



Standards
Certification
Education & Training
Publishing
Conferences & Exhibits

Setting the Standard for Automation™

AMERICAN NATIONAL STANDARD

ANSI/ISA-95.00.02-2018

Enterprise-Control System Integration – Part 2: Objects and Attributes for Enterprise-Control System Integration

Approved 24 May 2018

NOTICE OF COPYRIGHT

This is a copyright document and may not be copied or distributed in any form or manner without the permission of ISA. This copy of the document was made for the sole use of the person to whom ISA provided it and is subject to the restrictions stated in ISA's license to that person. It may not be provided to any other person in print, electronic, or any other form. Violations of ISA's copyright will be prosecuted to the fullest extent of the law and may result in substantial civil and criminal penalties.

ANSI/ISA-95.00.02-2018

Enterprise-Control System Integration – Part 2: Objects and Attributes for Enterprise-Control System Integration

ISBN: 978-1-64331-033-6

Copyright © 2018 by the International Society of Automation (ISA). All rights reserved. Not for resale. 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 T.W. Alexander Drive
P.O. Box 12277
Research Triangle Park, North Carolina 27709

PREFACE

This preface, as well as all footnotes and annexes, is included for information purposes and is not part of ANSI/ISA-95.00.02-2018.

The standards referenced within this document may contain provisions, which, through reference in this text, constitute requirements of this document. At the time of publication, the editions indicated were valid. All standards are subject to revision, and parties to agreements based on this document are encouraged to investigate the possibility of applying the most recent editions of the standards indicated within this document. Members of IEC and ISO maintain registers of currently valid International Standards. ANSI maintains registers of currently valid U.S. National Standards.

This document has been prepared as part of the service of ISA, the International Society of Automation, 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 T.W. Alexander Drive; P. O. Box 12277; Research Triangle Park, NC 27709; Telephone (919) 549-8411; 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, recommended practices, and technical reports to the greatest extent possible. *Standard for Use of the International System of Units (SI): The Modern Metric System*, published by the American Society for Testing & Materials as IEEE/ASTM SI 10-97, 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, recommended practices, and technical reports. Participation in the ISA standards-making process by an individual in no way constitutes endorsement by the employer of that individual, of ISA, or of any of the standards, recommended practices, and technical reports that ISA develops.

CAUTION — ISA ADHERES TO THE POLICY OF THE AMERICAN NATIONAL STANDARDS INSTITUTE WITH REGARD TO PATENTS. IF ISA IS INFORMED OF AN EXISTING PATENT THAT IS REQUIRED FOR USE OF THE STANDARD, IT WILL REQUIRE THE OWNER OF THE PATENT TO EITHER GRANT A ROYALTY-FREE LICENSE FOR USE OF THE PATENT BY USERS COMPLYING WITH THE STANDARD OR A LICENSE ON REASONABLE TERMS AND CONDITIONS THAT ARE FREE FROM UNFAIR DISCRIMINATION.

EVEN IF ISA IS UNAWARE OF ANY PATENT COVERING THIS STANDARD, THE USER IS CAUTIONED THAT IMPLEMENTATION OF THE STANDARD MAY REQUIRE USE OF TECHNIQUES, PROCESSES, OR MATERIALS COVERED BY PATENT RIGHTS. ISA TAKES NO POSITION ON THE EXISTENCE OR VALIDITY OF ANY PATENT RIGHTS THAT MAY BE INVOLVED IN IMPLEMENTING THE STANDARD. ISA IS NOT RESPONSIBLE FOR IDENTIFYING ALL PATENTS THAT MAY REQUIRE A LICENSE BEFORE IMPLEMENTATION OF THE STANDARD OR FOR INVESTIGATING THE VALIDITY OR SCOPE OF ANY PATENTS BROUGHT TO ITS ATTENTION. THE USER SHOULD CAREFULLY INVESTIGATE RELEVANT PATENTS BEFORE USING THE STANDARD FOR THE USER'S INTENDED APPLICATION.

HOWEVER, ISA ASKS THAT ANYONE REVIEWING THIS STANDARD WHO IS AWARE OF ANY PATENTS THAT MAY IMPACT IMPLEMENTATION OF THE STANDARD NOTIFY THE ISA STANDARDS AND PRACTICES DEPARTMENT OF THE PATENT AND ITS OWNER.

ADDITIONALLY, THE USE OF THIS STANDARD MAY INVOLVE HAZARDOUS MATERIALS, OPERATIONS, OR EQUIPMENT. THE STANDARD CANNOT ANTICIPATE ALL POSSIBLE APPLICATIONS OR ADDRESS ALL POSSIBLE SAFETY ISSUES ASSOCIATED WITH USE IN HAZARDOUS CONDITIONS. THE USER OF THIS STANDARD MUST EXERCISE SOUND PROFESSIONAL JUDGMENT CONCERNING ITS USE AND APPLICABILITY UNDER THE USER'S PARTICULAR CIRCUMSTANCES. THE USER MUST ALSO CONSIDER THE APPLICABILITY OF ANY GOVERNMENTAL REGULATORY LIMITATIONS AND ESTABLISHED SAFETY AND HEALTH PRACTICES BEFORE IMPLEMENTING THIS STANDARD.

THE USER OF THIS DOCUMENT SHOULD BE AWARE THAT THIS DOCUMENT MAY BE IMPACTED BY ELECTRONIC SECURITY ISSUES. THE COMMITTEE HAS NOT YET ADDRESSED THE POTENTIAL ISSUES IN THIS VERSION.

ISA (www.isa.org) is a nonprofit professional association that sets the standard for those who apply engineering and technology to improve the management, safety, and cybersecurity of modern automation and control systems used across industry and critical infrastructure. Founded in 1945, ISA develops widely used global standards; certifies industry professionals; provides education and training; publishes books and technical articles; hosts conferences and exhibits; and provides networking and career development programs for its 40,000 members and 400,000 customers around the world.

ISA owns Automation.com, a leading online publisher of automation-related content, and is the founding sponsor of The Automation Federation (www.automationfederation.org), an association of nonprofit organizations serving as “The Voice of Automation.” Through a wholly owned subsidiary, ISA bridges the gap between standards and their implementation with the ISA Security Compliance Institute (www.isasecure.org) and the ISA Wireless Compliance Institute (www.isa100wci.org).

This standard was approved by the ISA Standards and Practices Board on 16 April 2018.

Contents

1	Scope	21
2	Normative references	21
3	Terms, definitions, abbreviations, and conventions	22
3.1	Terms and definitions	22
3.2	Abbreviations	23
3.3	Conventions	24
4	Manufacturing operations information models	36
4.1	Information models	36
4.2	Relationship of exchange information models to operations management activity categories	37
4.3	Cross-model relationships for MOM activity context in exchanges	38
4.4	Abstract UML model information	39
4.5	Attributes of an object in an information model	40
5	Common object models	45
5.1	Hierarchy scope	45
5.2	Spatial definition	47
5.3	Operational location information	49
5.4	Personnel information	55
5.5	Role-based equipment information	63
5.6	Physical asset information	71
5.7	Material information	81
5.8	Process segment information	101
5.9	Operations test information	126
5.10	Operations record information	139
5.11	Operations event information	145
5.12	Containers, tools, and software	165
6	Operations management information	166
6.1	Operations definition information	166
6.2	Operations schedule information	196
6.3	Operations performance information	224
6.4	Operations capability information	248
6.5	Process segment capability information	272
6.6	Operations segment capability information	277
7	Interrelationships between object models	282
8	List of objects	287
9	Compliance	290
	Annex A (Informative) Implementation naming convention for object relationships	291
	A.1 Naming strategies	291
	Annex B (Informative) Value syntax	293
	Annex C (Informative) Use and examples	295
	C.1 Use and examples	295

C.2	Application of the standard.....	300
C.3	Database mapping of the models	300
C.4	XML usage.....	301
Annex D	(Informative) Example data sets	305
D.1	General.....	305
D.2	Material model example	305
D.3	Equipment model examples	307
D.4	Personnel model example	311
D.5	Operations capability example	312
D.6	Operations performance example.....	313
D.7	Operations test model use case examples	313
D.8	Example of planning and response state attributes and defined values	320
D.9	Operations event definition record specification example	323
D.10	Resource acquired example	323
D.11	Work commenced/redirected/completed/aborted example	326
Annex E	(Informative) Questions and answers about object use	329
E.1	General.....	329
E.2	Inflow materials.....	329
E.3	Multiple products per process segment	329
E.4	Process segments vs. operations segments	330
E.5	Segment parameter references	331
E.6	Use of hierarchy scope in parameter objects	332
E.7	Use of spatial definition in personnel objects.....	332
E.8	How class name and property IDs are used to identify elements	333
E.9	Possible capability overcounts	334
E.10	Routing and process capability	336
E.11	Product and process capability dependencies	337
E.12	Representation of dependencies	338
E.13	How a material transfer is handled	339
E.14	How to extend the standard when properties cannot be used	340
E.15	Modeling of tools.....	340
E.16	What is equipment and what is a physical asset?	340
E.17	How should dependencies in the operations schedule and operations response be handled?	341
E.18	How are “mixed” operations types used?.....	341
E.19	What is the relationship between this standard and MESA’s B2MML?	342
Annex F	(informative) Logical information flows	345
Annex G	(Informative) Abstract model to implementation model transformations	347
Annex H	(informative) Abstract to implementation model examples	351
H.1	Personnel abstract UML model	351
H.2	MESA – B2MML 6.0 (XSD) implementation model.....	352
H.3	Simplified XSD implementation model	353
H.4	Object Management Group (OMG) – Interface Definition Language (IDL) – Common Data Representation (CDR) implementation model	353

H.5	OPC Unified Architecture (UA) implementation model	354
H.6	Flat buffers – Interactive Data Language (IDL) implementation model	356
H.7	Internet Engineering Task Force (IETF) – JavaScript Object Notation (JSON) – Schema implementation model	356
H.8	Open Source Robotics Foundation (OSRF) as so known as Open Robotics, Robot Operating System (ROS) message description specifications (MDS)	357
H.9	World Wide Web Consortium (W3C) Resource Description Framework (RDF) schema	357
H.10	SQL database model.....	358
H.11	Transport Protocols.....	358

Figures

Figure 1 – Example: UML Diagram for an information model, personnel model.....	28
Figure 2 – Convention for simplified cross-model relationships to resource models	31
Figure 3 – Operations information models for operations management	37
Figure 4 – Defined cross-model MOM relationships between operations & work models	38
Figure 5 – Hierarchy scope model	46
Figure 6 – Example, WKT in 2D (3D is equally supported)	49
Figure 7 – Operational location model	50
Figure 8 – Personnel model.....	56
Figure 9 – Role-based equipment model.....	64
Figure 10 – Physical asset model	72
Figure 11 – Physical asset and equipment relationships	73
Figure 12 – Material model	81
Figure 13 – Example, material with an assembly	100
Figure 14 – Process segment model.....	102
Figure 15 – Example, Segment dependency	126
Figure 16 – Operations test model.....	127
Figure 17 – Operations record model (abstract).....	140
Figure 18 – Operations event model	146
Figure 19 – Example, Relationship of operations event definition with operations events	152
Figure 20 – Operations definition model	168
Figure 21 – Operations schedule model.....	197
Figure 22 – Operations performance model	225
Figure 23 – Operations capability model	249
Figure 24 – Process segment capability object model.....	273
Figure 25 – Operations segment capability object model	278
Figure 26 – Object model interrelationships	283
Figure C.1 – Personnel model	296
Figure C.2 – Instances of a person class	297

Figure C.3 – UML model for class and class properties	298
Figure C.4 – Class property	298
Figure C.5 – Instances of a person properties	299
Figure C.6 – Instances of person and person properties	299
Figure C.7 – XML schema for a person object	302
Figure C.8 – XML schema for person properties	302
Figure C.9 – Example of person and person property	303
Figure C.10 – Example of person class information	303
Figure C.11 – Adaptor to map different property IDs and values	304
Figure D.1 – Example of simplified job order state model	322
Figure D.2 – Typical MOM functions subscribing to the <i>ResourceAcquired</i> event.....	324
Figure D.3 – Typical MOM functions subscribing to the <i>WorkCommenced</i> , <i>WorkRedirected</i> , <i>WorkCompleted</i> and <i>WorkAborted</i> events	327
Figure E.1 – Class and property IDs used to identify elements	334
Figure E.2 – A property defining overlapping subsets of the capability	335
Figure E.3 – Routing for a product	336
Figure E.4 – Routing with co-products and material dependencies	337
Figure E.5 – Product and process capability relationships	338
Figure E.6 – Time-based dependencies	339
Figure E.7 – Mixed operation example	342
Figure F.1 – Enterprise to manufacturing system abstract information flows	345
Figure F.2 – Abstract information flows among multiple systems	346
Figure H.1 – Abstract UML Model Example: Personnel model	351
Figure H.2 – OPC UA Specification Notation	355
Figure H.3 – OPC UA representation of the personnel model	356

Tables

Table 1 – UML notation used	24
Table 2 – Object color convention for object relationship between UML models	26
Table 3 – Example: Relationship table for an information model, personnel model relationships	28
Table 4 – Example: Relationship role table, Personnel class relationship roles	29
Table 5 – Example: Object attribute table	30
Table 7 – Four possible cross-model relationships or paired relationships for a <i>resource</i> <i>reference object</i> instance	33
Table 8 – Resource reference object cross-model relationship roles	33
Table 9 – Resource reference object property cross-model relationship roles	34
Table 10 – Resource group cross-model relationship roles	34

Table 11 – Resource group property cross-model relationship roles	35
Table 12 – Resource cross-model relationship roles	35
Table 13 – Resource property cross-model relationship roles	36
Table 14 – Cross-model MOM relationship description	38
Table 15 – Commonly used CCTS types for exchange	42
Table 16 – Hierarchy scope relationships	46
Table 17 – Hierarchy scope relationship roles	46
Table 18 – Hierarchy scope attributes	47
Table 19 – Attributes of spatial definition	47
Table 20 – Operational location model relationships	50
Table 21 – Operational location class relationship roles	51
Table 22 – Operational location class attributes	52
Table 23 – Operational location class property relationship roles.....	52
Table 24 – Location class property attributes	53
Table 25 – Operational location relationship roles	53
Table 26 – Operational location attributes	54
Table 27 – Operational location property relationship roles	55
Table 28 – Operational location property attributes	55
Table 29 – Personnel model relationships	56
Table 30 – Personnel class relationship roles	57
Table 31 – Personnel class attributes	58
Table 32 – Personnel class property relationship roles	58
Table 33 – Personnel class property attributes	59
Table 34 – Person relationship roles	60
Table 35 – Person attributes	60
Table 36 – Person property relationship roles	62
Table 37 – Person property attributes	62
Table 38 – Role based equipment model relationships	64
Table 39 – Equipment class relationship roles	65
Table 40 – Equipment class attributes	66
Table 41 – Equipment class property relationship roles	67
Table 42 – Equipment class property attributes	67
Table 43 – Equipment relationship roles	68
Table 44 – Equipment attributes	69
Table 45 – Equipment property relationships	70
Table 46 – Equipment property attributes	71
Table 47 – Physical asset model relationships	72
Table 48 – Physical asset and equipment relationships	73
Table 49 – Physical asset class relationship roles	74

Table 50 – Physical asset class attributes	74
Table 51 – Physical asset class property relationship roles	75
Table 52 – Physical asset class property attributes	76
Table 53 – Physical asset relationship roles	77
Table 54 – Physical asset attributes	77
Table 55 – Physical asset property relationship roles	79
Table 56 – Physical asset property attributes	79
Table 57 – Equipment asset mapping relationship roles	80
Table 58 – Equipment asset mapping attributes	80
Table 59 – Material model relationships	81
Table 60 – Material class relationship roles	83
Table 61 – Material class attributes	84
Table 62 – Material class property relationship roles	85
Table 63 – Material class property attributes	86
Table 64 – Material definition relationship roles	86
Table 65 – Material definition attributes	87
Table 66 – Material definition property relationship roles	89
Table 67 – Material definition property attributes	89
Table 68 – Material lot relationship roles	90
Table 69 – Material lot attributes	91
Table 70 – Material lot property relationship roles	95
Table 71 – Material lot property attributes	95
Table 72 – Material subplot relationship roles	96
Table 73 – Material subplot attributes	97
Table 74 – Process segment model relationships	103
Table 75 – Process segment relationship roles	106
Table 76 – Process segment attributes	107
Table 77 – Process segment parameter relationship roles	108
Table 78 – Process segment parameter attributes	108
Table 79 – Personnel segment specification relationship roles	108
Table 80 – Personnel segment specification attributes	109
Table 81 – Personnel segment specification property relationship roles	111
Table 82 – Personnel segment specification property attributes	111
Table 83 – Equipment segment specification relationship roles	112
Table 84 – Equipment segment specification attributes	112
Table 85 – Equipment segment specification property relationship roles	114
Table 86 – Equipment segment specification property attributes	114
Table 87 – Physical asset segment specification relationship roles	115
Table 88 – Physical asset segment specification attributes	115

Table 89 – Physical asset segment specification property relationship roles	117
Table 90 – Physical asset segment specification property attributes	117
Table 91 – Material segment specification relationship roles	118
Table 92 – Material segment specification attributes.....	119
Table 93 – Material segment specification property relationships.....	123
Table 94 – Material segment specification property attributes.....	123
Table 95 – Segment dependency relationship roles	124
Table 96 – Segment dependency attributes	124
Table 97 – Operations test model relationships	127
Table 98 – Instances of operations test requirement.....	129
Table 99 – Operations test requirement relationship roles in the operations test model	130
Table 100 – Instances of testable object / testable object property pair	130
Table 101 – Testable object relationship roles in operations test model.....	131
Table 102 – Testable object property relationship roles in the operations test model	131
Table 103 – Test specification relationship roles	132
Table 104 – Test specification attributes.....	132
Table 105 – Test specification property relationship roles.....	134
Table 106 – Test specification property attributes.....	134
Table 107 – Test specification criteria relationship roles	134
Table 108 – Test specification criteria attributes	135
Table 109 – Evaluated property relationship roles	136
Table 110 – Evaluated property attributes	136
Table 111 – Test result relationship roles	136
Table 113 – Property measurement relationship roles	138
Table 114 – Property measurement attributes	138
Table 117 – Operations record model relationships	140
Table 118 – Operations record specification template relationship roles	141
Table 119 – Operations record specification template attributes	141
Table 120 – Operations record template relationship roles	143
Table 121 – Operations record template attributes	143
Table 122 – Operations record entry template relationships	144
Table 123 – Operations record entry template attributes.....	144
Table 124 – Operations record model relationships	147
Table 125 – Operations event class relationship roles	148
Table 126 – Operations event class attributes	148
Table 127 – Example of operations event class locked hierarchy.....	149
Table 128 – Operations event class property relationship roles	150
Table 129 – Operations event class property attributes	150
Table 130 – Operations event class record specification relationship roles	151

Table 131 – Operations event definition relationship roles	153
Table 132 – Operations event definition attributes	153
Table 133 – Operations event definition property relationship roles	155
Table 134 – Operations event definition property attributes	156
Table 135 – Operations event definition record specification relationship roles	157
Table 136 – Operations event relationship roles	158
Table 137 – Operations event attributes	159
Table 138 – Operations event property relationship roles	161
Table 139 – Operations event property attributes	162
Table 140 – Operations event record relationship roles	162
Table 141 – Operations event record attributes	162
Table 142 – Operations event record entry relationships	164
Table 143 – Operations event record entry attributes	164
Table 144 – Operations definition model relationships	169
Table 145 – Operations definition relationship roles	171
Table 146 – Operations definition attributes	172
Table 147 – Operations material bill relationship roles	173
Table 148 – Operations material bill attributes	173
Table 149 – Operations material bill item relationship roles	174
Table 150 – Operations material bill item attributes	175
Table 151 – Operations segment relationship roles	176
Table 152 – Operations segment attributes	178
Table 153 – Parameter specification relationship roles	179
Table 154 – Parameter specification attributes	179
Table 155 – Personnel specification relationship roles	180
Table 156 – Personnel specification attributes	181
Table 157 – Personnel specification property relationship roles	182
Table 158 – Personnel specification property attributes	183
Table 159 – Equipment specification relationship roles	184
Table 160 – Equipment specification attributes	184
Table 161 – Equipment specification property relationship roles	186
Table 162 – Equipment specification property attributes	186
Table 163 – Physical asset specification relationship roles	187
Table 164 – Physical asset specification attributes	188
Table 165 – Physical asset specification property relationship roles	189
Table 166 – Physical asset specification property attributes	189
Table 167 – Material specification relationship roles	190
Table 168 – Material specification attributes	191
Table 169 – Material specification property relationship roles	194

Table 170 – Material specification property attributes	194
Table 171 – Segment dependency relationship roles	195
Table 172 – Segment dependency attributes	195
Table 173 – Operations schedule model relationships	198
Table 174 – Operations schedule relationship roles	200
Table 175 – Operations schedule attributes	200
Table 176 – Operations request relationship roles	202
Table 177 – Operations request attributes	203
Table 178 – Segment requirement relationship roles	204
Table 179 – Segment requirement attributes	205
Table 180 – Segment parameter relationship roles	206
Table 181 – Segment parameter attributes	206
Table 182 – Personnel requirement relationship roles	207
Table 183 – Personnel requirement attributes	208
Table 184 – Personnel requirement property relationship roles	210
Table 185 – Personnel requirement property attributes	210
Table 186 – Equipment requirement relationship roles	211
Table 187 – Equipment requirement attributes	212
Table 188 – Equipment requirement property relationships	213
Table 189 – Equipment requirement property attributes	214
Table 190 – Physical asset requirement relationship roles	215
Table 191 – Physical asset requirement attributes	215
Table 192 – Physical asset requirement property relationship roles	217
Table 193 – Physical asset requirement property attributes	218
Table 194 – Material requirement relationship roles	218
Table 195 – Material requirement attributes	219
Table 196 – Material requirement property relationship roles	222
Table 197 – Material requirement property attributes	223
Table 198 – Requested segment response relationship roles	223
Table 199 – Operations performance model relationships	226
Table 200 – Operations performance relationship roles	227
Table 201 – Operations performance attributes	228
Table 202 – Operations response relationship roles	229
Table 203 – Operations response attributes	230
Table 204 – Segment response relationship roles	231
Table 205 – Segment response attributes	232
Table 206 – Segment data relationship roles	233
Table 207 – Segment data attributes	233
Table 208 – Personnel actual relationship roles	234

Table 209 – Personnel actual attributes	234
Table 210 – Personnel actual property relationship roles	236
Table 211 – Personnel actual property attributes	236
Table 212 – Equipment actual relationship roles	237
Table 213 – Equipment actual attributes	238
Table 214 – Equipment actual property relationship roles	239
Table 215 – Equipment actual property attributes	239
Table 216 – Physical asset actual relationship roles	240
Table 217 – Physical asset actual attributes	241
Table 218 – Physical asset actual property relationship roles	242
Table 219 – Physical asset actual property attributes	243
Table 220 – Material actual relationship roles	243
Table 221 – Material actual attributes	244
Table 222 – Material actual property relationship roles	247
Table 223 – Material actual property attributes	247
Table 224 – Operations capability model relationships	249
Table 225 – Operations capability relationship roles	251
Table 226 – Operations capability attributes	251
Table 227 – Personnel capability relationship roles	253
Table 228 – Personnel capability attributes	254
Table 229 – Personnel capability property relationship roles	257
Table 230 – Personnel capability property attributes	257
Table 231 – Equipment capability relationship roles	258
Table 232 – Equipment capability attributes	258
Table 233 – Equipment capability property relationships	261
Table 234 – Equipment capability property attributes	262
Table 235 – Physical asset capability relationship roles	262
Table 236 – Physical asset capability attributes	263
Table 237 – Physical asset capability property relationship roles	266
Table 238 – Physical asset capability property attributes	266
Table 239 – Material capability relationship roles	267
Table 240 – Material capability attributes	268
Table 241 – Material capability property relationship roles	271
Table 242 – Material capability property attributes	272
Table 243 – Process segment capability model relationships	273
Table 244 – Process segment capability relationship roles	275
Table 245 – Process segment capability attributes	275
Table 246 – Operations segment capability model relationships	278
Table 247 – Operations segment capability relationship roles	280

Table 248 – operations segment capability attributes	280
Table 250 – Model cross-reference.....	285
Table 251 – Common resource objects.....	287
Table C.1 – Personnel model relationships.....	296
Table C.2 – Person attributes	297
Table C.3 – Database structure for person	300
Table C.4 – Database structure for person property.....	301
Table C.5 – Database for person with data	301
Table C.6 – Database for person property with data	301
Table D.1 – Pulp and paper equipment model example	307
Table D.2 – Semiconductor manufacturing equipment model example	310
Table D.3 – Planning and job order execution states and defined values of this standard ...	321
Table D.4 – Operations event definition for the operations event, <i>ResourceAcquired</i>	324
Table D.5 – Entry set 1 for an operations event definition record specification.....	324
Table D.6 – Entry set 2 for minimal operations event definition record specification	325
Table D.7 – Entry set 3 for an operations event definition record specification	326
Table D.8 – Entry set 4 for operations event definition record specification 4	326
Table D.9 – Operations event definition record specification for <i>WorkCommenced</i> and <i>WorkRedirected</i>	327
Table D.10 – Operations event definition record specification for <i>WorkCompleted</i> and <i>WorkAborted</i>	328
Table E.1 – Definition of segment types.....	331
Table E.2 – Examples of materials and equipment.....	340
Table E.3 – Equipment and physical assets	341
Table H.1 – Example: Personnel class relationships	352
Table H.2 – Example, Personnel class attributes	352

This page intentionally left blank.

FOREWORD

This standard is Part 2 of a series of standards that defines the interfaces between enterprise activities and control activities. It follows ANSI/ISA-95.00.01-2010 (ISA-95.00.01 Mod), Enterprise-Control System Integration – Part 1: Models and Terminology.

The scope of this Part 2 standard is limited to defining the details of the interface content between manufacturing control functions and other enterprise functions. The scope of this Part 2 standard is limited to the definition of object models and attributes for the information defined in Part 1. The goal is to reduce the effort, cost, and errors associated with implementing these interfaces.

This standard may be used to reduce the effort associated with implementing new product offerings. The goal is to have enterprise systems and control systems that interoperate and easily integrate.

This standard has been prepared in accordance with the ISO/IEC Directives, Part 2. Therefore, the first three clauses present the scope of the standard, normative references, and definitions, in that order.

Clause 4 is informative. It describes the general information about the object models and examples defined in later clauses.

Clause 5 is normative. It defines the object models, relationships, and attributes of common information defined in Part 1.

Clause 6 is normative. It defines the object models, relationships, and attributes of operations management information defined in Part 1.

Clause 7 is informative. It defines the interrelationships between the object models.

Clause 8 is informative. It lists the objects defined in the standard as an aid to documenting conformance and compliance.

Clause 9 is normative. It defines completeness, conformance, and compliance criteria associated with the objects and attributes.

Annex A is informative. It defines implementation naming convention for object relationships.

Annex B is informative. It defines a value syntax best practice. The format for values in a value attribute of an object is not defined in the standard and are defined by implementations of the standard.

Annex C is informative. It provides examples to illustrate how the models, relationships, and attributes may be used.

Annex D is informative. It illustrates some example data sets for exchange.

Annex E is informative. It contains questions and answers on the use and reason for elements in the standard.

Annex F is informative. It discusses how the standard relates to logical information flows.

Annex G is informative. It discusses how the standard relates to abstract to implementation models.

Annex H is informative. It discusses how the standard relates to abstract to implementation examples.

INTRODUCTION

This ISA-95 Part 2 standard further defines formal object models for exchange information described in Part 1 using UML object models, tables of attributes, and examples. The models and terminology defined in this standard:

- a) emphasize good integration practices of control systems with enterprise systems during the entire life cycle of the systems;
- b) can be used to improve existing integration capability of manufacturing control systems with enterprise systems; and
- c) can be applied regardless of the degree of automation.

Specifically, this standard provides a standard terminology and a consistent set of concepts and models for integrating control systems with enterprise systems that will improve communications between all parties involved. Benefits produced will

- a) reduce the user's time to reach full production levels for new products;
- b) enable vendors to supply appropriate tools for implementing integration of control systems to enterprise systems;
- c) enable users to better identify their needs;
- d) reduce the cost of automating manufacturing processes;
- e) optimize supply chains; and
- f) reduce life-cycle engineering efforts.

This standard may be used to reduce the effort associated with implementing new product offerings. The goal is to have enterprise systems and control systems that interoperate and easily integrate.

It is not the intent of the standard to:

- a) suggest that there is only one way of implementing integration of control systems to enterprise systems;
- b) force users to abandon their current way of handling integration; or
- c) restrict development in the area of integration of control systems to enterprise systems.

This page intentionally left blank.

1 Scope

This standard specifies conceptual interface content exchanged between manufacturing control functions and other enterprise functions. The interface considered is between Level 3 manufacturing systems and Level 4 business systems in the hierarchical model defined in Part 1 of this standard. The goal is to reduce the risk, cost, and errors associated with implementing the interface.

Since this standard covers many domains and there are many different standards for those domains, the semantics of this standard are described at a level intended to enable the other standards to be mapped to these semantics. To this end, this standard defines a set of elements contained in the conceptual interface, together with a mechanism for extending the interface content for implementations.

The scope of this standard is limited to the definition of object models and attributes of the exchanged information defined in the Part 1 standard.

2 Normative references

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

ANSI/ISA-95.00.01-2010 (IEC 62264-1 Mod), *Enterprise-Control System Integration – Part 1: Models and terminology*

ANSI/ISA-95.00.03-2013, *Enterprise-Control System Integration – Part 3: Activity models of manufacturing operations management*

ANSI/ISA-95.00.04-2012, *Enterprise-Control System Integration – Part 4: Objects and attributes for manufacturing operations management integration*

ANSI/ISA-95.00.05-2013, *Enterprise-Control System Integration – Part 5: Business-to-manufacturing transactions*

ANSI/ISA-88.00.01-2010, *Batch Control – Part 1: Models and terminology*

IEC 62264-1, *Enterprise-control system integration – Part 1: Models and terminology*

IEC 61512-1, *Batch control – Part 1: Models and terminology*

ANSI/ISA-18.2-2009, *Management of alarm systems for the process industries*

ISO/IEC 19501, *Information technology – Open Distributed Processing – Unified Modeling Language (UML) Version 1.4.2*

ISO/IEC 19505-1, *Information technology – OMG Unified Modeling Language (OMG UML) Version 2.5 – Part 1: Infrastructure*

ISO/IEC 19505-2, *Information technology – OMG Unified modeling language (OMG UML) Version 2.5 – Part 2: Superstructure*

ISO 15000-5, *Core Components Technical Specification*

3 Terms, definitions, abbreviations, and conventions

3.1 Terms and definitions

For the purposes of this document, the terms and definitions given in Part 1 of this standard, as well as the following apply.

3.1.1

by-product

a secondary material, produced as a residual of, or incidental to the production of the primary product

3.1.2

co-product

a material usually manufactured together or sequentially with the primary product because of product or process similarities

3.1.3

equipment class

grouping of role-based equipment with similar characteristics

3.1.4

event

representation of a solicited or unsolicited fact indicating a change of relevance to Level 3 activities or Level 4 activities

3.1.5

material class

grouping of materials with similar characteristics

3.1.6

material definition

definition of a material type and type properties

NOTE 1 The *material definition* applies to *material lot* that can be identified as raw, intermediate, byproduct, finished, or consumable *material*.

3.1.7

material lot

uniquely identifiable amount of a material

NOTE 2 The *material lot* describes the actual or planned total quantity or amount of *material* available, its current state, and its specific property values.

3.1.8

material subplot

uniquely identifiable subset of a material lot

NOTE 3 The *material subplot* can be a single item.

3.1.9

personnel class

grouping of persons with similar characteristics

3.1.10

product

desired output of an operations process

NOTE 4 A product can be an intermediate product or end product from a business perspective.

NOTE 5 Product also defined in ISO 10303-1 as: a substance produced by a natural or artificial process.

3.1.11

property

implementation specific characteristic of an entity

3.2 Abbreviations

For purposes of this standard, the following abbreviations apply:

B2MML	Business to Manufacturing Markup Language
BOR	Bill of resources
CE	Common era
EBNF	Extended Backus-Naur form
EPSCG	European Petroleum Survey Group
FQN	Fully qualified name
GUID	Globally unique identifier
IEC	International Electrotechnical Commission
ISO	International Organization for Standardization
MIMOSA	Machinery information management open systems alliance (original) or operations and maintenance information open system alliance (current)
MOM	Manufacturing operations management
SOC	Standard operating conditions

SOP	Standard operating procedure
SRID	Spatial reference identifier
UML	Unified Modeling Language
XML	Extensible Markup Language
XSD	XML schema definition

3.3 Conventions

3.3.1 General

Italics are used for the use cases of names of objects used in exchanged data, beyond the use defined in ISO/IEC Directives Part 2, to emphasize the specific meaning of terminology of this standard.

The information object models are depicted using a simplified application of the Unified Modeling Language (UML) notational methodology, as defined in ISO/IEC 19501.

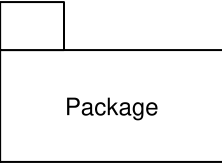
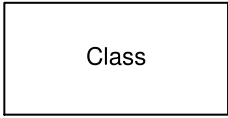
3.3.2 UML notation in information object models

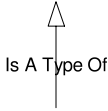
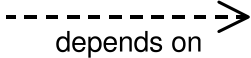
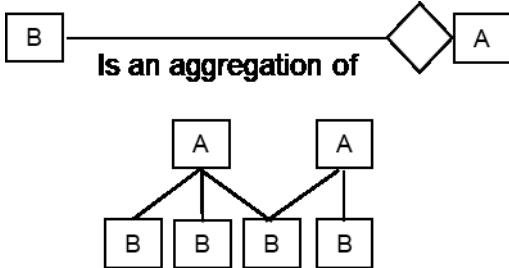
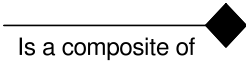
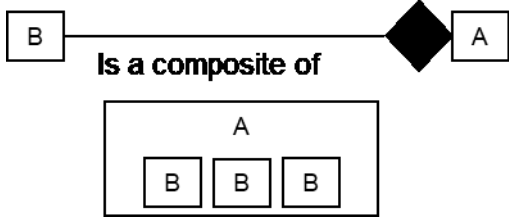
Table 1 defines the UML notations used in the simplified information object diagrams.

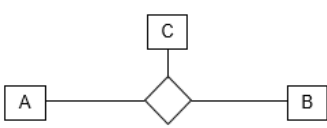
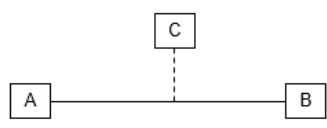
In this standard, the word “class” used as part of an object definition name is to be considered as a category, not as a “class” in the UML specification.

EXAMPLE 1 “Personnel class” is to be considered a “personnel category” in the sense of distinguishing between the kinds of personnel in the real world.

Table 1 – UML notation used

Symbol	Definition
 Package	Defines a package, a collection of object models, state models, use cases, and other UML models. Packages are general-purpose grouping mechanisms used to organize semantically related model elements. In this document, a package is used to specify an external model, such as operations definition model, or a reference to another part of the model.
 Class	Represents a UML class of objects, each with the same types of attributes. Each object is uniquely identifiable or enumerable. No operations or methods are listed for the classes.
 Role 1..1 0..n Association Name Role	An association between elements of a class and elements of another or the same class. Each association is identified. Can have the expected number or range of members of the subclass, when ‘*’ indicates an indeterminate number. EXAMPLE 2 0..* means that zero or more members of the subclass can exist.

Symbol	Definition
	The semantics of the association are defined in the object relationship table of each object participating in the relationship.
	Generalization (arrow points to the super class) shows that an element of the class is a specialized type of the super class.
	Dependence is a weak association that shows that a modeling element depends on another modeling element. The item at the tail depends on the item at the head of the relationship. The semantics of the association are defined in the object relationship table of each object participating in the relationship.
	Aggregation shows an element of the class (whole element) is made up of elements of other classes (part elements). EXAMPLE 3  The aggregation notation identifies that the lifetime of the part element is independent of the whole element. The part element is able to be a part of multiple parent elements. When a whole element is deleted, the part elements are not deleted. The parts' physical relationship is independent of its lifetime relationship. A part can be represented directly in the whole or referenced from the whole.
	A composition relationship represents a whole–part relationship and is a type of aggregation. A composition relationship specifies that the lifetime of the part classifier is dependent on the lifetime of the whole classifier. Composition shows a strong form of aggregation, which requires 1) a part instance is included in only one composition at a time and 2) the composition object has sole responsibility for disposition of its parts. EXAMPLE 4  The composition notation identifies that the lifetime of the part elements is dependent on the whole element. When a whole element is deleted, the part elements are deleted. The parts' physical relationship is independent of its lifetime relationship as a part represented directly in the whole or referenced from the whole.

Symbol	Definition
 Is an Nary association of	Three or more elements are strongly associated. If three elements are associated, the term “ternary” association may be used. The association object (diamond shape) may be an object, which can support attributes relevant to the association. There is no lifetime dependency between the associated objects.
 association class of	The association of A and B are refined by the class C, which is referred to as the association class. If A and B are the same class, A and B are part elements with no distinctions made between the A and B. A and B are referred to as source and target to allow distinction of parts.

NOTE UML life cycles are not applicable to individual information exchanges. The relationships in the UML diagram provides a view into object life cycles at the time of an information exchange. The navigation shown in the UML diagram is the most commonly applied; however, some implementation methods and technologies can require a bidirectional relationship or need to specify the relationship roles for both ends of the relationship.

The defined relationship in the information object diagrams in combination with the model relationship table and the object relationship tables for each object define the “minimum” compliance requirements. Additional relationship and roles required for an implementation are explicitly defined as noncompliant the implementation documentation.

3.3.3 Object color convention for object relationship between UML models

Table 2 defines the object color convention used in the UML diagrams of this part and Part 4 of this standard to illustrate the object relationships between information models. The objects defined in the clause containing the UML diagram have a white background; the relationships to objects not defined in the clause shown in the UML diagram have a gray background if defined in another clause of this part or a beige background if defined in Part 4 of this standard. For the objects not defined in the clause, the basic relationships between objects defined in the clause are shown for the complete context of the objects defined in the clause.

Table 2 – Object color convention for object relationship between UML models

Object type	Color convention
Object defined in clause	An object with a white background belongs to the information model defined in the clause containing the UML diagram.
Part 2 Object not defined In clause	An object with a gray background is not defined in the clause containing the UML diagram and is defined in another information model in this standard.
Part 4 Object not defined In clause	An object with a beige background is not defined in the clause containing the UML diagram and is defined in an information model in Part 4 of this standard.
External Object not defined In this standard	An object with a dark gray background belongs to an external information model not defined in this standard.
<<interface>> Object defined in clause	An object with a pink background belongs to an interface abstract object defined in the clause containing the UML diagram.

3.3.4 Relationship types

This subclause explains the use and meaning of the relationship types in relationship table and role table for each object in an information model.

The generic specification can be further qualified within the relationship table entry to clarify the relationship context.

- a) Association: the association type represents a general relationship between two objects. A dashed line indicates a weak association between objects.

EXAMPLE 1 A *material definition* defines 0..* *material lot* objects where, in contrast, a *material lot* is defined by 1..1 *material definition*. A *material lot* contains a relationship to its *material definition*, which is represented as a relationship attribute in the *material lot*. In a data exchange implementation, the *material lot* objects associated with a *material definition* is recorded in the *material definition* occurrence; but in another implementation, the *material lot* objects are recorded in separate systems using link table references.

- b) Dependency: the object has a semantic link to the object, which can be required to be satisfied before the object can be created. A relationship signifying that a single model object or a set of model objects require other model objects for their specification or implementation.
- c) Aggregation: the view of the UML relationship is qualified by the target end of the relationship link; the terminology used is “aggregation part” for the sub object and “aggregation whole” for the parent object. An aggregation type can be refined with the following keywords:
 - 1) Whole: the object is the whole object in the aggregation.
 - 2) Part: the object is the part object in the aggregation.
 - 3) Hierarchy: the object can be both a whole and part object.

EXAMPLE 2 Equipment hierarchy

This infers that implementations can utilize parent and child prefixes or link tables if navigation of the hierarchy is required. In an aggregation hierarchy when a parent object is deleted, the relationships to other child objects are removed if specified in link tables but the child objects are not deleted.

- d) Composition: the relationship view is qualified by the target end of the relationship; the terminology used is “composition part” for the sub object and ‘composition whole’ for the parent object. A composition type can be refined with the following keywords:
 - 1) Whole: the object is the whole object in the composition.
 - 2) Part: the object is the part object in the composition.
- e) Composition hierarchy: the objects can be both a whole and part object.

EXAMPLE 3 Material lot hierarchy

A composition hierarchy can use parent and child prefixes or link tables if navigation of the hierarchy is required. If the root parent object is deleted, all child objects and their child objects are deleted. Objects can be both a whole and part object.

3.3.5 Relationship table

Each information model's UML diagram is explicitly defined by the simplified UML model in combination with a relationship table. The relationship table lists each relationship and a role table for each object in UML model that lists the role of the object in its relationships.

The object relationship tables indicate an object name for the both end points, a type of relationship and name of relationship for each relationship between objects in an information model. The object relationship table uses the following UML object relationship types to identify the relationship between objects. In the object relationship table, the object relationship is defined specifically from each object's perspective.

EXAMPLE 1 The UML diagram of personnel model in Figure 1 below in combination with the associated relationship table in Table 3 shows the *personnel class* object's relationships as an example of how the standard represents each object's relationships in the context of the information model.

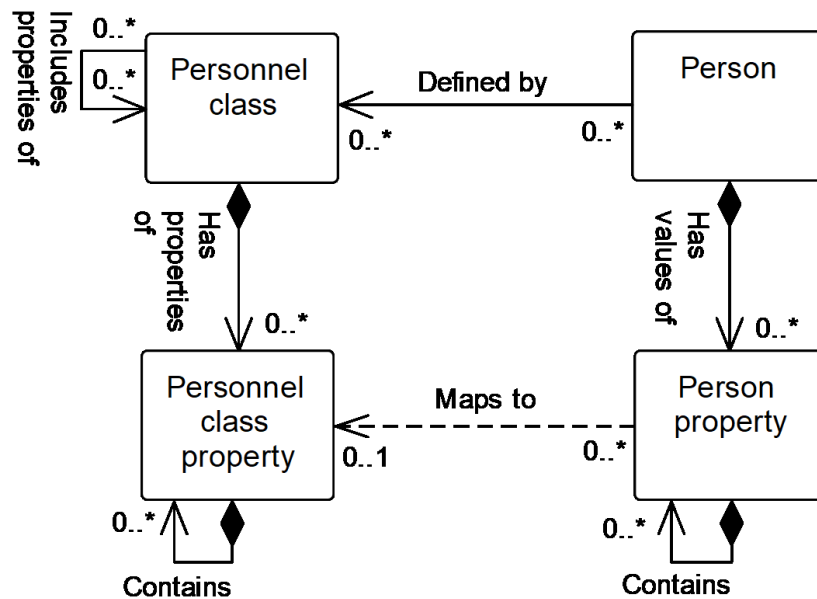


Figure 1 – Example: UML Diagram for an information model, personnel model

Table 3 – Example: Relationship table for an information model, personnel model relationships

From	To	Type	Relationship Name
Personnel class	Personnel class property	Composition	Has properties of
Personnel class	Personnel class	Association	Includes properties of
Person property	Personnel class property	Dependency	Maps to
Person property	Person property	Composition hierarchy	Contains
...	...	...	...

3.3.6 Relationship role table

To explicitly define object's relationships in an information model's UML diagram, each object's definition uses a relationship role table to lists role of the related object in each predominate relationship.

The column headers for the object relationship table are

- a) related object: for a given object, each row in the relationship role table defines a relationship of the object to another object in the information model;
- b) role: the relationship role of the related object in relationship to the given object (what the given object can use to identify the related object);
- c) multiplicity: the multiplicity in UML diagram that specifies cardinality of some collection of elements;

EXAMPLE 1 The number of elements.

Multiplicity element defines some collection of elements and includes both multiplicity as well as specification of order and uniqueness of the collection elements:

- d) description: the description of the object's relationship.

EXAMPLE 2 The example for role table for the personnel class object in the example of the UML diagram of personnel model is shown in Table 4.

Table 4 – Example: Relationship role table, Personnel class relationship roles

Related object	Role	Multiplicity	Relationship name	Description
Personnel class	Personnel class parent type	0..*	Includes properties of	This <i>personnel class(s)</i> include properties of the <i>personnel class</i> .
Personnel class property	Personnel class property	0..*	Has properties of	The <i>personnel class properties</i> of this <i>personnel class</i> .
Person	NA	0..*	Defined by	The <i>person</i> supports this <i>personnel class</i> . The <i>person</i> objects support the <i>personnel class property(s)</i> associated with this <i>personnel class</i> .

3.3.7 Object attribute tables

3.3.7.1 Object attribute tables

This subclause gives the meaning of the object attribute tables. This includes a listing of the object identification, presentation of the examples in the tables, references to resources, object relationships, related object naming convention, and object relationship implementations.

All attributes in the tables are be considered optional, except where specified as required in the attribute description.

3.3.7.2 Presentation of examples

Examples are included with each attribute given in object attribute tables. Examples are presented for each of the four main operation management categories (i.e. production, maintenance, quality and inventory operations) defined in Part 1 of this standard. Table 5 shows how the attribute example rows and columns are used.

Table 5 – Example: Object attribute table

Attribute name	Description	Production example	Maintenance example	Quality example	Inventory examples
Name of first attribute	Description of first attribute	Production example	Maintenance example	Quality example	Inventory example
Name of second attribute	Description of second attribute	Production example	Maintenance example	Quality example	Inventory example
Name of third attribute	Description of third attribute	Production example	Maintenance example	Quality example	Inventory example

When an example value is a set of values, or a member of a set of values, the set of values is given within a set of braces, {}.

The examples are purely fictional. They are provided to further describe attributes in the model. It is not intended to make the examples complete or representative of any manufacturing enterprise.

NOTE 1 Within a table the columns for Production, Maintenance, Quality and Inventory can be examples where the four operations management categories are coordinated, or they can be separate examples.

EXAMPLE 1 When one system is coordinating multiple operations management categories, the IDs used in each column can be the same. When different systems coordinate multiple operations management categories, the IDs can be different.

Example values of attribute are meant to be illustrative and do not imply requirements.

NOTE 2 Time and date attributes can illustrate a general or specific time horizon.

EXAMPLE 2 A yearly or quarterly plan can use general dates with no specific time, while a detailed schedule can include a specific timestamp down to the minute.

EXAMPLE 3 Data resolution used in the examples are typical examples.

Specific implementations of this part may specify the data resolution required for each attribute.

When <NA> (not applicable) is used as an example, this is only illustrative that there is not a value for this attribute in this example. It does not imply there can never be a value. This is also true when all four columns contain <NA>.

3.3.8 Relationships between information models and resource models

The information models in this standard use a simplified method to reference resource model. In order to make the UML diagrams less complex, the simplified UML model does not fully document the possible four cross-model relationships or paired relationships between a specific object (specification, requirement, actual, capability) in an information model and an associated object(s) in the resource information model. The UML models are not fully conformant to the Unified Modeling Language (UML) notational methodology as defined in ISO/IEC 19501.

The simplified UML techniques in this standard are intended as a visualization method that must be used in combination the object attribute tables, object model relationship table, and object role tables for an explicit definition of the abstract information exchanges supported by compliant integration implementations.

Figure 2 illustrates resource model referencing method in the simplified UML model in this standard in the left column to compare the detailed relationships per the ISO/IEC 19501 compliant UML model in the right column.

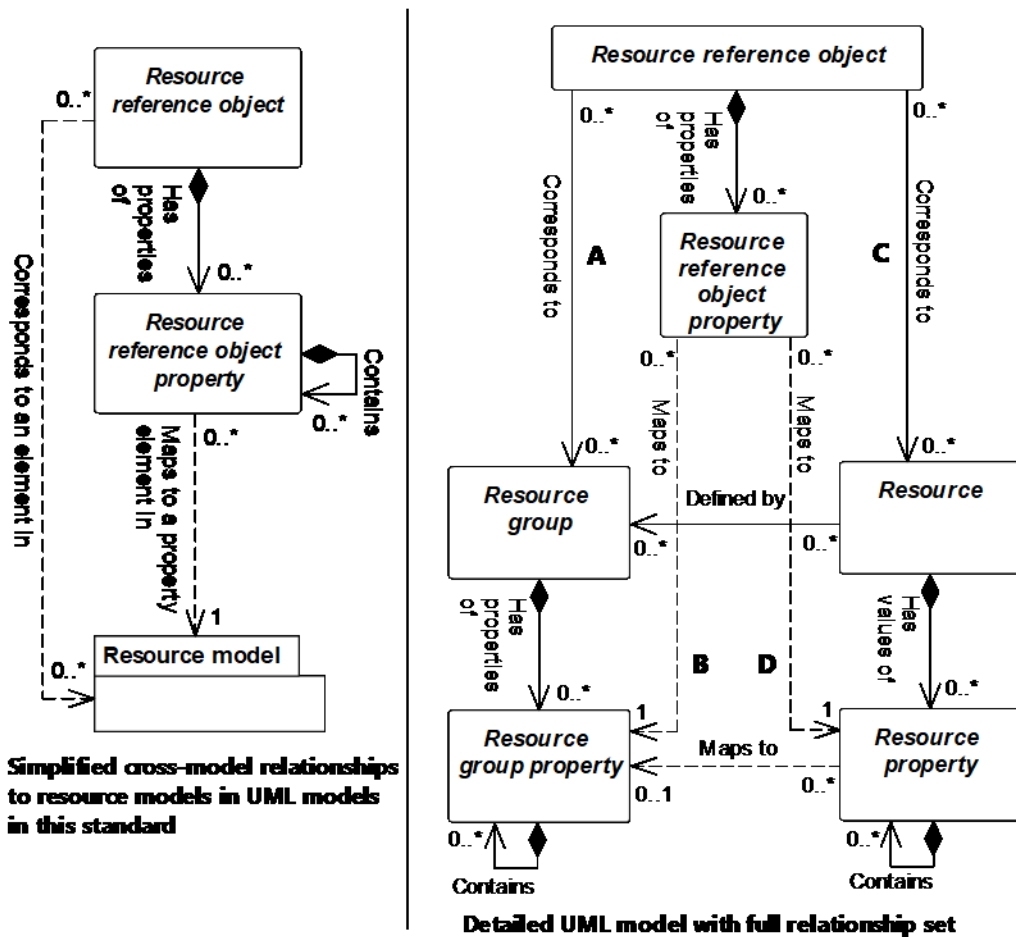


Figure 2 – Convention for simplified cross-model relationships to resource models

The implemented association relationship of the detailed UML model for a *resource reference object* is defined through only one of the following four relationships or paired relationships shown in Figure 2:

- a) A,
- b) A and B,
- c) C, and
- d) C and D.

Reference terms for resource objects in Figure 2 are

- a) *resource reference object* includes the *resource reference object* objects listed in Table 249,
- b) resource group includes personnel class, equipment class, physical asset class, material class, and material definition objects, and
- c) *resource* includes *person*, *equipment*, *physical asset*, *material lot* and *material sublot* objects.

Table 6 lists the object relationships in the detailed UML model example in Figure 2.

Table 6 – Detailed UML model relationships in Figure 2

From	To	Type	Relationship name
Resource reference object	Resource group	Association (A)	Corresponds to
Resource reference object	Resource	Association (C)	Corresponds to
Resource reference object property	Resource group property	Dependency (B)	Maps to
Resource reference object property	Resource property	Dependency (D)	Maps to
Resource reference object	Resource reference object property	Composition	Has properties of
Resource	Resource group	Association	Defined by
Resource	Resource property	Composition	Has values of
Resource group	Resource group property	Composition	Has properties of
Resource property	Resource group property	Dependency	Maps to
Resource property	Resource property	Composition hierarchy	Contains
Resource group property	Resource group property	Composition hierarchy	Contains

Table 7 provides the reference labels for each of the possible four cross-model relationships or paired relationships for referencing resource objects shown in Figure 2 for an implemented *resource reference object* instance such as Mixer 101 as an *equipment* object. The following conventions apply:

- a) only one of the four possible relationships or paired relationships are allowed for a given *resource reference object* instance;
- b) the paired relationships, Label A with B and Label C with D, show that a *resource reference object* cannot be represented by a single dependency relationship to either a *resource class property* (B) or to a *resource property* (D) since the properties are in a composition relationship;
- c) consequently, the *resource reference object* only references a resource model property if the property's parent object has an association relationship (A or D) with the *resource reference object*.

Table 7 – Four possible cross-model relationships or paired relationships for a *resource reference object* instance

Figure 3 relationship label	Possible cross-model relationships for a <i>resource reference object</i> instance
A	An association between the <i>resource reference object</i> and <i>resource group</i>
A with B	An association between the <i>resource reference object</i> and <i>resource group</i> and A dependency between the <i>resource reference object property</i> and <i>resource group property</i>
C	An association between the <i>resource reference object</i> and <i>resource</i>
C with D	An association between the <i>resource reference object</i> and <i>resource</i> and A dependency between the <i>resource reference object property</i> and <i>resource property</i>

Table 249 lists the information models of this part that apply the simplified resource models to reference the resource models and defines the set of possible relationships to the specific resource group and resource objects within each model.

Table 8 to Table 13 provide the role table for the following reference objects contained in the detailed UML model in Figure 2: *resource reference object*, *resource reference object property*, *resource group*, *resource group property*, *resource*, and *resource property*.

Table 8 – Resource reference object cross-model relationship roles

Related object (Figure 2 relationship label)	Role	Multiplicity	Relationship name	Description
Resource group (A)	Resource group	0..*	Corresponds to	The <i>resource group(s)</i> support this <i>resource reference object</i> . The <i>resource group(s)</i> support the <i>resource reference object property(s)</i> associated with this <i>resource reference object</i> .
Resource (C)	Resource	0..*	Corresponds to	The <i>resource(s)</i> support this <i>resource reference object</i> . The <i>resource(s)</i> support the <i>resource reference object property(s)</i> associated with this <i>resource reference object</i> .
Resource reference object property	Resource reference object property	0..*	Has properties of	The <i>resource reference object property(s)</i> defining in part this <i>resource reference object</i> .
EXAMPLE 1 <i>Personnel segment specification</i> cross-model relationships to personnel model				
Personnel class (A)	Personnel class	0..*	Corresponds to	The <i>personnel class(s)</i> support this <i>personnel segment specification</i> . The <i>personnel class(s)</i> support the <i>personnel segment specification property(s)</i> associated with this <i>personnel segment specification</i> .
Person (C)	Person	0..*	Corresponds to	The <i>personnel class(s)</i> support this <i>person</i> .

Related object (Figure 2 relationship label)	Role	Multiplicity	Relationship name	Description
				The <i>personnel class(s)</i> support the <i>person property(s)</i> associated with this <i>person</i> .

Table 9 – Resource reference object property cross-model relationship roles

Related object (Figure 2 relationship label)	Role	Multiplicity	Relationship name	Description
Resource group property (B)	Resource group property	1	Maps to	This resource reference object property defined by the resource group property.
Resource property (D)	Resource property	1	Maps to	This resource reference object property defined by the resource property.
Resource reference object	NA	1	Composition	This resource reference object property(s) of the resource reference object.

EXAMPLE 2 Personnel segment specification cross-model relationships to personnel model property(s)

Personnel class property (B)	Personnel class property	1	Maps to	The personnel class(s) support this personnel segment specification. The personnel class property(s) support the personnel segment specification property(s) associated with this personnel segment specification.
Person property (D)	Person property	1	Maps to	The person(s) support this personnel segment specification. The person property(s) support the personnel segment specification property(s) associated with this personnel segment specification.

Table 10 – Resource group cross-model relationship roles

Related object (Figure 2 relationship label)	Role	Multiplicity	Relationship name	Description
Resource reference object (A)	NA	0..*	Corresponds to	The resource reference object related to this resource group.
Resource	NA	0..*	Defined by	The resource defined by this resource group.
Resource group property	Resource group property	0..*	Has properties of	The resource group property(s) of this resource group property.

EXAMPLE 3 Personnel segment specification cross-model relationships with personnel class

Personnel segment specification (A)	NA	0..*	Corresponds to	The <i>personnel class(s)</i> support this <i>personnel segment specification</i> . The <i>personnel class property(s)</i> support the <i>personnel segment specification property(s)</i> associated with this <i>personnel segment specification</i> .
-------------------------------------	----	------	----------------	--

Personnel class property	Personnel class property	0..*	Has properties of	The personnel class(s) support this personnel segment specification. The personnel class property(s) support the personnel segment specification property(s) associated with this personnel segment specification.
--------------------------	--------------------------	------	-------------------	--

Table 11 – Resource group property cross-model relationship roles

Related object (Figure 2 relationship label)	Role	Multiplicity	Relationship name	Description
Resource reference object property (B)	NA	0..*	Maps to	The resource reference object property related to this resource group property.
Resource group	NA	1	Has properties of	The resource group defined in part by this resource property.
Resource property	NA	0..*	Maps to	The resource property defined by this resource group property.
Resource group property	Resource group property	0..*	Contains	The resource group property(s) of this resource group property.
EXAMPLE 4 Personnel class property cross-model relationships with a personnel segment specification property				
Personnel segment specification property (B)	NA	1	Maps to	The personnel class(s) support the personnel segment specification. This personnel class property(s) support the personnel segment specification property(s) associated with the personnel segment specification.
Personnel class property	Personnel class property	0..*	Contains	The personnel class(s) support this personnel segment specification. The personnel class property(s) support the personnel segment specification property(s) associated with this personnel segment specification.

Table 12 – Resource cross-model relationship roles

Related object (Figure 2 relationship label)	Role	Multiplicity	Relationship name	Description
Resource reference object (C)	NA	0..*	Corresponds to	This resource related to the resource reference object.
Resource group	Resource group	0..*	Defined by	This <i>resource</i> defined by the resource group
Resource property	Resource property	0..*	Has values of	The <i>resource property(s)</i> containing the values of this <i>resource</i> .
EXAMPLE 5 Personnel segment specification cross-model relationships with a <i>person</i>				
Personnel segment specification (C)	NA	0..*	Corresponds to	This <i>person(s)</i> support the <i>personnel segment specification</i> .

				This <i>person property(s)</i> support the <i>personnel segment specification property(s)</i> associated with the <i>personnel segment specification</i> .
Person property	Person property	0..*	Has values of	This <i>person(s)</i> support the <i>personnel segment specification</i> . This <i>person property(s)</i> support the <i>personnel segment specification property(s)</i> associated with this <i>personnel segment specification</i> .

Table 13 – Resource property cross-model relationship roles

Related object (Figure 2 relationship label)	Role	Multiplicity	Relationship name	Description
Resource reference object property (D)	NA	0..*	Maps to	The <i>resource reference object property</i> related to this <i>resource group property</i> .
Resource	NA	1	Has values of	The <i>resource</i> contains the values of this <i>resource property(s)</i> .
Resource group property	Resource group property	0..1	Maps to	This <i>resource property</i> defined by the <i>resource group</i> .
Resource property	Resource property	0..*	Contains	The <i>resource property(s)</i> of this <i>resource property</i> .
EXAMPLE 6 Person property cross-model relationships with a personnel segment specification property				
Personnel segment specification property (D)	NA	0..*	Maps to	This <i>person(s)</i> support the <i>personnel segment specification</i> . This <i>person property(s)</i> support the <i>personnel segment specification property(s)</i> associated with the <i>personnel segment specification</i> .
Personnel class property	Personnel class property	0..1	Maps to	This <i>person(s)</i> support the <i>personnel segment specification</i> . This <i>person property(s)</i> support the <i>personnel segment specification property(s)</i> associated with this <i>personnel segment specification</i> . This <i>person property</i> is defined in part by the <i>personnel class property</i> .

4 Manufacturing operations information models

4.1 Information models

Information models in Figure 3 define a common way of representing information that can be exchanged between different manufacturing operations activities. They define the objects that can be used to represent information, attributes of the objects, and relationships of objects in the exchange models.

A generic operations management information model is used to represent the information that can be exchanged between manufacturing operations activities and business activities. This is

illustrated in Figure 3 to indicate the four common types of exchanged information, operations schedule information, operations performance information, operations definition information, and operations capability information. The specific object models for exchanged information are defined in Clause 6.

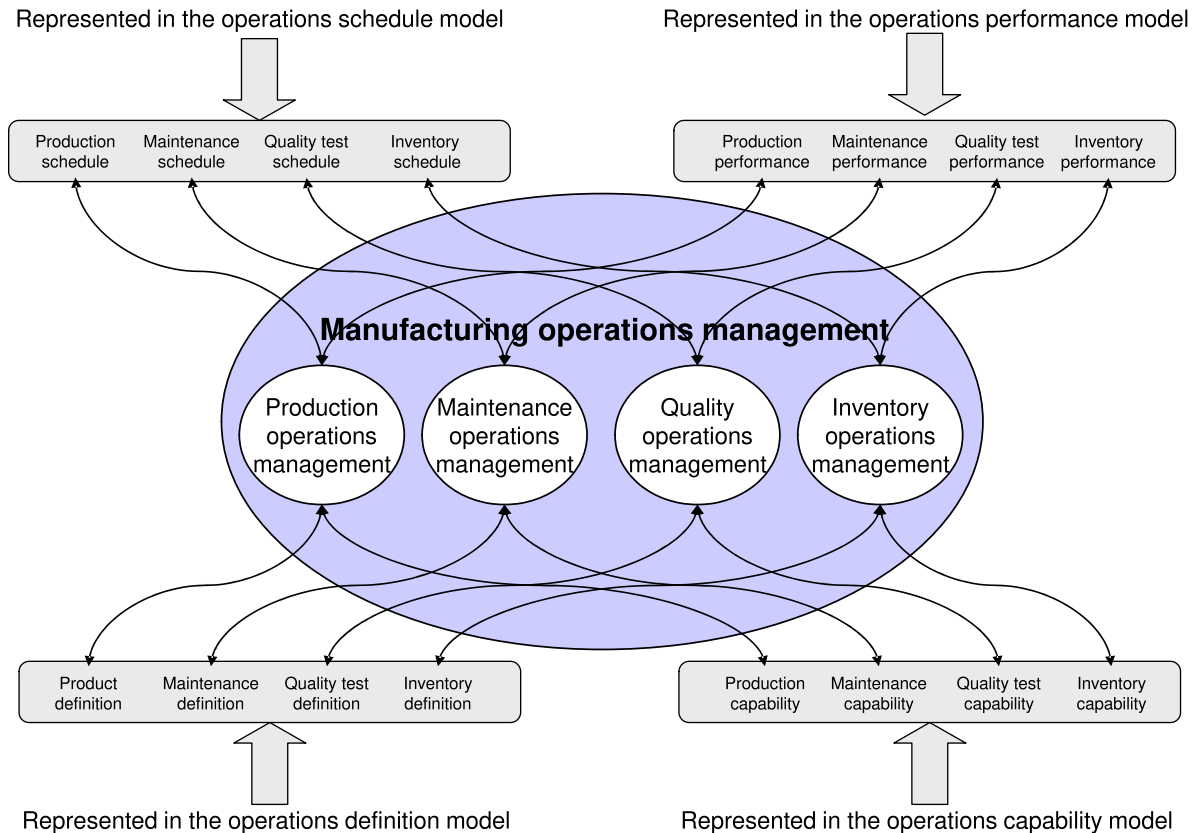


Figure 3 – Operations information models for operations management

4.2 Relationship of exchange information models to operations management activity categories

The common information models in Clause 5 describe the different types of resources and their uses in describing a (business) *process segment*. The Clause 5 common information models are also used to describe the other manufacturing operations management information models in Clause 6 and to describe the information models defined by the Part 4 of this standard.

NOTE 1 The operations management information object models described in this standard can be used for any manufacturing operations category, such as production, maintenance, quality, inventory, and inventory handling as defined in Part 1 of this standard.

NOTE 2 Although the generic information model can also be used to describe operations management information models for user-defined operations categories, conformance can be evaluated only if user-specific conformance testing scenarios are available.

Additional explanations are provided in Annex C to assist in applying these object definitions to describe the structure for the exchange information among the following operations management categories:

- production operations;

- maintenance operations;
- quality operations;
- inventory (handling) operations;
- mixed operations; and
- user-defined operations.

NOTE 3 Where Part 3 of this standard provide reference explanation for quality test operations, this standard applies across all Level 3 quality operations.

4.3 Cross-model relationships for MOM activity context in exchanges

The Part 2 operations information models and Part 4 work information models are aligned through defined cross-model relationships to provide the context of MOM activity or workflow for schedules/requirements, definitions, and performance/response exchanges. These cross-model relationships provide an integration framework for interactive manufacturing operations in the plant and between the plant and enterprise functions, Figure 4 and Table 14 shows the supported relationships between MOM objects that provide data references for schedule/request/requirement; *job order*, definitions; and responses/actuals objects in Part 2 and Part 4 of this standard.

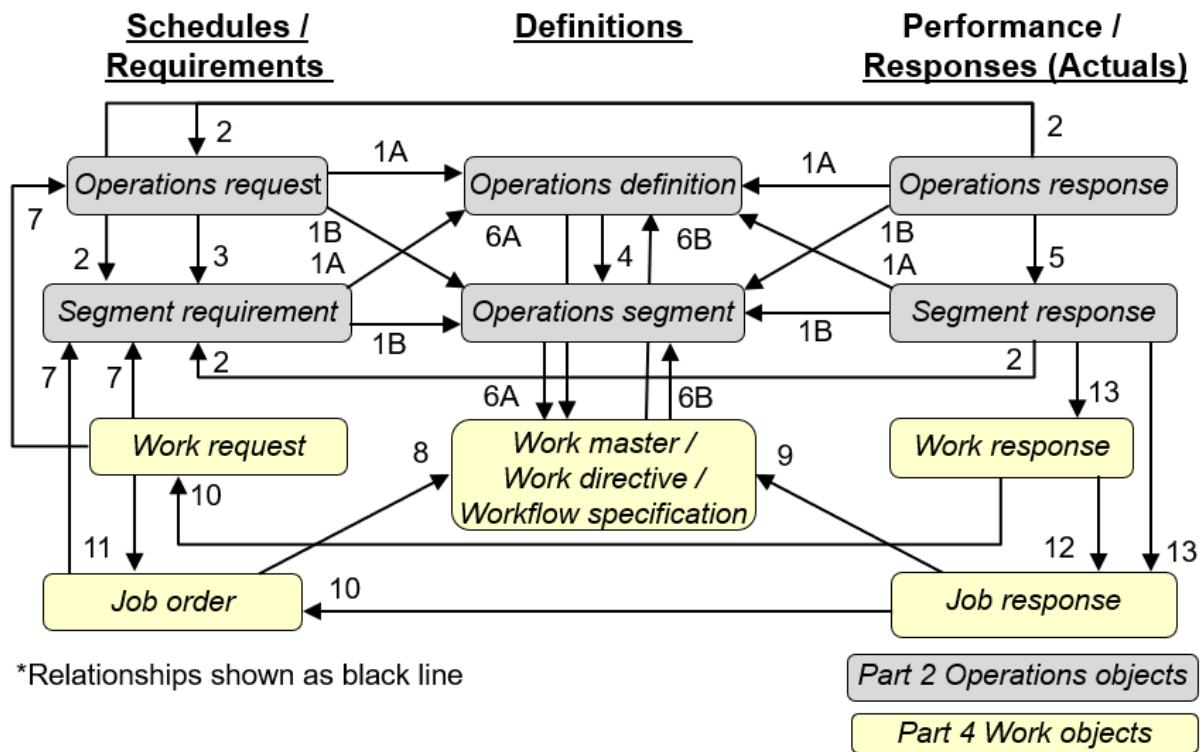


Figure 4 – Defined cross-model MOM relationships between operations & work models

Table 14 – Cross-model MOM relationship description

Reference No.	MOM context relationship description	
	From object	To object
1A	Part 2 schedule: <i>operations request</i>	Part 2 definition: <i>operations definition</i>

Reference No.	MOM context relationship description	
	From object	To object
	Part 2 requirement: <i>segment requirement</i> Part 2 response: <i>operations response, segment response</i>	
1B	Part 2 schedule: <i>operations request</i> Part 2 requirement: <i>segment requirement</i> Part 2 response: <i>operations response, segment response</i>	Part 2 definition: <i>operations segment</i>
2	Part 2 response: <i>operations response, segment response</i>	Part 2 schedule: <i>operations request</i> Part 2 requirement: <i>segment requirement</i> .
3	Part 2 schedule: <i>operations request</i>	Part 2 requirement: <i>segment requirement</i>
4	Part 2 definition: <i>operations definition</i>	Part 2 definition: <i>operation segment</i>
5	Part 2 response: <i>operations response</i>	Part 2 response: <i>segment response</i>
6A	Part 2 definition: <i>operations definition, operations segment</i>	L3 definition: <i>work master</i>
6B	Part 4 definition: <i>work directive</i>	Part 2 definition: <i>operations definition, operations segment</i>
7	Part 4 schedule: <i>job order, work request</i>	Part 2 requirement: <i>segment requirement</i> Part 2 schedule: <i>operations request</i>
8	Part 4 schedule: <i>job order</i>	Part 4 definition: <i>work master</i>
9	Part 4 response: <i>job response</i>	Part 4 definition: <i>work masters, work directives</i>
10	Part 4 response: <i>job response, work response</i>	Part 4 schedule: <i>job order, work request</i>
11	Part 4 schedule: <i>work request</i>	Part 4 requirement: <i>job order</i>
12	Part 4 response: <i>work response</i>	Part 4 response: <i>job response</i>
13	Part 2 response: <i>segment response</i>	Part 4 response: <i>work responses, job responses</i>

4.4 Abstract UML model information

The information models of this part and Part 4 of this standard represent exchanged data in an abstract UML model using

- a) objects,
- b) attributes of objects, and
- c) relationships and roles between objects.

Abstract UML model is a conceptual abstract overview of the structure of data. An abstract UML model does not provide information related to how the structure is to be implemented or the means (rules, technologies) that are needed to implement the structure shown.

NOTE 1 Abstract UML models can be represented as a logical model schema for the specific purpose to formally documents the contents of the abstract UML models.

NOTE 2 UML life cycles are not applicable to individual information exchanges. The relationships in the UML model provides a view into object lifecycles at the time of an information exchange.

Abstract UML models differ from the logical models typically used by IT groups in application development. Logical models in application development focus on a specific target. Abstract UML models are generic models, so they can be applied across many industries and applications. As such the abstract information models are focused on being abstract and generic with limited constraints on the representation. Constraints are applied during the transformation from abstract to implementation model.

NOTE 3 Implementation models are an interpretation/instance of the abstract UML model relevant as specific implementation scope. An implementation model can apply rules and requirements to the abstract model to develop a model implemented with a specific exchange technology.

4.5 Attributes of an object in an information model

4.5.1 General

Clause 4.5 describes the methods used to define object models and attributes for information exchanged between Level 3 and Level 4 activities and associated functions. The attributes are part of the definition of object models for exchanged information and terms.

4.5.2 Minimum object attribute sets

A minimum set of industry-independent information has been defined as attributes. However, values for all attributes are not be required depending on the actual usage of the models. If additional information, including industry- and application-specific information, is needed, it is presented as property objects. This mechanism is the extension capability referenced in the scope of this standard. This method increases the usability through the use of standard attributes and allows flexibility and extensibility through the use of properties. The use of properties is included to make the standard as widely applicable as practical.

4.5.3 Data types of object attributes

The attributes presented are abstract representations without any specific data type specified.

NOTE Specific implementations of this part can specify how the information is represented.

EXAMPLE 1 An attribute can be represented as a string in one implementation and as a numeric value in another implementation.

EXAMPLE 2 A date/time value can be represented in ISO standard format in one implementation and in Julian calendar format in another. Attributes for date or time can contain values for a date, a date and time or a time value, the standard does not enforce the value semantics. Each implementation will have to negotiate the value semantics.

EXAMPLE 3 An object or attribute relationship can be represented by key fields in data base tables, or by parent/child elements in an XML by nested hierarchy.

4.5.4 Object attribute extensibility

Accordingly, this standard provides mechanisms so that application or industry specific attributes can be modeled in terms of object properties and represented in object property classes in the model.

EXAMPLE The *personnel class property* can define application- or industry-specific attributes for *personnel classes* and *person property* can contain values for the properties.

NOTE The use of extensions through the addition of attributes can negate the advantage of using this standard.

4.5.5 Common object attributes

4.5.5.1 Object identification

Objects in the information model which require unique identifications contain an ID attribute, which are unique within the scope of the exchanged information.

NOTE 1 Translation of the ID attributes is possibly necessary for implementations

- from the internal ID of the source system to the interface content ID and
- from the interface content ID to the internal ID of the target system.

EXAMPLE 1 A unit can be identified as “X6777” in the interface content, as resource “R100011” in the business system, and as “East Side Reactor” in the control system.

A unique identification set should be agreed to support an information exchange.

The object IDs are used only to identify objects within related exchanged information sets.

EXAMPLE 2 Object ID values can be GUIDs or database index values.

NOTE 2 Generally, objects that are elements of composite aggregations or shared aggregation that are not referenced elsewhere in the model, does not require unique IDs.

4.5.5.2 Value type and use

4.5.5.2.1 General

Value attributes are used in properties, parameters, and data to exchange actual values.

Value attributes are also used to exchange the allowed or expected values in properties and parameters for *material definitions*, *material classes*, *equipment classes*, *personnel classes*, *physical asset classes*, *process segments*, and *operations definitions*. Value types thus represent actual single values, actual arrays of values, ranges of possible values, series values, and structured values, either as numerical or textual ranges or as sets of values.

4.5.5.2.2 Simple value types

Simple value types in Table 15 – Commonly used CCTS types for exchange should be derived from core component types that are compatible with the ISO-15000-5 Core Component Technical Specification (CCTS). The CCTS types are a common set of types that define specific terms with semantic meaning.

EXAMPLE 1 The meaning of a quantity, currency, amount, and identifier.

Table 15 – Commonly used CCTS types for exchange

AmountType	Used to define a number of monetary units specified in a currency where the unit of currency is explicit or implied.
BinaryObjectType	Used to define a data types representing graphics, pictures, sound, video, or other forms of data that can be represented as a finite length sequence of binary octets.
CodeType	Used to define a character string that is used to represent an entry from a fixed set of enumerations.
DateTimeType	Used to define a particular point in time together with the relevant supplementary information to identify the time zone information. This is a specific instance on time using the ISO 8601 CE (Common Era) calendar extended format and abbreviated versions.
IdentifierType	Used to define a character string to identify and distinguish uniquely, one instance of an object in an identification scheme from all other objects in the same scheme.
IndicatorType	Used to define a list of two mutually exclusive Boolean values that express the only possible states of a Property. EXAMPLE 2 “True” or “False”
MeasureType	Used to define a numeric value determined by measuring an object along with the specified unit of measure.
NumericType	Numeric information that is assigned or is determined by calculation, counting, or sequencing. It does not require a unit of quantity or unit of measure.
QuantityType	Used to define a counted number of nonmonetary units, possibly including fractions.
TextType	Used to define a character string generally in the form of words of a language. EXAMPLE 3 A finite set of characters

4.5.5.2.3 Array value types

Arrays of values may be represented following the syntax defined in Annex B.

EXAMPLE 1 A set of values for a single dimension array with 6 values would be represented as:

```
[ 1 , 2 , 3 , 4 , 5 , 6 ]
```

EXAMPLE 2 A set of values for a two-dimension array of size 2x3 would be represented as:

```
[ [ 1 , 2 ] , [ 3 , 4 ] , [ 5 , 6 ] ]
```

4.5.5.2.4 Range value types

Range specifications may be represented following the syntax defined in the EBNF above.

EXAMPLE 1 A simple range of values can be represented as:

```
{ 0 .. 100 }
```

EXAMPLE 2 A noncontinuous range of values can be represented as:

```
{ a .. z , A .. Z }  
{ 0 .. 100 , 200 .. 300 , 500 , 600 .. 650 }
```

4.5.5.2.5 Series value types

A specification defined as a set of allowed values may be represented following the syntax defined in Annex B.

EXAMPLE 1 A series of values that define colors can be represented as:

< Red, Green, Yellow, Blue>

EXAMPLE 2 A series of values that define equipment hierarchy levels can be represented as:

<Enterprise, Site, Area, WorkCenter, WorkUnit>

4.5.5.2.6 Structured value types

Structured data elements may be represented in this standard as properties, data and parameters as structured objects by

- a) the property model by representing the atomic elements of the structure in a flattened name space,
- b) using nested properties to represent the data structure, and
- c) a single value like arrays, ranges and series which are specific cases of structured data.

NOTE 1 The decision to use a flattened name space, nested properties, or a combination can be determined by the specific implementation.

A structure may be modeled by flattening the name space and having a single property for each structure element or by embedding a data structure within a single value attribute using EBNF notation.

EXAMPLE 1 To summarize, structured data can be represented in 3 different ways:

- a) EBNF notation,

(ABC(DEF(GHI = "x"))))

- b) Using properties/parameters in the same namespace,

ABC

ABC.DEF

ABC.DEF.GHI value = "x"

- c) Using fully qualified nested properties, and

ABC

DEF

GHI value = "x"

NOTE 2 This standard specifies how to exchange information without regard to the specific exchange element's mapping. With structured elements, there can be no guarantee that the communicating entities would have the same structure for the data. Therefore, flattening the structure to its individual elements provides a transportable format for structured data.

EXAMPLE 2 A structured element of data can be mapped to a flat name space as follows:

Structure Definition	Flattened Property Name
Struct ABC {	
Integer DEF;	ABC.DEF
Float GHI;	ABC.GHI
Array [3] of Integer JKI	ABC.JKI
}	

A structured data element may be represented by creating a property with no data value or unit of measure and with nested child properties and an identification of the element.

EXAMPLE 3 A structured data element can be mapped as follows:

C# Structure Definition	Equivalent Property
<pre>struct Simple { public int Position; public bool Exists; public double LastValue; };</pre>	<pre>Property [ID="Simple"] Property [ID="Simple"] \ Property [ID="Position"] Property [ID="Simple"] \ Property [ID="Exists"] Property [ID="Simple"] \ Property [ID="LastValue"]</pre>

A grouping or collection of related properties may be represented by creating a property with nested child properties.

EXAMPLE 4 A collection of related nominal properties can be mapped as follows:

Collection of Properties	Property Structure
<pre>NominalRate ExpectedRate LabelCode</pre>	<pre>Property [ID="Nominal"] Property [ID="Nominal"] \ Property [ID="NominalRate"] Property [ID="Nominal"] \ Property [ID="ExpectedRate"] Property [ID="Nominal"] \ Property [ID="LabelCode"]</pre>

Nested property objects are only shown in the personnel, equipment, physical asset, and material models. All property objects are also nested, as defined in the appropriate section in the text, but are not shown in the model figures in order to reduce the complexity of the figures.

4.5.5.3 Measurement uncertainty sub-attributes in value and quantity attributes

The exchange of measured quantity values, y , are supported in this standard using value, quantity, and duration attributes of objects within the information models. All value, quantity, and duration attributes defined in this standard may be subject to measurement uncertainty. When measurement uncertainty is required in the exchange, value, quantity, and duration attributes may contain the following sub-attributes:

- level of confidence;
- expanded uncertainty, U ;
- coverage factor, k ;
- precision of level of confidence; and
- accuracy of level of confidence.

NOTE 1 When a level of confidence attribute is required in the exchange, implementations can support the inclusion of one or more sets of values for the sub-attributes so that multiple levels of confidence can be exchanged for any given value, quantity, or duration attribute.

EXAMPLE Total capability of a piece of *equipment* has a y value of 100 widgets/hour. Two sets of uncertainty data are included in the exchange for each capability property value. One set has a level of confidence value of 95%, U value of 10 widgets/hour and k value of 2. The second set has a level of confidence value of 68%, U value of 5 widgets/hour and k value of 1. This information could be used in planning and scheduling in the development of possible and alternate schedules based on risk levels of 50%, 68% and 95%.

Measurement uncertainty for a measurand Y , can be expressed for a specified level of confidence in the following manner:

$$Y = y \pm U$$

where

y is the measured quantity value
 U is the expanded uncertainty

$$U = k u_c(y)$$

where

k is the coverage factor

$u_c(y)$ is the combined standard uncertainty

NOTE 2 Metrology, the science of measurement and its application, recognizes that all values which are obtained via a process of measurement are subject to measurement uncertainty. Measurement in this context refers broadly to the method by which quantities are obtained that represent the properties of a phenomenon, body or substance, where the properties have a magnitude that can be expressed as a number and a reference.

NOTE 3 Definitions pertaining to the measurement uncertainty explanation:

- a) measurand: quantity intended to be measured. [Source: ISO/IEC Guide 99, 2.3];
- b) measured quantity value: value of a measured quantity. [Source: ISO/IEC Guide 99, 2.10];
- c) measured value: quantity value representing a measurement result. [Source: ISO/IEC Guide 99, 2.10];
- d) standard uncertainty: uncertainty of the result of a measurement expressed as a standard deviation. [Source: ISO/IEC Guide 99, 2.3];
- e) measurement uncertainty: non-negative parameter characterizing the dispersion of the quantity values being attributed to a measurand, based on the information used. [Source: ISO/IEC Guide 99, 2.26];
- f) combined standard uncertainty: standard uncertainty that is obtained using the individual standard measurement uncertainties associated with the input quantities in a measurement model. [Source: ISO/IEC Guide 99, 2.31];
- g) expanded uncertainty: quantity defining an interval about the result of a measurement that can be expected to encompass a large fraction of the distribution of values that could reasonably be attributed to the measurand. The fraction can be viewed as the coverage probability or level of confidence of the interval. [Source: ISO/IEC Guide 98-3, 2.3.5];
- h) coverage probability: level of confidence probability that the set of true quantity values of a measurand is contained within a specified coverage interval. [Source: ISO/IEC Guide 99, 2.37]; and
- i) coverage factor: numerical factor used as a multiplier of the combined standard uncertainty in order to obtain an expanded uncertainty. A coverage factor, k , is typically in the range 2 to 3. [Source: ISO/IEC Guide 98-3, 2.3.6].

NOTE 4 As specified in ISO/IEC Guide 98-3, the intended purpose of expanded uncertainty, U , is to provide an interval about the result of a measurement that can be expected to encompass a large fraction of the distribution of values that could reasonably be attributed to the measurand. The choice of coverage factor k , which is usually in the range 2 to 3, is based on the coverage probability or level of confidence required of the interval.

NOTE 5 Practical methods for the expression of measurement uncertainty where the probability distribution function is asymmetric are provided in ISO/IEC Guide 98-3.

4.5.5.4 Unit of measure

This standard defines attributes for value, quantity, and duration where each has corresponding unit of measure attribute. The unit of measure is explicitly specified to ensure that it is not missed in information exchanges. Implementations of this standard can represent the unit of measure in the manner appropriate for the implementation.

5 Common object models

5.1 Hierarchy scope

The *hierarchy scope* identifies where the exchanged information fits within the role-based equipment hierarchy. It defines the scope of the exchanged information, such as a site or area for

which the information is relevant. The *hierarchy scope* identifies the associated instance in the role-based equipment hierarchy.

The *hierarchy scope* as an object attribute is not needed if the context of the exchanged information can be determined based on the exchange mechanism used.

EXAMPLE 1 A *hierarchy scope* can identify a Site, such as WEST-END. An *operations performance* can have a *hierarchy scope* attribute that identifies the WEST-END site.

EXAMPLE 2 A *hierarchy scope* can identify an Area within a Site, such as WEST-END/HOLDING-AREA. An *operations capability* can have a *hierarchy scope* attribute that identifies the area.

EXAMPLE 3 A *hierarchy scope* can identify a WORK CENTER within an Area or Site, such as WEST-END/HOLDING-AREA/CHIPPING-BIN #1.

EXAMPLE 4 A *hierarchy scope* can identify a WORK CENTER without an Area or Site identification because these are already known due to the exchange mechanism, such as CHIPPING-BIN #1.

EXAMPLE 5 A *hierarchy scope* can identify a complete hierarchy of Enterprise, Site, Area, Work Center.

Each *hierarchy scope* object defines one element in the equipment hierarchy. The *hierarchy scope* may be modeled using the model illustrated in Figure 5 in combination with Table 16 listing the relationships for *hierarchy scope*.

Table 17 listing the relationship roles for the *hierarchy scope* object.

Table 18 listing the attributes for the *hierarchy scope* object.

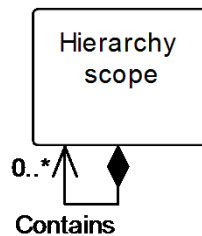


Figure 5 – Hierarchy scope model

Table 16 – Hierarchy scope relationships

From	To	Type	Relationship name
Hierarchy scope	Hierarchy scope	Composition hierarchy	Contains

Table 17 – Hierarchy scope relationship roles

Related object	Role	Multiplicity	Relationship name	Description
Hierarchy scope	Hierarchy scope child	0..*	Contains	The child <i>hierarchy scope(s)</i> contained within this <i>hierarchy scope</i> .

Table 18 – Hierarchy scope attributes

Attribute name	Description	Production examples	Maintenance examples	Quality examples	Inventory examples
Equipment ID	A unique identification of an <i>equipment</i> element	Work Center 23	West End	Ajax	North Size
Equipment level	Identification of the <i>equipment level</i> if the <i>equipment</i> element is defined	Work Center	Site	Enterprise	Area

5.2 Spatial definition

The spatial definition provides a means of communicating zero-dimensional point, one-dimensional line, or two-dimensional shape or three-dimensional solid geospatial location data for planning/scheduling, actuals, resources, and analytics. Spatial definition identifies a value and the predefined coordinate reference system using following attributes:

- a) value;
- b) format;
- c) spatial reference identifier (SRID); and
- d) SRID authority.

A fully qualified spatial definition exchange has all four attributes as a minimum. In some information exchanges, sending and receiving applications may have a specified agreement on format, SRID, and SRID authority.

Spatial definition is distinct from and separate to storage location and physical location. Storage location attribute is defined for *material lots* and *material sublots* as a nonspatial definition of location. The physical location attribute logically identifies a “place” which is actual physical location of the *physical asset*.

NOTE The spatial definition attribute is not needed if there is no requirement for exchanging geospatial location information.

Table 19 lists the attributes for the spatial definition object.

Table 19 – Attributes of spatial definition

Attribute name	Description	Production examples	Maintenance examples	Quality examples	Inventory examples
Value	A string to uniquely identify a spatial definition. EXAMPLE 1 UN/CEFACT CCTS: TextType) of the geospatial information in the format indicated in the format attribute.	lat="45.35" lon="24.15"	POINT (-1000 463)	POINT (3848472 96789)	POLYGON ((-2594646.99104 11815676.18456, -2594645.14848 11815683.08856, -2594646.99104 11815676.18456))
Format	An enumerator that determines the format of the value attribute.	GPX	WKT	WKT	WKT
SRID	SRID is the Spatial Reference Identifier which identifies the coordinate reference system to identify a predefined coordinate	4326 (WGS84)	5800 (Astra Minas)	4269 (North American	6283 (Geocentric Datum of Australia 1994)

Attribute name	Description	Production examples	Maintenance examples	Quality examples	Inventory examples
	reference system (preconfigured into communication systems). An identifier from the SRID Authority.			Datum 1983)	
SRID authority	The <i>SRID authority</i> identifies the authority that defines the coordinate reference system identified by the <i>SRID</i> . EXAMPLE 2 EPSG (see http://www.epsg-registry.org/). An organization can choose to define its own SRIDs. EXAMPLE 3 Reference its own authority.	EPSG	EPSG	EPSG	EPSG

EXAMPLE 4 Format attribute options can include but are not limited to:

- WKT (Well-known text): a text mark-up language defined by the open geospatial consortium (OGC) and used to represent vector geometry objects. WKT is defined in the ISO/IEC 13249-3:2011 standard, "Information technology – Database languages – SQL multimedia and application packages – Part 3: Spatial" (SQL/MM);
- WKB (Well-known binary): a binary representation of WKT defined in the ISO/IEC 13249-3:2011 standard, "Information technology – Database languages – SQL multimedia and application packages – Part 3: Spatial" (SQL/MM);
- GML (Geography markup language): GML defined by the Open Geospatial Consortium (OGC) and used to express geographical features. ISO 19136:2007 defines the OpenGIS® Geography Mark-up Language Encoding Standard (GML) or the Geography Markup Language (GML), version OGC 10-129r1;
- KML (Keyhole markup language): KML Version 2.2 is defined by the Open Geospatial Consortium (OGC) and used for the visualization of geographic information;
- GPX (GPS Exchange Format): GPX, 1.1 schema is the primary format for working with GPS data from mobile devices. GPX was released on August 9, 2004;
- GeoJSON: designed for representing simple geographical features based on JavaScript Object Notation (JSON). The current version is GeoJSON 1.0, 2008; and
- SVG (Scalable Vector Graphics): SVG is an XML-based vector image format for two-dimensional graphics with support for interactivity and animation. The SVG specification is an open standard developed by the World Wide Web Consortium (W3C) since 1999. The current version is SVG 2.0.

The spatial definition attribute can represent a point, line, 2D or 3D polygon or 2D or 3D solid for any resource. The value attribute is used to uniquely identify a point, line, polygon or solid. The advantage over other more complex spatial definition approaches is that the spatial definition attribute relies on an external format.

EXAMPLE 5 Figure 6 is example of WKT in 2D (3D is equally supported). WKT is defined in the ISO/IEC 13249-3:2011 standard, "Information technology – Database languages – SQL multimedia and application packages – Part 3: Spatial" (SQL/MM).

Geometry primitives (2D)		
Type	Examples	
Point		<code>POINT (30 10)</code>
LineString		<code>LINESTRING (30 10, 10 30, 40 40)</code>
Polygon		<code>POLYGON ((30 10, 40 40, 20 40, 10 20, 30 10))</code>
		<code>POLYGON ((35 10, 45 45, 15 40, 10 20, 35 10), (20 30, 35 35, 30 20, 20 30))</code>

Multipart geometries (2D)		
Type	Examples	
MultiPoint		<code>MULTIPOINT ((10 40), (40 30), (20 20), (30 10))</code>
		<code>MULTIPOINT (10 40, 40 30, 20 20, 30 10)</code>
MultiLineString		<code>MULTILINESTRING ((10 10, 20 20, 10 40), (40 40, 30 30, 40 20, 30 10))</code>
MultiPolygon		<code>MULTIPOLYGON (((30 20, 45 40, 10 40, 30 20)), ((15 5, 40 10, 10 20, 5 10, 15 5)))</code>
		<code>MULTIPOLYGON (((40 40, 20 45, 45 30, 40 40)), ((20 35, 10 30, 10 10, 30 5, 45 20, 20 35), (30 20, 20 15, 20 25, 30 20)))</code>

Figure 6 – Example, WKT in 2D (3D is equally supported)

5.3 Operational location information

5.3.1 Operational location model

Operational locations define the logical or physical places in which resources are located or are expected to be located within a plant. *Operational locations* may be made up of smaller *operational locations*. An *operational location* may belong to one or more *operational location classes*. An *operational location class* may be a specialization of one or more *operational location classes*.

The operational location model shown in Figure 7 contains the information about specific *operational locations* and classes of *operational locations*. Table 20 lists the relationships of the objects in the operational location model.

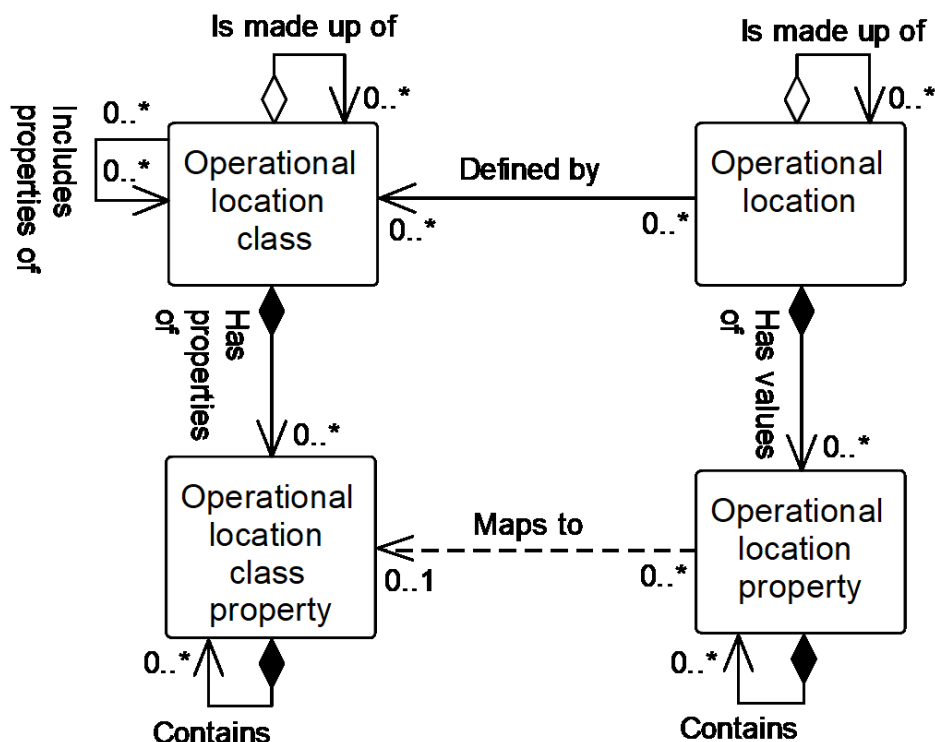


Figure 7 – Operational location model

Table 20 – Operational location model relationships

From	To	Type	Relationship name
Operational location class	Operational location class	Aggregation hierarchy	Is made up of
Operational location class	Operational location class	Association	Includes properties of
Operational location class	Operational location class property	Composition whole	Has properties of
Operational location	Operational location class	Association	Defined by
Operational location	Operational location	Aggregation hierarchy	Is made up of
Operational location	Operational location property	Composition whole	Has values of
Operational location class property	Operational location class property	Composition hierarchy	Contains
Operational location property	Operational location property	Composition hierarchy	Contains
Operational location property	Operational location class property	Dependency	Maps to

An *operational location* may specify a spatial definition attribute, which defines the position (0D) and/or boundaries (1D, 2D or 3D) of the *operational location*. A *material lot*, *material subplot*, *equipment*, *physical asset* or *person* may directly specify a spatial definition attribute, and/or it may have a corresponding *operational location*, which in turn may specify a spatial definition attribute.

Alternatively, a *material lot*, *material subplot*, *equipment*, *physical asset* or *person* may directly describe a location such as a street address instead of referencing an *operational location* object where it is not appropriate or required to define the location as an *operational location*.