



ISO/IEC JTC1/SC7
Software Engineering
Secretariat: CANADA (SCC)

ISO/IEC JTC1/SC7 N1542
July 14, 1996

DOC TYPE: CD Ballot

TITLE: CD 15475-1: Information Technology - Software Engineering Data Definition and Interchange - Transfer Format - Part 1: General Rules for Syntaxes and Encoding

SOURCE: WG11

PROJECT: 07.28.2.1

STATUS: First CD ballot

REFERENCES: Resolution 404, SC7 N1311, N1539

ACTION ID: ACT

DUE DATE: 1996-10-28. It would be most appreciated that a copy of the comments, if any, be send to the project 07.28 editor, Mr Jean Bérubé (jberube@ibm.net) before 1996-10-18.

DISTRIBUTION: P, O and L

MEDIUM: E-Mail - uuencoded compressed Word ; Diskette - compressed Word

NO. OF PAGES: 45 [169 kB]

DISK NUMBER: 8



ISO/IEC JTC1/SC7 CD 15475-1	
Date 1996-07-14	Reference number ISO/JTC 1/SC 7 N 1542
Supersedes document SC 7 N 1311	

THIS DOCUMENT IS STILL UNDER STUDY AND SUBJECT TO CHANGE. IT SHOULD NOT BE USED FOR REFERENCE PURPOSES.

ISO/JTC 1/SC 7 Committee Title Software Engineering Secretariat: Standards Council of Canada (SCC)	Circulated to P- and O-members, and to technical committees and organizations in liaison for: X discussion at Ottawa on 1996-10-21 X comment by 1996-10-21 X voting by (P-members only) 1996-10-28 Please return all votes and comments in electronic form directly to the SC 7 Secretariat by the due date indicated.
---	--

ISO/IEC JTC1/SC7

Title: Information Technology - Software Engineering Data Definition and Interchange - Transfer Format
Part 1: General Rules for Syntaxes and Encoding

Project: 07.28.2.1

Introductory note: See Attachements

Medium: E-Mail - uuencoded compressed Word ; Diskette - compressed Word

No. of pages: 45



Vote on Committee Draft ISO/IEC 15475-1	
Date of circulation 1996-07-14	Reference number ISO/JTC 1/SC 7 N 1542
Closing date 1996-10-28	

ISO/JTC 1/SC 7 Committee Title Software Engineering Secretariat: Standards Council of Canada (SCC)	Circulated to P-members of the committee for voting Please return all votes and comments in electronic form directly to the SC 7 Secretariat by the due date indicated.
---	--

ISO/IEC JTC1/SC7

Title: Information Technology - Software Engineering Data Definition and Interchange - Transfer Format - Part 1: General Rules for Syntaxes and Encoding

Project: 07.28.2.1

Vote:

- APPROVAL OF THE DRAFT AS PRESENTED
- APPROVAL OF THE DRAFT WITH COMMENTS AS GIVEN ON THE ATTACHED
 - general:
 - technical:
 - editorial:
- DISAPPROVAL OF THE DRAFT FOR REASONS ON THE ATTACHED
 - Acceptance of these reasons and appropriate changes in the text will change our vote to approval
- ABSTENTION (FOR REASONS BELOW):

P-member voting:
National Body (Acronym)

Date:
YYCC-MM-DD

Submitted by:

Name

ISO IEC CD 15475-1
EIA/IS-108
SC7 WG11 N040R1

<p>Information Technology - Software Engineering Data Definition and Interchange - Transfer Format Part 1: General Rules for Syntaxes and Encoding</p>

Author:	EIA CDIF	Source:	SC7 WG11
Project:	7.28.2.1	Date:	1996.05.31
Action:	Circulation to JTC1 SC7 For CD Ballot, CD Review and Comments	Status:	Committee Draft
Reference:	SC7 N1220	Resolution #343 Ottawa Plenary (94.06.17)	
	SC7 N1311	CDIF General Rules Working Draft	
	SC7 N1415	Resolution #404 Brisbane Plenary (96.06.01)	

Introduction

ISO/IEC JTC1 SC7 WG11, in its Project 7.28, has the objective of enabling the interchange of information between a variety of heterogeneous software engineering tools.

Many of the input documents for Project 7.28 are the result of developmental work performed by the CASE Data Interchange Format (CDIF) technical committee of the Electronics Industries Association (EIA). This is one of those documents, and is distributed as received from the CDIF editor.

At its Brisbane meeting, in June 95, SC7 approved the distribution of the committee drafts.

This document is intended to provide one part within the multipart standard resulting from Project 7.28. Documents will be reformatted to ISO Directives as part of the Committee Draft ballot and comment resolution.

Comments

Comments are requested within the timetable provided by the SC7 cover sheet (generally 100 days) to facilitate preparation for the next WG11 meeting. Comments should be sent by normal formal channels.

To facilitate planning and preparatory resolution proposals, WG11 would appreciate comments to be submitted to the WWW (www.cdif.org/comments.html) site identified for that purpose).

Comments are to be provided in electronic format, generally following the format used for the WWW page. If comments are also submitted on paper, they should use a similar format.

Please identify clearly the status of comments submitted, such as:

National Body Comments,
National Body Approved Expert Comments.

Jean Bérubé
Project Editor, 7.28

IDEgenic inc
87 Burton,
Greenfield Park, Québec,
Canada J4V 2Y6

Tél: 1 514 465 0245
Fax: 1 514 465 2398
jberube@ibm.net

711PRA-11
Explanatory Notes for SC7 Reviewers
of the
WG11-CDIF Documents
(Project 07.28)

- The “Overview” document (WG11 N047R1, SC7 N1540) should be read first since it describes all the other project 7.28 CD documents to be reviewed at this time (as well as several documents planned for the future). It also describes the overall architecture that applies to these documents.
- The “Framework” document (WG11 N046R1, SC7 N1541) should be read second, as it describes the entity-relationship-attribute modeling notation that is used in many of the documents.
- The “Data Modeling” document (WG11 N036, SC7 N1548) and the “Data Flow Modeling” (WG11 N035, SC7 N1549) document provide the substantive coverage of software engineering techniques within the first set of documents. These documents permit the structured representation of software designs prepared by the use of any of a number of well established software engineering techniques that produce what are commonly called data flow diagrams and/or entity-relationship- attribute diagrams. These two documents will be of the greatest interest to software engineering practitioners.
- The “Foundation”, “Common”, and “Data Definition” documents (WG11 N048R1, N049, N050 and respectively SC7 N1545, N1546, 1547) are “modeling infrastructure” documents that support the contents of the software engineering techniques document, Data Modeling and Data Flow Modeling. Elements in these three infrastructure documents are incorporated by reference into the technique documents as necessary.
- The Transfer Format documents, “General Rules”, “Syntax”, and “Encoding” (WG11 N040R1, N041R1, N042R1 and respectively SC7 N1542, N1543, N1544) describe how character sequences are formed in order to communicate software design representations. These may be read separately from the technique and infrastructure documents described above.
- Of special importance for this review is the “Foundation” document (WG11 N048R1, SC7 N1545). This document illustrates the format intended to be applied to all the documents in this set for subsequent balloting. WG11 specifically solicits reviewer comments related to the format of this document so that the reformatting effort for the other documents in this set need be done only once.

WG11 will greatly appreciate the national body review comments on these documents, and thanks in advance the reviewers for their efforts. Any early submissions of comments will be very helpful.

WG11 Foreword

In order to fully ensure that all comments received are processed efficiently and consistently, WG11 requests of all reviewers to please utilize the comments template that is provided with each document. Submitting comments in electronic form rather than on paper will be especially helpful.

If assistance with the usage of the comment forms is required, or if there are any other questions, please contact either Jean Berube, the Project 7.28 coordinating editor, or myself.

Sincerely,

Peter Eirich
WG11 Convenor

CD 15475-1

**Information Technology - Software Engineering Data
Definition and Interchange - Transfer Format**

Part 1: General Rules for Syntaxes and Encoding

CDIF TITLE:

***CDIF - Transfer Format
General Rules for Syntaxes and Encodings***

2 EIA/IS-108 JANUARY, 1994

3 Revised 4/18/96 to incorporate ISO/IEC JTC1 SC7/WG11 comments submitted as
4 input to the 1995 Brisbane meeting.

5 ELECTRONIC INDUSTRIES ASSOCIATION

6 engineering department

1

NOTICE

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

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

14

EIA INTERIM STANDARDS

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

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

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

26

COPYRIGHT 1994

27 Published by
28 ELECTRONIC INDUSTRIES ASSOCIATION
29 Engineering Department
30 2500 Wilson Blvd. Suite 203
31 Arlington, VA 22201
32

33 PRICE: Please refer to the current
34 Catalog of EIA & JEDEC STANDARDS & ENGINEERING PUBLICATIONS
35 or contact the EIA Electronic Information Group
36 (Phone: (703) 527-7001, Fax: (703) 527-0685)
37

38 All rights reserved
39 Printed in U.S.A.
40

41 ISBN 0-7908-0014-4

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.

CDIF - CASE Data Interchange Format - Overview (ISO CD 15474-1: Information Technology - Software Engineering Data Definition and Interchange - Overview and Framework: Part 1 : Overview) and *CDIF - Framework for Modeling and Extensibility (ISO CD 15474-2: Information Technology - Software Engineering Data Definition and Interchange - Overview and Framework: Part 2 : Framework for Modeling and Extensibility)* should be read first when initially exploring CDIF. The first explains the overall CDIF Architecture and how the family of standards fits together. The second explains the scope, and modeling approach in CDIF. The CDIF Meta-meta-model and extensibility mechanism are also defined in that document.

This standard defines how CDIF supports multiple exchange Syntaxes and Encodings, and describes how CDIF meta-models are concretely represented during a transfer. *EIA/IS-109 CDIF - Transfer Format - Syntax SYNTAX.1* and *EIA/IS-110 CDIF - Transfer Format - Encoding ENCODING.1* define one specific CDIF Syntax and Encoding.

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 has been copyrighted by the CASE Data Interchange Format (CDIF) Division of the Electronic Industries Association (EIA). Permission to copy this material in paper and electronic form has been granted to the international software engineering standards development organization (ISO/IEC JTC1/SC7) for the sole purpose of international standards development work. Organizations receiving copies of this document may not duplicate and distribute additional copies internally for any reason other than to facilitate their organization's review and comment back to ISO/IEC JTC1/SC7.

This page is intentionally blank

TABLE OF CONTENTS

1		
2	1. SCOPE.....	1
3	2. INTRODUCTION.....	2
4	2.1 Audience	2
5	2.2 Structure of this Document	2
6	2.3 Conventions	3
7	2.3.1 Global Conventions	3
8	2.3.2 BNF Conventions.....	4
9	3. CONCEPTS AND DEFINITIONS.....	6
10	3.1 Separation of Syntax and Encoding.....	6
11	3.2 Independence of Architecture Components	6
12	3.3 Data Types	6
13	3.4 Character Sets.....	6
14	4. GENERAL STRUCTURE OF THE CDIF TRANSFER	7
15	4.1 Introduction.....	7
16	4.2 The Transfer Envelope	8
17	4.2.1 Introduction	8
18	4.2.2 CDIF Signature	8
19	4.2.3 Syntax Identifier Structure.....	9
20	4.2.4 Encoding Identifier Structure.....	9
21	4.2.5 Character Set Identifier Structure	10
22	4.2.6 Transfer Envelope Example	11
23	4.3 The Transfer Contents	11
24	4.3.1 Introduction	11
25	4.3.2 The Header Section.....	11
26	4.3.3 The Meta-model Section	12
27	4.3.3.1 Introduction.....	12
28	4.3.3.2 CDIF Subject Area References.....	13
29	4.3.3.3 Meta-model Extensions.....	13
30	4.3.4 The Model Section	13
31	5. TRANSFER ENVELOPE FORMAL GRAMMAR	15
32	APPENDIX A QUESTIONS AND ANSWERS	17
33	GLOSSARY	21
34	REFERENCES.....	29

1 **INDEX.....31**

2 **RELATED EIA STANDARDS.....INSIDE BACK COVER**

3

FIGURES

1	
2	Figure 1 Position in the CDIF Family of Standards 1
3	Figure 2 The General Structure of a CDIF Transfer 7
4	Figure 3 The Transfer Envelope 8
5	Figure 4 The Transfer Contents 11
6	Figure 5 The Transfer Contents Header Section 12
7	Figure 6 The Transfer Contents Meta-model Section 12
8	Figure 7 The Transfer Contents Model Section..... 14

This page is intentionally blank

1. SCOPE

The CDIF Family of Standards is primarily designed to be used as a description of a mechanism for transferring information between CASE tools. It facilitates a successful transfer when the authors of the importing and exporting tools have nothing in common except an agreement to conform to CDIF. The language that is defined for the Transfer Format also has applicability as a general language for Import/Export from repositories. The CDIF Integrated Meta-model defined for CASE also has applicability as the basis of standard definitions for use in repositories.

The standards which form the complete family of CDIF Standards are documented in *EIA/IS-106 CDIF - CASE Data Interchange Format - Overview*. These standards cover the overall framework, the transfer format and the CDIF Integrated Meta-model.

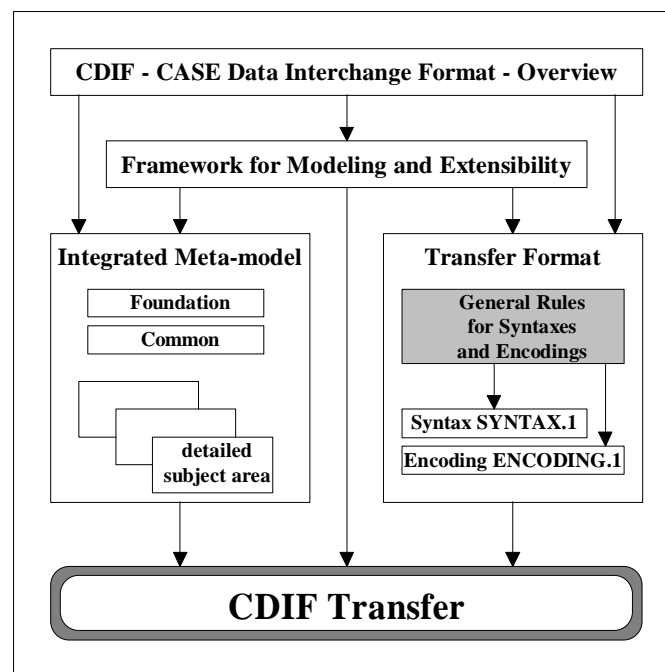


Figure 1
Position in the CDIF Family of Standards

The diagram in Figure 1 depicts the various standards that comprise the CDIF Family of Standards. The shaded box depicts this Standard and its position in the CDIF Family of Standards.

This document describes the way that CDIF meta-models are concretely represented during a transfer and the way that CDIF supports multiple exchange Syntaxes and Encodings. No specific exchange Syntaxes or Encodings are described in this document. *EIA/IS-109 CDIF - Transfer Format - Syntax SYNTAX.1* and *EIA/IS-110 CDIF - Transfer Format - Encoding ENCODING.1* describe one specific CDIF Syntax and one specific CDIF Encoding.

1 **2. INTRODUCTION**

2 **2.1 Audience**

3 This document is intended to be used by anyone wishing to understand and/or use CDIF. This
4 document provides a definition of the general rules for syntaxes and encodings for CDIF transfers.
5 It is suitable for:

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

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

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

- 14 • Entity-Relationship-Attribute modeling
- 15 • CASE tools
- 16 • Information repositories
- 17 • Data dictionaries
- 18 • Multiple meta-layer modeling
- 19 • Formal grammars
- 20 • Transfer formats.

21 **2.2 Structure of this Document**

22 This document is organized into the following sections:

23 **Section 1 'Scope'**

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

1 **Section 2 'Introduction'**

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

3 **Section 3 'Concepts and Definitions'**

4 This describes the concepts of separation of Syntax and Encoding and independence of
5 architecture components, and defines the general rules for data types and character sets in
6 a transfer.

7 **Section 4 'General Structure of the CDIF Transfer'**

8 This describes the purpose of a CDIF transfer, the syntax of the transfer envelope and the
9 general syntax of a CDIF transfer.

10 **Section 5 'Transfer Envelope Formal Grammar'**

11 This defines the precise format of the transfer envelope.

12 **Appendix A 'Questions and Answers'**

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

16 **Glossary**

17 The Glossary contains definitions of terms used within this Standard.

18 **References**

19 This contains details of other standards referenced within this Standard.

20 **Index**

21 A comprehensive index is provided.

22 **2.3 Conventions**

23 **2.3.1 Global Conventions**

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

- 26 • Double quotes are used to introduce new terms (e.g., "model layer")
- 27 • Bold type is used for emphasis (e.g., this is **important**)

- 1 • All meta-objects and meta-meta-objects in CDIF (in meta-models and meta-meta-
2 models) are named by concatenating all the words that name the meta-object or
3 meta-meta-object; the first letter of each word is upper-case, the rest are lower-
4 case (e.g., *MetaAttribute*, *AttributeDerivation*, *IsDrawnUsing*, *IsOptional*)

5 **2.3.2 BNF Conventions**

6 The grammar in this document is defined using an extended version of "Backus Naur Form"
7 (BNF).

8 In BNF, each syntactic element of the language is defined by means of a Production Rule. A
9 Production Rule defines a syntactic element in terms of an expression, ordered left to right, that
10 can be used to form an instance of the element. The expression can consist of characters,
11 character strings, and other syntactic elements.

12 The version of BNF used in this document has the following conventions:

13 <i>Symbol</i>	<i>Meaning</i>
14 < >	15 Angle braces delimit character strings that are the names of syntactic elements, i.e., Non-terminal Symbols.
16 ::=	17 This is the definition operator. It is used in a Production Rule to 18 separate the element defined by the rule from its definition. The 19 element being defined is to the left of the operator and the expression that defines the element is to the right of the operator.
20 []	21 Square braces enclose optional elements in an expression. The portion 22 of the expression within the braces may be explicitly specified or may be omitted.
23 { }	24 Curly braces group elements in an expression, so that the group can be treated as a single syntactic element.
25	26 This is the alternative operator. The vertical bar indicates that the 27 portion of the expression to the right of the bar is an alternative to the 28 portion to the left of the bar. If the vertical bar appears at a position 29 where it is not enclosed in curly or square braces, it specifies a 30 complete alternative for the element defined by the Production Rule. If 31 the vertical bar appears in a portion of the expression enclosed in curly 32 or square braces, it specifies alternatives for the contents of the innermost pair of such braces.

1	...	The ellipsis indicates that the element to which it applies in a formula
2		may be repeated any number of times (but note that optionality is
3		covered by the square braces convention). If the ellipsis appears
4		immediately to the right of a closing curly or square brace, "}" or "]",
5		then it applies to the portion of the expression enclosed between that
6		closing brace and the corresponding opening brace, "{" or "[". If an
7		ellipsis appears immediately to the right of any other element, then it
8		applies only to that element.
9	!!	This is the comment operator. This operator introduces normal English
10		text. It is used as a comment in the BNF and when the definition of a
11		syntactic element is not expressed further in BNF.
12	< TokenName >	Anything emboldened and surrounded by angle braces is a token , a
13		BNF Non-terminal Symbol that is not further defined in the Syntax.
14		The detailed representation of a token is specified in the encoding.
15	Terminal	Terminal would be a Terminal Symbol. Anything emboldened, and not
16		surrounded by angle braces, is a literal value (i.e., it represents the
17		character string itself).
18	x .. y	This is the range operator. It refers to any symbol in the range x to y.
19		For example, 0..255 refers to all integers 0 to 255 inclusive.

3. CONCEPTS AND DEFINITIONS

3.1 Separation of Syntax and Encoding

Because CDIF transfers are designed to be handled by different transport mechanisms (e.g., a file, an inter-process link, etc.), it is necessary to separate the definitions of the Syntax of a transfer format from the definition of that Syntax's Encoding. A **Syntax** is a definition of the grammar and structure of an exchange format. An **Encoding** is a description of how a Syntax is physically represented. This allows the use of a common Syntax, while permitting selection of an optimal encoding for the transport mechanism being used. The dividing line between the Syntax and Encoding is somewhat arbitrary, and the design of each may impinge on the other.

3.2 Independence of Architecture Components

The CDIF Family of Standards supports the concept of multiple Transfer Formats, each consisting of a Syntax and corresponding Encoding. For an individual transfer Syntax, multiple Encodings can be supported. The semantics (i.e., the meta-model) are **completely separated** from the definition of the Transfer Format. The meta-model has been defined only by using the concepts defined in the meta-meta-model. **Any transfer format that can, in the general case, represent all of the modeling concepts defined by the meta-meta-model, can exchange all the information defined in the meta-model**, and thus transfer any model that conforms to the meta-model.

The CDIF Family of Standards provides at least one transfer Syntax and one Encoding for that Syntax. This document describes the general rules for any Syntax or Encoding which may be defined for a CDIF transfer.

3.3 Data Types

Meta-attribute values must be of a given data type. The meta-model description of the relevant meta-entity or meta-relationship will define the data type for each of its meta-attributes.

There is a defined set of meta-attribute data types in CDIF. These types are defined in *EIA/IS-107 CDIF - Framework for Modeling and Extensibility*. Any Syntax and Encoding that supports CDIF transfers shall provide the definitions for how each of these data types occur in a transfer.

3.4 Character Sets

Each Syntax shall support the use of the characters in the character set (CODESET) specified in the Transfer Envelope for values of string and text meta-attributes within a transfer. If no character set is specified in the Transfer Envelope, then the ISO 8859-1 (Latin Alphabet no. 1) character set [1] shall be assumed.

4. GENERAL STRUCTURE OF THE CDIF TRANSFER

4.1 Introduction

The purpose of a CDIF transfer is to support the transfer of:

- References to the CDIF Subject Areas to be used during the transfer
- Extensions to the CDIF Meta-model to be used during the transfer
- Instances of meta-entities, meta-relationships and meta-attributes, as defined in the meta-model section of the transfer.

Every transfer is expressed using one Syntax and one Encoding. Figure 2 shows the general structure of all CDIF transfers. A CDIF transfer consists of the transfer envelope and the transfer contents as described below.

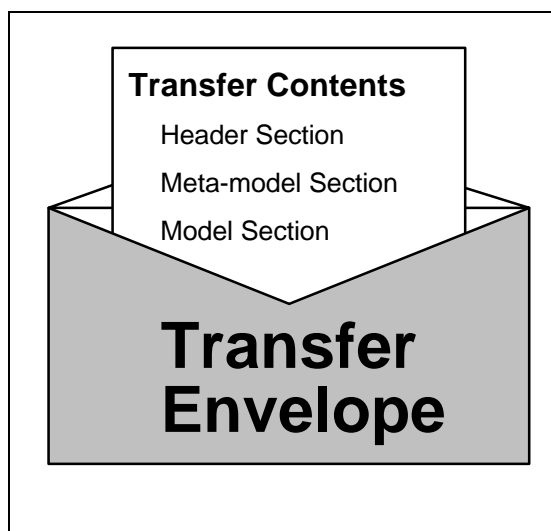


Figure 2
The General Structure of a CDIF Transfer

The general syntax of a CDIF Transfer is as follows:

```
<CDIFTransfer> ::=  
    <TransferEnvelope>  
    <TransferContents>
```

4.2 The Transfer Envelope

4.2.1 Introduction

The Transfer Envelope consists of the CDIF Signature, the Syntax identifier, the Encoding identifier and the optional Character Set (CODESET) identifier.

The syntax of the transfer envelope is as follows:

```

6     <TransferEnvelope> ::=
7         <CDIFSignature> <TransferEnvelopeComma> <SyntaxIdentifier>
8         <TransferEnvelopeComma> <EncodingIdentifier>
9         [ <TransferEnvelopeComma> <CharacterSetIdentifier> ]

```

The Transfer Envelope is the only section of the transfer that has exactly the same format for all combinations of CDIF Transfer Syntaxes, Encodings and Character Sets. It is used to identify the Syntax, Encoding and Character Set that are used in the remainder of the transfer. The format of the Transfer Envelope is restricted to that defined -- i.e., no whitespace characters are allowed anywhere in the transfer envelope section. All characters in a transfer envelope are as defined in ISO 646, because the transfer envelope must be understandable for any CDIF parser. The format of this section is shown in Figure 3.

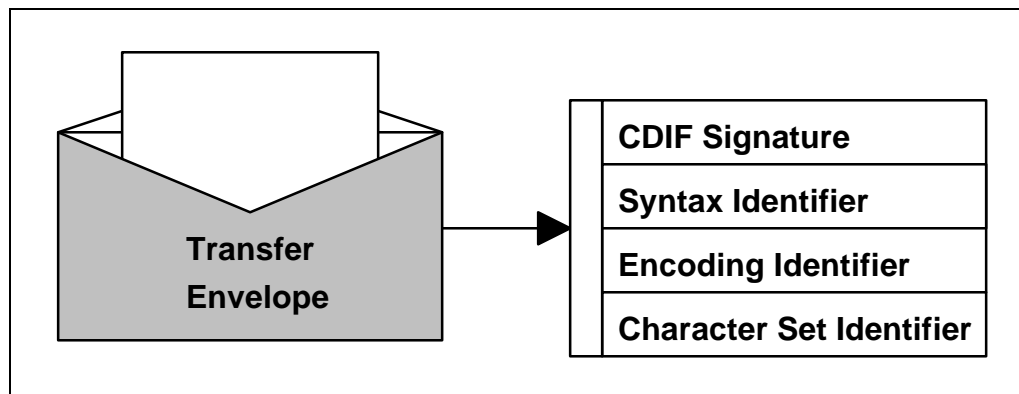


Figure 3
The Transfer Envelope

4.2.2 CDIF Signature

The CDIF Signature is a marker that uniquely identifies the transfer as a CDIF transfer.

The syntax of the CDIF Signature is as follows:

```

24     <CDIFSignature> ::=
25         CDIF

```

That is, the CDIF Signature must appear exactly as follows:

```

28     CDIF

```

4.2.3 Syntax Identifier Structure

The syntax of the syntax identifier is as follows:

```

<SyntaxIdentifier> ::=
    SYNTAX <TransferEnvelopeSpace> <SyntaxID>
    <TransferEnvelopeSpace> <SyntaxVersion>
<SyntaxID> ::=
    <TransferEnvelopeString>
<SyntaxVersion> ::=
    <TransferEnvelopeString>
<TransferEnvelopeString> ::=
    <TransferEnvelopeDoubleQuote>
    [ <TransferEnvelopePrintableCharacter> ] ...
    <TransferEnvelopeDoubleQuote>
<TransferEnvelopeDoubleQuote> ::=
    "
<TransferEnvelopePrintableCharacter> ::=
    a | b | c | d | e | f | g | h | i | j |
    k | l | m | n | o | p | q | r | s | t |
    u | v | w | x | y | z |
    A | B | C | D | E | F | G | H | I | J |
    K | L | M | N | O | P | Q | R | S | T |
    U | V | W | X | Y | Z |
    0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
    \ | " | ] | # |
    <Space> | <NBSP> | <Soft Hyphen> |
    ! | $ | % | & | ' | ( | ) | * | + | , |
    - | . | / | : | ; | < | = | > | ? | @ |
    [ | ^ | _ | ` | { | } | ~
!! These characters are as defined in ISO 646.

```

An example of a syntax identifier would be:

```
SYNTAX "SYNTAX.1" "02.00.00"
```

4.2.4 Encoding Identifier Structure

The syntax of the encoding identifier is as follows:

```

<EncodingIdentifier> ::=
    ENCODING <TransferEnvelopeSpace> <EncodingID>
    <TransferEnvelopeSpace> <EncodingVersion>
<EncodingID> ::=
    <TransferEnvelopeString>
<EncodingVersion> ::=
    <TransferEnvelopeString>
<TransferEnvelopeString> ::=
    <TransferEnvelopeDoubleQuote>
    [ <TransferEnvelopePrintableCharacter> ] ...
    <TransferEnvelopeDoubleQuote>
<TransferEnvelopeDoubleQuote> ::=
    "
<TransferEnvelopePrintableCharacter> ::=
    a | b | c | d | e | f | g | h | i | j |
    k | l | m | n | o | p | q | r | s | t |

```

```

1      u | v | w | x | y | z |
2      A | B | C | D | E | F | G | H | I | J |
3      K | L | M | N | O | P | Q | R | S | T |
4      U | V | W | X | Y | Z |
5      0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
6      \ | " | ] | | # |
7      <Space> | <NBSP> | <Soft Hyphen> |
8      ! | $ | % | & | ' | ( | ) | * | + | , |
9      - | . | / | : | ; | < | = | > | ? | @ |
10     [ | ^ | _ | ` | { | } | ~ |
11     !! These characters are as defined in ISO 646.
12

```

13 An example of an encoding identifier would be:

```
14     ENCODING "ENCODING.1" "02.00.00"
```

16 4.2.5 Character Set Identifier Structure

17 The syntax of the character set identifier is as follows:

```

18     <CharacterSetIdentifier> ::=
19         CODESET <TransferEnvelopeSpace> <CharacterSetID>
20     <CharacterSetID> ::=
21         <TransferEnvelopeString>
22     <TransferEnvelopeString> ::=
23         <TransferEnvelopeDoubleQuote>
24         [ <TransferEnvelopePrintableCharacter> ] ...
25         <TransferEnvelopeDoubleQuote>
26     <TransferEnvelopeDoubleQuote> ::=
27         "
28     <TransferEnvelopePrintableCharacter> ::=
29         a | b | c | d | e | f | g | h | i | j |
30         k | l | m | n | o | p | q | r | s | t |
31         u | v | w | x | y | z |
32         A | B | C | D | E | F | G | H | I | J |
33         K | L | M | N | O | P | Q | R | S | T |
34         U | V | W | X | Y | Z |
35         0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
36         \ | " | ] | | # |
37         <Space> | <NBSP> | <Soft Hyphen> |
38         ! | $ | % | & | ' | ( | ) | * | + | , |
39         - | . | / | : | ; | < | = | > | ? | @ |
40         [ | ^ | _ | ` | { | } | ~ |
41     !! These characters are as defined in ISO 646.
42

```

43 Some examples of a character set identifier are:

```

44     CODESET "EUC"
45     CODESET "ISO-2022-JP"
46     CODESET "ShiftJIS"
47

```

48 The Transfer Contents following the Transfer Envelope are encoded according to the identified
49 Syntax , Encoding and Character Set. Each transfer Syntax may have multiple Encodings. The
50 Encoding identifier and Character Set identifier in the envelope uniquely identifies the Encoding.
51 A Transfer Syntax and Encoding pair may have multiple Character Sets.

4.2.6 Transfer Envelope Example

Using the syntax rules defined in the preceding sections, here is an example of a complete envelope description:

```
CDIF,SYNTAX "SYNTAX.1" "02.00.00",ENCODING "ENCODING.1" "02.00.00",
CODESET "ShiftJIS"
```

4.3 The Transfer Contents

4.3.1 Introduction

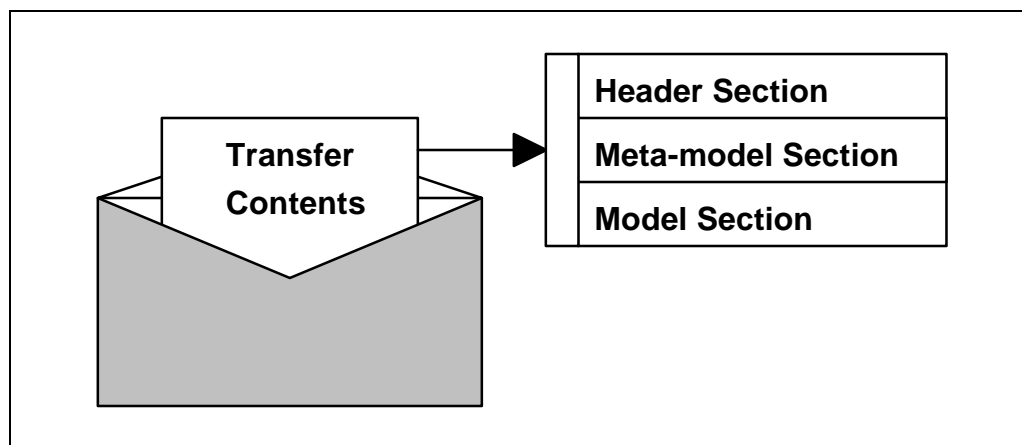


Figure 4
The Transfer Contents

The transfer contents consist of the header section, meta-model section and model section. Figure 4 shows the transfer contents.

The syntax of the Transfer Contents shall be the same for all Syntax definitions:

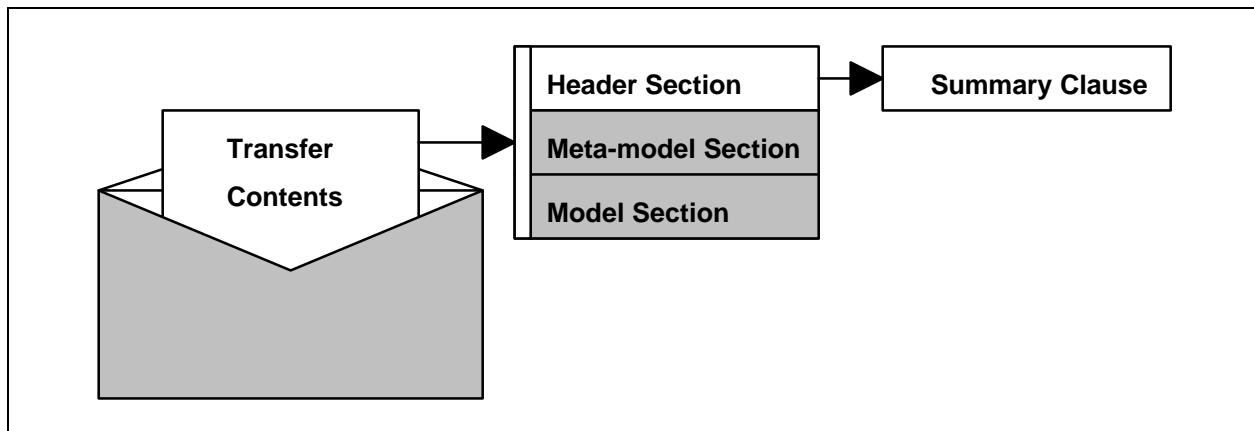
```
<TransferContents> ::=
    <HeaderSectionClause>
    <MetaModelSectionClause>
    [ <ModelSectionClause> ]
```

4.3.2 The Header Section

The Header Section contains information about the transfer. All Syntax definitions shall provide a mechanism for expressing at least the following information, although it is not mandatory to transfer it:

- Identification of the exporting tool, both name and version
- Configuration management information including, as a minimum, publisher and date & time of export.

- 1 Each Syntax definition shall provide an ability for the exporting tool to express additional
 2 information of this type. Figure 5 shows the Transfer Contents Header Section.



3

Figure 5

The Transfer Contents Header Section

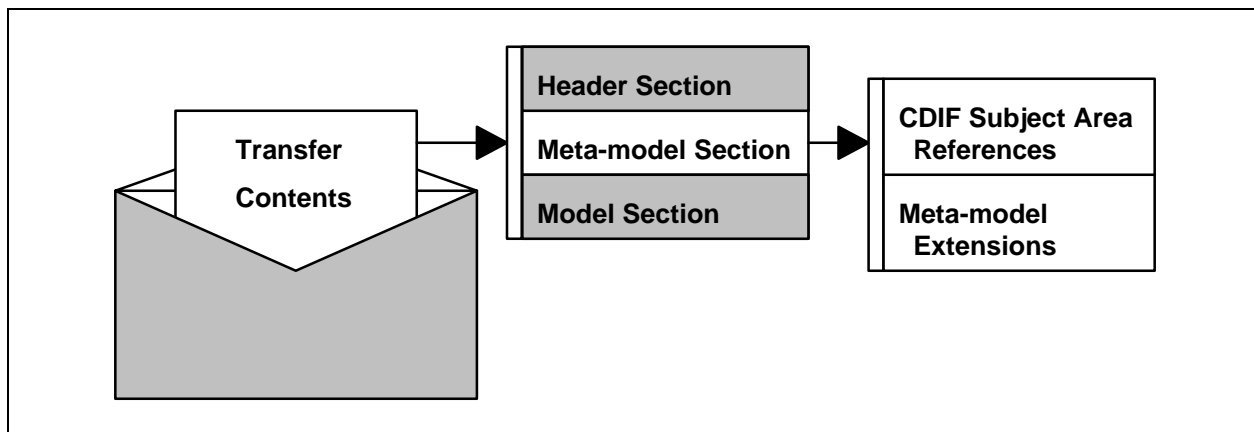
4

5

6 4.3.3 The Meta-model Section

7 4.3.3.1 Introduction

- 8 The Meta-model Section defines the meta-model used in the transfer. It contains references to the
 9 standardized CDIF Subject Areas used in the transfer. It may also contain extensions to the CDIF
 10 Integrated Meta-model. The general structure of this section is shown in Figure 6.



11

Figure 6

The Transfer Contents Meta-model Section

12

13

1 4.3.3.2 CDIF Subject Area References

2 This section identifies the standardized CDIF Subject Areas that should be used by the importer
3 when interpreting model data. The appropriate version of each of these Subject Areas is also
4 identified (as defined in the relevant Subject Area standard).

5 4.3.3.3 Meta-model Extensions

6 This section contains meta-model extension information that must be communicated to the
7 importer before it encounters model data. This section must be empty if importers and exporters
8 use only the standardized CDIF Subject Areas.

9 When an exporter needs to extend the standardized CDIF meta-model or to provide its own meta-
10 model definition(s), it places these extensions in this section. All Syntaxes shall provide
11 mechanisms for extension.

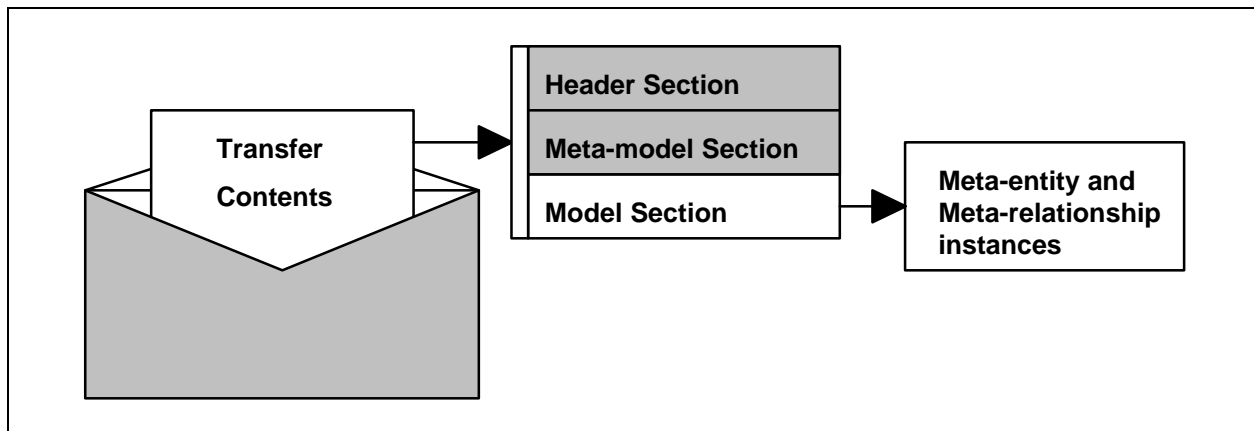
12 The forms of the extensibility that must be supported and the exact information that must be
13 communicated for each meta-object definition is determined by the CDIF meta-meta-model. For
14 a complete description of this information, see *EIA/IS-107 CDIF - Framework for Modeling and*
15 *Extensibility*.

16 Forward references are not allowed within the meta-model extensions part of a transfer. This
17 means that all meta-model objects referenced must be defined in the identified Subject Areas or in
18 previous extensions within the transfer.

19 All meta-entities, meta-attributes and meta-relationships that occur in the Meta-model Section
20 shall be uniquely identified within a single transfer. This identification is accomplished by
21 providing a unique identifier, known as a CDIF Meta-identifier, for every meta-object in a
22 transfer. These identifiers shall not be used to convey any semantic information between an
23 exporter and an importer and need not be retained by a CDIF importer. Meta-identifiers
24 beginning with a numeric character are reserved for use in the CDIF Integrated Meta-model, to
25 avoid clashes.

26 4.3.4 The Model Section

27 The Model Section contains instances of meta-entities and meta-relationships, along with their
28 associated meta-attributes, as defined in the Meta-model Section. Figure 7 shows the structure of
29 the Transfer Contents Model Section.



1

2

3

Figure 7

The Transfer Contents Model Section

4

All meta-entity and meta-relationship instances that occur in the Model Section shall be uniquely identified within a single transfer. This identification is accomplished by providing a unique identifier, known as a CDIF Identifier, for every meta-entity and meta-relationship instance in a transfer. These identifiers shall not be used to convey any semantic information between an exporter and an importer and need not be retained by a CDIF importer.

6

9

There shall be no forward references, implicit or explicit, within the Model Section.

10

The Model Section of a CDIF transfer may be omitted if only meta-model information is being transferred.

11

12

Meta-attribute values are present in a transfer only if they are non-null.

5. TRANSFER ENVELOPE FORMAL GRAMMAR

The exact representation of information in the transfer envelope is described in the following grammar:

```

4     <CDIFSignature> ::=
5         CDIF
6     <CharacterSetID> ::=
7         <TransferEnvelopeString>
8     <CharacterSetIdentifier> ::=
9         CODESET <TransferEnvelopeSpace> <CharacterSetID>
10    <EncodingID> ::=
11        <TransferEnvelopeString>
12    <EncodingIdentifier> ::=
13        ENCODING <TransferEnvelopeSpace> <EncodingID>
14        <TransferEnvelopeSpace> <EncodingVersion>
15    <EncodingVersion> ::=
16        <TransferEnvelopeString>
17    <SyntaxID> ::=
18        <TransferEnvelopeString>
19    <SyntaxIdentifier> ::=
20        SYNTAX <TransferEnvelopeSpace> <SyntaxID>
21        <TransferEnvelopeSpace> <SyntaxVersion>
22    <SyntaxVersion> ::=
23        <TransferEnvelopeString>
24    <TransferEnvelope> ::=
25        <CDIFSignature> <TransferEnvelopeComma> <SyntaxIdentifier>
26        <TransferEnvelopeComma> <EncodingIdentifier
27        [ <TransferEnvelopeComma> <CharacterSetIdentifier> ]
28    <TransferEnvelopeComma> ::=
29        ,
30    <TransferEnvelopeDoubleQuote> ::=
31        "
32    <TransferEnvelopePrintableCharacter> ::=
33        a | b | c | d | e | f | g | h | i | j |
34        k | l | m | n | o | p | q | r | s | t |
35        u | v | w | x | y | z |
36        A | B | C | D | E | F | G | H | I | J |
37        K | L | M | N | O | P | Q | R | S | T |
38        U | V | W | X | Y | Z |
39        0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
40        \ | " | ] | | # |
41        <Space> | <NBSP> | <Soft Hyphen> |
42        ! | $ | % | & | ' | ( | ) | * | + | , |
43        - | . | / | : | ; | < | = | > | ? | @ |
44        [ | ^ | _ | ` | { | } | ~
45        !! These characters are as defined in ISO 646.
46    <TransferEnvelopeSpace> ::=
47        !! A space character as defined in ISO 646.
48    <TransferEnvelopeString> ::=
49        <TransferEnvelopeDoubleQuote>
50        [ <TransferEnvelopePrintableCharacter> ] ...
51    <TransferEnvelopeDoubleQuote>
52
53

```

The envelope has a rigid format, with no arbitrary whitespace allowed.

1 **Question 2:**

2 **Can I use CDIF to transfer only meta-model extensions?**

3 **Answer:**

4 Yes. In such a case, the model section of the transfer would be omitted.

5 **Question 3:**

6 **How does an importer know if the CDIF Meta-model is provided in a transfer?**

7 **Answer:**

8 The CDIF Integrated Meta-model is never explicitly transferred, but implicitly through CDIF
9 Subject Area References. The exporter specifies the CDIF Subject Areas that are used in the
10 transfer. The importer must understand those Subject Areas of the CDIF Integrated Meta-model.

11 **Question 4:**

12 **What has to be transferred if the exporter wishes to export models that are currently**
13 **outside the scope of CDIF?**

14 **Answer:**

15 The exporter needs to transfer the meta-model definition in the meta-model extensions section of
16 the transfer, and then transfer the model instances of those definitions in the model section of the
17 transfer.

18 The meta-model section of the transfer should include references to any CDIF Subject Areas used
19 in the transfer. Where meta-model extensions are being transferred, a reference to the CDIF
20 Foundation Subject Area is normally necessary.

21 **Question 5:**

22 **Why do we have multiple Encodings for a single Syntax?**

23 **Answer:**

24 Different encodings address different requirements. For example:

- 25 • A **Clear Text** encoding is human readable but very bulky.
- 26 • A **Character** encoding (of which Clear Text is a special case) is also human
27 readable, more compact than Clear Text, and can pass freely down
28 communications lines. However, it is more terse than Clear Text and requires
29 detailed knowledge of the encoding to decipher.

- 1 • A **Binary** encoding is more compact than Character, but is not human readable,
2 and may not pass freely down communications lines.
- 3 • An **Encrypted** encoding may be based on any of the three previous classifications,
4 but is defined with an encryption algorithm for security purposes.

5 **Question 6:**

6 **What is the distinction between the syntax and the encoding?**

7 **Answer:**

8 The syntax is the sequence of the objects, while the encoding is the actual values of the objects. A
9 common syntax is "<subject> <verb> <object>". Several examples, using different encodings, of
10 a statement, with this syntax, that expresses affection for the person being addressed are:

- 11 • I love you. (English)
- 12 • Ich liebe dich. (German)

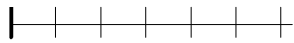
13 Japanese has a different syntax as well as encoding.

1

This page is intentionally blank

1

GLOSSARY

ATTRIBUTE	A single-valued characteristic of an entity or relationship.
CDIF	See <i>CDIF Family of Standards</i> .
CDIF CLEAR TEXT ENCODING	A human readable encoding for the CDIF syntax, SYNTAX.1.
CDIF COORDINATE FRAME	<p>The system of graphics addressing used by CDIF. Graphics space is defined as a coordinate system that is a three dimensional rectilinear grid. The intersections of grid lines define addresses (CDIF Points). The intersection of any two grid lines defines a coordinate plane. Each grid cell in a coordinate plane is a CDIF Pixel. All grid coordinates are integers and all grid lines are infinitely thin.</p> <p>The figure below describes the CDIF Coordinate Frame.</p> 
CDIF FAMILY OF STANDARDS	The CDIF Family of Standards is composed of a set of standards that, when used together, provide a standard definition for the interchange of information between CASE tools.
CDIF IDENTIFIER	An attribute that uniquely identifies an object in the Model Section of a transfer.
CDIF INTEGRATED META-MODEL	The description of the set of concepts and notations used to define a model. The CDIF Integrated Meta-model defines an Entity-Relationship-Attribute model that is used to construct and define models used in systems development.

¹This glossary is not a formal part of the attached EIA standard, but is included for purposes of information only.

CDIF METAIDENTIFIER

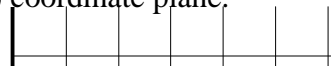
A meta-meta-attribute that uniquely identifies a meta-object in the Meta-model Section of a transfer.

CDIF META-META-MODEL

The description of the set of concepts and notations used to define a meta-model. Specifically, the CDIF Meta-meta-model defines an Entity-Relationship-Attribute model that is used to construct and define both meta-models and the CDIF Meta-meta-model itself.

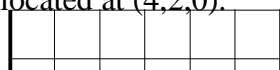
CDIF PIXEL

A grid cell on the coordinate plane in the CDIF Coordinate Frame. The shaded area in the figure below is a CDIF Pixel in the $(x,y,4)$ coordinate plane.



CDIF POINT

The location of a point in 3-dimensional space as specified by the x,y,z coordinates. The circle in the figure below encloses a point located at $(4,2,0)$.



CDIF TRANSFER

A combination of a particular syntax, a particular encoding of that syntax, and a meta-model. In other words, a complete definition of the format and contents of a transfer.

CDIF TRANSFER FORMAT

A combination of a particular syntax and a particular encoding of that syntax which together provides a complete definition of the transfer format.

CDIF TRANSFER SYNTAX AND ENCODING

A standard vehicle format supported by CDIF. The combination of SYNTAX.1 and ENCODING.1 forms the initial CDIF Transfer Syntax and Encoding.

CHARACTER SET

A character set is a collection of characters used in an Encoding to represent terminal symbols. The character set used is significant in the encoding of text and string meta-attributes for a CDIF Transfer.

CLEAR TEXT	A form of encoding where the resulting physical file is human-readable.
COORDINATE PLANE	See <i>CDIF Coordinate Frame</i> .
ENCODING	An encoding defines how the elements of a syntax are physically represented using an identified character set. Details of representation of the various terminal symbols and data types in the syntax's grammar are provided.
ENCODING.1	The CDIF Family of Standards supports multiple transfer formats, each composed of a syntax and an encoding. ENCODING.1 is the primary encoding defined within the CDIF Family of Standards.
ENTITY	An object (i.e., thing, event or concept) that occurs in a model (i.e., transfer).
INFORMATION CONTENT	The Information Content is the set of meta-model and model instances found in a CDIF Transfer.
INSTANCE	<p>An individual occurrence of a type (e.g., the diagram <i>MyDiagram</i> is an instance of the <i>Diagram</i> type).</p> <p>Usually, an instance is an object which occurs one meta-level beneath its (type's) definition. This is not always true, however (e.g., within the CDIF presentation subject areas where icon type definitions and their instances are both contained within the same meta-level).</p>
INTEGRATED META-MODEL	See <i>CDIF Integrated Meta-model</i> .

META

According to Webster's Ninth New Collegiate Dictionary: *more comprehensive: transcending - used with the name of a discipline to designate a new but related discipline designed to deal critically with the original one.*

Meta- is used in CDIF generally as a prefix to a concept to imply definition information about the concept. Specifically, used to designate the location of an object in the three model layers.

META-ATTRIBUTE

A definition of a characteristic of a meta-entity or meta-relationship. Instances of a meta-attribute occur in a model as data values.

META-ENTITY

A definition of a type of data object that occurs in CDIF models. Specifically, a meta-entity represents a set of zero or more meta-attributes, stored together to represent a thing, event or concept that has instances in a model.

META-LAYERS

See *Model Layers*.

META-META-ATTRIBUTE

A definition of a characteristic of a meta-meta-entity or meta-meta-relationship. Instances of a meta-meta-attribute occur in a meta-model as meta-data values.

META-META-ENTITY

A definition of the behavior and structure of meta-entities, meta-relationships, meta-attributes, or subject areas (i.e., a definition of the meta-object definitions used to describe information in models).

META-META-MODEL

See *CDIF Meta-meta-model* .

META-META-RELATIONSHIP

A definition of a type of data object that occurs in CDIF meta-models. Specifically, a meta-meta-relationship represents the definition of a relationship between instances of meta-meta-entities.

META-MODEL

A meta-model contains detailed definitions of the meta-entities, meta-relationships and meta-attributes whose instances represent an actual CDIF transfer.

The CDIF Integrated Meta-model (as defined in the set of standards that comprise the CDIF Family of Standards) is a definition of all of the types of information that can be transferred in a CDIF Transfer without using the CDIF extensibility mechanism.

META-OBJECT

A meta-object is a generic term for meta-entities and meta-relationships

META-RELATIONSHIP

A definition of a type of data object that occurs in CDIF models. Specifically, a meta-relationship represents the definition of a relationship between meta-entities that has instances in a model. A meta-relationship may also define a set of zero or more meta-attributes, stored together to represent characteristics of a relationship between meta-entities.

MODEL

A specific collection of software engineering data. This is called a model because it usually represents a model of a software system under development.

A collection of instances of meta-objects.

MODEL LAYERS

The different layers of definition (or abstraction) used in defining the CDIF Family of Standards. The four model layers in CDIF are: user data, model, meta-model, meta-meta-model.

Any given model layer provides an accurate and complete definition of all the instances that may occur one layer below the given layer. For example, the meta-meta-model provides a set of definitions that are used to construct and understand the meta-model; the meta-model provides a set of definitions that are used to construct and understand a model.

NON-TERMINAL SYMBOL	A part of the hierarchical definition of a syntax that is further decomposed in the hierarchy.
PRODUCTION RULE	The definition of a syntactic element in the form of an expression consisting of characters, character strings, and other syntactic elements.
RELATIONSHIP	A real-world association among one or more entities. Where the association is between an entity and itself, the relationship is said to be recursive.
SUBJECT AREA	A related collection of meta-object instance definitions. Subject areas are used to define scoped areas of interest. Subject areas overlap to ensure the integration of the overall Meta-model, but a tool need only use those subject areas relevant to the data to be exported/imported.
SYNTAX	A syntax is a definition of the format of information in a CDIF transfer. The definition is in the form of a grammar. It is specified with regard to ordering and repetition, down to the level of terminals, but does not specify the representation of any of the terminal objects in the grammar. Those decisions are defined in the encoding of the syntax.
SYNTAX.1	The CDIF Family of Standards supports multiple transfer formats, each composed of a syntax and an encoding. SYNTAX.1 is the primary syntax defined within the CDIF Family of Standards.
TERMINAL SYMBOL	A part of the hierarchical definition of a syntax that is not further decomposed in the hierarchy.
TRANSFER	See <i>CDIF Transfer</i> .
TRANSFER FORMAT	See <i>CDIF Transfer Format</i> .

WORKING META-MODEL

The working meta-model is the definition of the specific meta-objects that may be instantiated in the model section of a CDIF Transfer. The working meta-model comprises the meta-objects in the CDIF Integrated Meta-model that are used by the subject areas referenced in the meta-model section of the transfer, and the meta-objects defined as extensions in the meta-model section.

REFERENCES

1

2

[1] ISO 8859 Part 1: 1987 8-bit Single-byte Coded Graphic Character Sets:

3

Latin Alphabet No. 1.

4

1

This page is intentionally blank

INDEX

1

2	!	21	B
3	!! 5	22	Backus Naur Form (See BNF)
4	.	23	Binary encoding 19
5	... 5	24	BNF 4
6	:	25	(See also Formal grammar)
7	::= 4	26	conventions 4-5
8	<	27	Bold type 3, 5
9	<> 4	28	Braces
10	[29	angle 4, 5
11	[] 4	30	curly 4
12	{	31	square 4
13	{ } 4	32	C
14		33	CDIF
15	4	34	glossary entry 21
16	A	35	CDIF Clear Text Encoding 18
17	Alternative operator 4	36	glossary entry 21
18	Angle braces 4, 5	37	CDIF Coordinate Frame
19	Attribute	38	glossary entry 21
20	glossary entry 21	39	CDIF Family of Standards 1
		40	glossary entry 21
		41	CDIF Identifier 14
		42	glossary entry 21
		43	CDIF Integrated Meta-model
		44	glossary entry 21
		45	CDIF MetaIdentifier 13
		46	glossary entry 22
		47	CDIF Meta-Meta-Model
		48	glossary entry 22
		49	CDIF Meta-model 18
		50	CDIF Point
		51	glossary entry 22
		52	CDIF Signature 8-9
		53	formal grammar 8
		54	CDIF Subject Area references (See Subject
		55	Area:reference)
		56	CDIF Transfer 6, 7
		57	components 7
		58	contents 7, 11-14
		59	formal grammar 11
		60	envelope 7, 8-11, 16
		61	example 11
		62	formal grammar 8, 15-16
		63	formal grammar 7
		64	format 1, 6
		65	multiple 6
		66	glossary entry 22

1	CDIF Transfer Format	50	Export
2	glossary entry 22	51	date of 12
3	CDIF Transfer Syntax and Encoding	52	time of 12
4	glossary entry 22	53	Exporter
5	Character encoding 18	54	name 12
6	Character set 6	55	version 12
7	identifier 8, Error! Not a valid bookmark in entry	56	Expression 4
8	on page 10	57	Extensability (See Meta-model:extension)
9	formal grammar 10	58	Extensions
10	Clear Text 18	59	to meta-model 18
11	glossary entry 23		
12	Clear text encoding 18		
13	CODESET 6, 8, 10	60	F
14	Comment operator 5		
15	Configuration management 12		
16	Conventions 3	61	Formal grammar 15–16
17	BNF 4–5	62	(See also BNF)
18	Global 3	63	CDIFSignature 8, 15
19	Coordinate Plane	64	CDIFTransfer 7
20	glossary entry 23	65	CharacterSetID 10, 15
21	Curly braces 4	66	CharacterSetIdentifier 10, 15
		67	CODESET 10, 15
22	D	68	EncodingID 9, 15
		69	EncodingIdentifier 9, 15
23	Data type 6	70	EncodingVersion 9, 15
24	encoding of 17	71	SyntaxID 9, 15
25	Definition operator 4	72	SyntaxIdentifier 9, 15
26	Document	73	SyntaxVersion 9, 15
27	structure 2	74	TransferContents 11
		75	TransferEnvelope 8, 15
28	E	76	TransferEnvelopeComma 15
		77	TransferEnvelopeDoubleQuote 9, 10, 15
29	Element	78	TransferEnvelopePrintableCharacter 9, 10, 15
30	syntactic 4	79	TransferEnvelopeSpace 15
31	Ellipsis 5	80	TransferEnvelopeString 9, 10, 15
32	Encoding 1, 6, 7	81	Forward reference 13, 14
33	binary 19		
34	character 18	82	G
35	clear text 18		
36	different from syntax 19	83	Grammar 4
37	encrypted 19		
38	glossary entry 23	84	H
39	identifier 8, 9–10		
40	formal grammar 9	85	Header section 11, 12
41	multiple 6, 11, 18–19		
42	of data types 17		
43	version 9	86	I
44	<i>ENCODING.1</i>		
45	glossary entry 23	87	Information Content 6
46	<i>standard</i> i	88	(See also CDIF Transfer
47	Encrypted encoding 19	89	contents 6
48	Entity	90	glossary entry 23
49	glossary entry 23		

1 Instance 18
 2 glossary entry 23
 3 meta-attribute 7
 4 meta-entity 7, 14
 5 meta-relationship 7, 14
 6 Integrated Meta-model 1, 6, 12, 13, 14, 18
 7 glossary entry 23
 8 ISO 8859 29

J

9

K

10

L

11

M

12

13 Meta-
 14 glossary entry 24
 15 Meta-attribute
 16 glossary entry 24
 17 value 6, 14
 18 Meta-entity
 19 glossary entry 24
 20 instance 7, 14
 21 Meta-language 4
 22 Meta-layers
 23 glossary entry 24
 24 Meta-meta-attribute
 25 glossary entry 24
 26 Meta-meta-entity
 27 glossary entry 24
 28 Meta-meta-model
 29 glossary entry 24
 30 Meta-meta-object
 31 name 4
 32 Meta-meta-relationship
 33 glossary entry 24
 34 Meta-model
 35 CDIF 18
 36 extension 13–14
 37 extension of 18
 38 glossary entry 25
 39 section 7, 11, 12–14
 40 Meta-object
 41 glossary entry 25

42 name 4
 43 Meta-relationship
 44 glossary entry 25
 45 instance 14
 46 Model
 47 glossary entry 25
 48 Section 11, 14
 49 Model Layers
 50 glossary entry 26
 51 Multiple encodings (See Encoding:multiple)
 52 Multiple syntaxes (See Syntax:multiple)

N

53

54 Name
 55 meta-meta-object 4
 56 meta-object 4
 57 Non-terminal Symbol 4
 58 glossary entry 26
 59 Notational conventions 4

O

60

P

61

62 Prerequisites 2
 63 Production Rule 4
 64 glossary entry 26
 65 Publisher 12

Q

66

67 Questions and answers 17–19
 68 Quotes 3

R

69

70 Range 5
 71 Relationship
 72 glossary entry 26
 73 Rule
 74 production 4

1	S	33	X
2	Square braces 4		
3	String meta-attribute 6		
4	Subject Area		
5	glossary entry 26	34	Y
6	reference 7, 12, 13, 18		
7	Syntactic element 4		
8	Syntax 1, 6, 7		
9	different from encoding 19		
10	glossary entry 26	35	Z
11	identifier 8, 9		
12	formal grammar 9		
13	multiple 6	36	
14	version 9		
15	<i>SYNTAX.1</i>		
16	glossary entry 27		
17	<i>standard</i> i		
18	T		
19	Terminal Symbol 5		
20	glossary entry 27		
21	Text meta-attribute 6		
22	Transfer (See CDIF Transfer)		
23	glossary entry 27		
24	Transfer Format (See CDIF Transfer Format)		
25	glossary entry 27		
26	U		
27	V		
28	Vertical bar 4		
29	W		
30	Whitespace 8, 16		
31	Working Meta-model 12		
32	glossary entry 27		

RELATED EIA STANDARDS

1

2 EIA/IS-106 CDIF - CASE Data Interchange Format - Overview

3 EIA/IS-107 CDIF - Framework for Modeling and Extensibility

4 EIA/IS-109 CDIF - Transfer Format - Syntax SYNTAX.1

5 EIA/IS-110 CDIF - Transfer Format - Encoding ENCODING.1