



|                                                 |                                      |  |
|-------------------------------------------------|--------------------------------------|--|
| <b>TECHNICAL REPORT</b>                         | <b>EIA-IS-107™</b>                   |  |
|                                                 | Issued 1994-01<br>Stabilized 2015-07 |  |
| CDIF - Framework for Modeling and Extensibility |                                      |  |

#### NOTICE

This document has been taken directly from the original TechAmerica document and contains only minor editorial and format changes required to bring it into conformance with the publishing requirements of SAE Technical Standards. The release of this document is intended to replace the original with the SAE International document. Any numbers established by the original document remain unchanged.

The original document was adopted as an SAE publication under the provisions of the SAE Technical Standards Board (TSB) Rules and Regulations (TSB 001) pertaining to accelerated adoption of specifications and standards. TSB rules provide for (a) the publication of portions of unrevised specifications and standards without consensus voting at the SAE committee level, and (b) the use of the existing specification or standard format.

SAE Technical Standards Board Rules provide that: "This report is published by SAE to advance the state of technical and engineering sciences. The use of this report is entirely voluntary, and its applicability and suitability for any particular use, including any patent infringement arising therefrom, is the sole responsibility of the user."

SAE reviews each technical report at least every five years at which time it may be revised, reaffirmed, stabilized, or cancelled. SAE invites your written comments and suggestions.

Copyright © 2016 SAE International

All rights reserved. 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 SAE.

**TO PLACE A DOCUMENT ORDER:** Tel: 877-606-7323 (inside USA and Canada)  
Tel: +1 724-776-4970 (outside USA)  
Fax: 724-776-0790  
Email: CustomerService@sae.org  
<http://www.sae.org>

**SAE WEB ADDRESS:**

Full document details: [SAE EIAIS107](#)

SAE values your input. To provide feedback on this Technical Report, please visit <http://www.sae.org/technical/standards/EIAIS107>

Documents  
Visit Document Sharing Hub

**Interim Standard**

**EIA/IS-107**

**CDIF**

**CASE Data Interchange Format**  
A Division of Electronic Industries Association

**CDIF**  
**Framework for Modeling**  
**and Extensibility**

**January 1994**

**Electronic Industries Association**  
**CDIF Technical Committee**

## NOTICE

EIA Engineering standards and publications are designed to serve the public interest through eliminating misunderstandings between manufacturers and purchasers, facilitating interchangeability and improvement of products, and assisting the purchaser in selecting and obtaining with minimum delay the proper product for his particular need. Existence of such standards and publications shall not in any respect preclude any member or nonmember of EIA from manufacturing or selling products not conforming to such standards and publications, nor shall the existence of such standards and publications preclude their voluntary use by those other than EIA members, whether the standard is to be used either domestically or internationally.

Recommended standards and publications are adopted by EIA without regard to whether or not their adoption may involve patents on articles, materials or processes. By such action, EIA does not assume any liability to any patent owner, nor does it assume any obligation whatever to parties adopting the standard or publication.

## EIA INTERIM STANDARDS

EIA Interim Standards contain information deemed to be of technical value to the industry, and are published at the request of the originating Committee without necessarily following the rigorous public review and resolution of comments which is a procedural part of the development of an EIA Recommended Standard.

EIA Interim Standards should be reviewed on an annual basis by the formulating Committee and a decision made on whether to proceed to develop an EIA Recommended Standard on this subject. EIA Interim Standards must be canceled by the Committee and removed from the EIA Standards Catalog before the end of their third year of existence.

This EIA standard is considered to have international standardization implication, but the International Standardization Organization activity has not progressed to the point where a valid comparison between the EIA standard and the ISO work can be made.

## COPYRIGHT 1994

Published by  
ELECTRONIC INDUSTRIES ASSOCIATION  
2500 Wilson Blvd.  
Arlington, VA 22201

PRICE: Please refer to the current  
Catalog of EIA & JEDEC STANDARDS & ENGINEERING PUBLICATIONS  
or contact the EIA Electronic Information Group  
Phone: (703) 907-7545, FAX: (703) 907-7501

All rights reserved  
Printed in U.S.A.

ISBN 0-7908-0013-6

## PREFACE

This standard will assist the vendors and users of CASE tools in developing mechanisms for interchanging information between CASE tools. This standard specifies an element of a family of related standards. When used together, these standards specify a mechanism for transferring information between CASE tools.

***EIA/IS-106 CDIF - CASE Data Interchange Format - Overview* should be read first when initially exploring CDIF.** It explains the overall CDIF Architecture and how the family of standards fits together.

**This standard, *EIA/IS-107 CDIF - Framework for Modeling and Extensibility*, should also be read before the other standards in the CDIF Family of Standards.** It defines the CDIF Meta-meta-model and the modeling concepts used throughout CDIF and the extensibility mechanism.

This standard has been developed with the wide support and participation of vendors, users, academia and government involved in or familiar with the CASE industry, its products and the general requirements associated with interchanging information between these products.

The material contained in this publication was formulated under the cognizance of the CDIF Technical Committee.

# CONTENTS

|                                                                            |          |
|----------------------------------------------------------------------------|----------|
| <b>1. SCOPE</b>                                                            | <b>1</b> |
| <b>2. INTRODUCTION</b>                                                     | <b>3</b> |
| 2.1 Audience                                                               | 3        |
| 2.2 Structure of this Document                                             | 3        |
| 2.3 Conventions                                                            | 5        |
| <b>3. META-META-MODEL CONCEPTS</b>                                         | <b>6</b> |
| 3.1 Introduction                                                           | 6        |
| 3.2 Modeling Approach                                                      | 6        |
| 3.2.1 Introduction                                                         | 6        |
| 3.2.2 Naming Conventions                                                   | 7        |
| 3.2.3 Many-to-many Meta-relationships                                      | 7        |
| 3.2.4 Attributed Meta-relationships                                        | 7        |
| 3.2.5 Subtyping of Meta-entities                                           | 8        |
| 3.2.6 Multiple Inheritance in Meta-entities                                | 9        |
| 3.2.7 Subtyping of Meta-relationships                                      | 11       |
| 3.2.8 Multiple Inheritance of Meta-relationships                           | 12       |
| 3.2.9 Arity of Relationships                                               | 13       |
| 3.2.10 Mutual Exclusivity of Meta-relationships                            | 13       |
| 3.3 CDIF Graphical Notation                                                | 13       |
| 3.3.1 Introduction                                                         | 13       |
| 3.3.2 Meta-entity Symbol                                                   | 14       |
| 3.3.3 Meta-relationship Symbol                                             | 14       |
| 3.3.4 Meta-relationship Cardinality Symbol                                 | 15       |
| 3.3.5 Meta-entity Subtype Hierarchy Symbol                                 | 16       |
| 3.3.6 Meta-relationship Subtype Hierarchy                                  | 17       |
| 3.3.7 Mutual Exclusivity of Meta-relationships                             | 17       |
| 3.4 Subject Areas                                                          | 18       |
| 3.4.1 Definition                                                           | 18       |
| 3.4.2 Working Meta-model                                                   | 19       |
| 3.4.3 Explicit and Implicit Use of CollectableMetaObjects in Subject Areas | 19       |
| 3.4.4 Rules for Defining Subject Area Scope                                | 20       |
| 3.4.5 Subject Area Diagrams                                                | 21       |
| 3.5 Data Types                                                             | 21       |
| 3.5.1 Introduction                                                         | 21       |
| 3.5.2 Bitmap                                                               | 22       |
| 3.5.3 Boolean                                                              | 23       |
| 3.5.4 Date                                                                 | 23       |
| 3.5.5 Enumerated                                                           | 23       |
| 3.5.6 Float                                                                | 23       |
| 3.5.7 Identifier                                                           | 23       |
| 3.5.8 Integer                                                              | 23       |

|           |                                                                        |           |
|-----------|------------------------------------------------------------------------|-----------|
| 3.5.9     | IntegerList. . . . .                                                   | 23        |
| 3.5.10    | Point . . . . .                                                        | 24        |
| 3.5.11    | PointList. . . . .                                                     | 24        |
| 3.5.12    | String. . . . .                                                        | 24        |
| 3.5.13    | Text . . . . .                                                         | 24        |
| 3.5.14    | Time . . . . .                                                         | 24        |
| <b>4.</b> | <b>META-MODEL EXTENSIBILITY. . . . .</b>                               | <b>25</b> |
| 4.1       | Introduction . . . . .                                                 | 25        |
| 4.2       | Extensibility . . . . .                                                | 26        |
| 4.2.1     | Add SubjectArea. . . . .                                               | 26        |
| 4.2.2     | Add CollectableMetaObject.IsUsedIn.SubjectArea. . . . .                | 26        |
| 4.2.3     | Add Meta-entities . . . . .                                            | 26        |
| 4.2.4     | Add Meta-relationships . . . . .                                       | 26        |
| 4.2.5     | Add AttributableMetaObject.HasSubtype.AttributableMetaObject . . . . . | 27        |
| 4.2.6     | Add Meta-attribute to a Meta-entity or Meta-relationship . . . . .     | 27        |
| 4.2.7     | Add Legal Enumerated Value to a Meta-attribute. . . . .                | 27        |
| <b>5.</b> | <b>EXPORTER RESPONSIBILITIES. . . . .</b>                              | <b>28</b> |
| 5.1       | Introduction . . . . .                                                 | 28        |
| 5.2       | Extensibility . . . . .                                                | 28        |
| 5.3       | Maximum Output . . . . .                                               | 28        |
| 5.4       | Meta-attributes . . . . .                                              | 28        |
| <b>6.</b> | <b>IMPORTER RESPONSIBILITIES . . . . .</b>                             | <b>29</b> |
| 6.1       | Introduction . . . . .                                                 | 29        |
| 6.2       | Information Retention. . . . .                                         | 29        |
| 6.3       | Working Meta-model . . . . .                                           | 29        |
| 6.4       | Meta-attributes . . . . .                                              | 29        |
| <b>7.</b> | <b>META-META-MODEL OVERVIEW . . . . .</b>                              | <b>30</b> |
| 7.1       | Introduction . . . . .                                                 | 30        |
| 7.2       | Meta-meta-model Overview Format . . . . .                              | 30        |
| 7.3       | AttributableMetaObject Hierarchy . . . . .                             | 30        |
| 7.3.1     | Introduction. . . . .                                                  | 30        |
| 7.3.2     | Hierarchy . . . . .                                                    | 31        |
| 7.4       | Meta-meta-model Description. . . . .                                   | 31        |
| 7.4.1     | Description . . . . .                                                  | 31        |
| 7.4.2     | Uniqueness of Names . . . . .                                          | 32        |
| 7.5       | Meta-meta-model Diagram. . . . .                                       | 33        |
| 7.6       | Meta-meta-object Summary Format. . . . .                               | 34        |
| 7.7       | Meta-meta-entity Summary . . . . .                                     | 35        |
| 7.8       | Meta-meta-relationship Summary . . . . .                               | 37        |
| <b>8.</b> | <b>META-META-MODEL DEFINITIONS. . . . .</b>                            | <b>38</b> |
| 8.1       | Meta-meta-object Definition Format. . . . .                            | 38        |
| 8.2       | Meta-meta-entity and Meta-meta-attribute Definitions . . . . .         | 41        |
| 8.2.1     | AttributableMetaObject. . . . .                                        | 41        |

|                                                   |                                                                       |                          |
|---------------------------------------------------|-----------------------------------------------------------------------|--------------------------|
| 8.2.2                                             | CollectableMetaObject . . . . .                                       | 43                       |
| 8.2.3                                             | MetaAttribute . . . . .                                               | 44                       |
| 8.2.4                                             | MetaEntity . . . . .                                                  | 47                       |
| 8.2.5                                             | MetaObject . . . . .                                                  | 49                       |
| 8.2.6                                             | MetaRelationship . . . . .                                            | 53                       |
| 8.2.7                                             | SubjectArea . . . . .                                                 | 57                       |
| <b>8.3</b>                                        | <b>Meta-meta-relationship Definitions . . . . .</b>                   | <b>59</b>                |
| 8.3.1                                             | AttributableMetaObject.HasSubtype.AttributableMetaObject . . . . .    | 59                       |
| 8.3.2                                             | CollectableMetaObject.IsUsedIn.SubjectArea . . . . .                  | 60                       |
| 8.3.3                                             | MetaAttribute.IsLocalMetaAttributeOf.AttributableMetaObject . . . . . | 61                       |
| 8.3.4                                             | MetaRelationship.HasDestination.MetaEntity . . . . .                  | 62                       |
| 8.3.5                                             | MetaRelationship.HasSource.MetaEntity . . . . .                       | 63                       |
| <b>APPENDIX A META-MODEL EXAMPLE . . . . .</b>    |                                                                       | <b>65</b>                |
| <b>A1.</b>                                        | <b>Introduction . . . . .</b>                                         | <b>65</b>                |
| <b>A2.</b>                                        | <b>The Use of the Meta-meta-model Definitions . . . . .</b>           | <b>65</b>                |
| A2.1                                              | Instance of SubjectArea . . . . .                                     | 65                       |
| A2.2                                              | Instances of MetaEntities . . . . .                                   | 65                       |
| A2.3                                              | Instances of MetaRelationships . . . . .                              | 66                       |
| A2.4                                              | AttributableMetaObjects and MetaAttributes . . . . .                  | 66                       |
| A2.5                                              | Instances of MetaEntity Supertypes . . . . .                          | 66                       |
| <b>A3.</b>                                        | <b>Reading the Example Subject Area . . . . .</b>                     | <b>67</b>                |
| <b>A4.</b>                                        | <b>Subject Area Object Definitions . . . . .</b>                      | <b>69</b>                |
| A4.1                                              | Subject Area called ProjectSpecification . . . . .                    | 69                       |
| A4.2                                              | Meta-entity called Project . . . . .                                  | 70                       |
| A4.3                                              | Meta-relationship Project.IsComposedOf.Task . . . . .                 | 72                       |
| <b>APPENDIX B QUESTIONS AND ANSWERS . . . . .</b> |                                                                       | <b>73</b>                |
| <b>B1.</b>                                        | <b>Introduction . . . . .</b>                                         | <b>73</b>                |
| <b>B2.</b>                                        | <b>Extensibility . . . . .</b>                                        | <b>73</b>                |
| <b>B3.</b>                                        | <b>Subject Areas . . . . .</b>                                        | <b>76</b>                |
| <b>B4.</b>                                        | <b>Relationships . . . . .</b>                                        | <b>79</b>                |
| <b>B5.</b>                                        | <b>Identifiers and Keys . . . . .</b>                                 | <b>82</b>                |
| <b>B6.</b>                                        | <b>Other Points . . . . .</b>                                         | <b>83</b>                |
| <b>GLOSSARY . . . . .</b>                         |                                                                       | <b>87</b>                |
| <b>REFERENCES . . . . .</b>                       |                                                                       | <b>95</b>                |
| <b>INDEX . . . . .</b>                            |                                                                       | <b>97</b>                |
| <b>RELATED EIA STANDARDS . . . . .</b>            |                                                                       | <b>inside back cover</b> |

|                                                   |                                                                       |                          |
|---------------------------------------------------|-----------------------------------------------------------------------|--------------------------|
| 8.2.2                                             | CollectableMetaObject . . . . .                                       | 43                       |
| 8.2.3                                             | MetaAttribute . . . . .                                               | 44                       |
| 8.2.4                                             | MetaEntity . . . . .                                                  | 47                       |
| 8.2.5                                             | MetaObject . . . . .                                                  | 49                       |
| 8.2.6                                             | MetaRelationship . . . . .                                            | 53                       |
| 8.2.7                                             | SubjectArea . . . . .                                                 | 57                       |
| <b>8.3</b>                                        | <b>Meta-meta-relationship Definitions . . . . .</b>                   | <b>59</b>                |
| 8.3.1                                             | AttributableMetaObject.HasSubtype.AttributableMetaObject . . . . .    | 59                       |
| 8.3.2                                             | CollectableMetaObject.IsUsedIn.SubjectArea . . . . .                  | 60                       |
| 8.3.3                                             | MetaAttribute.IsLocalMetaAttributeOf.AttributableMetaObject . . . . . | 61                       |
| 8.3.4                                             | MetaRelationship.HasDestination.MetaEntity . . . . .                  | 62                       |
| 8.3.5                                             | MetaRelationship.HasSource.MetaEntity . . . . .                       | 63                       |
| <b>APPENDIX A META-MODEL EXAMPLE . . . . .</b>    |                                                                       | <b>65</b>                |
| <b>A1.</b>                                        | <b>Introduction . . . . .</b>                                         | <b>65</b>                |
| <b>A2.</b>                                        | <b>The Use of the Meta-meta-model Definitions . . . . .</b>           | <b>65</b>                |
| A2.1                                              | Instance of SubjectArea . . . . .                                     | 65                       |
| A2.2                                              | Instances of MetaEntities . . . . .                                   | 65                       |
| A2.3                                              | Instances of MetaRelationships . . . . .                              | 66                       |
| A2.4                                              | AttributableMetaObjects and MetaAttributes . . . . .                  | 66                       |
| A2.5                                              | Instances of MetaEntity Supertypes . . . . .                          | 66                       |
| <b>A3.</b>                                        | <b>Reading the Example Subject Area . . . . .</b>                     | <b>67</b>                |
| <b>A4.</b>                                        | <b>Subject Area Object Definitions . . . . .</b>                      | <b>69</b>                |
| A4.1                                              | Subject Area called ProjectSpecification . . . . .                    | 69                       |
| A4.2                                              | Meta-entity called Project . . . . .                                  | 70                       |
| A4.3                                              | Meta-relationship Project.IsComposedOf.Task . . . . .                 | 72                       |
| <b>APPENDIX B QUESTIONS AND ANSWERS . . . . .</b> |                                                                       | <b>73</b>                |
| <b>B1.</b>                                        | <b>Introduction . . . . .</b>                                         | <b>73</b>                |
| <b>B2.</b>                                        | <b>Extensibility . . . . .</b>                                        | <b>73</b>                |
| <b>B3.</b>                                        | <b>Subject Areas . . . . .</b>                                        | <b>76</b>                |
| <b>B4.</b>                                        | <b>Relationships . . . . .</b>                                        | <b>79</b>                |
| <b>B5.</b>                                        | <b>Identifiers and Keys . . . . .</b>                                 | <b>82</b>                |
| <b>B6.</b>                                        | <b>Other Points . . . . .</b>                                         | <b>83</b>                |
| <b>GLOSSARY . . . . .</b>                         |                                                                       | <b>87</b>                |
| <b>REFERENCES . . . . .</b>                       |                                                                       | <b>95</b>                |
| <b>INDEX . . . . .</b>                            |                                                                       | <b>97</b>                |
| <b>RELATED EIA STANDARDS . . . . .</b>            |                                                                       | <b>inside back cover</b> |

|                                                   |                                                                       |                          |
|---------------------------------------------------|-----------------------------------------------------------------------|--------------------------|
| 8.2.2                                             | CollectableMetaObject . . . . .                                       | 43                       |
| 8.2.3                                             | MetaAttribute . . . . .                                               | 44                       |
| 8.2.4                                             | MetaEntity . . . . .                                                  | 47                       |
| 8.2.5                                             | MetaObject . . . . .                                                  | 49                       |
| 8.2.6                                             | MetaRelationship . . . . .                                            | 53                       |
| 8.2.7                                             | SubjectArea . . . . .                                                 | 57                       |
| <b>8.3</b>                                        | <b>Meta-meta-relationship Definitions . . . . .</b>                   | <b>59</b>                |
| 8.3.1                                             | AttributableMetaObject.HasSubtype.AttributableMetaObject . . . . .    | 59                       |
| 8.3.2                                             | CollectableMetaObject.IsUsedIn.SubjectArea . . . . .                  | 60                       |
| 8.3.3                                             | MetaAttribute.IsLocalMetaAttributeOf.AttributableMetaObject . . . . . | 61                       |
| 8.3.4                                             | MetaRelationship.HasDestination.MetaEntity . . . . .                  | 62                       |
| 8.3.5                                             | MetaRelationship.HasSource.MetaEntity . . . . .                       | 63                       |
| <b>APPENDIX A META-MODEL EXAMPLE . . . . .</b>    |                                                                       | <b>65</b>                |
| <b>A1.</b>                                        | <b>Introduction . . . . .</b>                                         | <b>65</b>                |
| <b>A2.</b>                                        | <b>The Use of the Meta-meta-model Definitions . . . . .</b>           | <b>65</b>                |
| A2.1                                              | Instance of SubjectArea . . . . .                                     | 65                       |
| A2.2                                              | Instances of MetaEntities . . . . .                                   | 65                       |
| A2.3                                              | Instances of MetaRelationships . . . . .                              | 66                       |
| A2.4                                              | AttributableMetaObjects and MetaAttributes . . . . .                  | 66                       |
| A2.5                                              | Instances of MetaEntity Supertypes . . . . .                          | 66                       |
| <b>A3.</b>                                        | <b>Reading the Example Subject Area . . . . .</b>                     | <b>67</b>                |
| <b>A4.</b>                                        | <b>Subject Area Object Definitions . . . . .</b>                      | <b>69</b>                |
| A4.1                                              | Subject Area called ProjectSpecification . . . . .                    | 69                       |
| A4.2                                              | Meta-entity called Project . . . . .                                  | 70                       |
| A4.3                                              | Meta-relationship Project.IsComposedOf.Task . . . . .                 | 72                       |
| <b>APPENDIX B QUESTIONS AND ANSWERS . . . . .</b> |                                                                       | <b>73</b>                |
| <b>B1.</b>                                        | <b>Introduction . . . . .</b>                                         | <b>73</b>                |
| <b>B2.</b>                                        | <b>Extensibility . . . . .</b>                                        | <b>73</b>                |
| <b>B3.</b>                                        | <b>Subject Areas . . . . .</b>                                        | <b>76</b>                |
| <b>B4.</b>                                        | <b>Relationships . . . . .</b>                                        | <b>79</b>                |
| <b>B5.</b>                                        | <b>Identifiers and Keys . . . . .</b>                                 | <b>82</b>                |
| <b>B6.</b>                                        | <b>Other Points . . . . .</b>                                         | <b>83</b>                |
| <b>GLOSSARY . . . . .</b>                         |                                                                       | <b>87</b>                |
| <b>REFERENCES . . . . .</b>                       |                                                                       | <b>95</b>                |
| <b>INDEX . . . . .</b>                            |                                                                       | <b>97</b>                |
| <b>RELATED EIA STANDARDS . . . . .</b>            |                                                                       | <b>inside back cover</b> |

This document (*EIA/IS-107 CDIF - Framework for Modeling and Extensibility*) includes the definition of the CDIF Meta-meta-model and describes the rules and notations used throughout the CDIF Family of Standards. The rules for extending the CDIF Integrated Meta-model are also defined.

---

Scope

## 2. INTRODUCTION

### 2.1 Audience

This document is intended to be used by anyone wishing to understand and/or use CDIF. This document provides a definition of the framework of the entire CDIF Family of Standards. It is suitable for:

- those evaluating CDIF
- those who wish to understand the principles and concepts of a CDIF transfer
- those developing importers and exporters.

The document *EIA/IS-106 CDIF - CASE Data Interchange Format Overview* and this document, *EIA/IS-107 CDIF - Framework for Modeling and Extensibility*, should be read first when initially exploring CDIF and before attempting to read other documents in the CDIF Family of Standards.

While there are no specific prerequisites for reading this document, it will be helpful for the reader to have familiarity with the following:

- Entity-Relationship-Attribute modeling
- CASE tools
- Information repositories
- Data dictionaries
- Multiple meta-layer modeling.

### 2.2 Structure of this Document

This document is organized into the following sections:

#### Section 1 'Scope'

This describes the CDIF Family of Standards and positions the current document within the set of documents.

#### Section 2 'Introduction'

This describes the intended audience, structure and conventions used in the document.

### **Section 3 'Meta-meta-model Concepts'**

This defines the Meta-meta-model, modeling rules and graphical conventions that are used. The data types supported in meta-models are introduced. The concept of the Working Meta-model is also explained.

### **Section 4 'Meta-model Extensibility'**

This describes the types of extensibility that may be used when extending the CDIF Integrated Meta-model.

### **Section 5 'Exporter Responsibilities'**

This describes the responsibilities of a tool with regard to overlapping semantics and maximum output.

### **Section 6 'Importer Responsibilities'**

This describes the responsibilities of a tool with regard to the retention of information.

### **Section 7 'Meta-meta-model Overview'**

This contains an overview of the CDIF Meta-meta-model. This section includes an AttributableMetaObject Hierarchy, a Meta-meta-model Diagram, a Meta-meta-entity Summary and a Meta-meta-relationship Summary.

### **Section 8 'Meta-meta-model Definitions'**

This contains the full definitions of all the objects in the CDIF Meta-meta-model.

### **Appendix A 'Meta-model Example'**

This shows how the modeling approach is applied.

### **Appendix B 'Questions and Answers'**

This includes detailed responses to questions that have been raised by those reviewing this document. It is intended that the inclusion of these will aid the reader in understanding this document, and related issues concerning CDIF.

### **Glossary**

The Glossary contains definitions of terms used within this Standard.

### **References**

This contains details of other standards referenced within this standard.

### **Index**

A comprehensive index is provided.

## 2.3 Conventions

The typographical and naming conventions defined below are used throughout the CDIF Family of Standards.

- Double quotes are used to introduce new terms (e.g., "model layer")
- Bold type is used for emphasis (e.g., this is **important**)
- All meta-objects and meta-meta-objects in CDIF (in meta-models and meta-meta-models) are named by concatenating all the words that name the meta-object or meta-meta-object; the first letter of each word is upper-case, the rest are lower-case (e.g., *MetaAttribute*, *AttributeDerivation*, *IsDrawnUsing*, *IsOptional*)

Full details of the CDIF Graphical Notation used in the Meta-model and the Meta-meta-model can be found in Section 3.3, CDIF Graphical Notation.

### 3. META-META-MODEL CONCEPTS

#### 3.1 Introduction

This section describes the concepts of the CDIF Meta-meta-model. An overview and a diagram of the CDIF Meta-meta-model are included in Section 7, Meta-meta-model Overview. The formal definition can be found in Section 8, Meta-meta-model Definitions.

The CDIF Meta-meta-model is defined using the same facilities that are used to define the CDIF Meta-model (and could be used to define models). It is, therefore, quite common to get the names of the objects at the different meta-layers confused. To minimize this problem, the reader should remember the following:

- The components of the Meta-meta-model are known as meta-meta-objects (i.e., meta-meta-entities, meta-meta-relationships and meta-meta-attributes)
- The components of the Meta-model are known as meta-objects (i.e., meta-entities, meta-relationships and meta-attributes)
- The components of models are known as objects (e.g., processes, entities, flows)
- The names of the components in the Meta-meta-model usually have a single meta- as a prefix (e.g., there is a meta-meta-entity, in the Meta-meta-model, known as *MetaAttribute*)
- The names of the components in the meta-model usually have no meta prefix.

The CDIF Meta-meta-model is the definition of the concepts used to develop CDIF-conformant meta-models. This includes the Entity-Relationship-Attribute diagramming technique and information definition requirements pertinent to the meta-model development process.

#### 3.2 Modeling Approach

##### 3.2.1 Introduction

Because the CDIF Meta-meta-model is used to define meta-models, it must be capable of expressing a complete description of a meta-model. A CDIF-conformant meta-model contains three types of objects: entities, relationships and attributes; these objects in the meta-model are properly known as meta-entities, meta-relationships and meta-attributes. These objects are used to model the information content of the meta-model, and in fact, it is a rule in CDIF, that any single concept, or set of information may only be represented one way; that is, although there may be several ways of modeling a concept, only one will be included in the model.

The Meta-meta-model is composed of two things:

- An Entity-Relationship-Attribute model that is used to describe the CDIF Meta-model. This Entity-Relationship-Attribute model is also self-describing (i.e., is defined using itself).
- A graphical notation for representing occurrences (i.e., instances) of the Meta-meta-model. This notation is used in CDIF in two ways; to depict the contents of subject areas in the CDIF Meta-model, and to depict the CDIF Meta-meta-model itself.

It is important for the reader to remember that the rules in this section apply only to the meta-models defined using the CDIF Meta-meta-model. In particular, these rules do not relate to Entity-Relationship-Attribute models that are transferred during a CDIF transfer (see *EIA/IS-114 CDIF - Integrated Meta-model - Data Modeling Subject Area* for further information).

This section describes some general modeling rules that are part of the definition of the CDIF Meta-meta-model. They are described with a meta- prefix. The rules apply to the CDIF Integrated Meta-model and all extensions defined using the extensibility mechanism. The rules are also adhered to by the CDIF Meta-meta-model<sup>1</sup>.

### 3.2.2 Naming Conventions

All meta-entities, meta-relationships and meta-attributes in CDIF (in either the Meta-model or the Meta-meta-model) are named by concatenating together all the words that name the object. The first letter of each word in a name is upper-case and the rest are lower case. This convention can be relaxed in those cases where the "words" that the object name is composed of, already have capitalization or word-separator information within them. This is especially common when one of the "words" is an acronym (e.g., *VCRMModelName*). Other examples of names that conform to this convention include: *Person*, *HasSubtype*, *IsUsedToRepresent*.

The name shall have a maximum of 32 characters. Names shall conform to the Identifier data type, see Section 3.5.7, Identifier.

### 3.2.3 Many-to-many Meta-relationships

The CDIF Meta-meta-model supports many-to-many meta-relationships.

### 3.2.4 Attributed Meta-relationships

The CDIF Meta-meta-model supports attributed meta-relationships.

---

<sup>1</sup> The CDIF modeling rules may be used at any meta-level; they may be used to create models.

### 3.2.5 Subtyping of Meta-entities

The CDIF Meta-meta-model provides support for subtyping (sometimes known as specialization) and its converse, supertyping (sometimes known as generalization).

In particular, a meta-entity may be defined to be a "subtype" of another meta-entity. The subtype meta-entity inherits all of the meta-attributes and meta-relationships of its immediate and indirect supertype meta-entities.

Supertyping is used as a classification mechanism to group meta-objects with a substantial level of common semantics; an importer can use this classification as an aid to deciding how to represent any meta-object that is not directly representable within the importing tool.

Subtype meta-entities have unique *CDIFMetaIdentifiers*; that is, the *CDIFMetaIdentifier* is not inherited from the parent. The *Description*, *Constraints*, *Usage*, and *Aliases* that were defined for the parent meta-entity are inherited. These textual fields may also be expanded in the local definition of the subtype meta-entity. Subtype meta-entities inherit all the meta-attributes that have been defined for the parent meta-entity, and all those of its antecedents. The characteristics of the meta-attribute (for example, *Name*) are unchanged when inherited by a subtype meta-entity. Additional, local, meta-attributes may be defined specific to the subtype meta-entity.

In the same way that meta-attributes are inherited, any meta-relationships that are defined with the parent meta-entity as the source or destination are inherited by subtype meta-entities. None of the characteristics of the meta-relationship are changed by its use by a subtype; that is, the *Name*, *Cardinality*, *CDIFMetaIdentifier*, *Aliases*, *Constraints*, *Description*, and *Usage*, are unchanged. Notice that even the full name of the meta-relationship is unchanged and will contain the name of the supertype meta-entity rather than that of the subtype.

A summary of the inheritance rules for subtype meta-entities is shown in Table 1.

Table 1  
Inheritance of Meta-meta-attribute Values by Subtype Meta-entities

| Meta-meta-attribute | Inherited by Subtype Meta-entity |
|---------------------|----------------------------------|
| Name                | No                               |
| CDIFMetaIdentifier  | No                               |
| Description         | Yes, may be extended             |
| Usage               | Yes, may be extended             |
| Aliases             | Yes, may be extended             |
| Constraints         | Yes, may be extended             |
| Type                | No                               |

### 3.2.6 Multiple Inheritance in Meta-entities

A meta-entity may have multiple supertypes, allowing for "multiple inheritance". In this case, the subtype inherits all of the meta-attributes and meta-relationships of all of its supertypes (and their supertypes, etc.).

There are several situations that occur when using multiple-inheritance that either seem to, or do, present inheritance conflicts.

When several inheritance threads of a meta-entity converge at a common supertype, the inheriting meta-entity does not inherit multiple copies of the meta-attributes and meta-relationships from that point to the top of the hierarchy; it only inherits one copy. Figure 2 shows meta-entity *A* having two subtype meta-entities (*B* and *C*), which are both supertype meta-entities of meta-entity *D*. Both local and inherited meta-attributes for all meta-entities are shown.

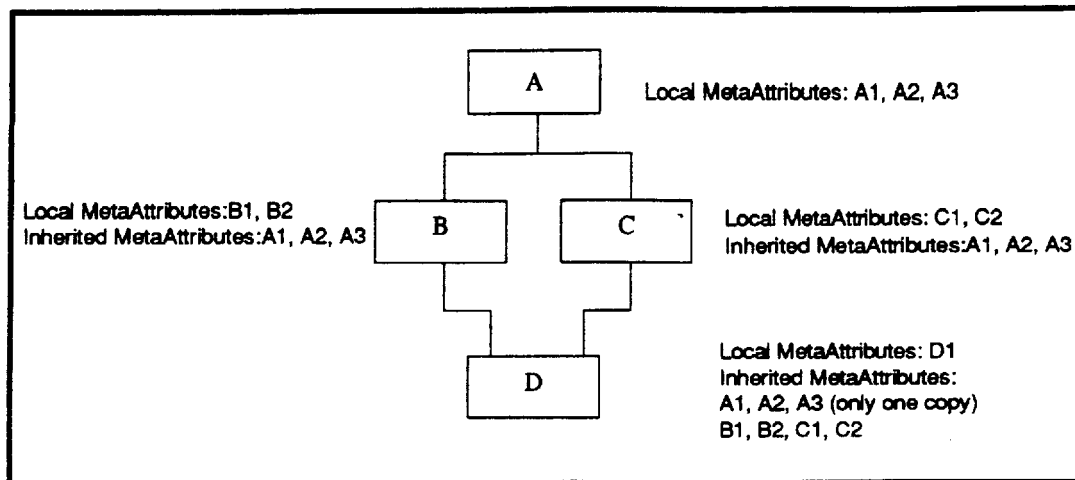


Figure 2  
Multiple Inheritance

The other potential conflict occurs when a meta-entity inherits meta-attributes from two supertypes, both having a meta-attribute with the same name that are, in fact, different meta-attributes. The behavior for CDIF is undefined in these circumstances. To guarantee proper (predictable) behavior from importers and exporters, this situation must be avoided. The CDIF Integrated Meta-model and CDIF Meta-meta-model are defined ensuring that this does not occur.

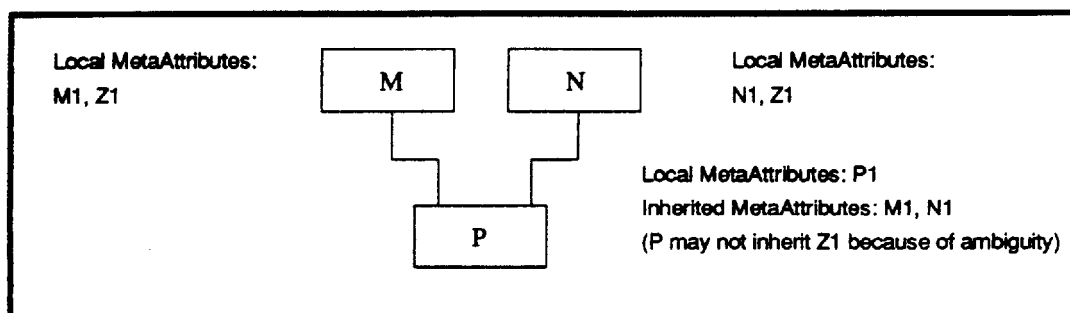


Figure 3  
Multiple Inheritance - Invalid Meta-attribute, *Z1*

Figure 3 shows an invalid inheritance example, where the meta-entity *P* has two supertype meta-entities (*M* and *N*) that both have an identically named meta-attribute, *Z1*.

### 3.2.7 Subtyping of Meta-relationships

A meta-relationship may be defined to be a subtype of another meta-relationship.

A subtype meta-relationship may be defined at the 'level' in the meta-entity hierarchy where the supertype meta-relationship was defined (that is, with the same source and destination meta-entities as the supertype). Alternatively, the source meta-entity or destination meta-entity, or both, may be subtypes of the source and destination meta-entities in the supertype meta-relationship.

Each subtype meta-relationship has a unique *CDIFMetaIdentifier*; that is, it does not inherit the *CDIFMetaIdentifier* from the supertype meta-relationship.

The cardinalities of the subtype meta-relationship are defined locally; however, they may only be as constrained, or more constrained than at the supertype level. Therefore, the minimum cardinality of the subtype must be greater than, or equal to, that of the supertype, and the maximum cardinality must be less than, or equal to, that of the supertype.

The other, textual, characteristics of the subtype meta-relationship are inherited from the supertype meta-relationship but may be further refined for the subtype.

The immediate and indirect subtype meta-relationship inherits any meta-attributes that have been defined for the supertype meta-relationship. None of the characteristics of the meta-attributes are changed. Additional meta-attributes may be created for the subtype meta-relationship.

A summary of the inheritance rules for subtype meta-relationships is shown in Table 2.

Table 2  
Inheritance of Meta-meta-attribute Values by Subtype Meta-relationships

| Meta-meta-attribute | Inherited by Subtype Meta-relationship |
|---------------------|----------------------------------------|
| Name                | No                                     |
| CDIFMetaIdentifier  | No                                     |
| MinSourceCard       | No, must be locally defined            |
| MaxSourceCard       | No, must be locally defined            |
| MinDestCard         | No, must be locally defined            |
| MaxDestCard         | No, must be locally defined            |
| Description         | Yes, may be extended                   |
| Usage               | Yes, may be extended                   |
| Aliases             | Yes, may be extended                   |
| Constraints         | Yes, may be extended                   |

Between any pair of meta-entities, there may be both a supertype meta-relationship (possibly due to inheritance by a subtype meta-entity) and a subtype meta-relationship defined. There may not be an instance of both the supertype meta-relationship and the subtype meta-relationship between the same pair of meta-entity instances. However, in most cases it is expected that the supertype meta-relationship will be abstract (not instantiable), which means that the condition of an instance of both supertype and subtype meta-relationships between the same instance pair will not arise.

### 3.2.8 Multiple Inheritance of Meta-relationships

Meta-relationships may inherit from multiple supertypes in the same way that meta-entities may inherit from multiple supertypes. The discussion in Section 3.2.6, Multiple Inheritance in Meta-entities, that describes potential inheritance conflicts arising from multiple inheritance applies equally to multiple inheritance of meta-relationships, and the diagrams, although depicting meta-entities (in boxes) may be read as though the boxes represent meta-relationships.

### 3.2.9 Arity of Meta-relationships<sup>2</sup>

The CDIF Integrated Meta-model, and any extensions, must be defined using binary meta-relationships rather than n-ary meta-relationships. (Note that the functionality of n-ary relationships is covered in the Data Modeling subject area in the CDIF Integrated Meta-model, see *EIA/IS-114 CDIF - Integrated Meta-model - Data Modeling Subject Area*.)

### 3.2.10 Mutual Exclusivity of Meta-relationships

In meta-models mutual exclusivity of meta-relationships is documented as a constraint on the meta-relationships that are in the exclusivity group.

## 3.3 CDIF Graphical Notation

### 3.3.1 Introduction

The same graphical notation is used for both the CDIF Meta-meta-model and CDIF Meta-model. The notation is described in this section with regard to meta-models, but applies equally to the Meta-meta-model defined in this document<sup>3</sup>. For example, the meta-entity symbol is defined; however, the definition is equally valid for the meta-meta-entity symbol.

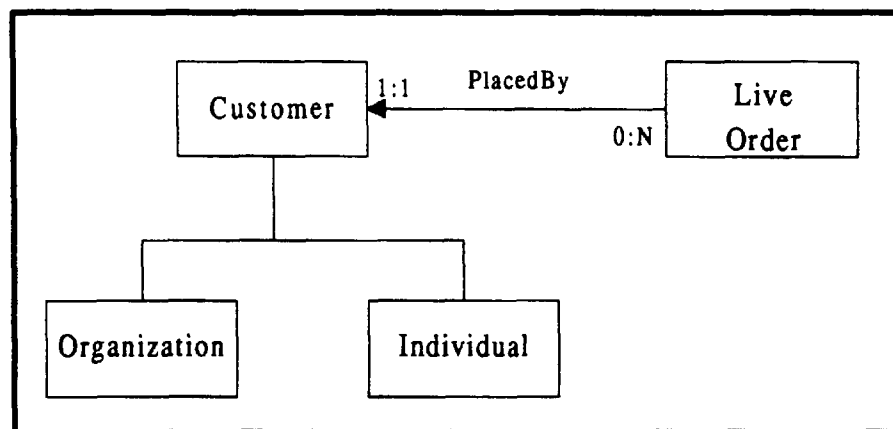


Figure 4  
Example of CDIF Graphical Notation

<sup>2</sup> The arity of a relationship may also be called the degree of the relationship. A binary relationship has an arity of two; a ternary relationship has an arity of three, etc.

<sup>3</sup> CDIF Graphical Notation can be used for any meta-level. This particular example, shown in Figure 5, is of a model.

Figure 4 is an example model that uses the CDIF Graphical Notation. A further example is shown in Appendix A.

### 3.3.2 Meta-entity Symbol

Meta-entities (that is, instances of the meta-meta-entity called *MetaEntity*) used in a subject area are drawn as rectangular boxes, with the name of the meta-entity on a white background. Examples are '*Meta-entity A*' in Figure 5, and *Customer* in Figure 4.

Additionally, a special notations may be used to represent meta-entities on diagrams of meta-models. If the meta-entity is not used in a subject area but is used in another subject area and is included in the diagram for clarity, the box shall be shaded. An example is '*Meta-entity B*' in Figure 5.

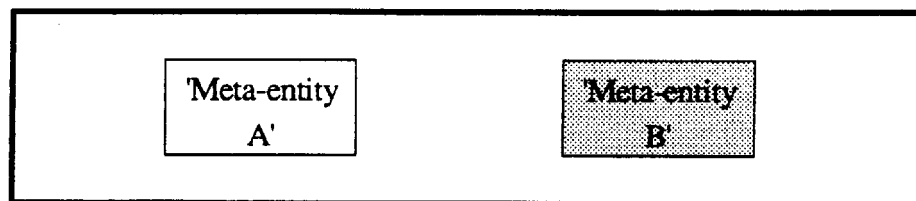


Figure 5  
The Meta-entity Symbol

### 3.3.3 Meta-relationship Symbol

This symbol names and depicts an instance of the meta-meta-entity *MetaRelationship* which exists between two meta-entities. It is usually shown as a straight line (possibly multiple segments) joining two meta-entities/rectangles. A single arrowhead must appear somewhere along the meta-relationship line to indicate the direction in which the meta-relationship name shall be read. The two meta-entities participating in a meta-relationship are referred to as the "source" and "destination" as indicated by the arrowhead. Examples of meta-relationships are '*Meta-relationship A*' in Figure 6, and *PlacedBy* in Figure 4.

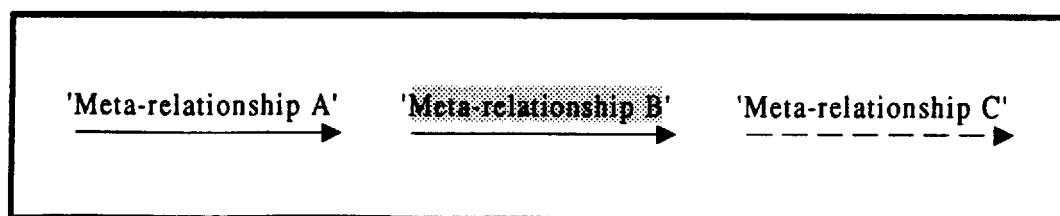


Figure 6  
The Meta-relationship Symbol

Additionally, three special notations may be used to represent meta-relationships on diagrams of meta-models.

If a meta-relationship is included in the diagram for clarification, but it is not used in the subject area, the text of the meta-relationship name shall be back-shaded. This is shown for '*Meta-relationship B*' in Figure 6.

If a supertype meta-entity that participates in a meta-relationship is not included in the diagram, the inherited meta-relationship may be shown on the diagram as a broken line to/from the inheriting subtype meta-entity. This is shown for '*Meta-relationship C*' in Figure 6.

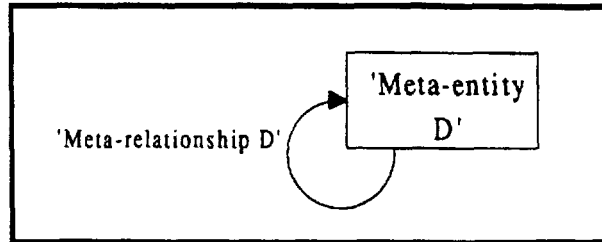


Figure 7  
A Reflexive Meta-relationship

Where a reflexive (recursive) meta-relationship is drawn from a meta-entity to itself, it may be drawn as an arc (often referred to as a pig's ear or scorpion) as shown in Figure 7.

### 3.3.4 Meta-relationship Cardinality Symbol

The cardinality and optionality of each meta-relationship is defined by colon-separated labels (x:y) at both ends of the meta-relationship line symbol. The labels indicate the minimum (x) and the maximum (y) number of instances of the meta-entity nearest the label from the perspective of a single instance of the other meta-entity participating in the meta-relationship. The cardinalities of the *PlacedBy* meta-relationship are shown in Figure 8.

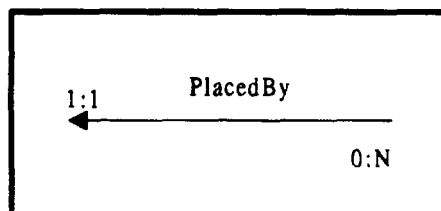


Figure 8  
The Meta-relationship Cardinality Symbol

The expansion of the notation into text for the *PlacedBy* meta-relationship shown in Figure 4 can be read:

"Each *LiveOrder* must be *placed by* one (and only one) *Customer*."

"must be" because the minimum cardinality at the *Customer* end of the meta-relationship is 1, and "one (and only one)" because the maximum cardinality is also 1.

Reading the meta-relationship in the other direction gives us:

"Each *Customer* may be *placer of* one or more *Live Orders*."

"may be" because the minimum cardinality at the *Customer* end is zero, and "one or more" because the maximum cardinality at *LiveOrder* is N, for many.

### 3.3.5 Meta-entity Subtype Hierarchy Symbol

A special type of symbol is used to show that a meta-entity is a subtype of another meta-entity. It is shown as a hierarchy from the supertype meta-entity to each subtype. The supertype will always occur closer to the top of the diagram than its subtypes. The lines that connect the supertype to its subtypes are **not** labeled and are constrained to vertical and horizontal. Figure 9 shows an example of a single supertype meta-entity with two subtype meta-entities; that is, customers may either be organizations or individuals.

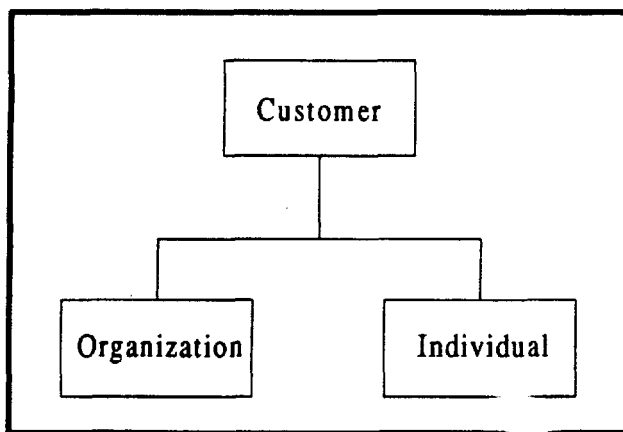


Figure 9  
The Meta-entity Subtype Hierarchy Symbol

Figure 10 shows an example where a single subtype meta-entity is inheriting characteristics from two supertype meta-entities; an example of multiple inheritance.

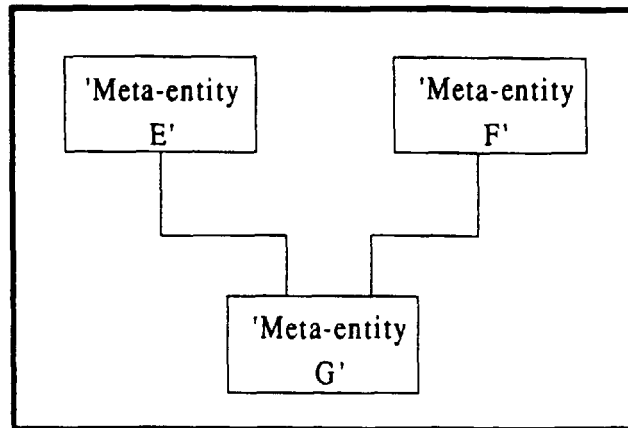


Figure 10  
The Meta-entity Subtype Hierarchy Symbol - Multiple Inheritance

### 3.3.6 Meta-relationship Subtype Hierarchy

The meta-relationship subtype hierarchy is not represented diagrammatically. The inheritance of meta-relationships is shown in the *AttributableMetaObject* hierarchy that is provided for each subject area. For example the following meta-relationship hierarchy is defined in the Common subject area (*EIA/IS-112 CDIF - Integrated Meta-model - Common Subject Area*):

```
RootEntity.IsRelatedTo.RootEntity
  ProjectionComponent.IsProjectionOf.SemanticObject
    ProjectionComponent.IsFullProjectionOf.DataObject
      ProjectionComponent.IsProjectionOf.Attribute
```

### 3.3.7 Mutual Exclusivity of Meta-relationships

Mutual exclusivity of meta-relationships may be shown diagrammatically by placing an arc across the mutually exclusive meta-relationships.

Each instance of the meta-entity on which the mutual exclusivity group is specified can be connected to instances of only one of the meta-entities at the far end of all the included meta-relationships. If a connection to an instance of a meta-entity at the far end is required for each meta-entity instance at the near end, then the CDIF convention will be to indicate this with a minimum cardinality value of one (1:x) at the far end of all the included meta-relationships. This does not mean an instance of each is required; only one instance is required across the entire mutual exclusivity group. More than one instance can participate for the same meta-entity if the maximum cardinality is greater than one on the connecting meta-relationship.

Figure 11 shows that each *ContactName* must be for either an *Organization* or an *Individual*, but that a single instance of *ContactName* would not be for both an *Organization* and an *Individual*. (The "must be" can be deduced because of the minimum cardinality of one at the destination end of the *For* meta-relationship.)

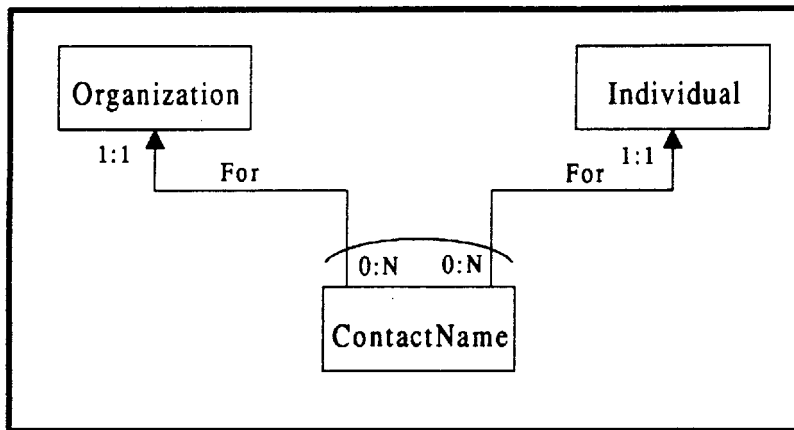


Figure 11  
Mutual Exclusivity of Meta-relationships

## 3.4 Subject Areas

### 3.4.1 Definition

The CDIF Integrated Meta-model is a single integrated model that is a description of the set of concepts and notations used to define an information model. This model is huge; therefore, it is partitioned into manageable sections by building views of the underlying meta-model. These partitions, or "subject areas" as they are known, group together collectable meta-objects (meta-entities, meta-relationships and meta-attributes) that address a particular aspect within the Systems Development Life Cycle (e.g., Data Flow Modeling), or that provide similar functionality (e.g., Data Definition). Meta-objects may be shared by many subject areas.

### 3.4.2 Working Meta-model

A CDIF Transfer may contain several, possibly overlapping subject areas. The actual subject areas, both standard and those created using extensibility, are identified in the transfer. An importer can then create what is called the Working Meta-model for a transfer by integrating the objects from standard subject areas and any additional objects created using extensibility.

When a transfer uses a standard subject area, the reference to that subject area is sufficient to define both explicitly and implicitly used meta-objects. This means that it is not necessary to give a subject area reference for the subject area where supertype meta-objects are defined purely to support those supertypes. If other meta-objects from that subject area are used in their own right, then the subject area reference must also be defined. In a CDIF Transfer where the meta-model is being defined entirely using extensibility, a subject area reference for the Foundation Subject Area as defined in *EIA/IS-111 CDIF - Integrated Meta-model - Foundation Subject Area* must be used since it contains the definitions for the eventual roots of all meta-entity and meta-relationship hierarchies used in CDIF.

### 3.4.3 Explicit and Implicit Use of *CollectableMetaObjects* in Subject Areas

Subject areas may contain some *CollectableMetaObjects* that are explicitly used and, may contain some that are implicitly used through inheritance.

In the standardized CDIF subject areas, which are published as standards, the explicitly used *CollectableMetaObjects* are defined in Section 5, Subject Area Detailed Definitions, of each subject area standard. The implicitly used *CollectableMetaObjects* are defined in Section 6, Referenced Meta-object Definitions, of each subject area standard; these are both supertype meta-entities whose meta-attributes and meta-relationships are inherited by meta-entities that are explicitly used, and supertype meta-relationships for those meta-relationships that are explicitly used.

When a new subject area is being created using extensibility, instances of the *CollectableMetaObject.IsUsedIn.SubjectArea* meta-relationship must be created for all explicitly used *CollectableMetaObjects*. Instances of meta-relationships should not be created for implicitly used *CollectableMetaObjects* since inheritance of meta-attributes and meta-relationships is assumed all the way up the inheritance hierarchies to *RootObject*.

### 3.4.4 Rules for Defining Subject Area Scope

Subject area views are built by instantiating the *CollectableMetaObject.IsUsedIn.SubjectArea* meta-meta-relationship for any *MetaEntity*, *MetaRelationship*, and *MetaAttribute* that is explicitly used. The following rules should be observed when defining the scope of a subject area:

- Where a meta-relationship is explicitly used in a subject area, both the source and destination meta-entities must also be used.
- Where both source and destination meta-entities for a meta-relationship happen to be explicitly used in a subject area, there is no obligation to use the meta-relationship.
- Where a meta-entity is explicitly used in a subject area and it participates in the integrated meta-model in a mandatory meta-relationship (at the mandatory end), the meta-relationship need not be used in the subject area.
- Where a meta-attribute is explicitly used in a subject area, the meta-entity or meta-relationship that the meta-attribute describes must also be explicitly used in the subject area.
- Where a meta-entity or meta-relationship is explicitly used in a subject area, any local meta-attributes that describe it that are required by the subject area must be explicitly used.
- Where a meta-entity or meta-relationship is explicitly used in a subject area, there is no obligation to use any of its meta-attributes, even mandatory ones.
- All inherited meta-attributes for both meta-entities and meta-relationships will be implicitly used in the subject area all the way up the inheritance hierarchy. The *CollectableMetaObject.IsUsedIn.SubjectArea* meta-meta-relationship should not be instantiated for inherited meta-attributes.
- For subtype meta-entities, any meta-relationships in which the supertype meta-entity participates will be implicitly used in a subject area, provided the other meta-entity, or a subtype of it, is explicitly used.

Note that where two subject areas are used in a transfer, a meta-relationship between two implicitly used supertype meta-entities (one in each subject area) will be inherited between the appropriate subtype meta-entities in the Working Meta-model for the transfer.

### 3.4.5 Subject Area Diagrams

Diagrams of subject areas will show explicitly used meta-entities. Meta-entities that are included in a diagram for clarification shall be shaded. Explicit meta-relationships will be shown in diagrams. Implicit meta-relationships (i.e. inherited meta-relationships) may be shown on the diagram even if the supertype meta-entity is not used explicitly. (See Section 3.3, CDIF Graphical Notation for further details.)

## 3.5 Data Types

### 3.5.1 Introduction

This section describes the data types that are available for the meta-attributes of meta-entities and meta-relationships, in all instances of the CDIF Meta-meta-model (i.e., the CDIF meta-model, the CDIF Meta-meta-model itself, and any additional meta-models built using the extensibility facilities of CDIF). The reader is reminded that the encoding of the information represented by these data types is defined in the specific syntax and encoding being used and is not defined here (e.g., a Boolean is defined here to hold either true or false, not *1* or *0*, not *Y* or *N*). The explicit limits for some of the data types is defined by the syntax being used.

Although some of the data types are defined as structured types, none of the components are optional. There are no defaults defined for any data types or for any of the components.

Table 3 shows the data types that are available for meta-meta-attributes in the Meta-meta-model and for meta-attributes in meta-models.

Table 3  
Data Types Supported by CDIF

| Data Type          | Description                                      |
|--------------------|--------------------------------------------------|
| <b>Bitmap</b>      | A bounded matrix of RGB <sup>4</sup> triples     |
| <b>Boolean</b>     | TRUE or FALSE                                    |
| <b>Date</b>        | A date (relative or absolute)                    |
| <b>Enumerated</b>  | A list of valid values                           |
| <b>Float</b>       | A floating point number                          |
| <b>Identifier</b>  | A name of an object                              |
| <b>Integer</b>     | A signed integer                                 |
| <b>IntegerList</b> | A list of integers                               |
| <b>Point</b>       | A location in three dimensions                   |
| <b>PointList</b>   | A list of points                                 |
| <b>String</b>      | A group of printable characters                  |
| <b>Text</b>        | A group of characters, not necessarily printable |
| <b>Time</b>        | A time (relative or absolute)                    |

Where a domain is not specified for an attribute, it shall be assumed to be the full range of values valid within the specified data type.

### 3.5.2 Bitmap

The bitmap data type is used to represent a two-dimensional array of graphical point specifications. Each graphical point is a RGB triple composed of an 8-bit unsigned red value, an 8-bit unsigned green value and an 8-bit unsigned blue value. The bitmap has a stated unsigned height and width (h and w).

---

<sup>4</sup> RGB stands for Red, Green, Blue

### 3.5.3 Boolean

The Boolean data type is used to represent a single item that can hold one of two values: TRUE or FALSE.

### 3.5.4 Date

The date data type is used to represent a date to the unit of granularity of a day. The class is specified to indicate whether the date is absolute or relative. An absolute date is given as *Absolute*. Relative dates are either positive or negative to their (implied) point of reference; this is indicated with the *RelativePositive* and *RelativeNegative* classes.

### 3.5.5 Enumerated

An enumerated data type specifies a list of possible values for a meta-attribute. There is no implicit ordering or sequence. Each value must conform to the rules of a valid identifier data type.

### 3.5.6 Float

A decimal number with up to 16 decimal digits of precision within the range  $10^{-1023}$  to  $10^{1023}$ .

### 3.5.7 Identifier

The identifier data type is used to represent a name. It shall be between one and 32 characters long. It shall consist of the characters A-Z, a-z, 0-9, underscore (\_) and hyphen (-). It must begin with an alphabetic character (i.e., A-Z or a-z) or a numeric character (i.e., 0-9).

When identifiers are used as the names of CDIF meta-objects, special capitalization rules apply. See Section 3.2.2, Naming Conventions, for more details.

### 3.5.8 Integer

An integer is a signed number capable of holding values in the range from -2147483648 to 2147483647 ( $-2^{31}$  to  $2^{31}-1$ ).

### 3.5.9 IntegerList

An integer list is an ordered list of values conforming to the integer data type.

### 3.5.10 Point

A point is a triple of (x, y, z) coordinates and represents an address in the Coordinate Frame. Each coordinate is represented using an integer. (See *EIA/IS-118 CDIF - Integrated Meta-model - Presentation Location and Connectivity Subject Area* for more information about the CDIF Transfer Coordinate Frame.)

### 3.5.11 PointList

A point list is an ordered list of points. Each point is a triple of (x, y, z) coordinates and represents an address in the Coordinate Frame. (See *EIA/IS-118 CDIF - Integrated Meta-model - Presentation Location and Connectivity Subject Area* for more information about the CDIF Transfer Coordinate Frame).

### 3.5.12 String

The string data type is represented as a group of printable characters of some maximum length, where  $1 \leq \text{maximum length} \leq 1024$ .

### 3.5.13 Text

The text data type is represented as a group of characters, not necessarily printable, of any pre-determined maximum length.

### 3.5.14 Time

The time data type is used to represent a time to the granularity of fractional seconds with a class to indicate whether the time is absolute or relative. Absolute times are given either as ***AbsoluteUTC***<sup>5</sup> or ***AbsoluteLocal***. Relative times are either positive or negative relative to their (implied) point of reference; this is indicated with the ***RelativePositive*** and ***RelativeNegative*** classes.

Times recorded in a CASE tool are exported as the values recorded by the tool. No conversions are specified by CDIF for time values where more than one time zone is involved.

---

<sup>5</sup> UTC stands for Universal Coordinated Time (English translation of French term) as defined in Recommendation 460 of the Consultative Committee on International Radio (CCIR) [1].

---

## 4. META-MODEL EXTENSIBILITY

### 4.1 Introduction

Extensibility is an important aspect of the CDIF Family of Standards. In general, extensibility means allowing two (or more) tools that support CDIF to exchange information that is not defined in the CDIF Integrated Meta-model. Specifically, this involves having the exporter warn the importer about extensions to the standard CDIF Integrated Meta-model before transferring information that uses those extensions, see Section 5, Exporter Responsibilities. CDIF makes no statement about whether or not the importer needs to be able to retain the contents of information transferred (especially where extensibility has been used). For a more detailed discussion of the responsibilities of an importer, see Section 6, Importer Responsibilities.

The following rules must be adhered to when using extensibility:

- Standardized subject areas may not be extended. This rule ensures that a consistent basis for importers to interpret is retained.
- A new, additional, subject area must be created to contain the extended meta-objects.
- The valid values of enumerated types are invariant by subject area. When an additional enumerated value is added it is added across the working meta-model, not for a specific subject area.
- Standardized meta-objects may also be used in exporter-defined subject areas.
- When reusing an existing meta-object in extensibility, the transfer must contain a reference to a subject area that contains the full definition of the meta-object.
- When any meta-object is added (i.e., a meta-entity, meta-relationship, meta-attribute, or subject area) a value must be defined for the meta-meta-attribute called *CDIFMetaIdentifier*. This identifier may have up to 32 characters but shall start with an alphabetic character and must be unique. The *CDIFMetaIdentifiers* of meta-objects in the CDIF Integrated Meta-model begin with a numeric character. No further structure to the identifier shall be given.