

# INTERNATIONAL STANDARD

**ISO/IEC  
9506-1**

First edition  
1990-10-15

**AMENDMENT 1**  
1993-07-01

## Industrial automation systems — Manufacturing Message Specification —

**Part 1:**  
Service definition

**AMENDMENT 1: Data exchange**

*Systèmes d'automatisation industrielle — Spécification de messagerie  
industrielle —*

*Partie 1: Définition de service*

*AMENDEMENT 1: Échange de données*

IECNORM.COM: Click to view the full PDF of ISO/IEC 9506-1:1990/AMD1:1993



Reference number  
ISO/IEC 9506-1:1990/Amd.1:1993(E)

## Foreword

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

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

Amendment 1 to International Standard ISO/IEC 9506-1:1990 was prepared by Technical Committee ISO/TC 184, *Industrial automation systems*, Sub-Committee SC 5, *System integration and communication*.

ISO/IEC 9506 consists of the following parts, under the general title *Industrial automation systems — Manufacturing Message Specification*:

- *Part 1: Service definition*
- *Part 2: Protocol specification*

# Industrial automation systems — Manufacturing Message Specification —

## Part 1: Service definition

### AMENDMENT 1: Data exchange

Page 20

**Subclause 7.2.6.2**

Insert

"GetDataExchangeAttributes"  
into the list of services between GetCapabilityList and GetDomainAttributes.

Page 25

**Subclause 7.3.2**

Table 1, add the following to the end of the table:

Data Exchange Objects	X		X	20
-----------------------	---	--	---	----

Page 27

**Subclause 7.4.2**

Change

"such names must be defined over the association"

to

"AA-specific objects shall only exist as a result of a definition of the name over the association by an MMS Service. As there is no MMS service to create AA-specific Data Exchange objects, these objects will be created by local action at the time of association establishment."

Page 29

**Subclause 8.1.2.4**

In the first sentence, change

"8 to 16"

to

"8 to 16 and 20"

Page 49

**Subclause 9.4.1.1.1.1**

Add the following to the end of the list:

"Data Exchange"

Page 52

**Subclause 9.6.1.1.1.1**

Add the following to the end of the list:

"Data Exchange"

Page 280

**Subclause 19.2.1.2**

Insert

"GetDataExchangeAttributes  
ExchangeData"

into the list between DeleteJournal and FileOpen.

Add a new clause 20 as follows:

## 20 Data Exchange Management Services

The Data Exchange management services provide facilities which allow a client MMS-user to invoke a procedure on a VMD. This remote procedure call is modeled as an exchange of data between two MMS users. 20.1 describes the MMS Model for Data Exchange. The ExchangeData service is described in 20.2 and the GetDataExchangeAttributes service is described in 20.3. These services are intended to provide functions not available through use of other MMS services, and shall not be used to circumvent the spirit or intent of those services.

### 20.1 The Data Exchange management model

This clause defines the MMS Model for the Data Exchange object and the related Data Exchange function. A data exchange object is an abstract element of a VMD which is capable of invoking (when requested) a real procedure. This procedure may require data as input and may produce data as output.

NOTE – Implementation of Data Exchange on a real device may take different forms. One form could be realization as a remote procedure call. Another form could be message function blocks on a programmable device. These may be used for synchronization of processing by having a logical thread of execution stalled waiting for receipt of the Data Exchange message.

#### 20.1.1 D-Exchange Function

The D-Exchange function represents the processing of a data exchange service at the VMD. Parameters of the D-Exchange function are the state of the VMD and the values of the input parameters. The relationship between the real procedure and the Data Exchange object which is used to invoke it is modeled by the D-Exchange function. If the processing is successful, the result is a set of values for the output parameters.

#### 20.1.2 The Data Exchange Object

##### 20.1.2.1 The Data Exchange Object Model

Object: Data Exchange  
 Key Attribute: Data Exchange Name  
 Attribute: In Use (TRUE, FALSE)  
 Attribute: MMS Deletable (FALSE)  
 Attribute: List Of Request Type Specifications  
 Attribute: List Of Response Type Specifications  
 Attribute: Linked (TRUE, FALSE)  
 Constraint: Linked = TRUE  
 Attribute: Program Invocation Reference

#### Data Exchange Name

The Data Exchange Name shall be the principal identifier of the Data Exchange Object. It shall be formed according to the rules for the specification of Object Names as described in 7.4 and 7.3.2.

#### In Use

This attribute shall indicate whether (true) or not (false) the particular Data Exchange object is performing the D-Exchange function.

#### MMS Deletable

This attribute shall indicate that the particular Data Exchange object may not be deleted through the use of an MMS service.

#### List of Request Type Specifications

The List of Request Type Specifications attribute shall specify the datatypes of input data of the underlying procedure. If no inputs are required, this list shall be empty.

#### List of Response Type Specifications

The List of Response Type Specifications attribute shall specify the datatypes of output data to the underlying procedure. If no outputs are required, this list shall be empty.

#### Linked

This attribute shall indicate whether (true) or not (false) the Data Exchange object is linked to a Program Invocation.

#### Program Invocation Reference

This attribute, which only exists for a Data Exchange object having Linked attribute equal to TRUE, shall reference the Program Invocation to which the Data Exchange object is linked.

#### 20.1.2.2 Operations on Data Exchange Objects

The services which operate on Data Exchange object are listed below.

##### ExchangeData

A client may use this service to invoke the processing defined by a Data Exchange Object on the VMD. This service is described in 20.2.

##### GetDataExchangeAttributes

A client may use this service to determine the attributes of a Data exchange object. This service is described in 20.3.

## 20.2 Exchange Data Service

The Exchange Data Service allows an MMS client to invoke a predefined procedure referenced by a Data Exchange Object.

### 20.2.1 Structure

The structure of the component service primitives is shown in Table 95.

Table 95 — ExchangeData Service

Parameter Name	Req	Ind	Rsp	Cnf	CBB
Argument (COMP) Data Exchange Name	M	M(=)			
List of Request Data	M	M(=)			
Result(+) (COMP) List of Response Data			S M	S(=) M(=)	
Result(–) Error Type			S M	S(=) M(=)	

#### 20.2.1.1 Argument

This parameter shall contain the parameters of the Exchange Data Service request.

##### 20.2.1.1.1 Data Exchange Name

This parameter, of type Object Name, shall specify the name of the Data Exchange Object which is to be invoked.

##### 20.2.1.1.2 List of Request Data

This parameter, of type List of Data, shall specify the list of data values to be delivered to the Data Exchange object. The data values shall correspond in type and number to that specified by the List of Request Type Specifications attribute of the Data Exchange Object specified by the Data Exchange Name parameter. If no data values are specified by this List of Request Data parameter, an empty list shall be transmitted.

##### 20.2.1.2 Result(+)

The Result(+) parameter shall indicate that the service request succeeded. When success is indicated, the following parameters shall be included.

###### 20.2.1.2.1 List of Response Data

This parameter, of type List of Data, shall specify the list of data values to be returned to the MMS client. The data values shall correspond in type and number to that

specified by the List of Response Type Specifications attribute of the Data Exchange Object specified by the Data Exchange Name parameter. If no data values are specified by this List of Response Data parameter, an empty list shall be transmitted.

#### 20.2.1.3 Result(–)

The Result(–) parameter shall indicate that the service request failed. The Error Type parameter, which is defined in detail in clause 17, provides the reason for failure.

### 20.2.2 Service Procedure

The MMS Server shall verify that the Data Exchange object referenced does exist. The MMS Server shall verify that the list of request data in the service request conforms to the data types specified in the List of Request Type Specifications attribute of the Data Exchange Object. If the Data Exchange object is linked to a Program Invocation, the MMS server shall verify that the associated Program Invocation does exist and is in the RUNNING state. The effect of this service on the Program Invocation, if any, shall be a local matter. If any of these conditions is not satisfied, the MMS Server shall return a Result(–).

The MMS Server shall set the value of the In Use attribute of the Data Exchange Object to true and perform a data exchange (see 20.1.1). After the D-Exchange function completes, the MMS Server shall change the value of the In Use Attribute to false and issue a Result(+) service response, which shall convey the output values of the D-Exchange function in the List of Response Data. The list of data values conveyed in the List of Response Data parameter shall conform in type and number to the List of Response Type Specification attribute of the Data Exchange object.

If the Data Exchange object is linked to a Program Invocation and the Program Invocation transitions out of the running state while the Data Exchange function is being performed, then success or failure of the service is a local matter.

The method of implementation of the D-Exchange function is a local matter. If the D-Exchange function allows multiple concurrent instances of execution of the procedure, then the value of the In Use attribute shall be true if any instance is active.

Success or failure of this service shall not be conditioned on the result of the procedure. Result of the procedure, if any, shall be returned as part of List of Response Data.

## 20.3 GetDataExchangeAttributes Service

The GetDataAttributes service may be used to request that an MMS server return the attributes associated with the specified Data Exchange Object.

### 20.3.1 Structure

The structure of the component service primitives is shown in Table 96.

**Table 96 — GetDataExchangeAttributes Service**

Parameter Name	Req	Ind	Rsp	Cnf	CBB
Argument (COMP) Data Exchange Name	M M	M(=) M(=)			
Result(+) (COMP) In Use List Of Request Type Specifications List of Response Type Specifications Program Invocation			S M M M C	S(=) M(=) M(=) M(=) C(=)	
Result(–) Error Type			S M	S(=) M(=)	

#### 20.3.1.1 Argument

This parameter shall contain the parameter of the GetDataExchangeAttributes service request.

##### 20.3.1.1.1 Data Exchange Name

This parameter, of type Object Name, shall be the name of the Data Exchange object whose attributes are requested.

##### 20.3.1.2 Result(+)

The Result(+) parameter shall indicate that the service request has succeeded. When success is indicated, the following parameters shall be included.

#### 20.3.1.2.1 List of Request Type Specifications

This parameter, of type List of Type Specification, shall indicate the value of the List of Request Type Specifications attribute. If no types are specified by this List of Request Type Specifications, an empty list shall be transmitted.

#### 20.3.1.2.2 List of Response Type Specifications

This parameter, of type List of Type Specification, shall indicate the value of the List of Response Type Specifications attribute. If no types are specified by this List of Response Type Specifications, an empty list shall be transmitted.

#### 20.3.1.2.3 In Use

This parameter, of type boolean, shall indicate the value of the In Use attribute.

#### 20.3.1.2.4 Program Invocation

This optional parameter, of type Identifier, shall indicate the value of the Program Invocation Reference attribute if present. If the value of the Linked attribute is true, this parameter shall be present, otherwise it shall be omitted.

#### 20.3.1.3 Result(–)

The Result(–) parameter shall indicate that the service request failed. The Error Type parameter, which is defined in detail in clause 17, provides the reason for failure.

### 20.3.2 Service Procedure

The MMS Server shall verify that the specified Data Exchange Object exists. If the Data Exchange Object does not exist a Result(–) response shall be returned. Otherwise the attributes of the Data Exchange Object shall be returned.