name attribute of a referencing Event Condition List object.

### 7.2.5.3.4 Display Enhancement

This parameter indicates the value of the Display Enhancement Class attribute of the Event Condition object. Depending on its value, one of the following parameters shall be selected.

#### 7.2.5.3.4.0.1 Display Enhancement string

This parameter, of type character string, is the string form of the Display Enhancement parameter.

#### 7.2.5.3.4.0.2 Display Enhancement index

This parameter, of type integer, is the numeric form of the Display Enhancement parameter.

#### 7.2.5.3.4.0.3 No Enhancement

This parameter, of type NULL, specifies that no Display Enhancement is present.

### 7.2.5.3.5 Service procedure extensions

The service procedure of 15.4.2 of ISO/IEC 9506 1 shall be performed. If the Display Enhancement Class attribute of the Event Condition has the value TEXT, the Display Enhancement string parameter shall be selected. If the Display Enhancement Class attribute has the value NUMBER, the Display Enhancement index shall be selected. If the Display Enhancement Class attribute has the value UNDEFINED, the No Enhancement parameter shall be selected.

The service procedure of the GetEventConditionAttributes service shall be extended to include the value of the Group Priority Override attribute of the Event Condition object as the Group Priority Override parameter and the value of the List of referencing Event Condition List references attribute as the value of the List of referencing Event Condition Lists parameter. These parameters shall only appear if and only if one or more of the services for Event Condition List objects are supported. The value of the Display Enhancement attribute of the Event Condition object shall be included as the Display Enhancement parameter.

### 7.2.5.3.6 Protocol extensions

The extensions to the GetEventConditionAttributes service response protocol shall be the CS GetEventConditionAttributes Response.

```
CS GetEventConditionAttributes Response ::= CHOICE {  
    supplied          SEQUENCE {  
        groupPriorityOverride    [0] CHOICE {  
            priority            [0] IMPLICIT Priority,  
            undefined           [1] IMPLICIT NULL  
        } OPTIONAL,  
        listOfEventConditionList [1] IMPLICIT SEQUENCE OF ObjectName  
        OPTIONAL,  
        displayEnhancement      [2] CHOICE {
```

```

        enhancementString      [0] IMPLICIT VisibleString,
        enhancementIndex       [1] IMPLICIT INTEGER,
        noEnhancement          [2] IMPLICIT NULL
    } },
    default                    NULL
}

```

The default choice shall be chosen if and only if the groupPriorityOverride is not transmitted, the listOfEventConditionList field is not transmitted, and the noEnhancement choice is selected for the display Enhancement field.

#### 7.2.5.3.7 GroupPriorityOverride

The priority choice shall be selected when the value of the Group Priority Override parameter of the GetEventConditionAttributes service response is not equal to UNDEFINED; otherwise, the undefined choice shall be selected.

#### 7.2.5.3.8 Display Enhancement

If the value of the Display Enhancement Class attribute of the Event Condition object is equal to TEXT, the enhancementString choice shall be selected. If the value of the Display Enhancement Class attribute of the Event Condition object is equal to NUMBER, the enhancementIndex choice shall be selected. Otherwise the noEnhancement choice shall be selected.

#### 7.2.5.4 AlterEventConditionMonitoring service

This subclause specifies the extensions to the AlterEventConditionMonitoring service.

##### 7.2.5.4.1 Parameter extensions

The AlterEventConditionMonitoring service shall be extended to include specification of the value of the Display Enhancement attribute of the Event Condition object. The structure of the AlterEventConditionMonitoring parameter extensions is specified in [Table 3](#).

**Table 3 — AlterEventConditionMonitoring parameter extensions**

| Parameter name             | Req | Ind  | Rsp | Cnf | CBB |
|----------------------------|-----|------|-----|-----|-----|
| Display Enhancement        | S   | S(=) |     |     |     |
| Display Enhancement string | S   | S(=) |     |     | DES |
| Display Enhancement index  | S   | S(=) |     |     | DEI |
| No Enhancement             | S   | S(=) |     |     |     |
| Unchanged Display          | S   | S(=) |     |     |     |

##### 7.2.5.4.1.1 Display Enhancement

Selection of this parameter indicates that the Display Enhancement attribute of the Event Condition shall be altered by this service. If this parameter is selected, one of the following parameters shall appear.

#### 7.2.5.4.1.1.1 Display Enhancement string

This parameter, of type character string, is the string form of the Display Enhancement parameter. This selection may be made only if the DES CBB has been negotiated.

#### 7.2.5.4.1.1.2 Display Enhancement index

This parameter, of type integer, is the numeric form of the Display Enhancement parameter. This selection may be made only if the DEI CBB has been negotiated.

#### 7.2.5.4.1.1.3 No Enhancement

This parameter, of type NULL, specifies that no Display Enhancement is present. This parameter shall be selected if neither DES nor DEI has been negotiated.

#### 7.2.5.4.1.2 Unchanged Display

If this parameter is selected, the Display Enhancement attribute of the Event Condition shall not be changed from its present value.

#### 7.2.5.4.2 Service procedure extensions

If the Display Enhancement selection is made, the value of the Display Enhancement attribute of the Event Condition object shall be altered. If the Display Enhancement string is selected, the Display Enhancement Class attribute of the Event Condition shall be set to TEXT and the Display Enhancement attribute shall be set to the value of the Display Enhancement string parameter. If the Display Enhancement index is selected, the Display Enhancement Class attribute of the Event Condition shall be set to NUMBER and the Display Enhancement attribute shall be set to the value of the Display Enhancement index parameter. Otherwise, the Display Enhancement Class shall be set to UNDEFINED.

#### 7.2.5.4.3 Protocol extensions

The extension to the AlterEventConditionMonitoring service argument shall be the CS AlterEventConditionMonitoring Request.

```
CS AlterEventConditionMonitoring Request ::= SEQUENCE {  
    changedDisplay          CHOICE {  
        enhancementString  [0] IMPLICIT VisibleString,  
        enhancementIndex   [1] IMPLICIT INTEGER,  
        noEnhancement       [2] NULL  
    } OPTIONAL }  
}
```

The choice of Display Enhancement for Display Option shall be indicated by the inclusion of the changeDisplay field in the PDU. If Unchanged Display is chosen, the changeDisplay field shall not occur.

#### 7.2.5.5 DefineEventEnrollment service

This subclause specifies the extensions to the DefineEventEnrollment service.

#### 7.2.5.5.1 Parameter extensions

The DefineEventEnrollment service shall be extended to include specification of the Display Enhancement attribute of the Event Enrollment object. The structure of the DefineEventEnrollment parameter extensions is specified in Table 4.

**Table 4 — DefineEventEnrollment parameter extensions**

| Parameter name             | Req | Ind  | Rsp | Cnf | CBB |
|----------------------------|-----|------|-----|-----|-----|
| Display Enhancement        | M   | M(=) |     |     |     |
| Display Enhancement string | S   | S(=) |     |     | DES |
| Display Enhancement index  | S   | S(=) |     |     | DEI |
| No Enhancement             | S   | S(=) |     |     |     |

##### 7.2.5.5.1.1 Display Enhancement

The value of this parameter indicates the type of Display Enhancement desired. Depending on its value, one of the following parameters shall be selected.

###### 7.2.5.5.1.1.1 Display Enhancement string

This parameter, of type character string, is the string form of the Display Enhancement parameter. This selection may be made only if the DES CBB has been negotiated.

###### 7.2.5.5.1.1.2 Display Enhancement index

This parameter, of type integer, is the numeric form of the Display Enhancement parameter. This selection may be made only if the DEI CBB has been negotiated.

###### 7.2.5.5.1.1.3 No Enhancement

This parameter, of type NULL, specifies that no Display Enhancement is present. This parameter shall be selected if neither DES nor DEI has been negotiated.

##### 7.2.5.5.2 Service procedure extensions

The service procedure of 15.12.2 of ISO/IEC 9506 1 shall be performed. If the Display Enhancement string has been selected, the value of the Display Enhancement Class attribute of the Event Enrollment shall be TEXT and the Display Enhancement attribute shall be set to the value of the Display Enhancement string parameter. If the Display Enhancement index is selected, the Display Enhancement Class attribute of the Event Enrollment shall be NUMBER and the Display Enhancement attribute shall be set to the value of the Display Enhancement index parameter. Otherwise the Display Enhancement Class attribute shall be set to UNDEFINED.

##### 7.2.5.5.3 Protocol extensions

The extensions to the DefineEventEnrollment service argument shall be the CS DefineEventEnrollment Request.

```
CS DefineEventEnrollment Request ::= [0] Choice {
    enhancementString      [0] IMPLICIT VisibleString,
    enhancementIndex       [1] IMPLICIT INTEGER,
    noEnhancement          NULL
}
```

**NOTE** — As a result of the way in which Confirmed RequestPDU is specified in ISO/IEC 9506 2, a NULL specified as a Companion Standard extension is not transmitted. The effect is as though the parameter were not included in the protocol. Specification of the NULL is required at the service interface, however.

#### 7.2.5.6 GetEventEnrollmentAttributes service

This subclause specifies the extension of the GetEventEnrollmentAttributes service.

##### 7.2.5.6.1 Parameter extensions

The GetEventEnrollmentAttributes service response shall be extended to include the Display Enhancement parameter. The structure of the GetEventEnrollmentAttributes parameter extensions is specified in [Table 5](#).

**Table 5 — GetEventEnrollmentAttributes parameter extensions**

| Parameter name             | Req | Ind | Rsp | Cnf  | CBB |
|----------------------------|-----|-----|-----|------|-----|
| Display Enhancement        |     |     | M   | M(=) |     |
| Display Enhancement string |     |     | S   | S(=) |     |
| Display Enhancement index  |     |     | S   | S(=) |     |
| No Enhancement             |     |     | S   | S(=) |     |

##### 7.2.5.6.1.1 Display Enhancement

The value of this parameter indicates the type of Display Enhancement desired. Depending on its value, one of the following parameters shall be selected.

###### 7.2.5.6.1.1.1 Display Enhancement string

This parameter, of type character string, is the string form of the Display Enhancement parameter.

###### 7.2.5.6.1.1.2 Display Enhancement index

This parameter, of type integer, is the numeric form of the Display Enhancement parameter.

###### 7.2.5.6.1.1.3 No Enhancement

This parameter, of type NULL, specifies that no Display Enhancement is present.

##### 7.2.5.6.2 Service procedure extensions

The service procedure of 15.14.2 of ISO/IEC 9506 1 shall be performed. If the Display Enhancement Class attribute of the Event Enrollment has the value TEXT, the Display Enhancement string parameter shall be selected. If the Display Enhancement Class attribute has the value NUMBER, the Display Enhancement index shall be selected. If the Display Enhancement Class attribute has the value UNDEFINED, the No Enhancement parameter shall be selected.

##### 7.2.5.6.3 Protocol extensions

The extensions to the GetEventEnrollmentAttributes service response protocol shall be the CS GetEventEnrollmentAttributes Response.

```

CS GetEventEnrollmentAttributes Response ::= [0] Choice {
    enhancementString      [0] IMPLICIT VisibleString,
    enhancementIndex       [1] IMPLICIT INTEGER,
    noEnhancement          NULL
}

```

**NOTE** — As a result of the way in which Confirmed ResponsePDU is specified in ISO/IEC 9506 2, a NULL specified as a Companion Standard extension is not transmitted. The effect is as though the parameter were not included in the protocol. Specification of the NULL is required at the service interface, however.

### 7.2.5.7 AlterEventEnrollment service

This subclause specifies the extension of the AlterEventEnrollment service.

#### 7.2.5.7.1 Parameter extensions

The AlterEventEnrollment service shall be extended to include specification of the Display Enhancement attribute of the Event Enrollment object. The structure of the AlterEventEnrollment parameter extensions is specified in [Table 6](#).

**Table 6 — AlterEventEnrollment parameter extensions**

| Parameter name             | Req | Ind  | Rsp | Cnf | CBB |
|----------------------------|-----|------|-----|-----|-----|
| Display Enhancement        | S   | S(=) |     |     |     |
| Display Enhancement string | S   | S(=) |     |     | DES |
| Display Enhancement index  | S   | S(=) |     |     | DEI |
| No Enhancement             | S   | S(=) |     |     |     |
| Unchanged Display          | S   | S(=) |     |     |     |

#### 7.2.5.7.1.1 Display Enhancement

Selection of this parameter indicates that the Display Enhancement attribute of the Event Condition shall be altered by this service. If this parameter is selected, one of the following parameters shall appear.

##### 7.2.5.7.1.1.1 Display Enhancement string

This parameter, of type character string, is the string form of the Display Enhancement parameter. This selection may be made only if the DES CBB has been negotiated.

##### 7.2.5.7.1.1.2 Display Enhancement index

This parameter, of type integer, is the numeric form of the Display Enhancement parameter. This selection may be made only if the DEI CBB has been negotiated.

##### 7.2.5.7.1.1.3 No Enhancement

This parameter, of type NULL, specifies that no Display Enhancement is present. This parameter shall be selected if neither DES nor DEI has been negotiated.

#### 7.2.5.7.1.2 Unchanged Display

If the parameter is selected, the Display Enhancement attribute of the Event Enrollment shall not be changed from its present value.

#### 7.2.5.7.2 Service procedure extensions

If the Display Enhancement selection is made, the value of the Display Enhancement attribute of the Event Enrollment object shall be altered. If the Display Enhancement string is selected, the Display Enhancement Class attribute of the Event Enrollment shall be set to TEXT and the Display Enhancement attribute shall be set to the value of the Display Enhancement string parameter. If the Display Enhancement index is selected, the Display Enhancement Class attribute of the Event Enrollment shall be set to NUMBER and the Display Enhancement attribute shall be set to the value of the Display Enhancement index parameter. Otherwise, the Display Enhancement Class shall be set to UNDEFINED.

#### 7.2.5.7.3 Protocol extensions

The extension to the AlterEventEnrollment service argument shall be the CS AlterEventEnrollment Request.

```
CS AlterEventEnrollment Request ::= SEQUENCE {  
    changeDisplay          CHOICE {  
        enhancementString [0] IMPLICIT VisibleString,  
        enhancementIndex  [1] IMPLICIT INTEGER,  
        noEnhancement     [2] NULL  
    } OPTIONAL }  

```

The choice of Display Enhancement for Display Option shall be indicated by the inclusion of the changeDisplay field in the PDU. If Unchanged Display is chosen, the changeDisplay field shall not occur.

#### 7.2.5.8 EventNotification service

This subclause specifies the extensions to the EventNotification service.

##### 7.2.5.8.1 Parameter extensions

The EventNotification service shall be extended to include specification of the Display Enhancement attribute of the Event Condition object. The structure of the EventNotification parameter extensions is specified in [Table 7](#).

**Table 7 — EventNotification parameter extension**

| Parameter name             | Req | Ind  | Rsp | Cnf | CBB |
|----------------------------|-----|------|-----|-----|-----|
| Display Enhancement        | M   | M(=) |     |     |     |
| Display Enhancement string | S   | S(=) |     |     | DES |
| Display Enhancement index  | S   | S(=) |     |     | DEI |
| No Enhancement             | S   | S(=) |     |     |     |

#### 7.2.5.8.1.1 Display Enhancement

The value of this parameter indicates the type of Display Enhancement desired. Depending on its value, one of the following parameters shall be selected.

##### 7.2.5.8.1.1.1 Display Enhancement string

This parameter, of type character string, is the string form of the Display Enhancement parameter. This selection may be made only if the DES CBB has been negotiated.

##### 7.2.5.8.1.1.2 Display Enhancement index

This parameter, of type integer, is the numeric form of the Display Enhancement parameter. This selection may be made only if the DEI CBB has been negotiated.

##### 7.2.5.8.1.1.3 No Enhancement

This parameter, of type NULL, specifies that no Display Enhancement is present. This parameter shall be selected if neither DES nor DEI has been negotiated.

#### 7.2.5.8.2 Service procedure extensions

The service procedure of 15.17.2 of ISO/IEC 9506 1 shall be performed, including the value of the Display Enhancement parameter in the response PDU. If the value of the Display Enhancement Class attribute of the Event Enrollment object referenced in the Event Condition request is not equal to UNDEFINED, the value of the Display Enhancement parameter shall be derived from the value of the Display Enhancement attribute of the Event Enrollment object. Otherwise, the value of the Display Enhancement parameter shall be derived from the value of the Display enhancement attribute of the Event Condition object referenced in the Event Notification request.

#### 7.2.5.8.3 Protocol extensions

The extensions to the EventNotification service argument shall be the CS EventNotification.

```
CS EventNotification ::= [0] Choice {  
    enhancementString      [0] IMPLICIT VisibleString,  
    enhancementIndex       [1] IMPLICIT INTEGER,  
    noEnhancement          NULL  
}
```

**NOTE** — As a result of the way in which Unconfirmed PDU is specified in ISO/IEC 9506 2, a NULL specified as a Companion Standard extension is not transmitted. The effect is as though the parameter were not included in the protocol. Specification of the NULL is required at the service interface, however.

#### 7.2.5.9 Event Condition Additional Detail

The Additional Detail parameter that is used in the Get Alarm Summary service and in the Get Alarm Enrollment Summary service is specified to be the Display Enhancement parameter defined in 6.4.1.1. The value of this parameter shall be derived from the attributes of the Event Condition object being referenced in the respective service response.

### 7.2.5.9.1 Additional Detail protocol

This part of ISO/IEC 9506 defines the EN Additional Detail production as follows:

```
EN Additional Detail ::= [0] Choice {  
    enhancementString      [0] IMPLICIT VisibleString,  
    enhancementIndex       [1] IMPLICIT INTEGER,  
    noEnhancement          NULL  
}
```

### 7.2.6 Other productions

The following productions, required for the MMS General Module and representing Companion Standard Extensions to the MMS services, are not used in the Process Control Companion Standard and are set equal to NULL.

```
CS Status Request ::= NULL  
CS Input Request  ::= NULL  
CS Output Request ::= NULL  
CS InitiateDownloadSequence Request ::= NULL  
CS DownloadSegment Request ::= NULL  
CS TerminateDownloadSequence Request ::= NULL  
CS InitiateUploadSequence Request ::= NULL  
CS UploadSegment Request ::= NULL  
CS TerminateUploadSequence Request ::= NULL  
CS RequestDomainDownload Request ::= NULL  
CS RequestDomainUpload Request ::= NULL  
CS LoadDomainContent Request ::= NULL  
CS StoreDomainContent Request ::= NULL  
CS DeleteDomain Request ::= NULL  
CS GetDomainAttributes Request ::= NULL  
CS CreateProgramInvocation Request ::= NULL  
CS DeleteProgramInvocation Request ::= NULL  
CS Start Request ::= NULL  
CS Stop Request ::= NULL  
CS Reset Request ::= NULL  
CS Resume Request ::= NULL  
CS Kill Request ::= NULL  
CS GetProgramInvocationAttributes Request ::= NULL  
CS DeleteEventCondition Request ::= NULL
```

CS GetEventConditionAttributes Request ::= NULL  
CS ReportEventConditionStatus Request ::= NULL  
CS TriggerEvent Request ::= NULL  
CS DefineEventAction Request ::= NULL  
CS DeleteEventAction Request ::= NULL  
CS GetEventActionAttributes Request ::= NULL  
CS ReportEventActionStatus Request ::= NULL  
CS DeleteEventEnrollment Request ::= NULL  
CS ReportEventEnrollmentStatus Request ::= NULL  
CS GetEventEnrollmentAttributes Request ::= NULL  
CS AcknowledgeEventNotification Request ::= NULL  
CS GetAlarmSummary Request ::= NULL  
CS GetAlarmEnrollmentSummary Request ::= NULL  
CS ReadJournal Request ::= NULL  
CS WriteJournal Request ::= NULL  
CS InitializeJournal Request ::= NULL  
CS ReportJournalStatus Request ::= NULL  
CS CreateJournal Request ::= NULL  
CS DeleteJournal Request ::= NULL  
CS GetCapabilityList Request ::= NULL  
CS Status Response ::= NULL  
CS Input Response ::= NULL  
CS Output Response ::= NULL  
CS InitiateDownloadSequence Response ::= NULL  
CS DownloadSegment Response ::= NULL  
CS TerminateDownloadSequence Response ::= NULL  
CS InitiateUploadSequence Response ::= NULL  
CS UploadSegment Response ::= NULL  
CS TerminateUploadSequence Response ::= NULL  
CS RequestDomainDownload Response ::= NULL  
CS RequestDomainUpload Response ::= NULL  
CS LoadDomainContent Response ::= NULL  
CS StoreDomainContent Response ::= NULL  
CS DeleteDomain Response ::= NULL  
CS CreateProgramInvocation Response ::= NULL  
CS DeleteProgramInvocation Response ::= NULL