109
12-28-94 JS (2)

SANDIA REPORT

SAND94–2375 • UC–706
Unlimited Release
Printed December 1994

Application Protocol, Initial Graphics Exchange Specification (IGES), Layered Electrical Product

Edited by Lawrence J. O'Connell

Prepared by
Sandia National Laboratories
Albuquerque, New Mexico 87185 and Livermore, California 94550
for the United States Department of Energy
under Contract DE-AC04-94AL85000
and by the United States Department of Commerce under Contracts
NIST 43 NANB 414068 and N66001-89-C 0047 CDRL, A003.

Approved for public release; distribution is unlimited.

MASTER

SF2900Q(8-81)

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

Issued by Sandia National Laboratories, operated for the United States Department of Energy by Sandia Corporation.

NOTICE: This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, nor any of their contractors, subcontractors, or their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government, any agency thereof or any of their contractors or subcontractors. The views and opinions expressed herein do not necessarily state or reflect those of the United States Government, any agency thereof or any of their contractors.

Printed in the United States of America. This report has been reproduced directly from the best available copy.

Available to DOE and DOE contractors from
Office of Scientific and Technical Information
PO Box 62
Oak Ridge, TN 37831

Prices available from (615) 576-8401, FTS 626-8401

Available to the public from
National Technical Information Service
US Department of Commerce
5285 Port Royal RD
Springfield, VA 22161

NTIS price codes
Printed copy: A12
Microfiche copy: A06

DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.

SAND 94-2375
Unlimited Release
Printed December 1994

Distribution
Category UC-706

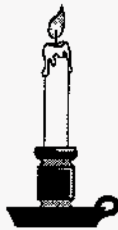
Application Protocol, Initial Graphics Exchange Specification (IGES), Layered Electrical Product

Committee Draft
Lawrence J. O'Connell, Editor
Project Development Department
Sandia National Laboratories
Albuquerque, NM 87185-0955

Abstract

An application protocol is an information systems engineering view of a specific product. The view represents an agreement on the generic activities needed to design and fabricate the product, the agreement on the information needed to support those activities, and the specific constructs of a product data standard for use in transferring some or all of the information required.

This application protocol describes the data for electrical and electronic products in terms of a product description standard called the Initial Graphics Exchange Specification (IGES). More specifically, the Layered Electrical Product IGES Application Protocol (AP) specifies the mechanisms for defining and exchanging computer-models and their associated data for those products which have been designed in two dimensional geometry so as to be produced as a series of layers in IGES format. The AP defines the appropriateness of the data items for describing the geometry of the various parts of a product (shape and location), the connectivity, and the processing and material characteristics. Excluded is the behavioral requirements which the product was intended to satisfy, except as those requirements have been recorded as design rules or product testing requirements.



> **"Information is neither Matter nor Energy"**
>
> *Norbert Wiener*

Initial Graphics Exchange Specification (IGES)

LAYERED ELECTRICAL PRODUCT APPLICATION PROTOCOL (AP)

VERSION 1.0: This document is the IGES Electrical Applications Committee release of a complete application protocol for layered electrical products technology. Included is the collected set of design objects which the committee recommends for transfer of electronic product models and which are within the IGES domains of logical (e.g., schematic) and physical product drawing and computer presentation, data supporting manufacturing, and product testing. A broad range of electrical and electro-mechanical product models may be exchanged when this AP is applied together with the general product model exchange capabilities of IGES. The Committee recommends this AP version for translator development and encourages comments on the quality of the documentation and in particular the set of design data objects. The Committee further expects this AP to facilitate a transition into the Standard for the Exchange of Product Model Data (STEP) through the specification of consistent IGES usage for electrical applications.

ABSTRACT

An application protocol is an information systems engineering view of a specific product. The view represents an agreement on the generic activities needed to design and fabricate the product, the agreement on the information needed to support those activities, and the specific constructs of a product data standard for use in transferring some or all of the information required.

This application protocol describes the data for electrical and electronic products in terms of a product description standard called the Initial Graphics Exchange Specification (IGES). More specifically, the Layered Electrical Product IGES Application Protocol (AP) specifies the mechanisms for defining and exchanging computer-models and their associated data for those products which have been designed in two dimensional geometry so as to be produced as a series of layers in IGES format. The AP defines the appropriateness of the data items for describing the geometry of the various parts of a product (shape and location), the connectivity, and the processing and material characteristics. Excluded is the behavioral requirements which the product was intended to satisfy, except as those requirements have been recorded as design rules or product testing requirements.

PREFACE

The Electricity Division of the Electronics and Electrical Engineering Laboratory and the Automated Manufacturing Research Facility of the Center for Manufacturing Engineering at the National Institute of Standards and Technology (NIST) completed a program in 1992 entitled "A Data Format Specification for Hybrid Microcircuit Assemblies." This project was sponsored by the Office of the Assistant Secretary of the Navy for Manufacturing Technology and was managed by the Naval Command, Control & Ocean Surveillance Center, Research Development Test & Evaluation Division (NCCOSC, RDT&E), San Diego, Contract No. N0001991IPAKC1R, MOD/ AMEND No. 0A-001. The project objective was the development of a specification for a neutral format to promote the exchange of design and manufacturing data for hybrid microcircuit products. The resulting document "Initial Graphics Exchange Specification (IGES) Hybrid Microcircuit Application Protocol," NIST TN 1295, January 1993 was distributed to Navy Contracting Officers, other program reviewers, and the Electrical Applications Committee (EAC) of the IGES Organization.

Reviews were held with industry, EAC, and the IGES Project from the Navy Program inception and throughout 1993. The discussions brought out the applicability of the Navy Hybrid Microcircuit work to a wide range of electrical product types. Following discussions in the EAC and with the Continuous Acquisition and Life Cycle Support (CALS) Interest Group, the editors of this document were tasked with bringing together the contributions of other authors and the NIST document contents mentioned above, resolving comments and experiences submitted during the years time.

The scope of this AP, formally stated in Section 1.3.2, includes the various aspects of all layered electrical products; the specification control drawings, circuitry and parts layout, and information concerning their fabrication. The specified data model is sufficiently detailed to support the design release, fabrication, and final assembly of a layered product system.

IGES is designed to support a broad range of applications and information, and it is recognized that few implementations will support all of the specification. An application protocol defines a logical subschema of the IGES specification, the usage of the subschema, and the necessary benchmarks for testing implementations. The application protocol for layered devices builds on the previous work of the Electrical Applications Committee of the IGES/PDES Organization. Such documents are seen as important for the development of STEP (Standard for the Exchange of Product Model Data) application protocols.

The data structures defined in this document are proposed in order to greatly improve the fidelity of layered electrical product applications data exchanged. The lack of fidelity has been termed "flavorings[1]" which occur from two primary sources. The first flavoring source is attributed to inconsistent use of computer aided design (CAD) systems, often because of differing ways to describe the product within different organizations. The second source of flavorings often results when the data structures of different CAD systems are converted to or from IGES data entities. Where an identified subset of the IGES entities did not address the flavoring issue, this document specifies product objects which constrain both sources of flavorings. This AP is proposed as a replacement of MIL-D-28000A[2] Class 3. This revision will define a new MIL-D-28000 class for the transfer of layered electrical product data by use of future versions of IGES.

[1]. The Sandia National Laboratories on behalf of the Department of Energy contributed a great deal of information on the flavorings problem and possible solutions during the mid 1980s.

The members of the EAC also note that other data formats may be employed. They would encourage continuation of work to achieve an industry consensus of the activity and reference model sections (Appendix A and 4.2 respectively) and additional interpretation (Section 5) models as needed for each additional format supported. Comments on this document should be sent to:

Sandia National Laboratories
Larry O'Connell
Mail Stop 0955
P. O. Box 5800
Albuquerque, NM 87185-0955

(505) 844-1061
E-Mail: ljoconn@sandia.gov

Future Directions

The Application Activity Model (Appendix A) and Application Reference Model (Section. 4.2) contain sufficient detail to support design release, fabrication, and final assembly of electrical product system. The IGES models (Section 5) in this document, however, are not meant to fully support product testing, analysis, and manufacturing. A full life-cycle data standard—possibly based on the Standard for the Exchange of Product Model Data (STEP)—will be necessary to support all phases of product processing.

This document does not complete the fabrication definition in terms of the sequence of processing steps and the parameters used in those steps.

This AP AIM is currently limited to layered electronic product (LEP) information contained in electronic computer-aided design (ECAD) systems. These systems are in widespread use. Their ability to translate data between dissimilar systems is a high priority in the user community. IGES serves as the implementation for this information because most ECAD suppliers are familiar with the specification and support it in their software packages.

A broadened capability for this AP that includes manufacturing, test, simulation, behavior, and documentation is desired by users and manufacturers of hybrids. Much of this information is outside the scope identified in most ECAD systems and the IGES specification.

About the Document

The style, format, and much of the material for this application protocol was taken from the "3D Piping IGES Application Protocol." That AP specifies the mechanisms for defining and exchanging 3D piping system models in IGES format.[3] The "3D Piping IGES Application Protocol" was the first IGES AP to be delivered to industry and led to the development of STEP (Standard for the Exchange of Product Model Data) application protocols. The authors of this

[2] MIL-D-28000 is the Department of Defense specification that identifies the requirements to be met when product definition data is delivered in IGES format.

[3] See Section 2.4, Reference 16 (NOTE: This reference was the source of AP material, but is not the most current version of the IGES Piping AP.).

document are indebted to Mark Palmer for his work and the guidance that it provides. It is the authors' belief that if the CAD/CAM industry is to use application protocols consistently, we need consistent application protocols.

Preparation of this document is a tribute to the ability to move text and graphics between applications and computer systems.[4] The text was originated in WordPerfect™ DOS™ and filtered into FrameMaker™ 2.1 on a Sun™ workstation for the original formatting and some of the graphics. The AAM model was developed in Design/IDEF™, and glossary in Hypercard; both on the Macintosh™. The ARM was drawn in MacDraw II™. The AIM was developed at International TechneGroup Inc. (ITI) where the text was written in Word™ for DOS and the graphics drawn in AutoCAD™. The graphics were transmitted to Raytheon in IGES format where they were edited and translated into FrameMaker™ MIF format. Final document assembly was done in FrameMaker™ 3.0/Mac, 4.0/PowerPC and LASER-printed from Frame's PostScript™ output.

ACKNOWLEDGMENTS

The editors thank the following individuals for contributions to the development of this document:

Authors and Contributors to the "IGES Hybrid Microcircuit Application Protocol":

Thomas Leedy—the author of the first work toward this document—and Curtis Parks, NIST; Roger McCollough and Charles Azu, NCCOSC; Larry Savage and Patrick Toomey, and Thomas Makoski, ITI, Inc. The product objects were developed by Diane Elmer, Scicards; Bernard Gilbert, Prime-ComputerVision; Dave Kehmeier, Mentor Graphics; Randy Lawson, Cadence Design Systems, Inc.; Thomas Makoski, ITI, Inc.; and Anthony Prince, Intergraph-Dazix.

Authors and Contributors to this Document:

Larry O'Connell (Chairman of the Electrical Applications Committee, IGES/PDES Organization), Sandia Laboratories; together with: Bill Loye, Loye & Associates; Curtis Parks, National Institute of Standards & Technology, John Russell, Honeywell Commercial Flight Systems; and the membership of the Electrical Applications Committee.

Special thanks to Gaye Garrison, PC Support Inc., for document editing and file preparations.

---

4. Word and DOS are trademarked by Microsoft Corporation; WordPerfect is trademarked by the WordPerfect Corporation; Framemaker is trademarked by Frame Technology Corporation; Sun is trademarked by Sun Microsystems, Incorporated; Design/IDEF is trademarked by Meta Software Corporation; Hypercard and Macintosh are trademarked by Apple Computer, Incorporated; MacDraw is trademarked by Claris Corporation; AutoCAD is trademarked by Autodesk, Incorporated; and PostScript is trademarked by Adobe Systems, Incorporated. Trademarked products were mentioned in this document as examples of the range of possible products and are not necessarily endorsed by the contributors to this document.

Table of Contents

Abstract i

Preface ii

Future Directions iii

About the Document iii

Acknowledgements iv

1. INTRODUCTION 1

1.1. Purpose 1

1.2. Background 1

1.2.1. Electrical Assembly Complexity 2
1.2.2. Consistent Information for Concurrent Engineering 3

1.3. APPLICATION PROTOCOL CONTENTS 4

1.3.1. Terms and Definitions 4
1.3.2. Scope 4

2. APPLICATION AND CORE REQUIREMENTS 6

2.1. Application Information Requirements 6

2.2. Application and ARM Correspondence 7

2.3. Normative References 9

2.4 Informative References 9

3. DEFINITIONS 11

3.1. IGES Application Protocol Definitions 11

3.2. LEP Application Definitions 13

4. INFORMATION REQUIREMENTS AND APPLICATION REFERENCE MODEL 31

4.1 Application Data Views 31

4.2. Application Reference Model 31

4.2.1. Descriptions of the ARM Data Constructs 31
4.2.1.1. ARM Methods and Language 31
4.2.2. Data Construct Definitions 33
4.2.2.2. Data Construct Assertions 33
4.2.3. ARM Diagrams 34

5. IGES APPLICATION INTERPRETED MODEL 43

5.1. ARM to AIM Mapping 44

5.2. IGES Structure and Syntax of the Application Interpreted Model 44

5.2.1. IGES File Structure 44
5.2.1.1. START Section 45
5.2.1.2. TERMINATE Section 45
5.2.1.3. GLOBAL Section 45
5.2.1.4. DIRECTORY ENTRY and PARAMETER DATA Sections 45
5.2.2. Global Values and Entity List 45
5.2.3. Basic Syntax used in the AIM 50
5.2.3.1. Object Definition Block 50
5.2.3.2. Object Instance Block 50
5.2.3.3. Object Value Block 51
5.2.3.4. Object Reference Mechanism 52

5.3. AIM Object Models 53

5.3.1. Interface Object Models 54
5.3.1.1. LEP Part Library IGES File 54
5.3.1.2. LEP Physical Layout IGES File 55
5.3.1.3. LEP Technical Illustration IGES File 56
5.3.1.4. LEP Simulation/Analysis IGES File 57

5.3.2. LEP Features 59

5.3.2.1. Component Placement Keepin/Keepout 60
5.3.2.2. Component Thermal Outline 61
5.3.2.3. Drawing Definition 62
5.3.2.4. Drawing Instance 63
5.3.2.5. Generic Part Definition 64
5.3.2.6. Generic Part Instance 65
5.3.2.7. Hole 66
5.3.2.8. Join 67
5.3.2.9. Jumper Wire 68
5.3.2.10. Keepin/Keepout 69
5.3.2.11. Land 70

5.3.2.12. Layer Outline 71
5.3.2.13. LEP Definition 72
5.3.2.14. LEP Instance 73
5.3.2.15. Level to LEP Layer Map Entity (Type 406 Form 24) 74
5.3.2.16. Net 75
5.3.2.17. Package Figure Definition 77
5.3.2.18. Package Figure Instance 79
5.3.2.19. Padstack Definition 80
5.3.2.20. Padstack Instance 81
5.3.2.21. Panel Definition 82
5.3.2.22. Panel Instance 83
5.3.2.23. Part Placement Boundary 84
5.3.2.24. Pin 85
5.3.2.25. Routing Keepin/Keepout 86
5.3.2.26. Test Point 87
5.3.2.27. Trace Keepin/Keepout 88
5.3.2.28. Via 89
5.3.2.29. Via Keepin/Keepout 90
5.3.2.30. Wire Bond 91

5.3.3. Display Geometry Object Models 93

5.3.3.1. Arc 93
5.3.3.2. Circle 94
5.3.3.3. Closed Curve 95
5.3.3.4. Composite Curve 96
5.3.3.5. Geometry Figure Definition 97
5.3.3.6. Geometry Figure Instance 98
5.3.3.7. Line Entity (Type 110, Form 0) 99
5.3.3.8. Multi-Line 100
5.3.3.9. Non-Closed Curve 101
5.3.3.10. Polygon 102
5.3.3.11. Point Entity (Type 116, Form 0) 103
5.3.2.12. Predefined Planar Shape 104

5.3.4. Miscellaneous Object Models 105

5.3.4.1. Generic Data Property (Type 406, Form 27) 106
5.3.4.2. Hierarchy Property (Type 406, Form 10) 112
5.3.4.3. Name Property (Type 406 Form 15) 113
5.3.4.4. Network Subfigure Definition Entity (Type 320, Form 0) 114
5.3.4.5. Network Subfigure Instance Entity (Type 420, Form 0) 115
5.3.4.6. Object Locator 116
5.3.4.7. Part Number Property (Type 406, Form 9) 117
5.3.4.8. Sectioned Area Fill Definition 118
5.3.4.9. Subfigure Definition Entity (Type 308, Form 0) 119
5.3.4.10. Subfigure Instance Entity (Type 408, Form 0) 120
5.3.4.11. Text String 121

5.3.5. LEP Semantic Property 123

5.3.6. DE Referenced Object Models 124

5.3.6.1. Level Definition 125
5.3.6.2. Line Width Definition 126
5.3.6.5. Transformation Matrix 127
5.3.6.6. Transformation (Mirrored) Entity (Type 124, Form 1) 128
5.3.6.7. Transformation Matrix (Normal) Entity (Type 124, Form 0) 129

5.3.7. DE Section Object Models 131

5.3.7.1. Associativity DE 131
5.3.7.2. Definition DE 132
5.3.7.3. Geometry DE 133

5.3.7.4. Instance DE 134
5.3.7.5. Property DE 135
5.3.7.6. Transformation Matrix DE 136

6. Implementation and Conformance Testing Guidelines 139

6.1. PROCESSOR CONFORMANCE REQUIREMENTS 139

6.1.1. Testing Considerations 140
6.1.2. Conformance Considerations 140

6.2. Test Purposes and Test Groups 142

6.3. LEP Constructs 148

6.3.1. LEP Constructs Supporting Signal Conductivity 148
6.3.2. LEP Constructs Supporting a Physical Representation 149
6.3.3. LEP Constructs Supporting a Logical Representation 149

6.4. Entity Usage in a PDQ LEP Representation 149

6.4.1 DE Fields are Not Expected to Convey Meaning 150
6.4.2. Entities Which May Appear in a PDQ LEP Representation 151
6.4.2.1. LEP Usage of Geometry 151
6.4.2.2. LEP Usage of Annotation 152
6.4.2.3. LEP Usage of Structure (Excluding Properties) 153
6.4.2.4.LEP Usage of Existing Specific Property Entities 154
6.4.2.5. LEP Usage of the (Augmented) Generic Data Property Entity 154
6.4.2.6. LEP Usage of Simulation/Analysis Entities 155
6.4.3. Entities Which Shall NOT be Used in a PDQ LEP Representation 155

6.5. LEP Entity Usage for Core Applications 155

6.5.1. LEP Entity Usage for Padstack Representation 155
6.5.1.1. LEP Entity Usage for a Physical Padstack Definition 156
6.5.1.1.1. Entities Required to Add the Representation of Drilled Holes into a Padstack 156
6.5.1.1.2. Entities Required to Add Display Appearance of Holes in a Padstack 157
6.5.2. LEP Entity Usage for Physical Component Representation 159
6.5.2.1. LEP Entity Usage for Physical Component Definition 160
6.5.2.2. LEP Entity Usage for Physical Component Instance 161
6.5.2.3. LEP Usage for Physical Connect Point 162
6.5.2.4. LEP Entity Usage for Physical Via 163
6.5.3. LEP Entity Usage for Physical Mechanical Part Representation 165
6.5.4. LEP Entity Usage for Physical PWB Representation 166
6.5.4.1. LEP Physical PWB Definition 166
6.5.4.2. LEP Physical PWB Instance 167
6.5.4.3. LEP Constructs Supporting Conductivity 169
6.5.4.4. LEP Constructs Supporting Components 169
6.5.4.5. LEP Constructs Supporting Connectivity 170
6.5.4.6. LEP Constructs Supporting a Physical Representation 170
6.5.4.7. LEP Physical Flow Associativity (Signal/Netlist) 171
6.5.4.8. LEP Physical LEP Signal Carriers (Traces, Shapes) 172
6.5.4.9. LEP Physical PWB Features: Fiducials 173

6.6. LEP Entity Usage for Specific Applications 174

6.6.1. LEP Entity Usage for Design/Engineering Applications 174
6.6.1.1. LEP Entity Usage for Netlist Reports 174
6.6.1.2. LEP Entity Usage for Component Placement Reports 175
6.6.1.3. LEP Entity Usage for Bill of Materials Reports 176
6.6.2. LEP Entity Usage for Manufacturing Applications 177
6.6.2.1. LEP Entity Usage for Photo-lithography Tools 177
6.6.2.2. LEP Entity Usage for LEP Test Aids 178

6.6.2.3. LEP Entity Usage for Numerically Controlled (NC) Production Files 180
6.6.2.3.1. LEP Entity Usage for Component Insertion 180
6.6.2.3.2. LEP Entity Usage for Component Pick and Place 182
6.6.2.3.3. LEP Entity Usage for Component Verification 185
6.6.2.3.4. LEP Enitity Usage for Wire Wrap 185
6.6.2.3.5. LEP Entity Usage for Labelmaker 186
6.6.2.4. LEP Entity Usage for Panelization 186
6.6.3. LEP Entity Usage for Drafting/Documentation Applications 187
6.6.3.1. Major Differences Between the Documentation and the PDQ Model of a Conformant LEP AP 188
6.6.3.2. Changes and Additions to the PDQ LEP AP 188
6.6.3.3. LEP Entity Usage for Fabrication Documentation 188
6.6.3.4. LEP Entity Usage for Assembly Documentation 189
6.6.3.5. LEP Entity Usage for LEP Technical Illustrations 189
6.6.4. LEP Usage for CAD to CAD Exchange 189
6.6.4.1. LEP Entity Usage for Layer Assignment 190
6.6.4.2. LEP Entity Usage for Color 190
6.6.4.3. Semantic Properties for LEP Applications 190
6.6.4.4. Entity Usage for Presentation (Appearance) 198
6.6.5. Entity Usage for CAD Simulation/Analysis 198

A. Activity Model Contents A-1

B. Application Protocol Usage Guide B-1

C. Technical Discussions C-1

List of Figures & Diagrams

4. ARM Figures

4-1. Component Parts of an IDEF Relationship 32

4-2. Cardinality 33

4-3. The Categorization Structure 34

4. ARM Diagrams

View 4-1 Top-Level Entities 36

View 4-2 Expanded Dependents of Pattern 37

View 4-3 Expanded Dependents of LEP Version 38

View 4-4 Expanded Dependents of LEP Assembly Occurrence 39

View 4-5 Expanded Component Model 40

View 4-6 Expanded Dependents of Assembly Consumables and Deposition Component 41

5. AIM Figures (Note: AIM Objects are listed in Table of Contents)

5-1. AIM Object and Referencing 48

5-2. References to Previously Defined Object 49

5-3. Object Field Value Restrictions 50

5.-4. Reference Mechanism 51

5-5. Representation Mechanism 51

6. Implementation and Conformance Testing Figures

6-1. LEP Information Interface Requirements 141

6-2. General Structure of Objects Found in a Padstack 159

6-3 LEP Semantic Features 190

List of Tables

2-1. Application to ARM Entity Correspondence 8
6-1. Overview of DE Settings for PDQ 151
6-2. DE Settings of Annotation Entities in a PDQ IGES File 153
6-3. IGES Data Fragments Showing Padstack Structures 158
6-4. DE Settings of Via Entities in a PDQ IGES File 164
6-5. DE Settings of LEP Instance Entities in a PDQ IGES File 169
6-6. Level Function Properties (Type 406, Form 3) 191
6-7. Semantic Properties of Top Level LEP 192
6-8. LEP Semanic Properties of Network 193
6-9. LEP Semanic Properties of Substrates 193
6-10. Semantic Properties of Component and/or LEP 193
6-11. LEP Semantic Properties of Component Definitions (Type 308 or 320) 194
6-12. LEP Semantic Properties of Component Instances (Type 408 or 420) 194
6-13. LEP Semantic Properties of Connect Point (Type 132) 195
6-14. LEP Semantic Properties of Closed Curves 195
6-15. LEP Semantic Properties of Non-Closed Curves 196
6-16. LEP Semantic Properties of Feature Definitions 196
6-17. EP Semantic Properties of Vias 196
6-18. Production Engineering LEP Semantic Properties 197
6-19. Insertion Machinery LEP Semantic Properties 197
6-20. Placement Machinery LEP Semantic Properties 197
6-21. Wire Wrap Application LEP Semantic Properties 198
6-22. Production Test LEP Semantic Properties 198

BLANK PAGE

Initial Graphics Exchange Specification (IGES) LAYERED ELECTRICAL PRODUCT APPLICATION PROTOCOL

1. INTRODUCTION

1.1. Purpose

This application protocol (AP) for layered electrical product (LEP) assemblies and parts to be incorporated onto such assemblies specifies the structure of Initial Graphics Exchange Specification (IGES) data for the representation of the product definition and for the exchange of these definitions from one LEP defining application to another. Since the LEP application protocol makes use of a specific interpretation of entities in the IGES file, both the sending and receiving processors must conform to this AP with regard to the information they process. It will not suffice to simply use IGES entities listed in a subset in the AP or in IGES or other documents. Proper packaging of connectivity information, for example, is crucial to success of some subsequent operations.

1.2. Background

The layered electrical product is a complex assembly made up of both graphic and non-graphic information, interrelated through parent/child relations and associations.

LEPs are defined to be modules or subcircuits that are incorporated into larger electronic assemblies, in a hierarchy of devices for use in operational systems. The LEP may be either a monolithic device, such as an integrated circuit, or an assembly such as a hybrid microcircuit or printed circuit, or more than one assembly connected by means of a cable. Typically, an LEP is connected to the larger assembly with external leads or pins. An LEP assembly incorporates an insulating substrate onto which a mix of integrated circuits and other electronic components (such as thick- and thin-film devices) are interconnected.

A neutral data format serves several purposes: It permits the interchange of data between computer-aided design (CAD) and computer-aided manufacturing (CAM) systems. It allows the archiving of the LEP design in a format that can be used in the future, even if the original CAD or CAM systems or their software are no longer in use.

There are a number of motivations for developing a specified representation for LEPs. The most compelling motive is to reduce the errors created by different interpretations of a format as used for a particular product. A specified representation for LEPs can also minimize cost and maximize efficiency in the design and maintenance of translators. These uniform applications can provide means for coping with the increasing complexity of LEPs. In current practice, there is often the need for manual intervention in order to transfer the data between the CAD workstations and to the CAM stations that produce the LEP. Even in automated systems, the problem is compounded by the fact that a pair of data "translators" must be written for each pair of machines that must exchange data. If there are n machines in the system—*potentially* $n*(n-1)$—translators are needed.

Data hub concept

A neutral format serves as a data "hub." Each station served must only translate data from its own format to the neutral format of the hub to be able to interchange with any other station interfacing with the hub. For example, data could be exported from a CAD workstation to the hub and be available there in neutral format for extraction by other CAD workstations and CAM machines. In this arrangement, for n machines in the system, there are *potentially 2*n* translators needed, this is a substantial reduction if n is greater than four machines.

LEP Constructs Supporting Intelligent Representation

Intelligent Representation of a LEP includes the ability to uniquely identify different objects and attributes in the LEP. Subsequent applications may need to get "Product Data by Queries." Rather than refer to files and representations as "intelligent" we shall call them PDQ files or PDQ representations. For example, the ability to select the component by its reference designator of "R1" and its pin name of "1". This will be critical for all automated shop floor processes, such as inserters, testers, pick/place, inspection, and others.

Automated assembly graphics designed to facilitate production build need such PDQ intelligence. Nearly any automated (or partially automated) process needs the ability to query signals, Components, Pins, Pads, Test Points, Vias, fiducials, tooling holes, and mounting hardware. PDQ intelligence also relates to the organization of the semantic information in the file. Semantic properties indicate the context of the data. The LEP is conceptually the same as the "model" in a mechanical system. It reflects a real world physical structure, with an unscaled coordinate system, and reference points to align the product and various manufacturing with inspection equipment.

1.2.1. Electrical Assembly Complexity

Assemblies have become increasingly complex in nature. One measure of this increased complexity is the ratio of the area of active elements (usually silicon chips) to the unoccupied area of the substrate of a hybrid microcircuit. In a single-chip package of a monolithic integrated circuit, the ratio is about 1:20; in today's typical hybrid, the ratio is from 1:10 to 1:6. Present engineering efforts have the goal of raising this ratio to 1:1, i.e., 50% of the substrate would be covered by silicon chips. These high-density hybrids are often referred to as multichip modules, or MCMs. To cope with this expanding complexity, hybrid manufacturing will increasingly depend on CAD and CAM techniques. For practical implementation of combined CAD and CAM techniques, it is important to have a single electronic representation of the CAD data available for interfacing with CAM environments. The representations described in this AP apply to MCM and conventional hybrid technology plus integrated circuits and printed boards and related products. The technologies of integrated circuits, printed wiring assemblies, flex cables and flex circuits have comparable density increases.

Many assembly manufacturers still use extensive paper documentation, such as prints and drawings, to document their product during manufacturing. Often, these drawings are produced on CAD stations that contain information that is not represented on the drawing and yet may be useful during the design process. As automated manufacturing methods become more available, such "paper" documentation will impede manufacturing. Further, as the complexity of electrical products increases, it will become much more necessary to convey this information to manufacturing machines in computer-comprehensible form.

1.2.2. Consistent Information for Concurrent Engineering

Another benefit from a unified representation of the data describing a product is the ability to achieve concurrent engineering, which in the case of an LEP permits various automated and human resources to be applied to the design simultaneously. Since these resources share common data regarding the design, it is possible for various groups of engineers to refine the mechanical, electrical, thermal, and testability characteristics of the LEP in a much shorter time than would be required otherwise. In addition, concurrent engineering permits different application specialists to work in parallel with the designer. Thus, for example, those that are responsible for the manufacturing, assembly, quality, and reliability of an LEP design are able to provide suggestions concerning the design from its inception.

This method of business is in sharp contrast to the traditional methods where each department contributed sequentially to the design process of an LEP. Concurrent engineering methods promote a combined effort where all information builds on an existing model and changes can be easily accommodated through the separate functional areas. Since changes in the design are incorporated early in the design cycle, the costs of such changes is decreased. The result of effective concurrent engineering is a product at lower cost and with a shorter design cycle than would be realizable with traditional methods. Increased product quality results from accurate data transfer, as opposed to manual regeneration of CAD data on succeeding systems.

One of the ways in which the new method contrasts with the traditional methods is in the use of an architecture patterned after the ANSI-SPARC Three-Schema[4] structures. In this architecture, a "conceptual schema" is created which functions as a means for managing the coupling between the "user views" of the information (the external schema), and the "database implementation" needed (the internal schema) to support the information system. The language reference manual for a transfer format standard usually describes the internal format, and within the descriptions the internal schema—to varying degrees of formality— may be found. The AP provides the external schema (herein called the Application Reference Model or ARM) for a particular application area and the way the data structures (herein called the Application Interpreted Model or AIM) are to be built from the primitive constructs of the standard. In this case, as in many others, the conceptual schema is not provided explicitly. Lacking an explicit conceptual schema, the mappings between the ARM and AIM are subject to interpretation.

There are several existing neutral file specifications to describe electrical and electronic functions. These include specifications developed by the Institute for Interconnecting and Packaging Electronic Circuits (IPC), the Electronic Design Interchange Format (EDIF), the Initial Graphics Exchange Specifications (IGES), and the VHSIC Hardware Description Language (VHDL). These neutral file specifications may support many of the data elements needed to represent an LEP design. Adding these formats is encouraged, and may be accomplished by adding the appropriate models to Section 5 of this AP. During such additions, the remaining sections should be refined by agreement between the organizations that are responsible for the formats. Conflicting information requirements (e.g., Sections 3 and 4 of this document) among different formats are not considered appropriate to the goals of product data consistency.

[4] For more information on these data architectures see Reference 15.

1.3. APPLICATION PROTOCOL CONTENTS

Section 2 of the AP provides a list of documents which are used in the construction of this protocol. Definitions of terms used in the AP are listed in alphabetical order in Section 3. Part 3.1 of this section lists terms contained in the text of the AP while part 3.2 has terms contained in the various models (AAM, ARM, AIM) of the AP. Terms in both sections are listed along with reference to the models in which they appear. Section 4 contains the ARM; an engineer's view of data about the product being designed. In Section 5, the "IGES-exchangeable" information of the ARM is defined as product model objects defined in terms of IGES entities. Section 6 defines the tests for conformance to the IGES and the object definitions as required for information uses.

The following typesetting conventions have been used to identify the particular model referred to by semantic terms in the text of this document:

Roman, Initial Cap (followed by Type, Form numbers)	IGES entities
Italic, Initial Cap	ARM entities
ITALIC, ALL CAP	AIM objects

1.3.1. Terms and Definitions

In addition to the definitions listed in Section 3, there are many terms associated with electrical product technologies as discussed below.

Products to which this AP apply are also distinguished by many "technology" terms. In particular, some of the terms which are associated with integrated circuits include custom, application specific integrated circuit (ASIC), gate array, digital, analog, mixed, and monolithic microwave integrated circuit (MMIC). Some of the terms which are associated with hybrid microcircuits include multi-chip module (MCM), single-chip module (SCM), microstrip assembly, thin film circuit, thick film circuit, green tape design, and surface mount technology. Some of the terms which are associated with printed circuit assembly (PCA) include printed circuit board (PCB), printed wiring board (PWB) or assembly (PWA), Flexible Circuits, Flexible Cables, and microstrip board.

The characteristics which distinguish the above domain of product types are their physical product model defined by features on one or more strata. Some features of various strata may be associated with a signal or signal bundle and/or various electrical properties.

The AP-applicable product types may also include features common to mechanical product types. Such features may include base plates, milled pockets, routed edges, threads or threaded inserts, mounting brackets, and heat sinks.

The AP information is applicable to the various ways that parts and components are defined. One way is the (external) package, usually as defined in Joint Electron Device Engineering Council (JEDEC) specifications, or the "footprint" for such a package. Another way may be the depiction of wire bonds electrically connecting a silicon chip to the leads of the package. Yet another way is the depiction of the part assembled as a component on another LEP such as an MCM or a PCA.

1.3.2 Scope

Layered Electrical Products include Flexible (Flex) Cables, Flex Circuits, Printed Circuit Assemblies, (includes Printed Wiring Boards), Hybrid Microcircuit Assemblies, Multi-Chip Modules, and Integrated Circuit die. The unifying element common to all these LEP technologies is the se-

ries of photomasks used in their manufacture. Practitioners[5] of each technology use some jargon different from all or most of the others, but the concepts are remarkably similar.

The scope of this Application Protocol includes:

- a reduced set of two-dimensional geometry sufficient to describe physical features of Layered Electrical Products, deposited components, and incorporated parts;
- connectivity of Traces, Conductive Areas, wirebonds, and Vias built into the LEP plus Pins, Pads, and Sockets of components incorporated into the LEP;
- patterns of photoplots and masks used in fabricating the LEP substrate;
- data plus context to support automated and semi-automated fabrication (such as numerically controlled drilling, panel layout, and automatic part insertion), testing of some kinds (such as bare board and in-circuit), technical illustrations (including process pictures and maintenance manuals), and Engineering Drawings (including schematics, netlists, bills of material, assembly drawings and layouts). Notice that this AP is not sufficient for these activities (for instance, a drafting AP may be needed to control dimensions and other annotation) but this AP controls the provision of much of the technical information needed by the activities.

The scope of this Application Protocol does not include:

- circuit simulation;
- behavior analysis;
- finite element analysis;
- drafting of engineering drawings;
- production of full color illustrations;
- control of process operations;
- selection of component parts;
- control of automated testing;
- vendor qualification; nor
- dispensing of consumable properties.

The AP may cover some of the activities within its scope in less detail than users might like. If so, such users are invited to send constructive suggestions to the editor of this AP.

5. The Application Reference Model in 4.2.3 below was developed largely by people working in the domain of Hybrid Microcircuits. Some of the people involved had worked on the Cal Poly model years earlier. That model was developed by a team of people experienced with Printed Boards and Integrated Circuits.

2. APPLICATION AND CORE REQUIREMENTS

This AP applies to electrical designs which were originated in a computer program (herein called "ECAD" system) designed to capture electrical-specific information. (Technically, this term often refers to Electrical Computer Aided Engineering (CAE) systems as well as to Electrical Computer Aided Design (CAD) systems.) Some electrical designs, drawings, or descriptions may be originated on non-ECAD systems sometimes called MCAD systems; IGES files from these systems shall follow the guidelines established within the IGES Drafting Application Protocol. The following subsections provide guidelines for determining the content of those applications which depend on the electrical data (most importantly connectivity data) defined within ECAD systems.

This AP requires that non-ECAD systems either display the graphics associated with the Network Subfigure Instances and Connect Point IGES entities, or convert them to their internal version of IGES Subfigure Instance and Point entities. Non-ECAD systems are not required to write out IGES files with Network Subfigure Instances and Connect Point entities

2.1. Application Information Requirements

The core requirements were determined by examining a portion of the product life cycle. These life cycle tasks were:

- Physical Design Layout
- Manufacturing
- Visualization (for assembly, troubleshooting, etc.)
- Testing
- Logistic Support Documentation

A single LEP design file may be used as the source for several different kinds of processes. Each independent entity in the ARM may also constitute an application requirement. Further, the child entities of these entities may, together with their existence-dependent entities, be used in an application structure. For a view of the relationships between core requirements and the ARM, see table 1. The symbol "+" indicates that all child entities apply as well.

Parts are often involved in a lead preparation step such as lead bending and tinning prior to their being assembled on an LEP. Transfer of part models which specify these intermediate life cycle steps are considered within the domain of this AP.

During these life cycle stages there are data added such as pin swapping and netlist back annotation. These data are added within the CAD or CAE system; the IGES output file following such CAE operations is differentiated by a different file name. Different fabrication facilities have different capabilities and pre-defined processes. Thus it may be premature for the IGES file to contain detailed information governing the fabrication processes prior to the selection of a facility and receipt of a commitment at that facility. As concurrent engineering becomes a reality, this will be less of a concern because the fabrication planner will participate in the definition captured and then transferred using IGES.

This AP is not applicable to the electrical product life cycle phases preceding the capture of the structural design by way of a schematic drawing or netlist representation unless the behavior is presented as a graphic such as a waveform. Where product behavioral definition exists in a data format other than IGES, provisions will be defined for the association of behavior at a logical pin (or "port") with a schematic or physical pin defined within the IGES file. Such association is fre-

quently achieved in CAE systems by matching on the assembly component's reference designator and pin number occurring in each data definition.

The information for all life cycle phases may originate in a computer aided design (CAD) system. Some information may only be computer-processed within a Computer Aided Engineering (CAE) system. Or, part of the information for any category may be created by manual—non-CAD/ CAE— means. LEP information from any source must be constrained semantically for use within the AP, and further, the requirements for the IGES form defined herein is (syntactically) constrained by the constructs in the application interpretation model.

All functionality categories which have been originated in a CAD system are capable of being transmitted to another, or an intermediate CAD or CAE system. In this event, much of the information may be "exchanged" without distinction. When electrical-specific information has been added by a CAE system, such information may not be comprehensible by a CAD system. The CAD system receiving a CAE-created file is required to ignore (i.e., to not process) the connectivity and electrical property entities, but to process and display the physical product model.

All information originated from a CAE system shall be processed by another CAE system; whether through direct conversion into the CAE native data, or through display to the CAE operator in such a way that the information may be accommodated by manual entry means.

For example, a design exchanged between CAD systems may adopt a level of information content that is greater than the Drawing. The receiving CAD system would benefit by "knowing" the properties and rules defined in the source system in addition to the product as a drawing graphic.

LEP Entity Usage for Core Applications

Section 5. identifies objects, which in turn defines which entities are required to support a specific application. For example, production test requires Network Subfigure Instances (representing components) and Connect Points (representing pins), in order to identify each testable point uniquely.

Entities may not always map directly between various applications. Section 6.6. contains guidelines for specific applications. For example, within the photolithography application, some programs will accept complex geometry, text strings, tapered lines, and subfigure constructions. Other programs can understand no more than simple fixed width lines (no arcs), and a few predefined geometries called flashes.

There are two different ways of looking at this problem. One approach would have the LEP AP detail which entities may be found in conformant files, implying a method to process those entities into something the application would accept. The second approach would detail which entities will probably be accepted by the application. The LEP AP follows the first approach, and assumes there is a tool available to perform required tasks of simplifying and remapping entities from the LEP AP conformant file.

2.2. Application and ARM Correspondence

The LEP is defined in this AP by a combination of documents (called presentation) and CAD/ ECAD data (called representation). The CAD/ECAD data consists of the screen display data and the CAD/ECAD product model itself. The ARM reflects the data distinction. For example, the product may be defined through a drawing produced manually or on a CAD system, such as a Specification Control Drawing. CAD systems may be utilized in such a way as to produce a "pa-

per document" intended for human interpretation. Information such as the electrical schematic and waveforms are frequently treated as such drawing data. Some applications include the CAD or ECAD display information such as the color with which a particular system-layer is displayed. Some applications are developed such that lists may be extracted (e.g., net lists, test probe locations, and drill lists) from the ECAD or CAE model of the product.

For purposes of this AP, the following table indicates appropriate applications and the entities in the ARM which are related. The symbol "+" indicates that all child entities apply as well.

Table 2-1: Application to ARM Entity Correspondence

Application	CAD/ CAE	ARM Entities
Bill of Materials	both	Component, Assembly Consumables, Substrate
Package	CAE	Device Package
Visualization	CAE	LEP Documentation +
Geometry	both	Geometry
Artwork	both	Pattern +
Surface mount pick & place	both	Area+, Flashed Geometry, Via Hole, Probe Pad
Panel	both	Panel MultiUp, LEP Assembly
Test programs	CAE	LEP Assembly Component
Wire Wrap	CAE	Pin +
Flex tape	CAE	Device Package
DITMCO (continuity test) support	CAE	Test, Net
Thru-hole auto-insert	CAE	Process Step, LEP Assembly Component
Semi-automatic insertion	CAE	Process Step, LEP Assembly Component
Combination test programs & fixtures	CAE	Tool, Test
SMT LEP producibility	CAE	Process Step
Adhesive dispensing	both	Process Step
In-work assy graphics	both	LEP CAD Presentation

The Application Reference Model (ARM) is explained and presented in Section 4. The ARM is constructed in terms used within the electronic design and fabrication community (i.e., an external schema). For example Part and Component are used, and are given definitions appropriate to those found in engineering as related to the use of CAD systems and supporting libraries. The IDEF1X modeling technique is used to develop the ARM and a brief tutorial on how to read the model is included.

The ARM was constructed to depict the complete layout-to-manufacturing of the LEP; primarily to be understandable by the engineer and planner. The AIM, described below and closely related to the IGES file structures, may not contain all ARM structures but will be more detailed. The ARM entities which are not defined as AIM objects are graphically marked in the ARM.

The Application Interpretive Model (AIM) is described and presented in Section 5. The mapping of the ARM to AIM is explained. The objects declared in this section are associated to general and specific CAD data items (i.e., an internal schema). A description of the IGES file structure and the breakdown of its sections is given. A modeling syntax is used to develop the AIM and a brief tutorial on how to read the model is included.

Section 6 details implementation and conformance guidelines for this Application Protocol. Process conformance requirements and testing purposes are given. Appendix A contains the Activity Model which indicates the processes which utilize ARM information. Appendix B has a parameter listing of typical LEP processing parameters/variables and their values.

2.3. Normative References

This AP is specifically applicable to IGES version 5.2. The versions of IGES listed below include electrical product capability; any one of them may be cited in an exchange. Earlier versions of IGES may contain electrical product-defining objects compliant with this AP.

IGES versions 1.0, 3.0, 4.0, and 5.2 have been registered as an American National Standard (ANS) Y14.26M by the American National Standards Institute. Additionally, IGES version 4.0 is registered by NIST as a Federal Information Processing Standard; FIPS PUB 177. IGES electrical-related entities and the versions in which each was first published are identified in Section 5 of this document.

1. IPC-T-50E, "Terms and Definitions for Interconnecting and Packaging Electronic Circuits," February, 1991, Institute for Interconnecting and Packaging Electronic Circuits, Lincolnwood, IL 60646.
2. MIL-D-28000A, 10 February 1992, and Amendment 1, February 1993; *Digital Representation for Communication of Product Data; IGES Application Subsets and IGES Application Protocols,* Director, CALS Evaluation and Integration Office, Pentagon Room 3D833, Washington, DC 20301
3. MIL-H-38534, "Hybrid Microcircuits, General Specification for," 31 March 1989.
4. The Initial Graphics Exchange Specification (IGES) Version 5.2, an American National Standard, US PRO/IPO-100, November, 1993. Available from the National Computer Graphics Association (NCGA) Technical Services and Standards, 2722 Merilee Drive Suite 200, Fairfax, Virginia 22031.

2.4 Informative References

5. Smith, B., Wellington, J., The Initial Graphics Exchange Specification (IGES) Version 3.0, NBSIR 86-3359, U.S. National Bureau of Standards, April 1986. Available from the National Technical Information Service (NTIS), Fairfax, VA 22031 as PB86-199759.
6. Smith, B., Rinaudot, G., Wright, T.; Reed, K., The Initial Graphics Exchange Specification (IGES) Version 4.0, NBSIR 88-3813, U.S. National Bureau of Standards, June 1988. Available from the National Technical Information Service (NTIS), Fairfax, VA 22031 as PB88-235452.
7. Reed, K., Harrod Jr., D., Conroy, W., The Initial Graphics Exchange Specification (IGES) Version 5.0, NISTIR 4412, National Institute of Standards & Technology, September 1990. Available from the National Computer Graphics Association (NCGA) Technical Services and Standards, 2722 Merilee Drive Suite 200, Fairfax, Virginia 22031.
8. Reed, K.; Kelly, J.C., Harrod Jr., D., Conroy, W., The Initial Graphics Exchange Specification (IGES) Version 5.1, US PRO/IPO, September, 1991. Available from the National Computer Graphics Association (NCGA) Technical Services and Standards, 2722 Merilee Drive Suite 200, Fairfax, Virginia 22031.
9. O'Connell, L., "Guide to the IGES Electrical Entities," EACP-2.3, unpublished, June 24, 1989, Available from IPO Office, National Institute of Standards and Technology, Bldg. 220, Room A-127, Gaithersburg, Maryland 20899
10. Harrison, R.J. and Palmer, M. E., "Guidelines for the Specification and Validation of IGES Application Protocols" NISTIR 88-3646, January 1989. Available from the National Technical Information Service (NTIS), Springfield, VA 22161, October 1990, (110 pages).
11. "Integration Definition for Information Modeling (IDEF1X)," FIPS PUB 184, National Institute of Standards and Technology, Computer Systems Laboratory, Gaithersburg, Maryland 20899, December 21, 1993.
12. "Integration Definition for Function Modeling (IDEF0)," FIPS PUB 183, National Institute of Standards and Technology, Computer Systems Laboratory, Gaithersburg, Maryland 20899, December 21, 1993.
13. Parks, C., "Tutorial: Reading and Reviewing the Common Schema for Electrical Design and Analysis," Proceedings of the 24th Design Automation Conference; IEEE & ACM, June 30, 1987, paper 27.2.
14. Loomis, M., "Data Modeling - The IDEF1X Technique," Proceedings of the IEEE 1986 Phoenix Conference on Computers and Communications, Phoenix Arizona, page 146-149.
15. Tsichritzis, D.; Klug, A.; *The ANSI/X3/SPARC DBMS Framework Report of the Study Group on Database Management Systems*, X3 project 226, 1977, AFIPS Press, 210 Summit Avenue, Montvale, NJ 07645
16. Palmer, M.E. and Reed, K.A., "3D Piping IGES Application Protocol, Version 1.0," NISTIR 4420 (257 pages), October 1990, Available from the National Technical Information Service (NTIS), Springfield, VA 22161.
17. ISO/IEC 10641-1993(E) *Information Technology - Computer Graphics and Image Processing - Conformance Testing of Implementations of Graphics Standards*. First Edition, 31 pp., CH-1211 Genève 20, Switzerland.

3. DEFINITIONS

Section 3.1 contains those definitions which apply to the notion of an application protocol; they are included as an assist in understanding the content of this document. Section 3.2 combines the definitions of the functions (in Appendix A), the entities (in Section 4.2.3), and the objects (in Section 5).

3.1. IGES Application Protocol Definitions

APPLICATION: An enterprise process that puts product data to use. The scope of an application is defined by the class of product, the supported stages in the life cycle of the product, the uses of the product data, and the disciplines that participate in that use.

APPLICATION ACTIVITY MODEL (AAM) - A structured decomposition of the tasks which are followed to achieve the particular product defined by the AP. The AAM identifies the context of the AP as well as the activities which create and use the information being defined. The AAM is constructed to be generally representative of the industry, and includes processes which are, or could be, largely automated.

APPLICATION INTERPRETED MODEL (AIM) - An information model that identifies the information structures of a particular data format specification. The AIM is defined to accomplish a physical implementation of an associated application reference model (ARM). This AIM is prepared at a level of abstraction that is sufficient for selecting the necessary entities for an application.

APPLICATION PROTOCOL (AP): Defines the scope and information domain of an application and specifies the rules for using IGES, or some other standard, to enable unambiguous transfer of the application information, and the testing of those transfer implementations.

APPLICATION REFERENCE MODEL (ARM) - An information model that describes the information structures and constraints for an application area. The ARM is defined in terms of Entities, Attributes, and Relationships. The information model uses application specific terminology and rules familiar to an expert from the application area. The model is independent of any physical implementation and can be validated by an expert from the application area.

APPLICATION SUBSET - An unambiguous set of entities which span the data requirements of the specified application. The set of entities is determined on the basis of the Application Reference Model.

ASSEMBLY - In general (e.g., from Websters): Parts fitted together to make a whole. The assembly referred to in this AP is taken to mean parts placed on, and/or connected to, an interconnecting part generally in the form of a flat substrate on which has been deposited alternate layers of metalized patterns and insulation material; the whole of which has been defined as to provide a specified part of the function of a system. An assembly may be used as a part on another assembly.

ATTRIBUTE - A property or element of information associated with an Entity. An instance of the attribute must exist for any instance of the Entity to which it belongs.

CONFORMANCE - (Ref. 17) Adherence of an implementation to the requirements of one or more specific standards or technical specifications.

ENTITY - The basic unit of data classes in the ARM. Also the basic data types identified in the

AIM. The term applies to single units which may be individual elements of geometry, individual elements of annotation, or collections of geometry or annotation elements that are combined to form more complex data structures.

IGES POSTPROCESSOR - A software unit that transfers CAD information from the IGES format to the CAD database format of a particular system. The software is usually developed and maintained by a commercial CAD system vendor.

IGES PREPROCESSOR - A software unit that translates CAD information from the CAD database format of a particular CAD system to the IGES format. The software unit is usually developed and maintained by a commercial CAD system vendor.

INFORMATION - In mathematical communication theory[1], a measure (strongly related to entropy) of the freedom of choice available when selecting a message to be sent. In the context of this AP, each message consists of graphics, or numeric quantities, or text strings, or various combinations of these three forms of data. The messages—in a particular data format—are augmented by definitions, usages, and context(s) supplied in separately communicated semantic proscriptions (e.g., IGES and this AP). Messages are required to transmit data accurately, to convey meaning precisely, and to prompt appropriate reactions at the destination by a human or a computer.

INFORMATION CONFIGURATION CONTROL - An approach that consists of specifying, documenting, and controlling both the creation and modification of information and the subsequent translation and exchange of the information between different systems and formats. The approach requires substantial documentation for both the syntax (the format) and the semantics (the meaning) attached to an item of information.

INSTANCE - In general (e.g., from Websters): A detail or circumstance. In this AP an instance indicates that a specific member of an object class is being referenced. An example of an object class may be the assembly as identified by a drawing number. An example of an instance may be a completed assembly identified by a serial number.

LAYERED ELECTRICAL PRODUCT (LEP) - A broad category of electronic product types which are characterized by a two or more laminar strata stacked in sequential order. The completed structure serves to conduct electrical signals and may contain passive or active electrical functional elements. The structure is usually designed using two-dimension (2-D) graphics which result in photo-process tooling. Additional components or other LEPs may be added to complete the assembly. Descriptions within this document which apply additionally to products such as integrated circuits, printed circuit boards, flex harnesses are labeled with this term.

OCCURRENCE - In general (e.g., from Websters): Anything that happens or takes place. In this AP the term is sometimes used to label an object class which exists as a result of a specific grouping of two or more other object classes.

PRODUCT - A result produced by specified activities or used for specified activities.

PRODUCT DATA - The set of data elements that is necessary to fully support a product over its expected life cycle. The set of data elements includes all of the product definition data plus other

---

1. See especially Shannon, C.E., Weaver, W., *The Mathematical Theory of Communication*, Urbana, University of Illinois Press, 1949. Shannon's work includes insights on both semantics and behavior.

data pertaining to the operation and maintenance of the product until it is removed from service.

PRODUCT DEFINITION DATA - The set of data elements that completely define a product for a certain discipline; a subset of product data. This set of data elements includes the geometry, topology, features, tolerances, and relationships necessary to completely define a component product or an assembly of products and is structured such that it can be used by one or more applications.

RELATIONSHIP - The identified association of two entities to each other. The two entities and their relationship taken together constitute a natural language *role* expressed in the data model. Example: Assuming the entities "Car" and "Person" are associated with relationship "Owned" in a data model Then the role may be expressed in natural language as: "A Car is owned by a person." The exact role stated depends on the modeling language's interpretation specifications.

SEMANTICS - The meaning that is given or assigned to an item of information. The meaning is assigned to an item of information on the basis of its application area.

SYNTAX - The structure of expressions in a language. [3] This structure is described in a specification such as IGES.

UNIT of FUNCTIONALITY - A collection of entities, attributes and relationships that conveys one or more well-defined concepts (within the scope of the AP) such that removal of any component would render the concept(s) incomplete or ambiguous. (See "Current Issues on STEP APs," 10/9/91.)

3.2. LEP Application Definitions

The bracketed character following the term indicates that the definition applies primarily to the AAM [A], AIM [I], or the ARM [R].

Note: definitions preceded by (T50) are from IPC-T-50E, December 1991.

ANNOTATION - [R] Text or legend pertinent to a design; text may appear as data related to legend on a multilayer substrate, lettering on a drawing, or other types of symbols.

APERTURE CODE - [R] Code representing aperture number used in plot file intended for a plotter such as a Gerber Plotter.

APERTURE NUMBER - The number for a photoplotter aperture setting, number used in pin and pad tables, and formatter table corresponding to a position on the aperture wheel.

APERTURE SHAPE - [R] Shape of aperture, such as one of the following: round, square, donut.

APERTURE SIZE - [R] The x,y dimensions of the aperture such as width or diameter where appropriate.

ART WORK (T50) - [R] An accurately-scaled configuration that is used to produce the artwork master used for production.

ART WORK GEOMETRY (target) (outside circuit) - [R] A collection of primitive shapes such as arcs, areas, and lines.

ART WORK TEXT - [R] Identifying text which calls out the product number and configuration number as well as program configuration for the art work. It identifies layer, revision, product number.

ASPECT RATIO (of Deposition Resistor) - [R] The ratio of the length of a thick or thin film resistor to its width, or the ratio between the resistance of the resistor and the sheet resistivity of the ink. This is also the number of effective "squares" in the design of a resistor.

ASSEMBLY CONSUMABLES - [R] A type of consumable that becomes part of the finished product. See also Consumable.

ASSEMBLY ID - [R] All information needed to uniquely identify a particular assembly, manufactured in-house or purchased. The identification may include, but is not limited to, the Drawing Number, Revision, Revision Status, Dash Number, Cage Number, case style, and other product notation.

ASSEMBLY OCCURRENCE - [R] Either a single product which is defined to become part of an assembly, or two products which are identified as items to be assembled into the next higher assembly. The occurrence identified may be noted in a bill of materials, or may be in-process work which must be identified for manufacturing tracking.

ASSEMBLY OCCURRENCE ID - [R] All information required to uniquely identify (ID) a product and the assembly it is assembled to. The identification may include, but is not limited to, the Drawing Number, Revision, Revision Status, Dash Number, Cage Number, case style, and other product notation.

ASSEMBLY TYPE - [R] The name of the class of assembly which indicated the general stage of completion of the electronic product. Examples are screened substrate, subassembly, component assembly, and final assembly.

ATTACHMENT - [R] Method used to describe how a component can be held on to the substrate such as solder, or epoxy.

ATTACH MATERIAL PATTERN - [R] Pattern associated with a component for the purpose of specifying the location and shape of an attachment medium or material.

AUTOINSERTION - An automated process by which through-hole components are selected, grasped, and positioned into the appropriate component holes in a PCB. Bonding of the component to the PCB may employ solder paste, glue or other consumables.

BACK PAD POLARIZATION - [R] Description of which pin of a component such as a diode or IC chip is surface-mounted to an LEP. (The terminal so mounted is the *back pad*.)

BACK PAD POLARIZATION NAME - [R] Name associated with the back pad such as anode.

BEHAVIOR REQUIREMENT - [R] The performance needed from a circuit (specified through the use of a "hardware description language") which is to be or has been packaged in an LEP.

BILL OF MATERIALS - [R] A formatted list of products (instances of LEP Assembly Components) used for a particular assembly.

BLIND VIA - A via that is visible from only one side or the other of a design. (Derived from Via attributes.)

BONDING PADS - [R] Areas of metallization on the component and the LEP substrate that permits connection of the wire or circuit elements.

BURIED VIA - A via hole not extending to the surface. A buried via is completely contained within the inner layers of a design and therefore is not visible from the outside layers. May be referred to as an interstitial via hole. (Derived from Via attributes.)

CAD LAYER - [R] A method of separating information in a CAD system for the purpose of display or processing. Some CAD layers correspond to artwork layers.

CAGE NUMBER - [R] The military-assigned identification of the manufacturer responsible for the product definition.

CAPACITOR COMPONENT - [R] A kind of component used to add capacitance to a circuit.

CAPACITOR MATERIAL - [R] The type-name for a capacitor derived from the dielectric material of the capacitor

CATCH PAD - [R] Metallization patterns on a conductive layer associated with a via which aid in the interconnection of a via with its associated traces.

CHECKPLOT SCALE (T50) - [R] Drawing scale used on interim drawing for graphical data verification. (called a check plot.)

COMPONENT - A Part which has been installed in an LEP, thereby acquiring a Reference Designator and a location within the LEP. The Component does not lose its Part characteristics as described in the catalogue of parts.

COMPONENT ATTACH - [A] The placing of a component (chip) on a substrate or header in a manner which causes it to mechanically bond.

COMPONENT LOCATION PLACEMENT TOLERANCE - [R] The amount of displacement form the specified location which is acceptable.

COMPONENT LOCATION X Y - [R] The coordinates specified for the location of a component within an assembly.

COMPONENT ORIENTATION (T50) - [R] The direction in which the components on a printed board or other assembly are lined up electrically with respect to the polarity of polarized components, with respect to one another, and with respect to the board outline.

COMPONENT PLACEMENT RESTRICTION - [R] An area on substrate where components can not be placed.

COMPONENT POSITION - [R] The location at which a component is placed.

COMPONENT TOLERANCE - [R] The range of acceptable component values; usually specified as a percentage of the nominal specified value.

COMPONENT VALUE - [R] The parameter associated with some component types which may be used in the fabrication of the components.

COMPONENT VALUE TOLERANCE - [R] A range of permissible product's values.

COMPONENT VOLTAGE RATING - [R] Maximum voltage specified that can be placed across a component.

CONDUCTIVE PATTERN - [R] The configuration or design of the conductive material on the substrate. For a lep, the pattern includes conductors, lands, and through connections (vias) when these connections are an integral part of the manufacturing process.

CONDUCTIVITY, ELECTRICAL - [R] The capability of a material to carry an electrical current. The reciprocal of resistivity.

CONSUMABLE - [R] An in-process or an end item material supplied in bulk form.

CONSUMABLE ID - [R] All information needed to uniquely identify a particular consumable, manufactured in-house or purchased. The identification may include, but is not limited to, the Drawing Number, Revision, Revision Status, Dash Number, Cage Number, and other material notation.

CONSUMABLE NAME - [R] The name given to bulk-purchase materials used in the assembly process, such as solder, flux, etc.

CONSUMABLE PROPERTY - [R] Properties of a consumable such as shelf life, expiration date.

CONSUMABLE PROPERTY NAME - [R] The name of a Consumable Property.

CONSUMABLE PROPERTY VALUE - [R] The measurement value associated with a Consumable Property.

CONSUMABLE TYPE - [R] The generic classification given to consumables, such as epoxy.

CROSSOVER - The transverse crossing of metallization paths without mutual electrical contact achieved by the deposition of an insulating layer between the conducting paths at the area of crossing. Dielectrics can be used with crossover conductive patterns, acting as a bridge over previously screened conductive material. (derived from trace geometry.)

DEFAULT DESIGN RULE - [R] The specifications which are followed during layout of the patterns on the layers of an electronic assembly. Examples are the maximum trace width, or the minimum conductor spacing.

DEFAULT DESIGN RULE DESCRIPTION - [R] The description of a Design Rule that is employed when no other rule is specified.

DEFAULT DESIGN RULE NAME - [R] The name associated with a Design Rule.

DEPOSITION COMPONENT - [R] Components which are fabricated coincidentally with the interconnects as opposed to components which are attached to the substrate after the interconnect network is complete.

DEPOSITION RESISTOR - [R] The deposition of resistor material which is formed as part of the conductive circuitry on a substrate layer. The resistor is a component in the sense of the electrical function of the circuit, but not in the sense of a separate product to be assembled to the completed substrate.

DEPOSITION SEQUENCE NUMBER - [R] A particular layer's position on the LEP substrate with respect to the first layer.

DEPOSITION TYPE - [R] A category used to describe the technology used (THICK or THIN film) to create the component.

DERATING - [R] A factor which is applied to a product parameter to insure that reliability requirements are met; such as derate resistor power rating by 50%.

DESIGN RULE - [R] A guideline that determines behavior with respect to specified design parameters. Becomes a constraint on the functional and physical layout. An example is a rule that ensures that minimum distance between elements in a design file is maintained.

DESIGN RULE CHECK (DRC) - [R] A process to enforce the design rules.

DESIGN RULE VALUE - [R] The measurement assigned to a particular restriction for a product's design.

DEVICE PIN ASSOCIATIVITY- [R] Functional relationships among pins that are needed to allow operations such as gate swapping or pin swapping to improve trace routing

DIE - [A] The uncased and normally leadless form of an electronic component that is either active or passive, discrete active device or integrated circuit.

DIE BONDING - [A] The attachment of an integrated circuit chip to a substrate or header.

DIELECTRIC - [R] An insulating material used to separate conductors.

DIELECTRIC CONSTANT - The ratio of the capacitance, Cx, of a given configuration of electrodes with a specified dielectric, to the capacitance Cv, of the same electrode configuration having a vacuum as a dielectric.

DIELECTRIC LOSS ANGLE - [R] The difference between 90 degrees and the dielectric phase angle given a set of operating conditions. May be referred to as the dielectric phase difference.

DIELECTRIC OUTLINE - [R] The boundary of an insulating layer.

DIELECTRIC STRENGTH - The maximum voltage that a dielectric can withstand, under specified conditions, without resulting in a voltage breakdown. Usually expressed as volts per unit length.

DIGITAL LOGIC BEHAVIOR - [R] Performance of circuit elements having two known stable states of interest; a type of Behavior Requirement.

DISSIPATION - [R] The tendency for fluids or energy to distribute evenly within a medium. This tendency is usually associated with the ability of a material shape to remove heat or power from physical devices and circuits.

DISSIPATION FACTOR - [R] A measure of insulating or dielectric materials to absorb some of the energy in an alternating current signal.

DOCUMENTATION TEXT - [R] Words (strings) which exists on documentation.

DRAWING NUMBER -[R] The unique identification of a drawing within a manufacturing operation.

EDGE CONNECTOR - (from IPC: 'Edge-Board Connector' -) A patterned group of conductive terminals that is used specifically for making interconnections between the LEP traces and the next higher assembly or among regions of the LEP. An edge connector may be a set of PCB features or a discrete component bonded to the PCB. Not all discrete connectors are mounted at the edge of the PCB (i.e. within a few trace widths of the PCB outline).

EE COMPONENT - (from AP 210 3.4.2) An occurrence of a part having electrical characteristics used in an assembled layered electrical product. Each EE Component has one or more circuit connections and a unique reference designator.

EE CONNECTIVITY - (from AP 210 3.4.6) The flow or association of EE_Signal information, or energy between objects in an electrical product. Connectivity focuses more on the logical need for a 'link' between two or among many terminals. The physical 'join' guides the packaging and testing of the physical conductors providing the required connectivity.

EQUIPMENT - [R] The identification name of machine types that are used as the means to complete a process step during manufacture of a device. Examples are a pick-and-place machine or a solder-joint inspection machine.

ELECTRONIC PACKAGE - [R] (See Package.)

FIDUCIAL [I] - See Registration Mark

FILE ID - [R] The unique identification of a data file stored in or intended for a computer system. For files containing product information, the identification may include, but is not limited to, the Drawing Number, Revision, Revision Status, Dash Number, Cage Number, and other product notation.

FILM - (See Thick Film or Thin Film.)

FILM MATERIAL - [R] Material used in thin and thick film processes to fabricate components and circuitry.

FLEX CABLE - A type of LEP which conducts one or more signals between attached devices (e.g., connectors) which is fabricated on non-ridged substrate material.

FLEX CIRCUIT - A type of LEP which is fabricated on non-ridged material.

FOOTPRINT - (from IPC: 'Land Pattern' -) A pattern of lands and associated features that defines the allowable interconnection space for the mounting, interconnection, and possibly the testing of a particular component. The lands in the footprint provide part of the connectivity between the substrate conductors and the component terminals (or cell ports).

FUNCTION TYPE - [R] The category describing the use of a particular artwork geometry such as Pad, Registration Mark, Conductor, Dielectric Edge, Text.

GENERIC NAME - [R] A commonly accepted product identification, such as 54LS04 or 2N2222.

GEOMETRY - [R] The constructs which are used to describe shapes of objects to be presented, or to which a product is to be fabricated. The description is usually in terms of a set of spatial coordinates and the identification of the curve-type to be constructed from the coordinates.

GEOMETRY ELEMENT - [R] A feature used to identify the meaning of, or use for, a collection of geometries. Examples from assemblies are countersink, keyway or threads. Examples from electrical planar designs are line, circle, area, etc.

GEOMETRY LINE WIDTH - [R] A measurement of the distance across a line (i.e., normal to the line). This value is often specified for circuitry artwork lines to regulate or check the actual or the artwork image produced from a computer plot or other image master.

GEOMETRY TYPE - [R] A classification of the basic geometry such as Line, Arc, and Polygon.

GLAZE - [R] An insulating film material used to protect underlying circuitry.

HEAT SINK - (IPC: 'Heatsink' -) A mechanical device that is made of a high thermal-conductivity and low specific-heat material that dissipates heat generated by a component or assembly.

HYBRID MICROCIRCUIT ASSEMBLY (HMA) - In general an integrated circuit that is not monolithic. The term hybrid denotes that the circuit elements are made by two or more different technologies. A typical hybrid may consist of semiconductor chips and capacitors attached to a ceramic substrate with printed resistors and interconnections which are screen printed and fired.

INTEGRATED CIRCUIT (IC) - A type of LEP. An electronic part which has been fabricated upon a monolythic semiconductor substrate.

INTELLIGENCE - A capability to support queries about relationships among and values of uniquely specified layered electrical product data.

INTERNAL PART NUMBER - A part identifier unique within the assigning enterprise to enable distinctions based on the business needs of the enterprise.

JUMPER - A conductor expressly connected to a particular pair of circuit access locations to provide required connectivity.

KEEPIN - A design rule determining the definition of an area or volume within which specified features are to be contained.

KEEPOUT - A design rule determining the definition of an area or volume from which specified features are to be excluded.

LAND - [R] A portion of a conductive pattern (also called Pad by some) usually used for electrical connection, component attachment, or both, or a Via feature to one or more traces on the associated layer of the LEP.

LAYER - A strata of material such as conductor, dielectric, or resistor material which forms part of the LEP structure.

LAYER DISPLAY COLOR - [R] The display color of a layer on a color drawing or on a CAD machine.

LAYER NAME - [R] The name of a particular layer (dielectric or conductive) such as Glaze or top conductor.

LAYER SEQUENCE NUMBER - [R] A particular layer of material (conductor or dielectric) position on the LEP substrate with respect to the first layer. Usually begins with the first material placed on a substrate. These layers relate to process layers as contrasted with e.g., CAD layer.

LAYER TYPE - [R] A category used to describe the function performed by a deposited layer such as Conductor, Dielectric, Fill, Glaze, Resistor, Epoxy, Solder.

LAYER RESTRICTION - [R] A design restriction which is imposed on a particular layer of the product.

LEAD FRAME - [R] The metallic portion of the device package that completes the electrical connection path from the die or substrate conductors to the external circuitry.

LEAKAGE CURRENT - [R] The flow of electrons through the dielectric of a capacitor, usually expressed as a maximum which will be acceptable for the component.

LEP - [R] See Layered Electrical Product (Section 3.1).

LEP ASSEMBLY COMPONENT - [R] A product or an assembly which has been identified and located within an assembly.

LEP CAD PRESENTATION - [R] The design file that contains the electronically coded data available on a particular design.

LEP ID - [R] All information needed to uniquely identify a particular product, manufactured in-house or purchased. The identification may include, but is not limited to, the Drawing Number, Revision, Revision Status, Dash Number, Cage Number, case style, and other product notation.

LEP NAME - [R] A descriptive name given to a product that usually describes its function.

LEP NUMBER - [R] The product number of a particular type.

LEP SUBSTITUTE PART NUMBER - [R] Other organizations identification (i.e., the customer

may have alternate product number for the same product).

LEP VERSION - [R] The revision level of a product; each change to the product design results in a new product Version.

LEVEL OF PREFERENCE - [R] A relative rating assigned to a product in a collection of functionally similar products which indicates its ranking of desirability for use in a product to be produced for the rating organization.

LIBRARY NAME - [R] The name of part in CAD system files of part information.

LINE RESISTANCE - The resistance of a conductor line on a substrate measured in total trace resistance.

LINE WIDTH - [R] (See Geometry Line Width.)

LOG REQUIREMENTS - [R] Indicates what information about the process step must be recorded and retained such as Machine ID, Component Lot Identification, Start Time, Stop Time.

LOSS FACTOR - [R] A property of an insulating material that is equal to the product of its dissipation and dielectric constant.

MANUFACTURER NAME - [R] The name a company is registered to do business under; may also include a company division name.

MAX SUBSTRATE SIZE - [R] The largest size of substrate which can be accommodated.

MIL - [R] A unit equal to 0.0254 millimeter (mm), (or 0.001 in).

MOUNTING_HARDWARE - Fasteners applied to attach one or more PCA_component occurrence(s) to the PCB or the PCB to its next assembly.

MOUNTING PAD - [R] Metallization pattern on the substrate associated with a component for the purpose of indicating the region for physical attachment of the component to the substrate. See Attach Material Pattern.

MULTICHIP MODULE -[A] A complex multilayer circuit containing components (integrated circuit chips) which cover at least half of the available circuit area. Note that this term is not present in the CAD information.

MULTI-UP ID - [R] The unique identification of data which indicates the placement of an array of images nested together on a single photo tool artwork. For files containing product information, the identification may include, but is not limited to, the Drawing Number, Revision, Revision Status, Dash Number, Cage Number, and other product notation.

NET - An entire sequence of electrical connections from the first source point to the last target point, including lands and vias. The Net is identified by a Net Name which is unique within the product, and may also have a signal name.

NET DESIGN RULE - [R] Specified constraints on parameters related to the connectivity. Examples are maximum distributed inductance, or maximum circuit length.

NET DESIGN RULE NAME - [R] The assigned identification of a specification for some parameter of all conductive elements associated with an individual signal.

NET DESIGN RULE VALUE - [R] The measurement allocated to some parameter of all conductive elements associated with an individual signal.

NETLIST - A to-from listing of all the interconnections in the design. The netlist should correspond to the to-from listing of all the wires in the schematic. (Model; set of Net.)

NET NAME - A character string used to identify and distinguish nets; the name of the logical connection realized by the trace.

OBSTRUCTION - A physical entity which restricts where a component or a trace can be placed. See also Component Placement Restriction.

OPERATION VALUE - [R] A parameter associated with a Process Operation.

ORIENTATION - [R] The angle of rotation of a component relative to a predetermined direction.

PACKAGE - An integrated circuit, or a LEP circuit. It provides hermetic or nonhermetic protection, determines the form factor, and serves as the first-level interconnection externally for the device by means of package terminals.

PACKAGE DIMENSION - The physical measurements of the package. See Part Outline.

PACKAGE ID - [R] The unique identification of a component enclosure. For files containing product information, The identification may include, but is not limited to, the Drawing Number, Revision, Revision Status, Dash Number, Cage Number, case style, and other product notation (e.g., TO-18; a JEDIC Standard number for a metal transistor package).

PACKAGE PIN - [R] A Pin which is associated with a Part or Component.

PACKAGE OUTLINE - [R] The 3-dimensional curve and surface geometry of a component package.

PACKAGE PIN LOCATION - [R] The geometric position of a lead in a part package of an electrical product with respect to the part origin and reference axis.

PACKAGE PIN NAME - [R] A referencing word that describes a use for the lead of a component package; such as "Output1".

PACKAGE PIN NUMBER - [R] The identifying number assigned to the lead of a package for electronic components.

PACKAGE PIN TYPE - [R] The name given to the lead of a component package which indicates how the lead is to be attached to the assembly.

PAD - The metallized area on a substrate or on the face of an integrated circuit used for making electrical connections. Also see land.

PAD END FEATURE - [R] The type of joint formed on a wirebond wire at the place where the wire attaches to the circuitry of an assembly.

PAD TYPE - [R] The descriptive name that implies the function of a pad.

PADSTACK [I] - An ECAD construct for all of the pieces of information (features) about a component pin needed to establish the interconnection /layer geometry on the LEP.

PAD WIREBOND END FEATURE - [R] The method of attaching a wirebond to a pad as denoted by the appearance of the joint such as wedge, ball.

PANEL MULTI-UP (PANEL) - [R] A fabrication artifact (often with test coupons) intended to contain one or more printed board details. A panel may be defined to realize process efficiencies or to minimize wasted material. I.e., multiple of the same image on one piece of material that will

be separated after processing into multiple substrates

PART - [A, R, I] An individual functional element in a physically independent body that cannot be reduced or divided without destroying its stated function. When assembled onto the completed film network will become a Component of the circuit assembly. In a CAD system, represented as a graphic symbol with attached annotations. Parts are manufactured items; the information about which is usually stored in a library or listed in a catalogue.

PART GRAPHIC- [R] The graphic presentation of a component.

PART ID - [R] All information needed to uniquely identify a particular item, manufactured in-house or purchased. The identification may include, but is not limited to, the Drawing Number, Revision, Revision Status, Dash Number, Cage Number, case style, and other product notation.

PART KIND - [R] A category of component describing how it is used electrically such as mechanical, passive, active.

PART MANUFACTURERS NAME - [R] The corporate name used by the manufacturer of components.

PART NAME - [R] Description of the function of the component such as Coil, Cover, IC Chip, Transistor, Resistor Header, Substrate.

PART OUTLINE - [R] The boundary of component on the top assembly drawing or the CAD equivalent.

PART PROPERTY - [R] A pair of terms, one indicating the property name (see Part Property Name), the other the value of the property (see Part Property Value).

PART PROPERTY NAME - [R] The name of a measured parameter associated with a device used in an assembly. Examples are capacitance, inductance, gain, resistance, and temperature range.

PART PROPERTY VALUE - [R] The value associated with a Component Property. Examples would be 10K as a value for the resistance property, or 50 as a value for the low frequency gain property of a transistor.

PART SELECTION - [R] A set one or more alternate chips to be specified in the design of an assembly, one of which is selected at the time of manufacture of a particular product.

PASSIVE COMPONENT - [R] A component which does not exhibit signal gain but does affect one or more other parameters of a signal. Examples are resistor, inductor, and capacitor.

PATH VERTEX - The X, Y location which specifies one end point of the linear segment of a trace.

PATTERN [R]- The configuration of materials on a panel or printed board layer. Pattern also denotes the circuit configuration on related tools, drawings, and artwork masters.

PATTERN ID - [R] The unique identification for the artwork for a specified layer of a laminated product.

PATTERN VALUE - [R] The measurement value assigned to a deposited or etched component at some specified stage in the fabrication cycle. An example is 2300 ohms prior to trimming.

PATTERN VALUE MIN & MAX - [R] The tolerance associated to the deposited or etched component.

PHOTOLITHOGRAPHY [A]- The use of light or ultraviolet rays through a mask to define circuit conductor patterns. (Model; intractable taxonomy.)

PHOTOPLOT APERTURE- Mechanism which allow a certain amount and incidence of light to expose film, thereby making a "photograph" of design being plotted. See Aperture Size.

PHOTORESIST [A]- A photosensitive coating that is applied to a laminate and subsequently exposed through a photo tool (film) and developed to create a pattern that can be either plated or etched.

PHYSICAL LAYER (1 top) - A conductor, dielectric, or fill for each layer.

PICK/PLACE - An automatic process by which SMT parts are selected, grasped, and positioned onto an LEP.

PIN - The identification of a possible connection to a component providing a reference point for creating a netlist.

PIN ASSIGNMENT - [R] The name of the characteristic or electronic function assigned to the lead of a component package in a circuit.

PIN END FEATURE - [R] The type of joint formed on a wirebond wire at the place where the wire attaches to a component.

PIN LOCATION - The physical location of pins on a component package with reference to its assembly origin.

PIN NAME - [R] The general functional notation for the signal role at the lead of a component package such as E, B, C on a transistor.

PIN NUMBER - A number used to distinguish pins on a component or package.

PIN TYPE - The name of a characteristic on a pin, such as input, output, tri-state analog.

PIN WIREBOND END FEATURE - [R] The characteristic of the wire at a joint with the material to which the wire is attached. The usual characteristics given are wedge or ball.

PLACEMENT NAME/PHYSICAL NAME - The name of component in CAD library.

PLACEMENT TOLERANCE - [R] The permitted positional variation of a component on an assembly.

PLANE - A portion of a roughly planar physical conductive layer of an LEP assigned a particular function or signal implementation, i.e. Ground_Plane, Signal_Plane, and Power Plane.

PLATING - (1) The metallic deposit on a surface, formed either chemically or electrochemically. (2) the process of the chemical or electrochemical deposition of metal on a surface. (Model; Deposition taxonomy.)

POINT - [R] A singular location in a given coordinate system.

POINT X VALUE - [R] The numeric value in one (usually horizontal when viewed from a reference coordinate system) direction from a single point spatial reference.

POINT Y VALUE - [R] The numeric value of a single point reference, in a direction orthogonal to the x vector direction.

PORT ID - [R] All information needed to uniquely identify a external terminal of a Component or a product. The identification may be a number or string unique within the Part or Component. For

the identification to be unique within the product, the ID of the parent Component or Part will be concatenated, usually as a prefix.

PORT DECLARATION - [R] The boolean notation describing the behavior at a Port of a digital logic device.

POWER DENSITY - [R] A measure of the concentration of energy, such as 1 watt per square inch.

POWER DISSIPATION - (T50) [R] In watts/meter^2 (heat flux density), the energy used by an electronic device in the performance of its function.

POWER PLANE - A conductive area or set of closed areas on a particular physical layer of an LEP assigned the function of distributing supply voltage to the appropriate ports of components of the LEP.

PREFORM - [R] For soldering or adhesion functions, a form which is punched out of thin sheets of solder, epoxy, or eutectic alloy. The form is placed on the spot of attachment by soldering or by bonding prior to placing the object there to be heated. (a type of Part Name)

PRIMITIVE TYPE - [R] The category of simple geometric element that makes up artwork such as geometry or text.

PRINTED CIRCUIT ASSEMBLY - A PCB together with attached components.

PRINTED CIRCUIT BOARD - The Printed Circuit Board (PCB) is a processed substrate which provides most of the electrical connections among printed and packaged components. The PCB may be single-sided, double- sided, or multi-layered and may be rigid, flexible, or rigid-flex.

PRINTED WIRING ASSEMBLY - see Printed Circuit Assembly.

PRINTED WIRING BOARD - see Printed Circuit Board

PROBE PAD - A connection point for an external probe used to test some aspect of the LEP circuit function. For example, a probe pad may be used to make connections to a screened resistor during the trimming process.

PROCEDURE - [R] A step-by-step description of the activities needed to complete a given operation.

PROCESS ID - [R] The unique identification of the parameters used on one process step during manufacturing. The identification may include, but is not limited to, the Drawing Number, Revision, Revision Status, Dash Number, Cage Number, and other product notation.

PROCESSING CONSUMABLE - [R] Material or chemicals which are used in manufacturing processes which do not remain as part of the product being manufactured. Examples include cleaning fluids, flux, and photoresists.

PROCESS DESCRIPTION - [R] The narrative explanation of an identified step in the fabrication of a product or assembly.

PROCESS OPERATION - [R] The procedures sequentially described by a document identified by the Process ID.

PROCESS STEP - [R] A single operation used to manufacture an Assembly Occurrence. Each step corresponds to one block on a flow chart.

PROCESS STEP NUMBER - [R] A sequential number assigned to a process step.

PROCESS VARIABLE NAME - [R] The identifier assigned to a particular fabrication step.

PROCESSING CONSUMABLE NAME - [R] The generic identification of a substance used during the fabrication of a product or assembly, but which does not remain as part of the product. Examples are flux, solvent, or resist.

PRODUCT CONSUMABLE - [R] Material or chemicals which are used in manufacturing processes and which remain as part of the product being manufactured. Examples include solder, epoxy, and plated metals.

PRODUCT CONSUMABLE NAME - [R] The generic name of material or chemicals which is used in manufacturing processes and which remain as part of the product being manufactured.

PRODUCT TOLERANCE - [R] The allowable deviation from the specified location of a product feature.

PRODUCTION TOLERANCE - [R] The allowable deviation from the specified location of a product feature.

PROGRAM - [R] A text string indicating the system a LEP was designed for or intended to be used in.

QUALIFIED VENDOR - [R] A company from which components are purchased and which has been identified as meeting specified requirements for certification and/or inspection of the components being sold.

REFERENCE DESIGNATOR - [R] A character string which defines an instance of an assembly component; unique within the product or system. The guidelines generally followed are found in ANSI Y32.16-1975 (e.g., R1=Resistor 1, U7=Microcircuit 7, A10= Assembly 10; thus A10R6 is the 6th Resistor in Assembly 10).

REGISTRATION - The proper alignment of layers with respect to other layers or to the substrate. The accuracy of registration is measured as the concentricity or relative position to a datum.

REGISTRATION MARK - [A, R] The marks on a wafer or substrate that are used for aligning successive processing masks. Also known as alignment marks or fiducial marks.

REGISTRATION MARK NAME - [R] The term assigned to a locating orientation feature related to a particular use.

REQUIREMENT ID - [R] All information needed to uniquely identify a particular Behavior Requirement. The identification may include, but is not limited to, the Drawing Number, Revision, Revision Status, Dash Number, Cage Number, and other notation.

REQUIREMENT TYPE - [R] A categorization of Behavior Requirement.

RESISTIVITY - The ability of a material to resist the passage of electric current. See also Resistor, Sheet Resistance or Line Resistance.

RESISTOR - A device that offers resistance to the flow of electric current in accordance with Ohm's law: $R = E/I$, where R is resistance, E is voltage, and I is current.

RESISTOR GEOMETRY - The film resistor outline. The four system generated geometries for many CAD systems are as follows: rectangle, tophat, u - shaped, serpentine.

RESISTOR MAX ASPECT RATIO - A greatest-value factor obtained by dividing the specified width by the length and multiplying the numerator and denominator by a constant such that the re-

sulting factor is a simple number, such as 4:1. Such a design rule may constrain the automated design system, and is transferred with the design. The rule is not necessarily intended to be applied to the finished product.

RESISTOR MIN ASPECT RATIO - A least-value factor obtained by dividing the specified width by the length and multiplying the numerator and denominator by a constant such that the resulting factor is a simple number, such as 1:6. Such a design rule may constrain the automated design system, and is transferred with the design. The rule is not necessarily intended to be applied to the finished product.

RESISTOR MIN LENGTH - A design rule—or metric—which establishes the least acceptable dimension a resistance trace length, such as 40 mil. Such a design rule may constrain the automated design system, and is transferred with the design. The rule is not necessarily intended to be applied to the finished product.

RESISTOR MIN WIDTH - A design rule—or metric—which establishes the least acceptable dimension across a resistance trace, such as 20 mil. Such a design rule may constrain the automated design system, and is transferred with the design. The rule is not necessarily intended to be applied to the finished product.

RESISTOR OVERLAP - A thick-film conductor pad that overlaps and makes contact with a thick-film resistor area.

RESISTOR PASTE VALUE - [R] Material used to create thick film resistors rated with a nominal ohm per square value.

RESISTOR SHAPE - [R] The term which is characteristic of the topology used for a deposited or etched resistor component.

RESISTOR TERMINATION - The contact area between a thick- or thin-film resistor and an adjacent conductor layer.

REVISION - [R] Number(s) and/or character(s) that uniquely identify a version of documentation and the products built to that documentation.

ROUTE REGION - (T50) Region where routing paths can automatically be placed between points to be interconnected.

SCHEMATIC - (T50) A drawing that shows, by means of graphic symbols, the electrical connections, components, and functions of a specific circuit arrangement.

SCHEMATIC ID - [R] All information needed to uniquely identify a particular circuit functional diagram, manufactured in-house or purchased. The identification may include, but is not limited to, the Drawing Number, Revision, Revision Status, Dash Number, Cage Number, case style, and other product notation.

SCREEN ID - [R] All information needed to uniquely identify a particular screen, manufactured in-house or purchased. The identification may include, but is not limited to, the Drawing Number, Revision, Revision Status, Dash Number, Cage Number, and other product notation.

SCREEN PRINTING - [A] A process for depositing material in a pre-defined pattern by forcing material through a stencil screen with a squeegee.

SCREEN STENCIL - [A] A network of metal or fabric strands, mounted snugly on a frame, and upon which the film circuit pattern and configurations are superimposed by photographic means.

(from ISHM Industry Guide, 1990)

SHEET RESISTIVITY - [R] Electrical resistance of a thin sheet of a material with a uniform thickness as measured across opposite sides of a unit square pattern expressed in ohms per square. (from ISHM Industry Guide, 1990)

SIGNAL CONDUCTIVE PATTERN - [R] The association of geometry and the electrical characteristic that is intended to be carried on the material finished to that geometric shape.

SIGNAL NAME - [R] The user defined functional name, usually referring to an electrical impulse of predetermined voltage, current, polarity, and pulse width. See also Net Name.

SILK SCREEN - A physical layer of the LEP providing text and/or graphic annotation by means of screen printing. The resulting annotation is usually visible from the top side or from the bottom side of the LEP. (IPC: 'Screen Printing') - The transferring of an image to a surface by forcing a suitable medium with a squeegee through an imaged-screen mesh.

SMT - Surface Mount Technology (IPC: 'Surface Mounting' -) The electrical connection of components to the surface of a conductive pattern that does not utilize component holes.

SURFACE MOUNT - see SMT.

SOLDERING [A]- A process of joining metallic surfaces by melting solder.

SPECIFIED TOLERANCE - [R] The amount of (usually) dimensional deviation which has been called for in the product specification.

START LAYER - [R] The first of one or more layers through which a via or passage will pass.

STATION - [R] The place or the equipment which is identified (in planning) as the location of a processing or an assembly step during the manufacturing of a product.

STEP X COUNT - [R] The number of patterns in X direction in panel multi-up.

STEP X DISTANCE - [R] The X step distance between patterns in panel multi-up.

STEP Y COUNT - [R] Number of patterns in Y direction in panel multi-up.

STEP Y DISTANCE - [R] The Y step distance between patterns in panel multi-up.

STOP LAYER - [R] The last layer through which a via or passage will pass.

SUBSTRATE [R]- The supporting structural material into and/or upon which the passivation, metallization and circuit elements are placed. MIL-STD-883C (Usually aluminum oxide (alumina)).

SUBSTRATE OUTLINE [R]- The graphical representation of the substrate shape.

SUBSTRATE MATERIAL - [R] The name of the substance from which the base layer is formed.

SUBSTRATE THICKNESS - [R] The nominal material cross section of the base material of a laminated product.

SURFACE RESISTIVITY - [R] (See Sheet Resistivity.)

SWAPPING - [A] The process that exchanges connections from one pin to another pin to improve routing, provided the pins are in the same gate and in the same swap group.

TEMPERATURE COEFFICIENT OF CAPACITANCE - [R] A multiplier which represents the linear change of a capacitance value per degree change in temperature. It may be positive, negative, or zero and is usually expressed in parts per million per degree Celsius, or 10^{-6}.

TEMPERATURE COEFFICIENT OF RESISTANCE - [R] A multiplier which represents the linear change of a resistance value per degree change in temperature. It may be positive, negative, or zero and is usually expressed in parts per million per degree Celsius, or 10^{-6}.

TERMINATOR MOUNTING - [A] Metallization pattern on the substrate associated with a film component. The purpose of the pattern is to indicate the region for electrical connection of the film component to its connecting traces.

TEST - [R] [A] The examination of a product during or after fabrication for conformance to specified criteria for that product.

TEST CONDITION - [R] The specification of the setup needed to carry out a specific test of a product. Examples include supply voltage, temperature, and initializing signals.

TEST COUPON (IPC: -) A portion of quality conformance test circuitry that is used for a specific test, or group of related tests, in order to determine the acceptability of a product. (Test Coupons are often provided to enable detailed testing of adherence, positioning, plating, etching, and other process attributes even if some of the tests are destructive.)

TEST DESCRIPTION - [R] The narrative which is followed in performing the specified test.

TEST MAXIMUM - [R] Value of the upper limit of a range of test values.

TEST MINIMUM - [R] Value of the lower limit of a range of test values.

TEST POINT (from IPC: -) A designated point of access to an electrical circuit that is used for electrical testing purposes.

TEST SEQUENCE NUMBER - [R] A number assigned to a test in a series of tests.

TEST SYMBOL - [R] Short character sequence that allows easy reference to test such as Vin.

TEST TOLERANCE - [R] The acceptable deviation from the specified test results parameter.

TEST TYPE - [R] Category of test being performed such as Final, Pre-burn, Qual, Sample.

TEST UNITS - [R] Measurement units of the Test Minimum and Test Maximum values for the test.

TEXT - [R] An ordered sequence of characters (glyphs) to which a meaning is ascribed.

TEXT ATTRIBUTE - [R] The characteristics that describe the appearance of text such as size, font, slant, layer.

THERMAL OUTLINE - [I] The boundary of a region within which temperature is a significant packaging consideration.

THERMOCOMPRESSION BONDING - [R] The process involving the use of temperature and pressure to join two materials by interdiffusion across a boundary. (from ISHM Industry Guide, 1990)

THERMOSONIC BONDING - [R] The process involving the use of temperature, pressure and applied ultrasonic energy to join two materials by interdiffusion across a boundary. (from ISHM Industry Guide, 1990)

THICK-FILM CIRCUIT - [R] A circuit that is fabricated by the deposition of paste materials such as screen-printed ceramic material (cermet) pastes on a ceramic substrate, which are fired in a kiln to create permanent circuit patterns.

THIN-FILM CIRCUIT - [R] Conductive, resistive, or dielectric material, usually less than 50,000 angstroms in thickness, that is deposited onto a substrate by vacuum evaporation, sputtering, or other means. (from MIL-STD-883C, Notice 12)

THRU-Hole - see HOLE. ("Thru" is a spelling variant of the word "through.")

TOOL - [R] (See Tool Type.)

TOOL DESCRIPTION - [R] A narrative which indicates the use of the implement.

TOOL ID - [R] All information needed to identify uniquely a particular manufacturing implement—manufactured in-house or purchased. The identification may include, but is not limited to, the Drawing Number, Revision, Revision Status, Dash Number, Cage Number, and other product notation.

TOOL TYPE - [R] The name given to a device used, as part of product manufacturing, to accomplish an operation. Examples include the product traveler, a cutting device, or a machine.

TOOLING HOLE - (from IPC: -) A tooling feature in the form of a passage in an LEP or fabrication panel which may often be a non-plated hole for fixturing, measuring, or testing by means of the measuring feature(s) provided by the passage.

TRACE (IPC: 'Conductor' -) A single conductive path in a conductive pattern. 'Conductive Pattern' - The configuration or design of the conductive material on a base material. (This includes conductors, lands, vias, heatsinks, and passive components when these are an integral part of the printed board manufacturing process.) The role of 'join' which provides physical connections is the common characteristic here.

TRACE KEEPOUT - [R] Symbolic representation of a region (layer-specific) associated with component which may not be used for trace routing.

TRACE LAYER - [R] The physical layer on which the given trace exists.

TRACE WIDTH - [R] The physical width of the trace segment. The width data is typically used to control the photoplotter aperture or the pattern generator setting.

TRACE WIDTH TOLERANCE - [R] The allowable range of the dimension across a conductive line.

TRIM PATH - [R] The path made by trimming (using an abrasive or a laser) a component to obtain the design value.

UNITS (e.g., Volt, Meter) - [R] A definite amount or quantity used as a standard of measurement.

USER DEFINED PROPERTIES - [I] Structured data describing information that is not previously defined. This property is often used for information not yet defined for the data structure.

USER INFORMATION - [R] Unstructured text information inserted as user comments, such as drawing notes.

VACUUM DEPOSITION - The deposition of a metal film onto a substrate in a vacuum by metal evaporation techniques.

VARIABLE VALUE - [R] Quantitative data that is associated to a particular named data element.

VENDOR ADDRESS - [R] The postal location of the organization from which items are procured.

VENDOR NAME - [R] The identifying text of the company, which may include the company division, that supplied materials, equipment, or service.

VIA - An electrically conductive passage connecting two or more lands on distinctly specified layers of an LEP.

VIA HOLE - [R] A hole (i.e., a void or passage) through insulating layer (s) used to make a connection (electrical or thermal), but for which there is no intention to insert a component lead or other reinforcing material. The hole may be any shape as defined by a Via Shape.

VIA KEEP OUT - [R] The symbolic representation of a region (layer-specific) associated with a via where traces cannot exist.

VIA LAYER - [R] A layer which is used principally (in a CAD system) for placing vias holes.

VIA LOCATION - [R] The X, Y position on which a via is positioned.

VIA SHAPE - [R] The shape of the top view of a via.

VIA SIZE - [R] The width and length or the diameter of a via.

VOLTAGE ALLOWANCE - [R] A design rule that requires extra spacing between specified traces (e.g., to allow higher voltages to be carried).

VOLUME RESISTIVITY - [R] The volume resistance between two electrodes (of unit area and unit distance apart) that are in contact with or embedded in a specimen. The value is the ratio of the direct voltage applied to the electrodes in proportion to the current between them that is distributed through the volume of the specimen. Usually expressed in ohms per centimeter.

WIRE BOND - [A, R] A completed wire connection that provides electrical continuity. Bonding wires are used to connect between component pads, or package pin, or conductive pads on the LEP substrate.

WIRE BOND MAX LENGTH - [R] A design rule—or metric—to limit total length of a wirebond wire, such as 140 mils maximum.

WIRE BOND MIN LENGTH - [R] A design rule—or metric—to limit the least-length parameter of a wirebond wire, such as 10 mils minimum.

WIRE BONDING - [A] The method used to attach very fine wire to make electrical connections.

WIREBOND SEQUENCE NUMBER - [R] The number assigned to a wirebond that specifies the order in which wirebond is to be made.

WIRE SIZE - [R] The diameter of wire, such as 0.8, 1, 5 mils.

4. INFORMATION REQUIREMENTS AND APPLICATION REFERENCE MODEL

In this section, LEP information needed for the documentation and manufacture of the end item is described in the Application Reference Model (ARM). In addition, the task of identifying collections (or groupings) of information, termed Application Information Requirements, has been assembled into categories and described in section 4.1. In section 4.2.1, the methodology used to convey information and the method's implied constraints, are described. Lastly, in section 4.2.3, the results of utilizing the method to capture all of the data entities (i.e., information primitives) as they relate to each other are diagrammed.

It should be noted that information that describes the logical information structures required for accomplishing the physical implementation of an associated Application Reference Model is called an Application Interpreted Model (AIM) and is found in section 5. The indication of when the information is used during the overall design and manufacturing process is documented in the Application Activity Model (AAM) in Appendix A.

4.1 Application Data Views

Units of Functionality was an organization of data with a focus on the activities that are to be performed. A new construct, Application Information Requirements (see Section 2.1) has been created to shift the emphasis from the activities to the information needed to support those activities. In this context we are using information to mean both the data and the meaning (semantics) of that data.

4.2 Application Reference Model

The Application Reference Model was created to define the semantics (data and relationships) of all information needed for the manufacture and documentation of the end item product. The model may relate to information which is not typically captured during design. Examples are the *Test* environment (*TEXT STRING*) supplied in a procurement drawing, or the actual hand-written signatures found in the drawing title block. The model does not include data related to enterprise management or to the use of the LEP.

4.2.1 Descriptions of the ARM Data Constructs

The following sub-sections define the interpretation methods which shall be followed. These subsections are not intended to provide a tutorial on the subject. For further explanation of the syntax and methods the reader is directed to manuals, information, and tools references available from the IDEF-Users Group, 1900 Founders Drive, Kettering, OH 45420; telephone (513) 259-4702

4.2.1.1 ARM Methods and Language

The ARM is presented in the $IDEF_1X$[13] language. IDEF derives its name from the activity that developed the methodology, the Air Force Integrated Computer Aided Manufacturing (ICAM) Definition Program. The program developed several languages. $IDEF_1X$, was developed to model data—the semantics of information. A data model may be considered as a *connecting link* between the user's view of the data (referred to as the external schema) and the manner in which the data is stored in a computer, or database (the internal schema). Common to these two views of the

information is a conceptual schema and the data model is the visual presentation of the conceptual schema. In general, the data model is presented as a series of drawings that contain diagrams and accompanying text. The text forms a glossary that defines the terms used in the diagrams and more fully explains their relationships and context.

The component parts of an IDEF model are the entities, their attributes and their relationships as illustrated in figure 4-1. An entity is an item (an object or an event) about which we wish to keep data. In general, rectangles are used to graphically represent entities. If the entity is not within the scope of the present model, the outline of the entity is dashed, notifying the reader to look elsewhere for the definition of the entity. Each entity box has a name (the entity-class label) that is given above the rectangle and which is defined in the model glossary.

Attributes of an entity are shown inside the entity rectangle. An attribute will have one value

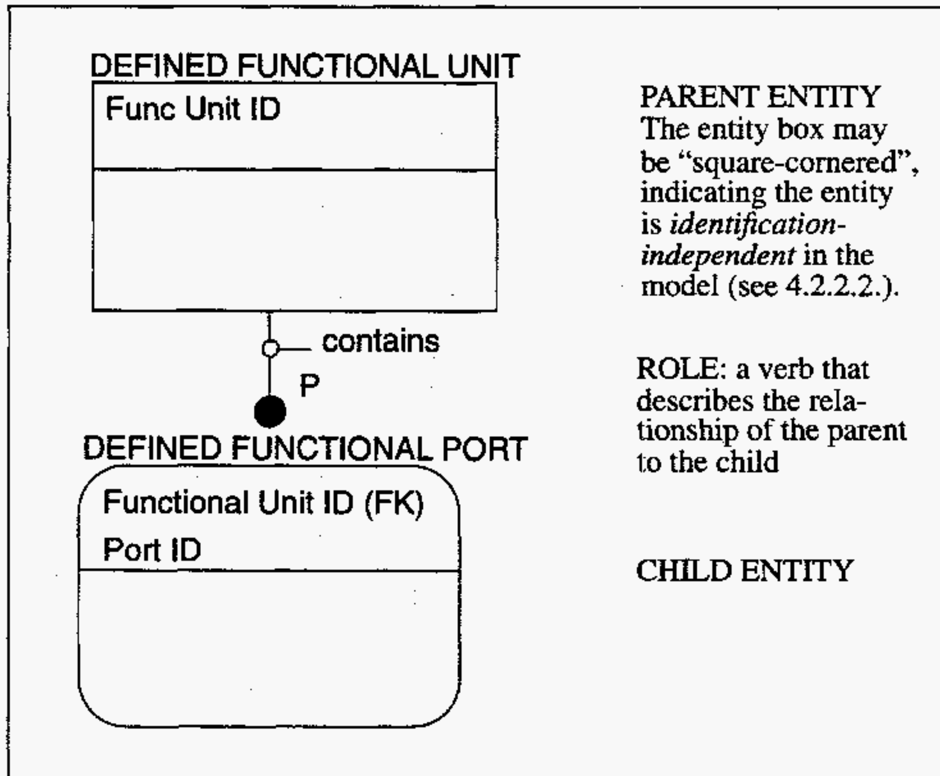


Figure 4-1. Component Parts of an IDEF Relationship.

(instance) for each instance of the subject entity. Some attributes that uniquely define the instance of an entity, called keys, are located above a line near the top of the entity rectangle. Keys are of several types and the general concepts of key migration follow those concepts as used to develop relational database structures. For example: the key attribute set from the parent entity is propagated downward through the model structure and such instances are called foreign keys. Attributes that are not keys are located below the horizontal line of the entity rectangle.

The relationships between entities are shown by the lines drawn between them in the model diagram. The relationship consists of the role and the cardinality. Text associated with the interconnecting lines provides a clear meaning (the role or predicate) of the relationship between the entities. One end of the line is connected to the entity rectangle denoting that this entity is the

independent entity. At the other end of the line—at the dependent entity end—a terminator symbol (a "big dot") may carry cardinality information on how to read the model as in figure 4.2. If there is no letter adjacent to the terminator symbol, this means that there may be "zero, one or many" entities associated with the relationship. Likewise, the letter Z designates "exactly zero or one," P designates "one or many," and the letter n designates some exact integer number of occurrences.

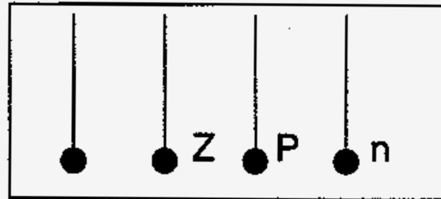


Figure 4-2. Cardinality.

Lastly, models may contain information that denote the "categorization" or types-of entities as diagramed in figure 4-3. Each instance of the generalization parent must be one of the types. A double horizontal bar indicates that all types of dependent entities are present (for example, all people are either male or female). If all of the various types of entities are not listed (the model equivalent to a written ellipsis (...)), then a single horizontal bar is used to show that there may be relationships not shown and the categorization is assumed to be incomplete.

A related pair of entities are read, that is, retrieved as two English sentences, in the form of assertions. The pair may be read in both directions. From the top down (i.e., toward the entity at the big dot end of the relation line), the set is read as follows: the independent entity becomes the subject; the relation label becomes the predicate; and the dependent entity becomes the object in the assertive sentence. From the bottom up the sentence is constructed using the dependent entity as the subject and the independent entity as the object; the usual relation is "always has exactly one."

Additional details may be obtained that will enable those who are interested in working with IDEF to become more proficient at reading the models. Several good references[15, 16] are available.

4.2.2 Data Construct Definitions

The definitions of entities for the Layered Electrical Product ARM are found in section 3.2.

4.2.2.2 Data Construct Assertions

The method and language (IDEF$_1$X) used for this AP ARM carries several assertions important to the interpretation and application of the ARM.

Attributes: All attributes which appear within the entity box exist whenever an instance of the entity occurs, unless designated by the (null OK) notation.

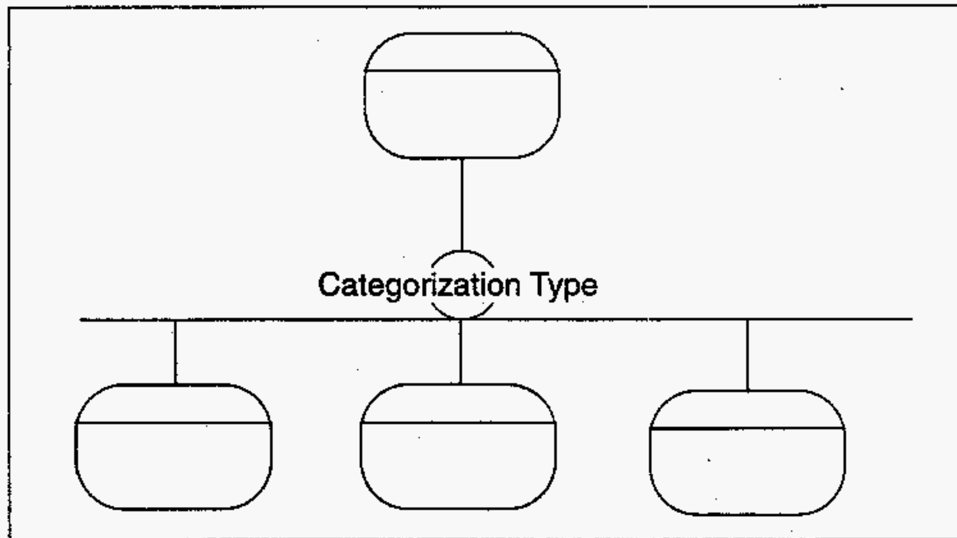


Figure 4-3. The Categorization Structure.

Relationships: The ARM is intended to be interpreted independently of any computer database structure. Thus, the entities may be considered as representing objects in an Object Oriented (OO) system, or as tuples in a Relational system. As such, the keys have not been migrated as required by relational notation. The relationships (connecting lines) may be used to define OO methods, or to migrate relational foreign keys. The entity at the *attached* end of a relationship line is considered the *parent* of the two entities. The following dependencies apply to the parent-child relation.

Existence Dependencies: The existence of a relationship (line) indicates that, should an instance of the child entity exist (at the "cardinality" end), the related parent must exist. An example of this dependency from the ARM is that: A Process Step instance cannot exist unless an Assembly Occurrence exists.

Identification Dependencies: In addition to the existence dependency above, if the relationship is a solid line, the unique identification of the child requires the key of the parent in addition to the key declared in the child. An example is the Process Step can only be uniquely identified by knowing the Assembly Occurrence to which it "belongs."

Round corners are used on an entity-boxes to indicate that there exists a parent entity which provides some of the key for identification. For entity-boxs with square corners, all of the entity's key is owned by the entity; the entity exists independently within the given domain of the model.

4.2.3 ARM Diagrams

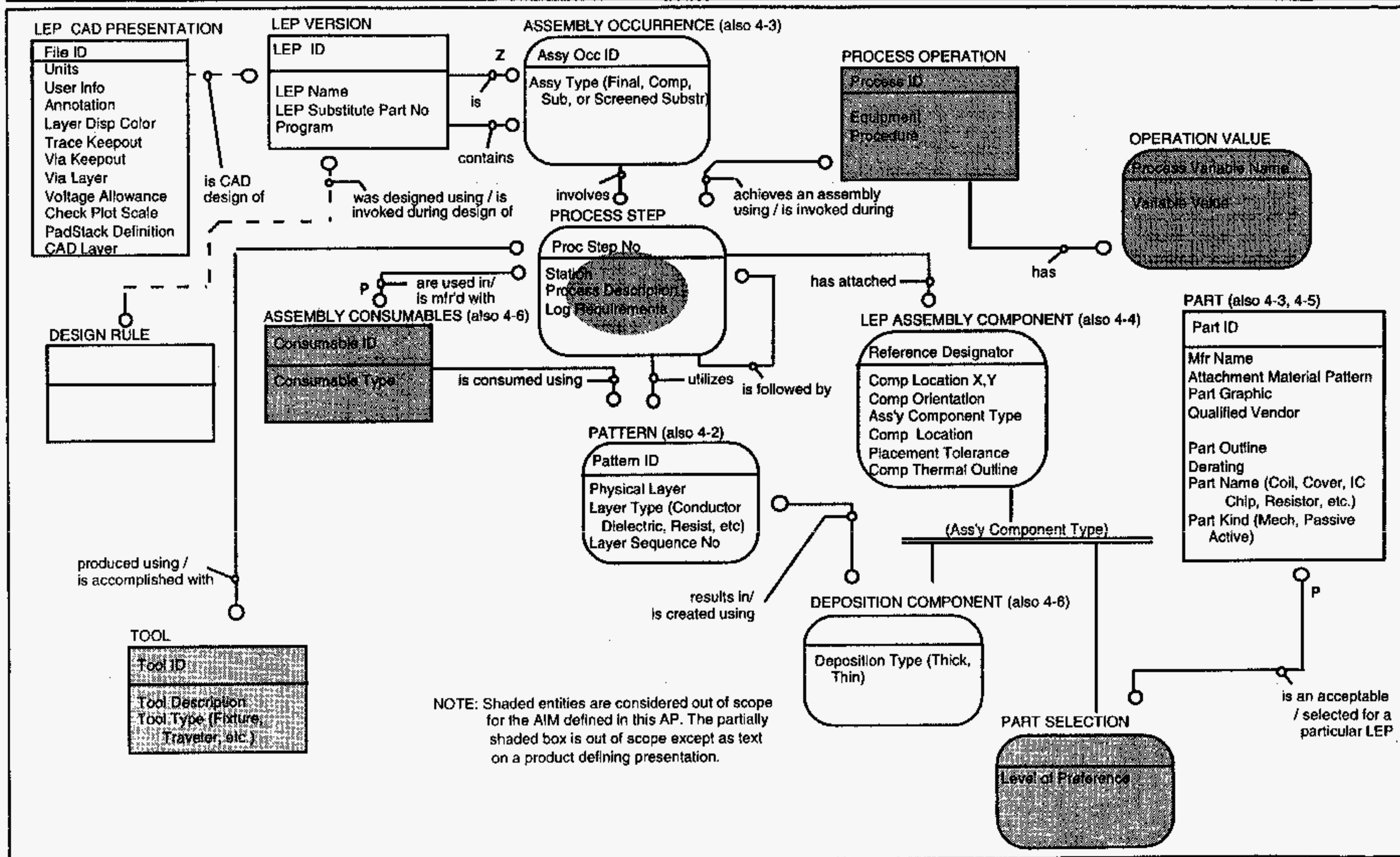
The ARM is intended to be an enterprise-view of the information about the LEP as a product. The

ARM serves as a reference-point for the data implementation as captured in drawings, in the CAD system, and specified for IGES exchange in the next section.The ARM diagrams (views) constitute an information model that documents the information structures needed to support the activities in the Application Activity Model (AAM) and to develop the corresponding IGES Application Interpreted Model (AIM).

Product Types or "technologies" associated to the ARM Entities

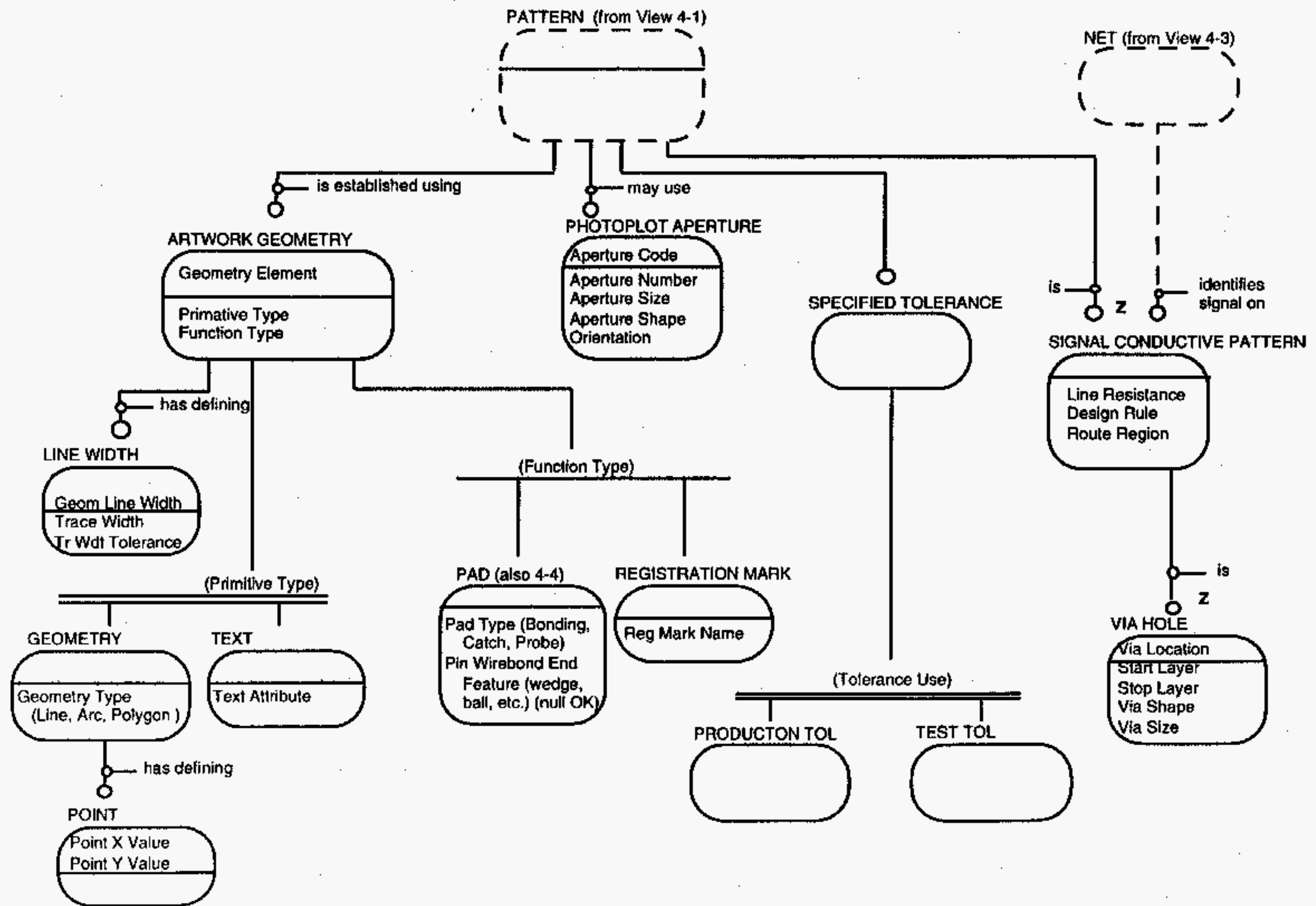
a) Integrated Circuits
b) Hybrid Microcircuit Assemblies (or Micro-chip Modules)
c) Printed Circuit Assemblies
d) Flex (FLEX) Circuits
e) Flexible (Flex) Cables

USED AT:	AUTHOR: Curt Parks, Team editor PROJECT: IGES EAC NOTES: 1 2 3 4 5 6 7 8 9 10	DATE: 3/4/91 REV: 10/9/91 12/4/91 1/22/92 9/11/93	WORKING DRAFT RECOMMENDED PUBLICATION	READER DATE	CONTEXT:



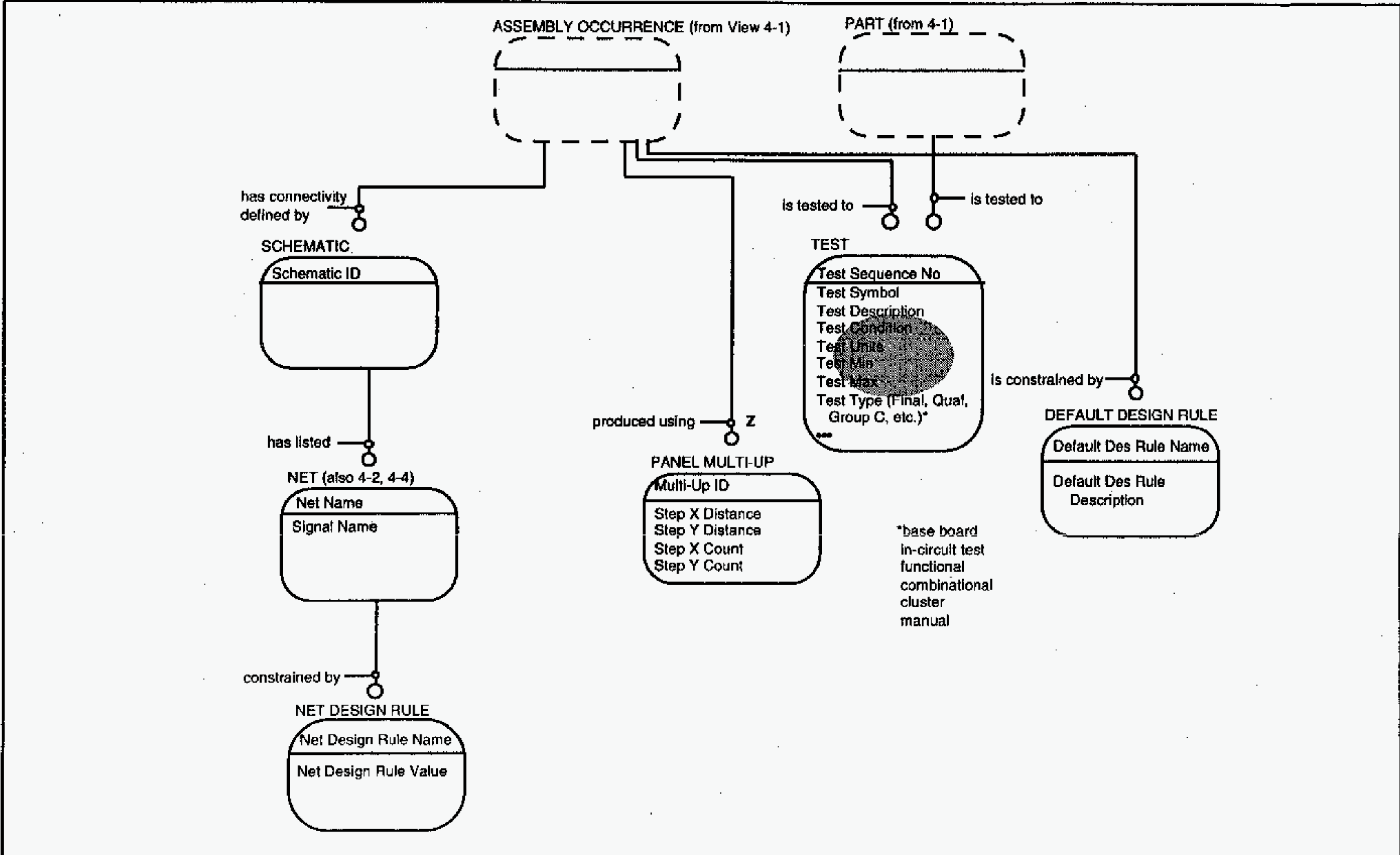
NODE:	TITLE: LEP APPLICATION REFERENCE MODEL	NUMBER: VIEW 4-1

USED AT:	AUTHOR: Curt Parks, Team editor PROJECT: IGES EAC NOTES: 1 2 3 4 5 6 7 8 9 10	DATE: 3/28/91 REV: 10/9/91 12/4/91 1/22/92	WORKING DRAFT RECOMMENDED PUBLICATION	READER DATE	CONTEXT:

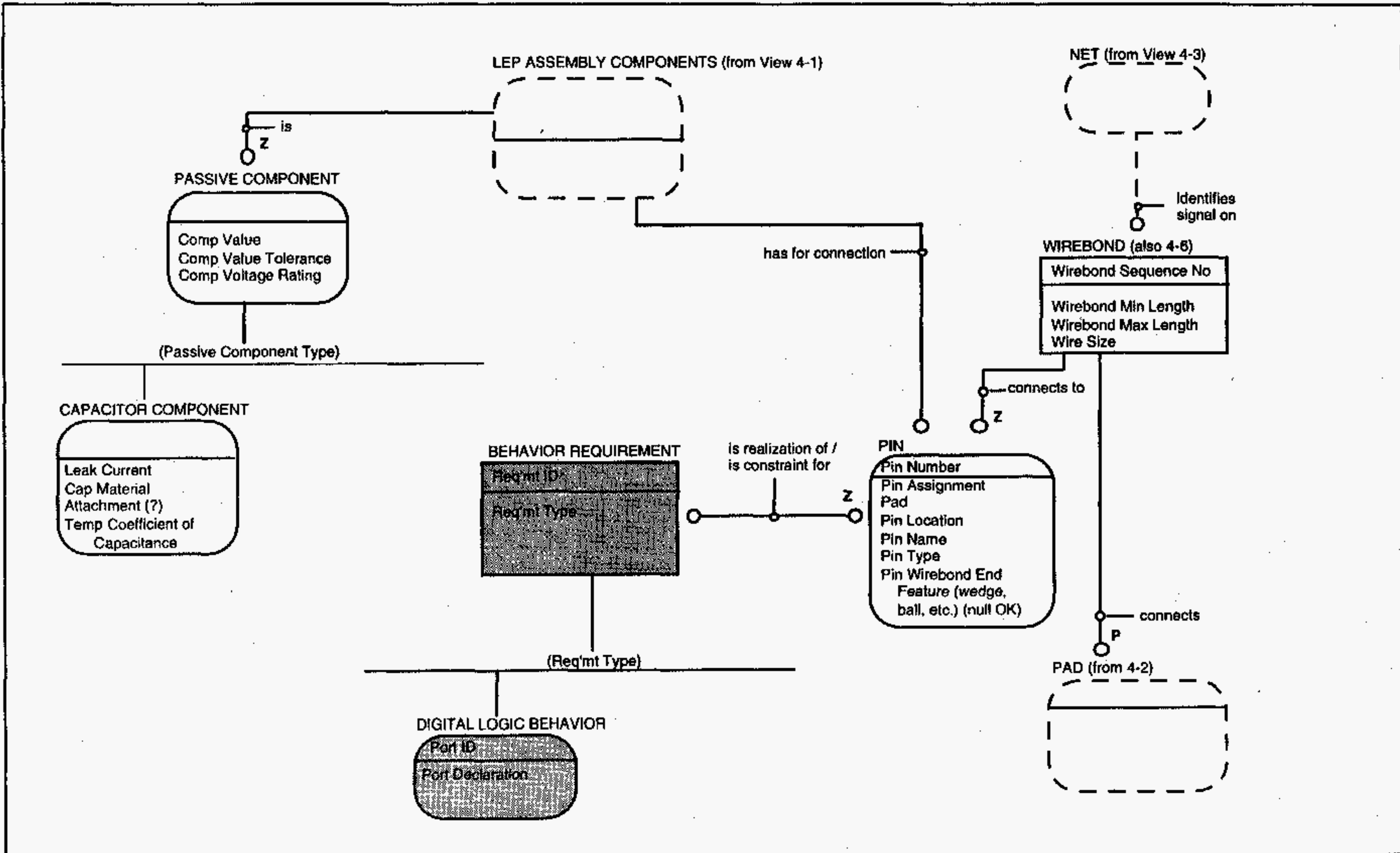


NODE:	TITLE: LEP APPLICATION REFERENCE MODEL EXPANDED DEPENDENTS OF PATTERN	NUMBER: VIEW 4-2

USED AT:	AUTHOR: Curt Parks, Team editor PROJECT: IGES EAC NOTES: 1 2 3 4 5 6 7 8 9 10	DATE: 10/5/91 REV: 10/9/91 12/4/91 1/22/92	WORKING DRAFT RECOMMENDED PUBLICATION	READER DATE	CONTEXT:



NODE:	TITLE: **LEP APPLICATION REFERENCE MODEL** **EXPANDED DEPENDENTS OF LEP VERSION**	NUMBER: **VIEW 4-3**



USED AT:
AUTHOR: Curt Parks, Team editor
PROJECT: IGES EAC
DATE: 10/5/91
REV: 12/4/91
9/15/93
11/8/94
NOTES: 1 2 3 4 5 6 7 8 9 10
WORKING
DRAFT
RECOMMENDED
PUBLICATION
READER
DATE
CONTEXT:
LEP ASSEMBLY COMPONENTS (from View 4-1)
NET (from View 4-3)
is
Z
PASSIVE COMPONENT
Comp Value
Comp Value Tolerance
Comp Voltage Rating
(Passive Component Type)
CAPACITOR COMPONENT
Leak Current
Cap Material
Attachment (?)
Temp Coefficient of
Capacitance
has for connection
Identifies
signal on
WIREBOND (also 4-6)
Wirebond Sequence No
Wirebond Min Length
Wirebond Max Length
Wire Size
connects to
Z
BEHAVIOR REQUIREMENT
Req'mt ID
Req'mt Type
is realization of /
is constraint for
Z
PIN
Pin Number
Pin Assignment
Pad
Pin Location
Pin Name
Pin Type
Pin Wirebond End
Feature (wedge,
ball, etc.) (null OK)
connects
P
(Req'mt Type)
PAD (from 4-2)
DIGITAL LOGIC BEHAVIOR
Port ID
Port Declaration
NODE:
TITLE:
LEP APPLICATION REFERENCE MODEL
EXPANDED DEPENDENTS OF LEP ASSEMBLY COMPONENTS
NUMBER:
VIEW 4-4

USED AT:

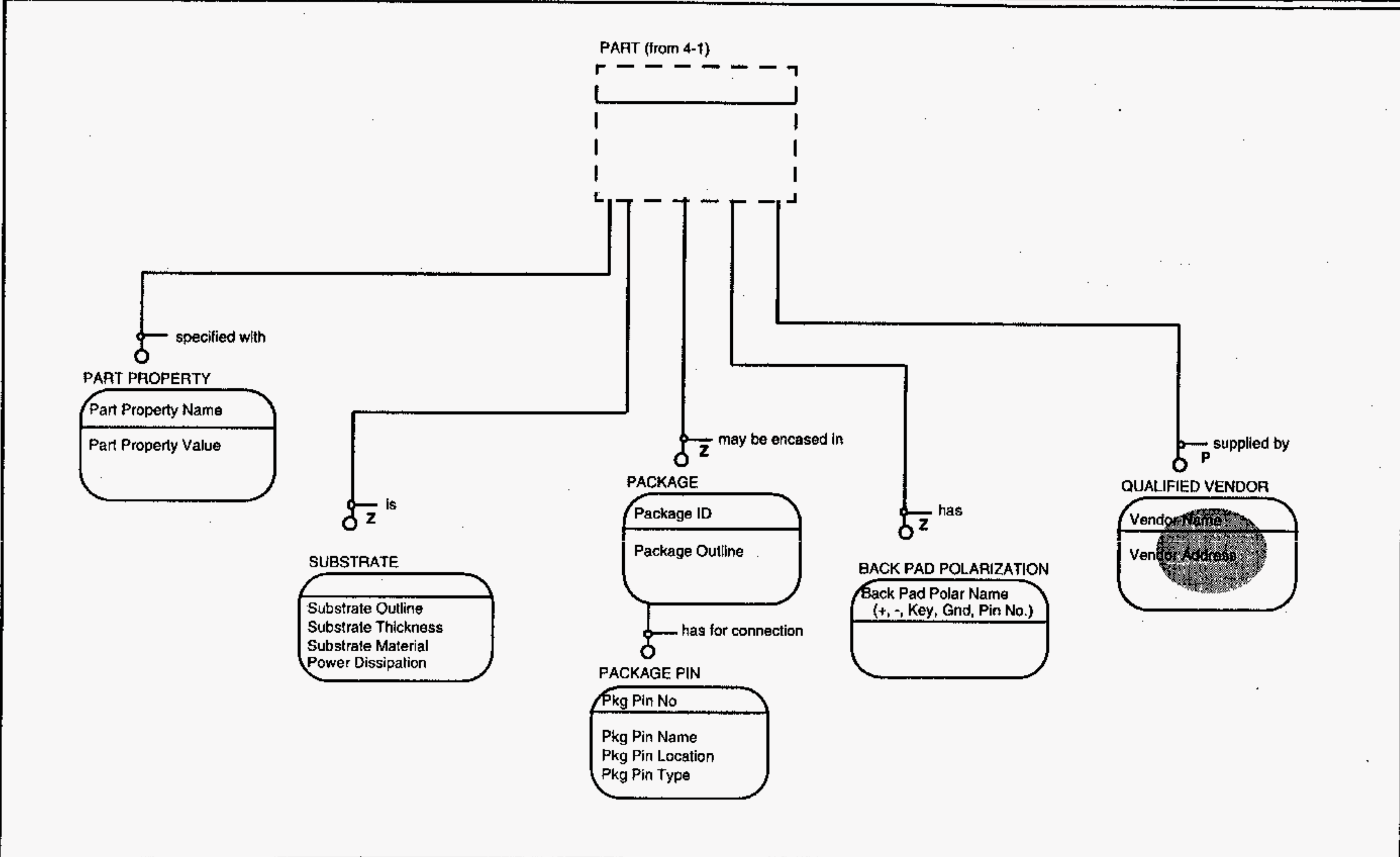
AUTHOR: Curt Parks, Team editor
PROJECT: IGES EAC

DATE: 11/18/91
REV: 12/2/91

NOTES: 1 2 3 4 5 6 7 8 9 10

		READER	DATE
X	WORKING		
	DRAFT		
	RECOMMENDED		
	PUBLICATION		

CONTEXT:

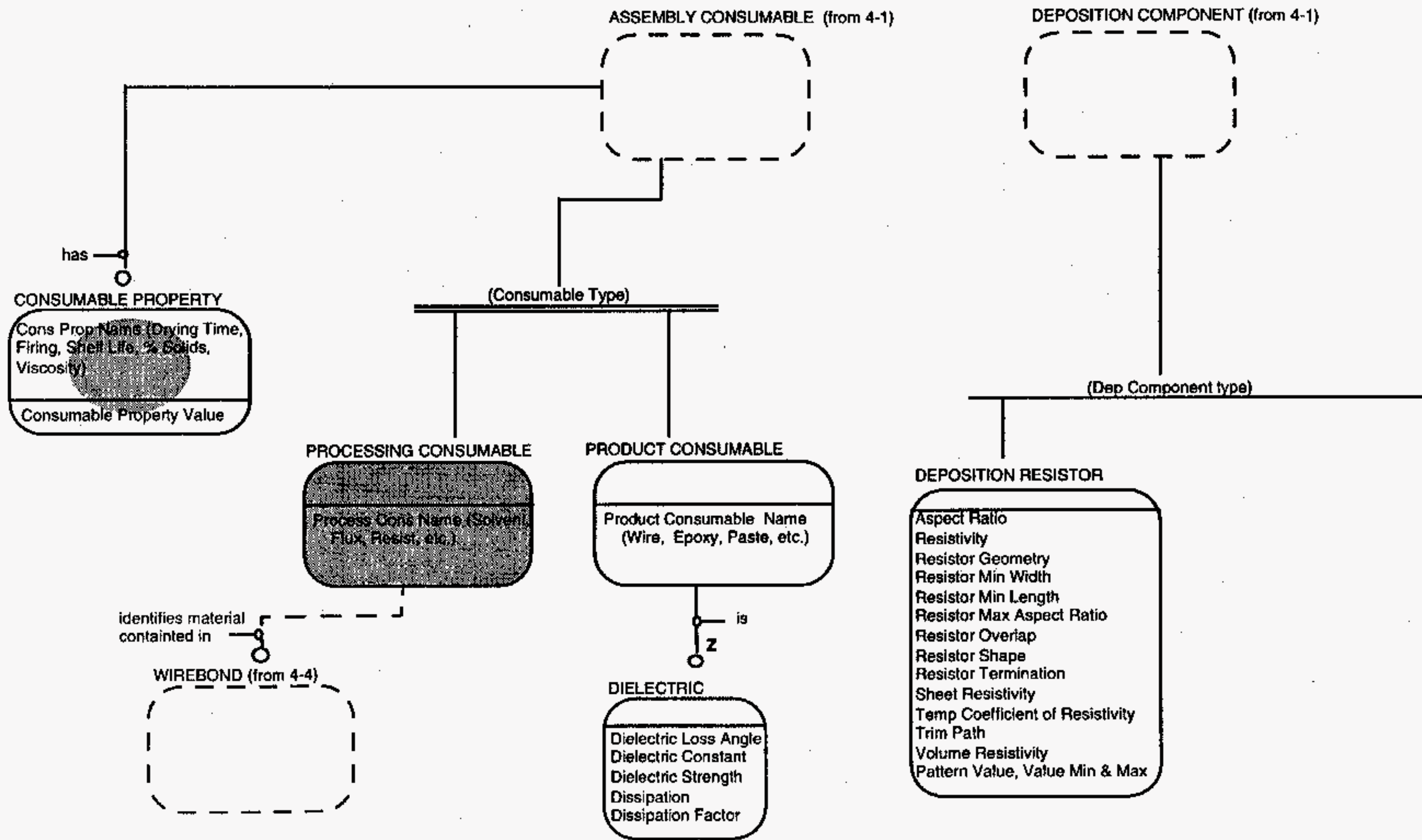


NODE:

TITLE: **LEP APPLICATION REFERENCE MODEL**
EXPANDED COMPONENT MODEL

NUMBER: **VIEW 4-5**

USED AT:	AUTHOR: Curt Parks, Team editor PROJECT: IGES EAC NOTES: 1 2 3 4 5 6 7 8 9 10	DATE: 10/5/91 REV: 10/9/91 11/29/91 12/2/91 1/14/92	WORKING DRAFT RECOMMENDED PUBLICATION	READER DATE	CONTEXT:



NODE:	TITLE: LEP APPLICATION REFERENCE MODEL EXPANDED DEPENDENTS OF ASSEMBLY CONSUMABLE & DEPOSITION COMPONENT	NUMBER: VIEW 4-6

Blank Page

5. IGES APPLICATION INTERPRETED MODEL

The Application Interpreted Model (AIM) is an information model that describes the logical information structures required for accomplishing a physical implementation of an associated application reference model. This AIM is prepared at a level of abstraction that is sufficient for selecting the necessary IGES[7] entities for an application protocol.

The scope of this AIM is currently limited[8] to LEP representation information contained in computer-aided design (CAD or CAE) systems. These systems are in widespread use and translation between dissimilar systems is a high priority in the user community. IGES serves as the implementation for this information because many CAE suppliers are familiar with the specification and support it in their software packages.

The scope is also inclusive of connectivity information (as contained in some CAE systems) in IGES files in addition to more conventional graphic (presentation) constructs. This connectivity information may be of significant value for future product representations. These future applications could combine display graphics with constructs better supporting concurrent engineering.

A broader scope for this AP that includes manufacturing, test, simulation, behavior and documentation is desired by users and manufacturers of LEPs. Much of this information is outside the scope (see also Abstract) of most CAE systems. Some of this information is beyond the scope of the IGES specification.

The graphical constructs of the AIM are presented in section 5.2 followed by the AIM diagrams in section 5.3. The AIM diagrams are divided into subsections as follows:

5.3.1 Interface	Interface object models are defined in this section. Each interface object model represents a "view" or "perspective" of an LEP in which one may exchange data. The interface object models describe the set of independent entities (DE Attribute - Subordination Status equals 0) that are in an IGES file which describe some aspect of an LEP. There are currently three interface objects; LEP Part Library IGES File, LEP Physical Layout IGES File and LEP Technical Illustration IGES File. Future interface objects might be LEP Schematic IGES File or LEP Manufacturing IGES File.
5.3.2 LEP Specific	This section defines objects that are unique and specific to LEPs.
5.3.3 Display Geometry	This section defines objects that are common to any CAE/CAD/CAM system that utilizes 2-dimensional geometry.
5.3 4 Miscellaneous	This section defines subordinate objects which, when used in the indicated combination, represent an LEP specific object.
5.3.5 DE Referenced	This section defines objects that are only referenced from the DE section of another object.
5.3.6 DE Section	This section defines value objects that represent pre-defined DE section values.

---

7. This section depends on the IGES document cited in section 2 for definitions and explanations of the format structure and the entities cited.

8. See Section 1.3.2. for the declaration of elements which are within the scope of this AP.

5.1. ARM to AIM Mapping

The ARM found in section 4 is an information model that documents the user view of the information structures of the subject application and provides the baseline from which candidate interpreted models are developed. For an IGES AP, the ARM is then used to develop the corresponding IGES Application Interpreted Model (AIM). The AIM shows how the information content from the ARM is to be expressed by a subset of IGES entities. Annotated outside the AIM object boxes (for reference) are the ARM entities which provided the requirement. (Note: The reader should not infer a one-to-one mapping; the AIM has a finer level of detail than the ARM.)

The PDQ (Product Data by Queries) IGES file represents the final product, fully assembled, with both electrical and mechanical parts installed. It has tooling holes, fiducials, and all the other PWB features needed by the product processes. This PDQ file serves information needs from conceptual design through packaging, shipping, and maintenance. In addition to being physically complete, the file also contains a significant amount of non-graphic information, including connectivity/netlist information, component values, traces, flows, properties, part numbers, and more.

The IGES entities selected for use in the AP specification should be selected to best convey the information needed in subsequent processes subject to constraints defined in this section. The options for the use of the entities within this subset have been restricted so that only one method is available for carrying each element of information from the ARM. The set of IGES entities and the necessary restrictions on the Global, Directory Entry, and Parameter Data Section field values have been developed by using the ARM and the AIM.

5.2. IGES Structure and Syntax of the Application Interpreted Model

Some graphical objects will be defined in this section to correspond closely to the constructs of the IGES file structure.

5.2.1. IGES File Structure

The following is the brief review of the IGES file structure from a predecessor[9] to this document:

> "IGES provides format guidance for data transfer/archive. It defines several data structures or entities, whose pre-defined purpose may be geometry, annotation, or structure. The individual vendor can then map information contained in his proprietary database into this neutral format, then transfer that information to other vendor's applications, or archive the file for future use. IGES is fully three dimensional, and includes finite element, graphics, documentation/annotation, manufacturing and electrical entities.
>
> ... The transfer mode will commonly be character (ASCII) ... The logical record length is 80 characters (block to 800 on tape) and is formatted for human readability and interaction. The advantage of the ASCII format are at the sacrifice of file size. However, there are standard file compression algorithms which may be used to achieve great reductions in file size, but the compressed files must be decompressed prior to translation. Also see ANSI X3.27-1978 labeled variable block.
>
> The IGES data file structure consists of five sections. The start section, contains 72 columns of user comments which are not to be processed by the program other than listing to

[9] see Appendix E of [11].

screen or printer. Next is the global section, a free format area specifying certain file attributes such as creation date, time, location, designer, units, display resolution, etc. The third section is the Directory Entry (DE) section containing fixed format attribute data (level, pen, translation, view, etc.) for each entity. The Parameter Data (PD) section follows and contains actual parameters for each of the entities in the DE section. Finally, the terminate section contains a single record containing the count of the records in each section. The entities come in two classes, those that are pre-defined and documented by the IGES committee (numbers 0001 to 5000), and those that are implementor definable (5001 to 9999). The constructs for a valid IGES file shall conform to those specified in the current version of the IGES Specification."

5.2.1.1. START Section

The IGES file Start section shall be in accordance with MIL-D-28000.

5.2.1.2. TERMINATE Section

The IGES file Terminate section shall be in accordance with the IGES document identified in Global parameter number 23.

5.2.1.3. GLOBAL Section

The IGES file Global section shall be in accordance with the IGES document identified in Global parameter number 23. Further information on particular global parameters are described in Section 5.2.2 below.

5.2.1.4. DIRECTORY ENTRY and PARAMETER DATA Sections

The IGES file Directory Entry and Parameter sections shall be in accordance with IGES version 5.2 or later. The values applicable to these section entries are dependent on the IGES entities and the objects (information structures) contained in the file. Those entities and objects shall be structured as defined in the remainder of this section.

5.2.2. Global Values and Entities List

The following are IGES constructs that are used to fulfill the requirements of the ARM (View 4) for LEP technology. The IGES entities have been selected to ensure the constructs uniquely attributable to an LEP will be correctly transferred when they are postprocessed. Other entities defined in the IGES Specification may be used for other purposes, however, these other entities should not be used for the purposes stated in the AIM. Parenthetical statements that follow the entities guide the reader to the relationships defined in the AIM. The IGES Specification should be consulted for the details of the syntactical constructs needed to produce valid IGES files.

Globals: (these are used to identify general product-file attributes)

Global No.	Description
3	Product identification from sending system
4	File name
5	System identification

12	Product identification for the receiving system
13	Model space scale
14	Unit flag
15	Units
21	Name of author
22	Author's organization
23	IGES Version Code number
25	Date and time the model was created or last modified

IGES Entities used in the AIM:

The AIM defines objects needed by the designer of translators and other software. The information requirements for these objects are defined within the ARM. The ARM information elements, excepting those identified as "out of scope," have been developed as detailed IGES-specific objects in the AIM.

All versions of IGES subsequent to version 3.0 are capable of transferring LEP information. To provide guidance needed to interpret files written to various versions, this section contains information on each applicable IGES entity; the initial version for the entity, and the existence of committee work which may extend or define its functionality. The following categories for IGES documented entities are utilized within this AIM. Individual object definition models in the AIM contain usage restrictions (i.e., usage constraints) appropriate to the application. Please refer to the applicable IGES document for detailed information regarding these entities. The "Version" column specifies the status of the entity. Applicable values are:

Vn — The entity is valid in this and subsequent versions and is not expected to be modified to be used within this AIM.

Gray — The entity currently exists and does not need to be modified to be used within this AIM. The entity is approved, but may not have been verified through testing.

+ — The entity either has proposed extensions and is being considered by the Electrical Applications Committee or has been submitted to the IGES Project for ballot. The number in parenthesis following the "+" relates to the "Request for Change" (RFC) document defining the new attributes proposed for the entity.

Please contact the IGES Organization for a copy of a cited Request for Change of interest. Implementors are encouraged to return results from implementation and exchange through the use of the constructs defined in an RFC. Implementors are also encouraged to write an RFC where user-defined IGES entities have demonstrated a capability.

Version	Type	Form	Description
V1.0	100	0	Circular Arc
V1.0	102	0	Composite Curve
V1.0	106	11	Copious Data - Planar Piecewise Linear Curve
V1.0	106	20, 21, 40	Copious Data*
V2.0	106	63	Copious Data - Simple Closed Planar Curve
V1.0	108	0, +1, -1	Plane

V1.0	110	0	Line
V1.0	116	0	Point
V1.0	124	0-1	Transformation Matrix
V2.0	125	0-4	Flash
V3.0	132	0	Connect Point
V1.0	212	0	General Note
V3.0	212	1,6,7&8	General Note*
V4.0	230	0	Sectioned Area (patterns 0 and 19 used as filled region)
	2xx	x	Other Annotation entities*
V1.0	308	0	Subfigure Definition
V3.0	312	0, 1	Text Display Template
V3.0	314		Color Definition*
V3.0	320	0	Network Subfigure Definition
V4.0	322	All	Attribute Table Definition**
V1.0	402	7	Group Associativity without Back Pointers
V3.0+(544)	402	18	Flow Associativity
V1.0	404	0	Drawing*
V1.0	406	1	Property - Definition Levels
V2.0	406	3	Level Function
V2.0	406	5	Property - Line Widening
V2.0	406	6	Property - Drilled Hole***
V2.0	406	9	Property - Part Number
V2.0	406	10	Hierarchy
V3.0	406	11	Tabular Data
V3.0	406	15	Property - Name
V5.0+(537)	406	24	Property - Level to LEP Layer Map
V5.1, Gray	406	27	Property - Generic Data
V1.0	408	0	Subfigure Instance
V1.0, Gray	410	All	View*
V3.0	420	0	Network Subfigure Instance
V4.0	422	All	Attribute Table Instance**

* Optional; used in drafting/documentation applications when appropriate.
** Optional; used in circuit analysis/simulation applications where appropriate.
*** May only be applicable to certain product types, in particular the Printed Wiring Board.

5.2.3 Basic Syntax used in the AIM

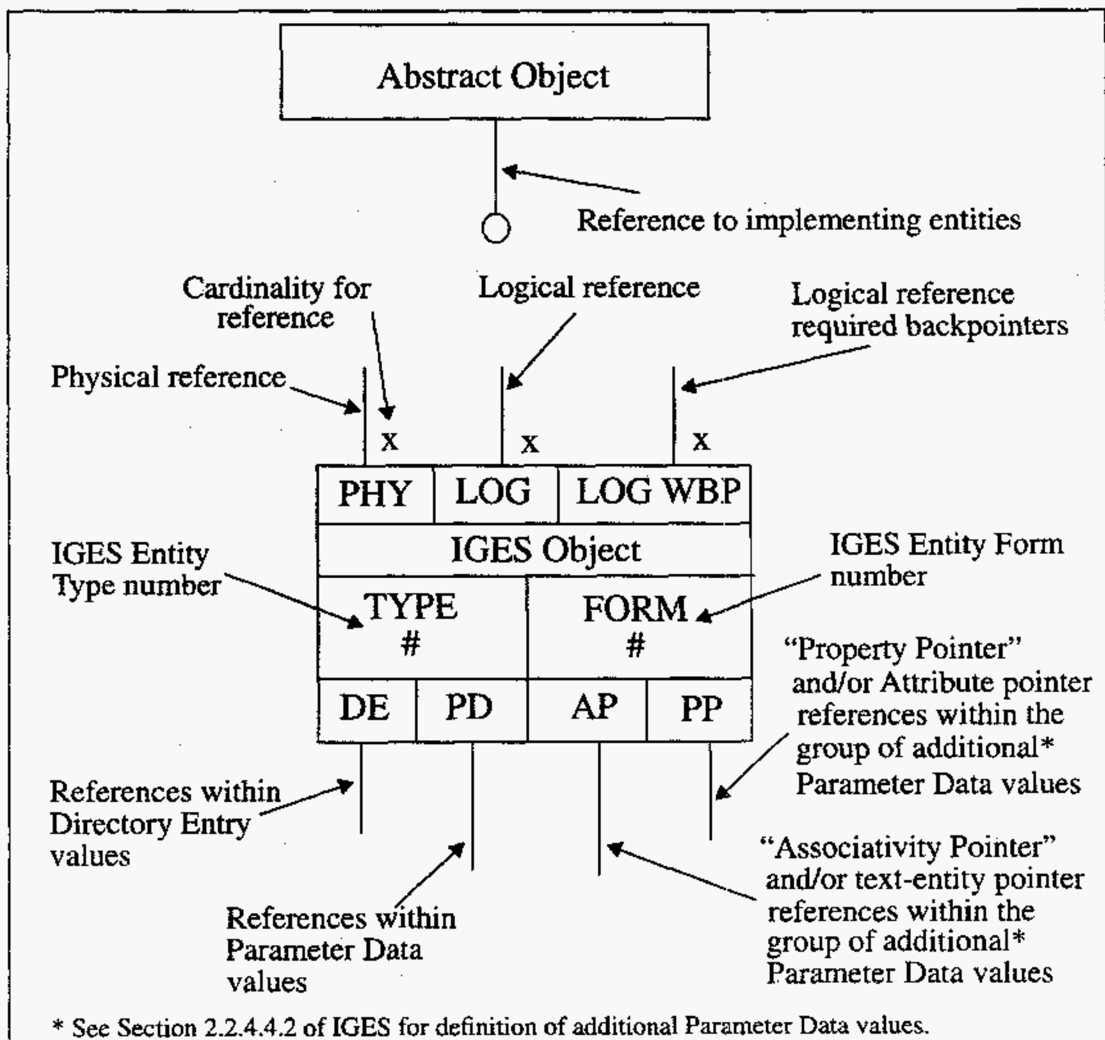
A graphic notation has been adopted to describe the IGES data structures.The objective of these diagrams is the ease of developing unambiguous IGES translators and proof-reading files. The notation is not intended for use as a conceptual modeling language. The brief descriptions which follow explain the use of principal elements of the notation; the Object Definition Block, the Object Instance Block, the Object Value Block, and the Cardinality code.

5.2.3.1 Object Definition Block

The Object Definition Block is used to depict an IGES entity type, form, directory entry values, parameter data values, and relationships to other IGES entities. There are several forms of the Object Definition Block depending upon how the object is being used within the model.

Each in-scope Object is depicted at least once using the Object Definition Block.

The graphic in Figure 5-1 is used to depict data and relationships to other objects for a particular "Abstract Object" (a high-level abstraction usually from the electronic design domain). The graphic shows the field name and required value(s)—if appropriate—for a particular field within the definition or instance of an object. A Value Block (see 5.2.3.3) may be referenced by either an Object Definition Block or an Object Instance block.



* See Section 2.2.4.4.2 of IGES for definition of additional Parameter Data values.

Figure 5-1. AIM Object and Referencing.

5.2.3.2 Object Instance Block

The graphic in Figure 5-2 illustrates a particular reference to a previously defined object. The long form is available if the particular instance needs to reference other IGES entities.

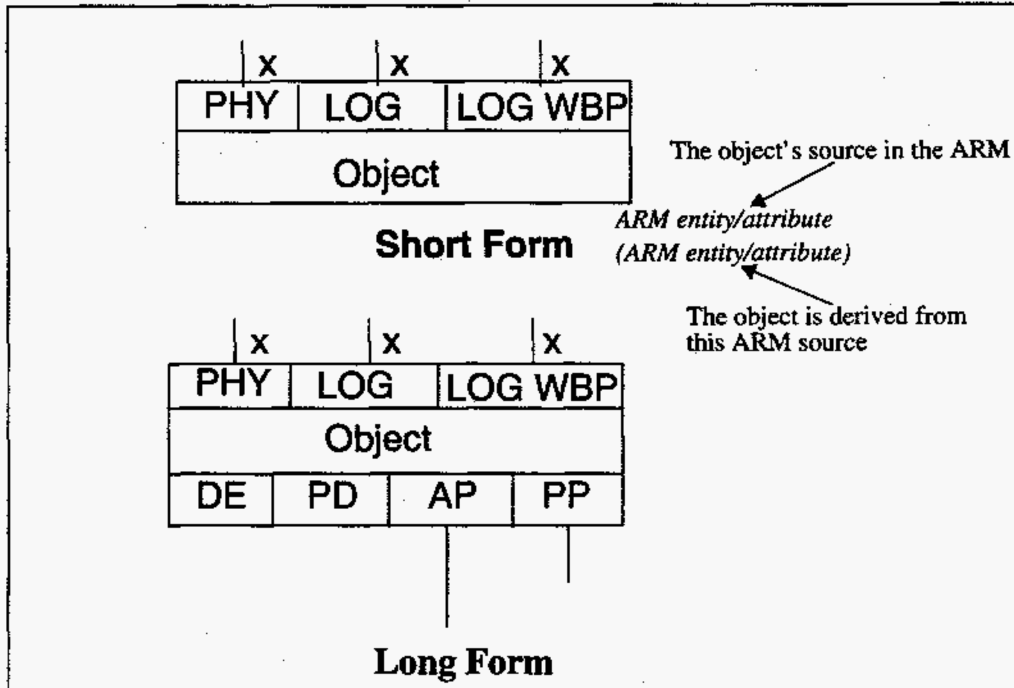


Figure 5-2. References to Previously Defined Object.

5.2.3.3 Object Value Block

The graphic in Figure 5-3 illustrates the field name and required value(s), if appropriate, for a particular field within the definition or instance of an object. This Value Block shown corresponds to the IGES parameter data section.

The Value Definition Block is used to show a set of Object reference mechanisms. The mechanisms provide ways to show either Directory Entry (DE) values or Parameter Data (PD) values. These values may reference other IGES Entities via their DE identifying numbers.

A Value Block may be referenced by either an Object Definition Block or an Object Instance Block.

Value			
Index	Name	Value	
1	name1	Value	Field value as appropriate
2	name2	*	Any acceptable value
3	name3	⇒	Pointer to a predefined Instance Block
4	name4	value |⇒	Value or pointer to a predefined Object Instance Block
5	name5	N/A	Value will be ignored
...	name...	...	

Figure 5-3. Object Field Value Restrictions.

5.2.3.4 Object Reference Mechanism

The Object Definition Block or the Object Instance Block each may reference subordinate /child objects. Each object block will distinguish a reference to a subordinate object by any of the following mechanisms:

1) a directory entry (DE) pointer

2) a parameter data (PD) pointer, or (when appropriate)

3) an associativity backpointer (AP) or

4) a property pointer (PP)

as shown in Figure 5-4.

The reference to the subordinate object may involve either of the two kinds of dependencies described in 2.2.4.3.9.2 of IGES. The three dependency mechanisms indicated in Figure 5-4 are:

1) physically subordinate (Physical),

2) logically subordinate (Logical), or

3) logically subordinate with backpointer (LogW/BP).

The backpointers for the third mechanism are never explicitly shown in the figures that follow.

The representations of cardinality indicators are shown in Figure 5-5

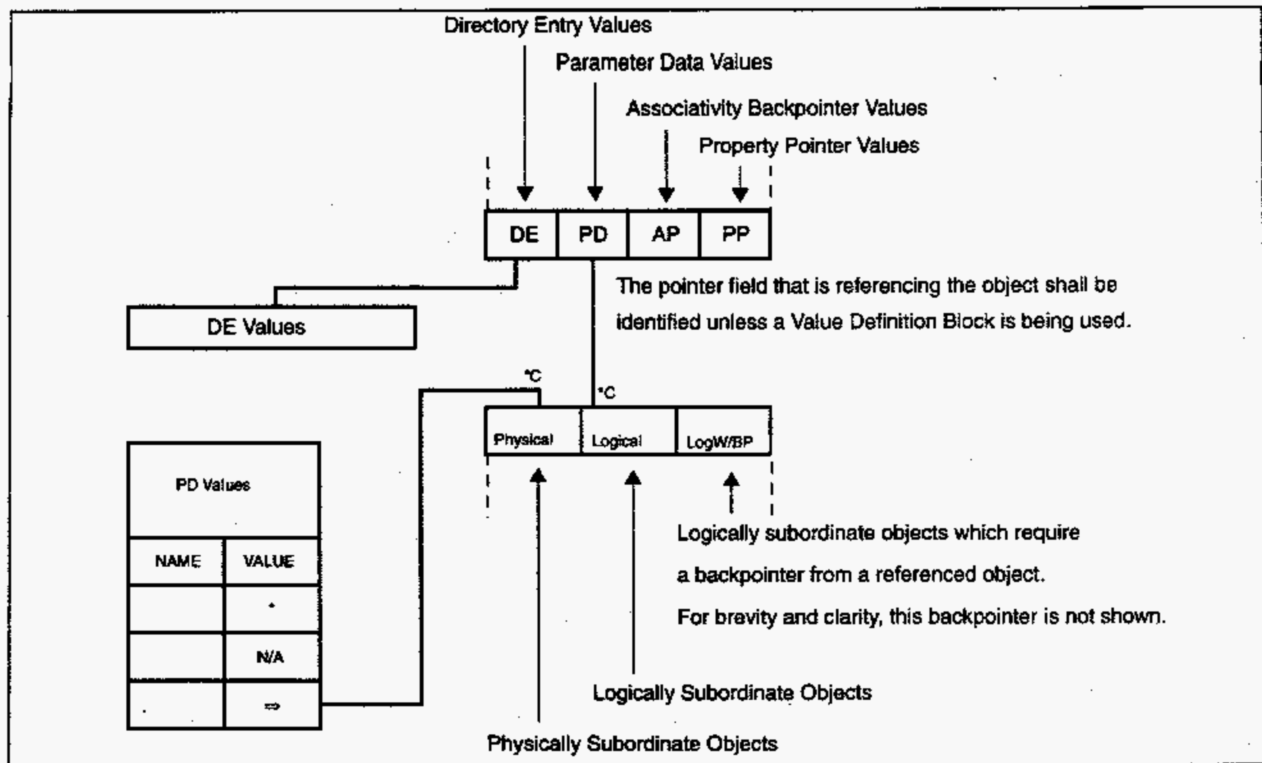


Figure 5-4. Reference Mechanism.

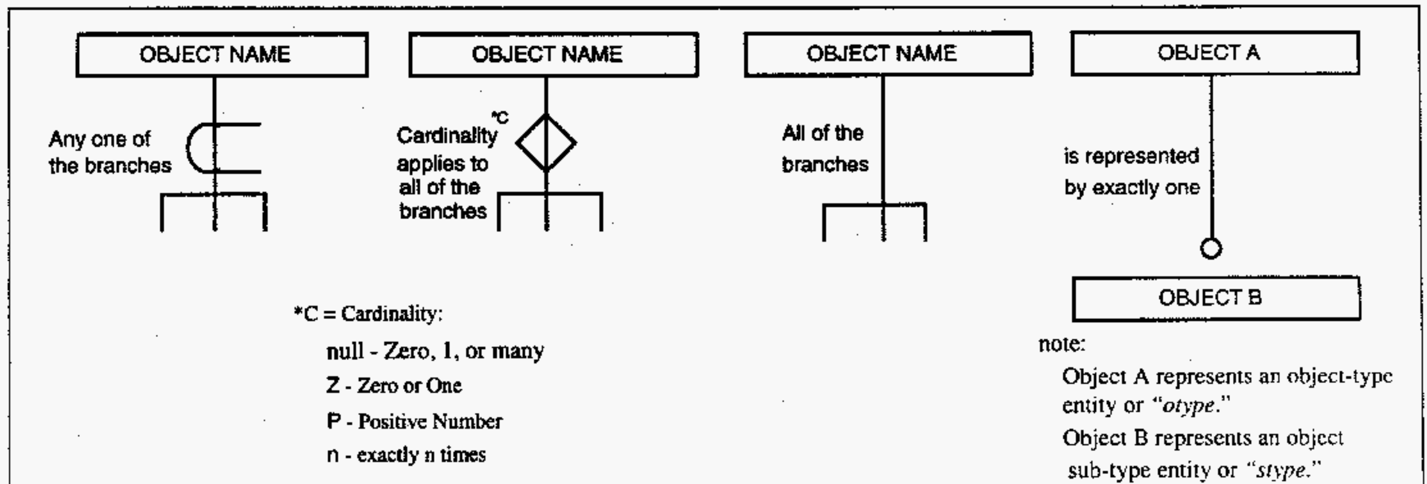


Figure 5-5. Representation Mechanism.

BLANK PAGE

5.3. AIM Object Models

Six categories of object models are documented in this Section.

NOTE: The AIM object requirements were derived from ARM entities; the associated ARM entity will be found next to each object. Some ARM entities do not relate to objects in this section on a one-to-one basis, and are identified here.

Arm Entity	AIM Object(s)
Artwork Geometry	All objects in Section 5.3.3
Back Pad Polarization	*PACKAGE FIGURE INSTANCE* and *GENERIC DATA PROPERTY*
Capacitor Component	*GENERIC PART DEFINITION* and *INSTANCE*
Component Property	*GENERIC DATA PROPERTY*
Default Design Rule	Various *KEEPIN/KEEPOUT* objects
Deposition Component	*GENERIC PART DEFINITION* and *INSTANCE*
Deposition Resistor	*GENERIC PART DEFINITION* and *INSTANCE*
Dielectric	*CLOSED CURVE* with appropriate *GENERIC DATA PROPERTIES*
LEP Version	*GENERIC DATA PROPERTY* (component_version)
Net Design Rule	*TRACE KEEPIN/KEEPOUT*
Package	*PACKAGE FIGURE DEFINITION* and *INSTANCE*
Passive Component	*GENERIC PART DEFINITION* and *INSTANCE*
Photoplot Aperture	*GENERIC DATA PROPERTY*
Processing Consumable	*GENERIC DATA PROPERTY*
Production Tolerance	*GENERIC DATA PROPERTY* (tolerance)
Registration Mark	*CLOSED CURVE* and *GENERIC DATA PROPERTY*
Schematic	*LEP TECHNICAL ILLUSTRATION IGES FILE*
Signal Cond. Pattern	*NON-CLOSED CURVE* (c.f. Table 6-15)
Specified Tolerance	*GENERIC DATA PROPERTY* (tolerance)
Substrate	*CLOSED CURVE* (c.f. Table 6-9)
Test Tolerance	*GENERIC DATA PROPERTY* (tolerance and others)

5.3.1. Interface Object Models

Three categories of object models have been defined for interfaces. Interfaces to a LEP part library IGES file, a LEP physical layout IGES file, and to a technical illustration IGES file are included.

5.3.1.1. LEP Part Library IGES File

Description:

The *LEP PART LIBRARY IGES FILE* object is an interface model which enables computer-aided systems to write or read a library of LEP parts (e.g., Package Figure Definitions, Padstack Definitions, or Generic Part Definitions).

Requirements/Restrictions:

Translation Usage Notes:

General:

Output:

Input:

LEP PART LIBRARY IGES FILE

Part, LEP CAD Representation

Please refer to IGES Section 3.6.4 for this object.

IGES graphic file: PARTLIB

5.3.1.2. LEP Physical Layout IGES File

Description:

The *LEP PHYSICAL LAYOUT IGES FILE* object is an interface model which enables computer aided systems to write or read LEP Physical Layout data.

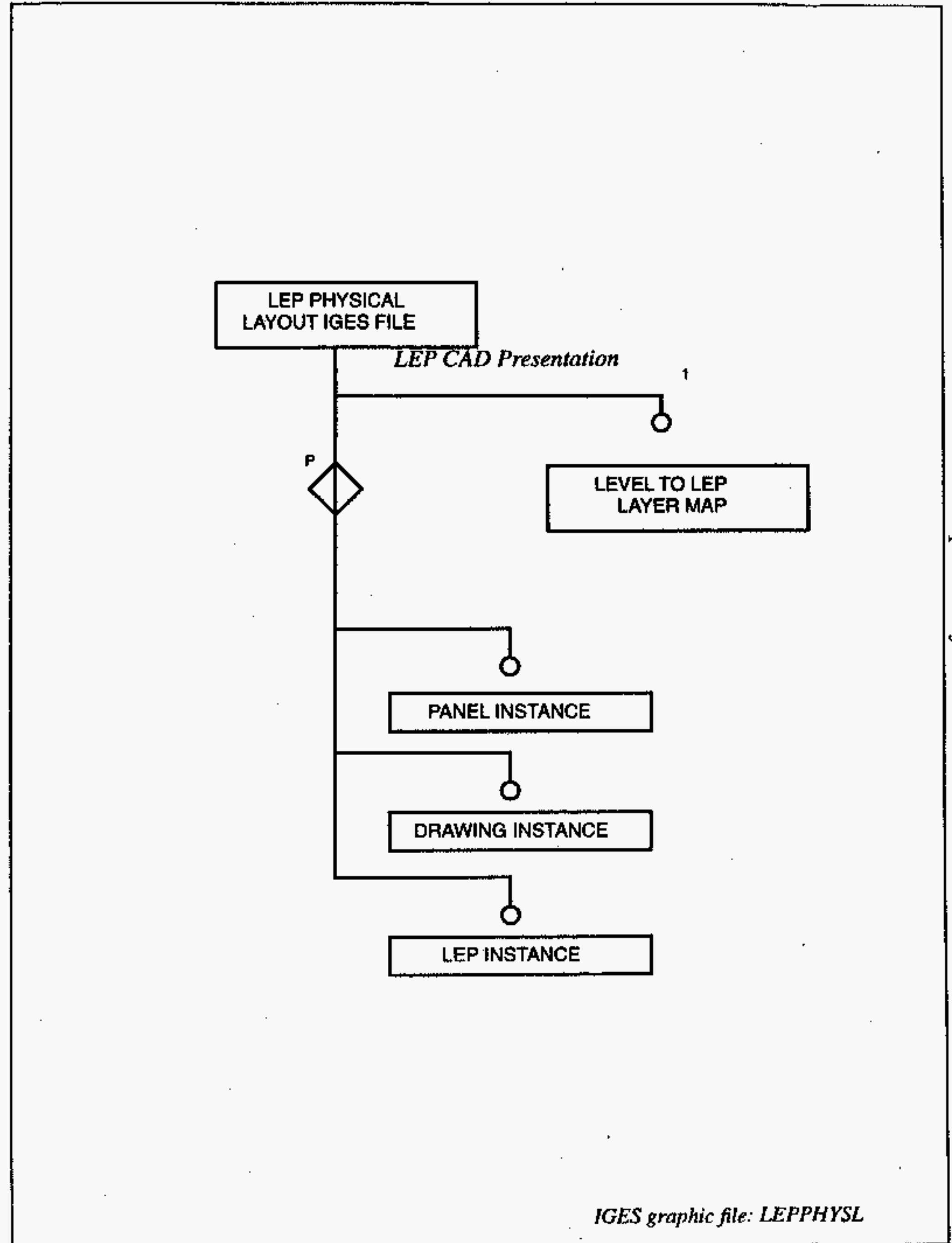
Requirements/Restrictions:

1. When this object defines the layout of an LEP Panel, the *LEP SEMANTIC PROPERTY*which is referenced from the *LEP INSTANCE* object shall specify
   panel_offset

Translation Usage Notes:

General:
These files contain geometry, annotation, structure, and connectivity information.
Output:

Input:



IGES graphic file: LEPPHYSL

5.3.1.3. LEP Technical Illustration IGES File

Description:

The *LEP TECHNICAL ILLUSTRATION IGES FILE* object is an interface model which enables computer-aided systems to write or read a graphic presentation of LEP data. This interface model describes only geometry and annotation constructs.

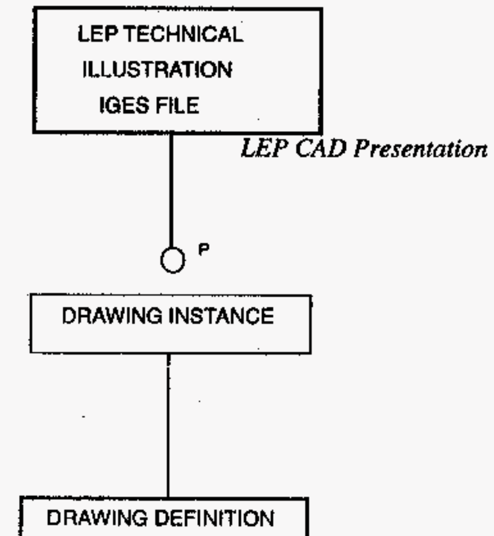
Requirements/Restrictions:

Translation Usage Notes:
See especially MIL-D-28000 Class 1.

General:

Output:

Input:



IGES graphic file: TECHILL

5.3.1.4. LEP Simulation/Analysis IGES File

Description:

The *LEP SIMULATION/ANALYSIS IGES FILE* object is an interface model which enables computer-aided systems to write or read attributes associated with the graphic presentation of LEP data. This interface model describes only attribute annotation constructs.

The use of the annotation entities is not required for an LEP file; the entities provide information intended primarily for Circuit Analysis and Finite Element Modeling applications. The entities may also be used within specification control drawings for annotation.

Requirements/Restrictions:

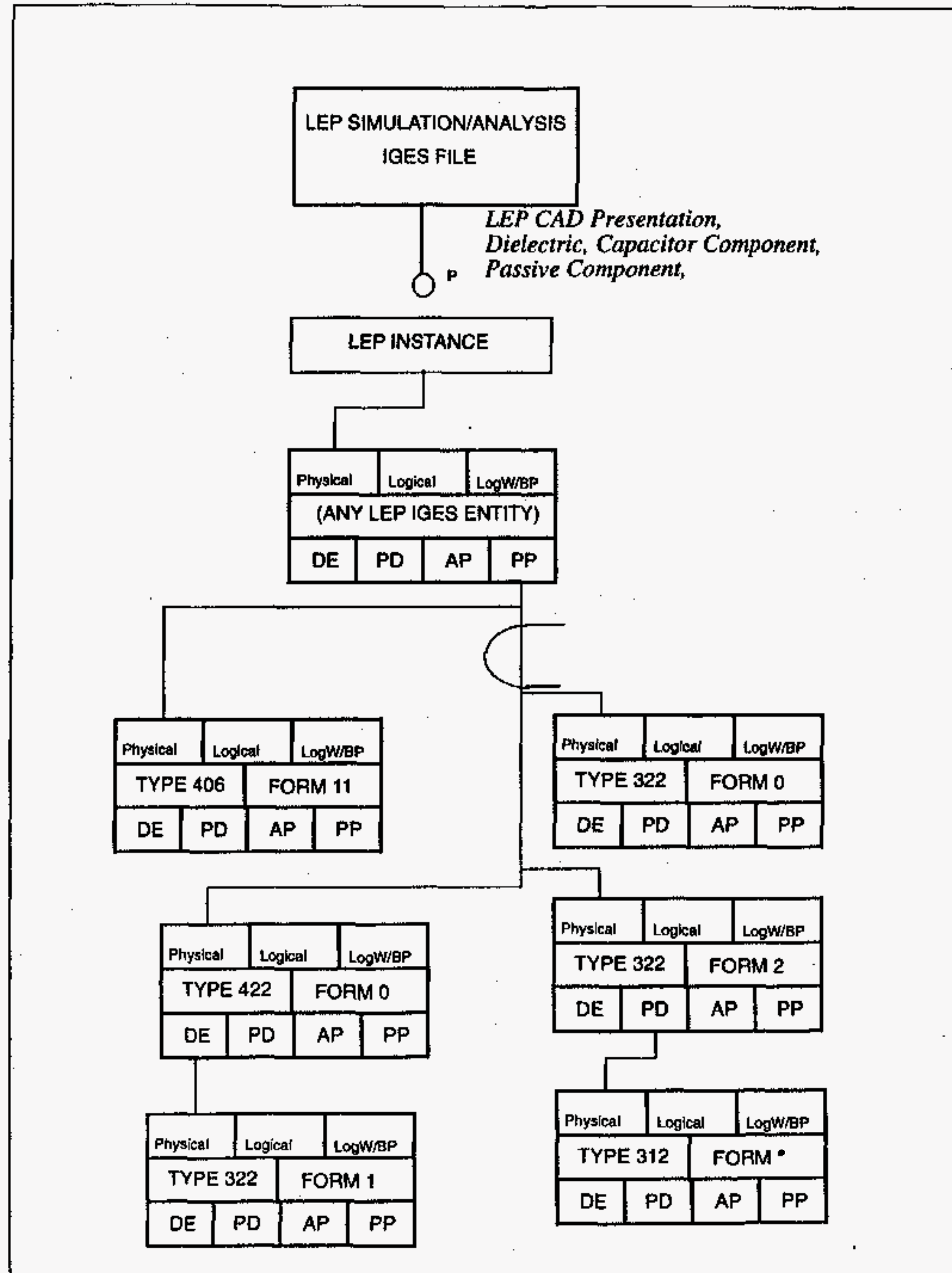
1. The Attribute Table Definition Entity (Type 322) and Attribute Table Instance Entity (Type 422) shall be associated with other IGES entities as defined in IGES Section 3.6.7.

Translation Usage Notes:

General:

Output:

Input:



BLANK PAGE

5.3.2. LEP Features

These objects are higher-level structures of entities which are identified with LEP designs and with CAE system usage terminology.

5.3.2.1. Component Placement Keepin/ Keepout

Description:

The *COMPONENT PLACEMENT KEEPIN/KEEPOUT* object is a *CLOSED CURVE*, which bounds within or excludes the instantiation of all Package Figure Instances.

Requirements/Restrictions:

The IGES file shall have one or more *COMPONENT PLACEMENT KEEPIN/KEEPOUT* per placement layer.

Translation Usage Notes:

General:
The values for the *LEP SEMANTIC PROPERTY* shall be as follows:

component_keep_outline

with a value of either "in," or "out" to state that the component shall be inside or outside the region defined by the closed curve. (NOTE: The CAE system is expected to utilize this object to provide component placement control or rules checking.)

Output:
Input:

see KEEPIN/KEEPOUT (Section 5.3.2.10.).

5.3.2.2. Component Thermal Outline

Description:

The *COMPONENT THERMAL OUTLINE* object is a *CLOSED CURVE*, which represents the area of the *PACKAGE FIGURE INSTANCE* that has associated thermal characteristics.

Requirements/Restrictions:

Translation Usage Notes:

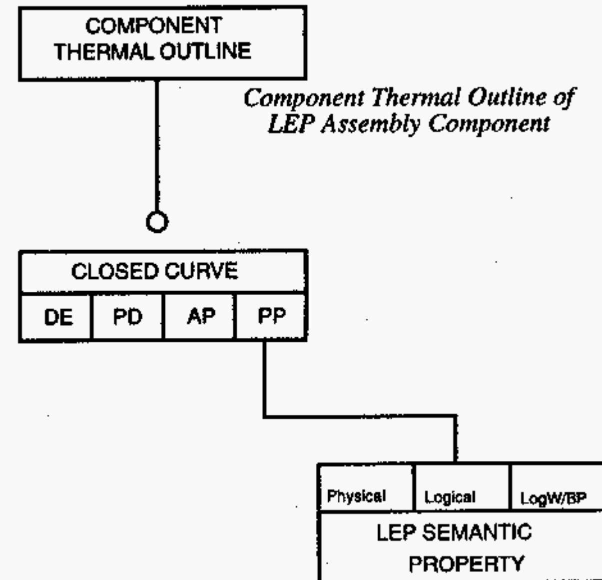
General:

The value for the *LEP SEMANTIC PROPERTY* shall be:
thermal_outline

to state that the component has a thermal boundary defined at the closed curve.

Output:

Input:



5.3.2.3. Drawing Definition

Description:

The *DRAWING DEFINITION* object is a Subfigure Definition which represents an engineering drawing of the *LEP*, a *PANEL*, or some portion of the *LEP*.

Requirements/Restrictions:

1. The *GENERIC DATA PROPERTY* which is referenced from the *SUBFIGURE DEFINITION* object shall specify
   - date
   - schematic_number or assembly_number.
2. The *GENERIC DATA PROPERTY* which is referenced from the *DISPLAY GEOMETRY* object, when appropriate, shall specify
   - physical_outline
3. The NAME field in the *SUBFIGURE DEFINITION* shall be unique within the scope of the IGES File.

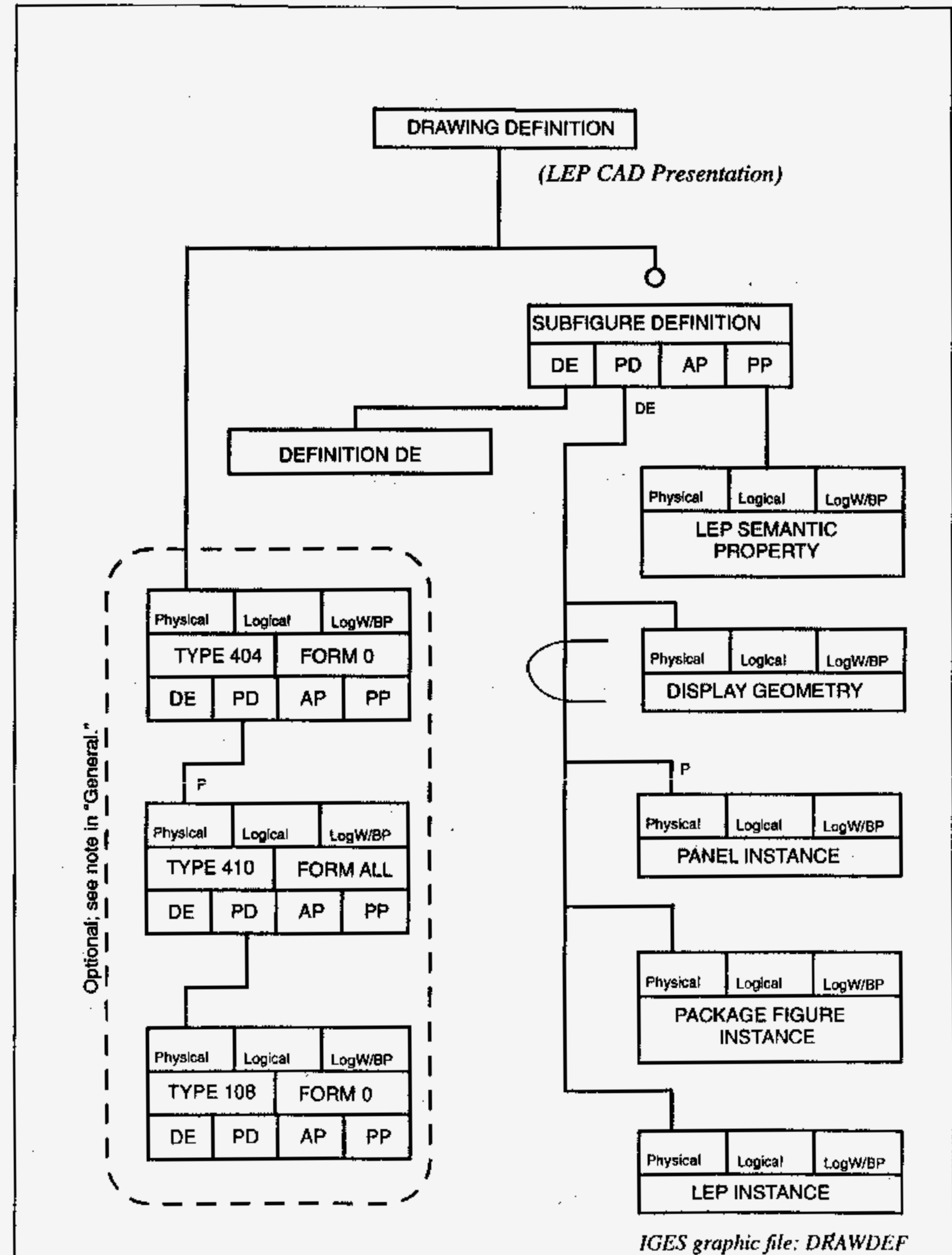
Translation Usage Notes:

General:

The Drawing Entity (Type 404), View Entity (Type 410) Plane Entity (Type 108) and properties as appropriate optionally appears in *LEP TECHNICAL ILLUSTRATION*. See Section 3.6.5 of IGES for association of these entities.

Output:

Input:



IGES graphic file: DRAWDEF

5.3.2.4. Drawing Instance

Description:

The *DRAWING INSTANCE* object is an instantiation of a *DRAWING DEFINITION*.

Requirements/Restrictions:

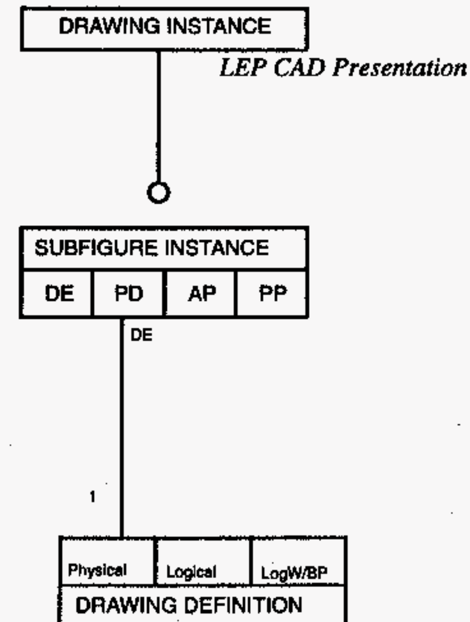
1. The *GENERIC DATA PROPERTY* which is referenced from the *SUBFIGURE INSTANCE* object shall specify
   top_level_assembly_instance.

Translation Usage Notes:

General:

Output:

Input:



IGES graphic file: DRAWINST

5.3.2.5. Generic Part Definition

Description:

The *GENERIC PART DEFINITION* object is a *SUB-FIGURE DEFINITION*, which lists a set of constituent objects for the purpose of treating them like a fixed set or group (e.g., heatsink).

Requirements/Restrictions:

1. A *ROUTING KEEPIN/KEEPOUT* or *COMPONENT PLACEMENT KEEPIN/KEEPOUT* shall not be subordinate to a *GENERIC PART DEFINITION.*
2. The *LEP SEMANTIC PROPERTY* values referenced from the *GENERIC PART DEFINITION* are specifed by the *PACKAGE SYMBOL DEFINITION.*
3. The NAME field in the Subfigure Definition Entity (Type 308) shall be unique (for all *GENERIC PART DEFINITIONS*) within the scope of the IGES File.
4. The *LEP SEMANTIC PROPERTY*
    substrate_physical_outline,
    and as appropriate placement_outline
   shall reference the appropriate *CLOSED CURVE* object.

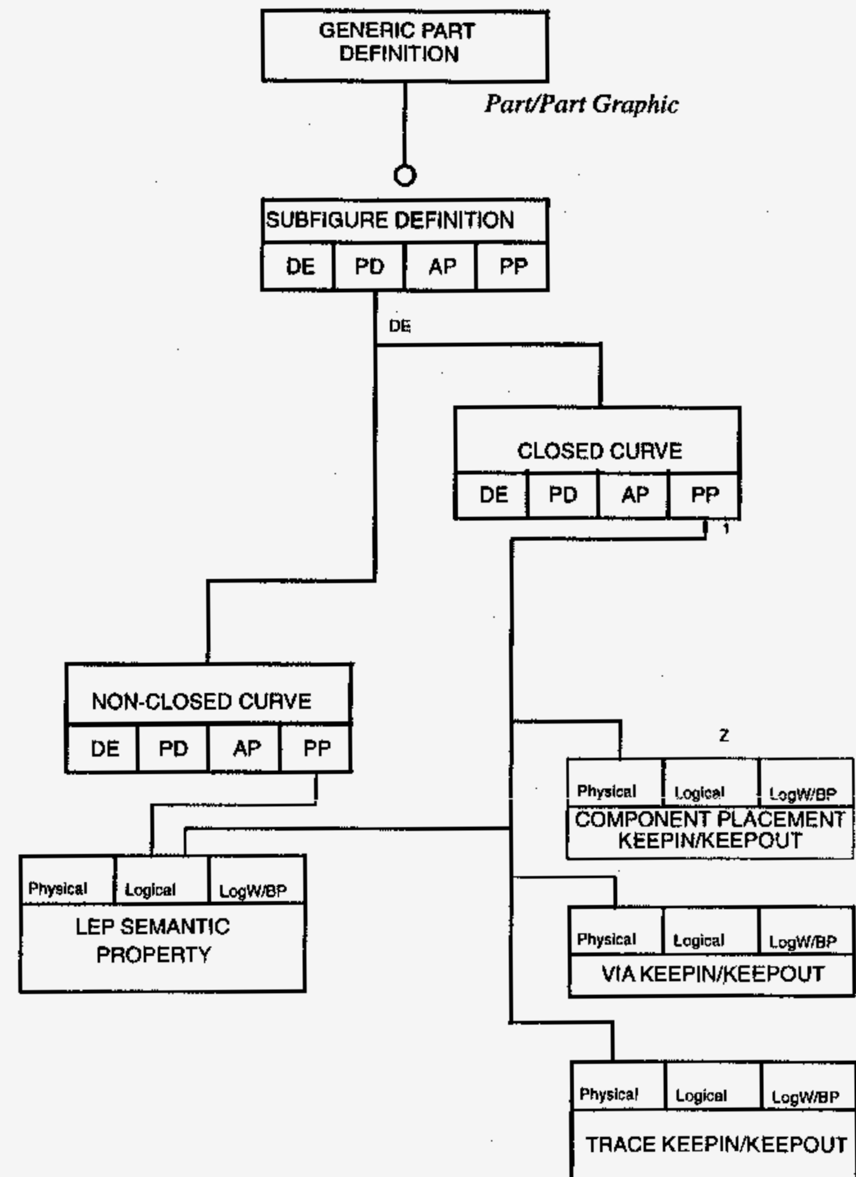
Translation Usage Notes:

General:

NOTE: This *GENERIC PART DEFINITION* is not applicable to electrical parts; see *NETWORK SUBFIGURE DEFINITION.*

Output:

Input:



IGES graphic file: PART

5.3.2.6. Generic Part Instance

Description:

The *GENERIC PART INSTANCE* object is an instantiation of a *GENERIC PART DEFINITION*.

Requirements/Restrictions:

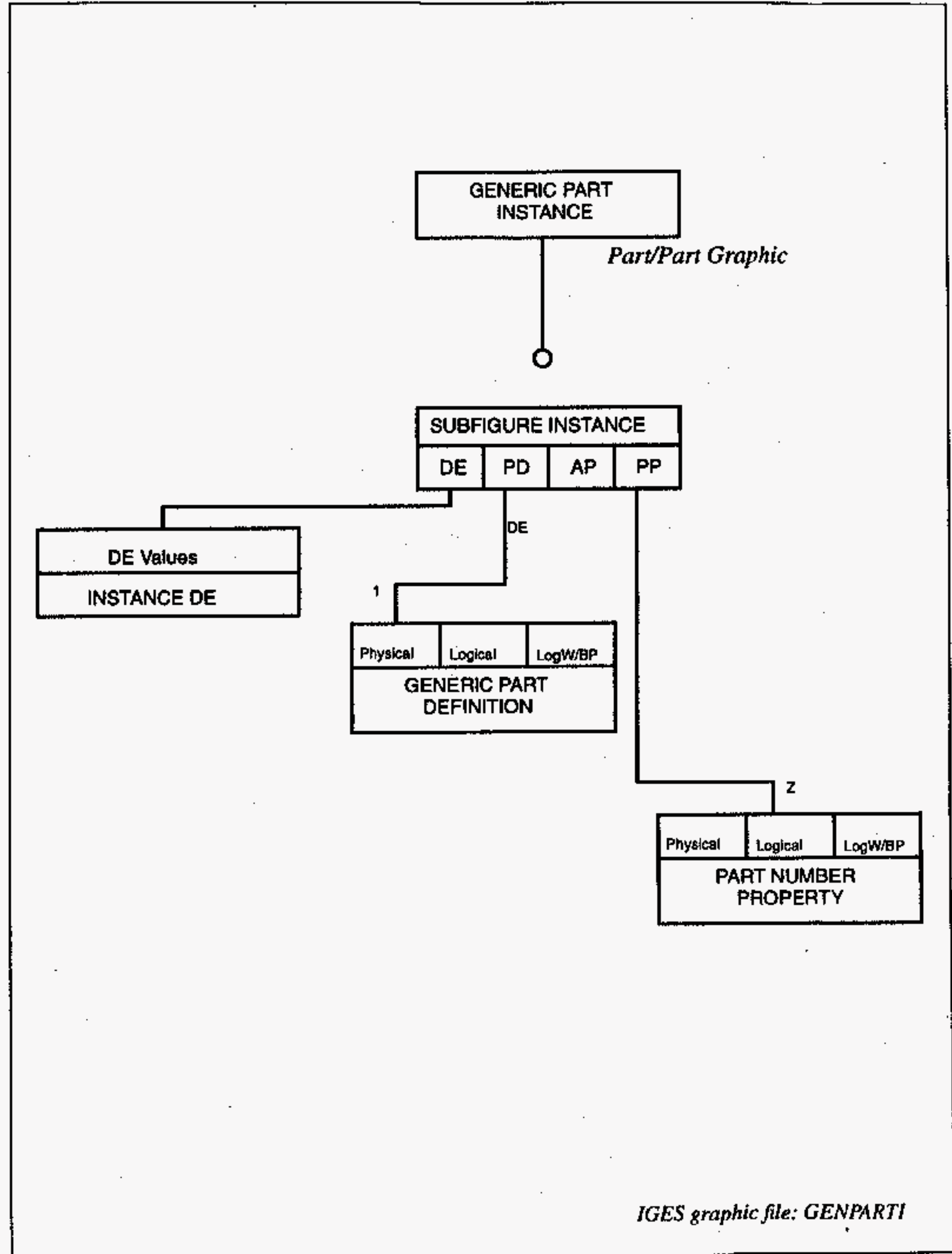
1. The *LEP SEMANTIC PROPERTY* which is referenced from the *SUBFIGURE INSTANCE* object shall specify the name and value appropriate to the usage of this object. In particular, the
   schematic_number,
   lep_number,
   assembly_number
   of the top level LEP shall be referenced by the *SUBFIGURE INSTANCE* depicting
   substrate_physical_outline

Translation Usage Notes:

General:

Output:

Input:



IGES graphic file: GENPART1

5.3.2.7. Hole

Description:

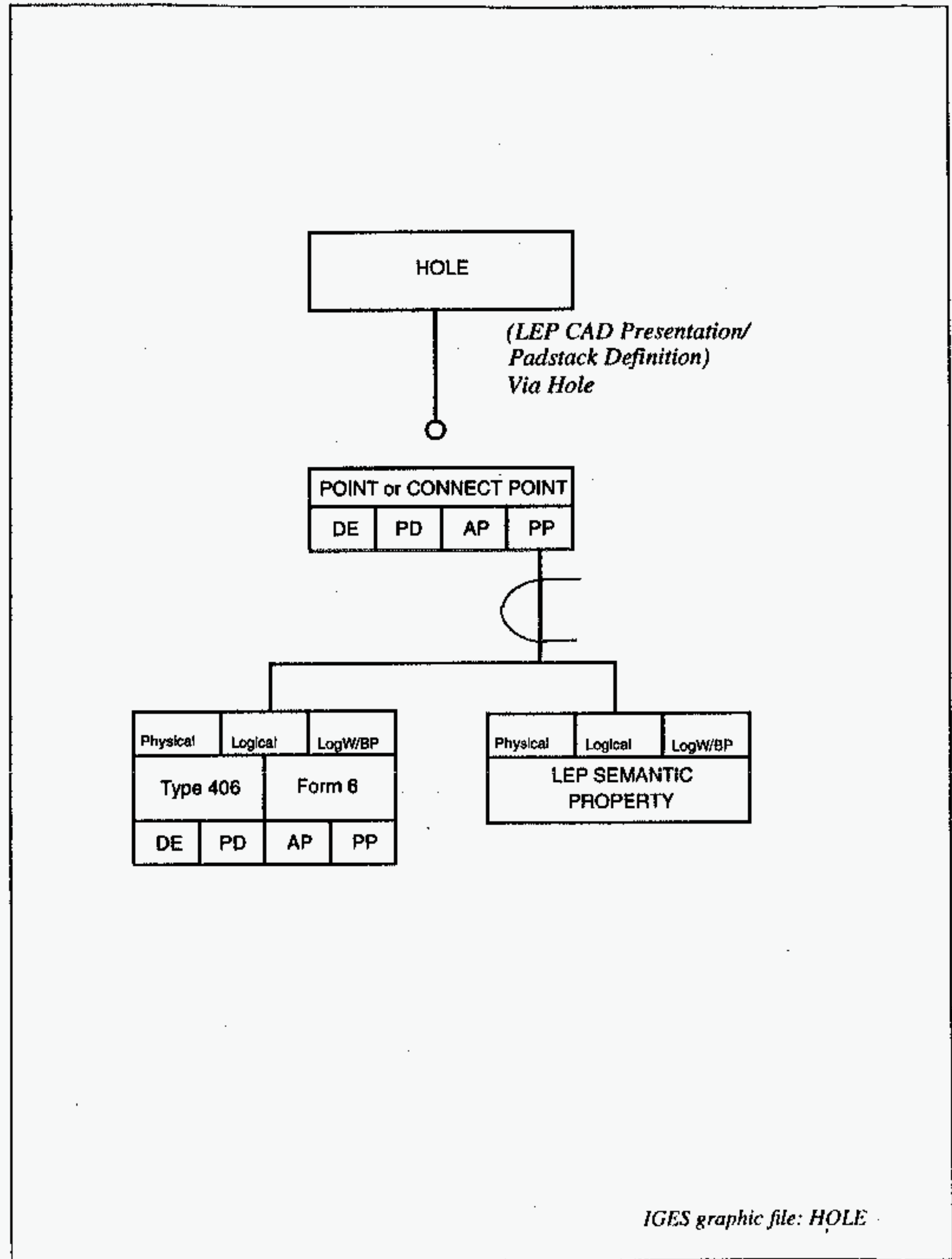
The *HOLE* object documents the location of a hole in one or more layers of the LEP. The hole may be any shape and may be manufactured in a variety of ways (e.g., drilled, punched, milled, or burned).

Requirements/Restrictions:

1. If the *HOLE* is subordinate to a definition that is intended to be instantiated on the LEP (e.g., Package Figure Definition), it shall reference a *PADSTACK DEFINITION*. If the *HOLE* is subordinate to a *LEP DEFINITION*, it shall reference a *PADSTACK INSTANCE*.
2. The *LEP SEMANTIC PROPERTY*, which is referenced from the *POINT* or *CONNECT POINT* object, shall specify

   physical_outline.
3. The Drilled Hole Property Entity (Type 406, Form 6) shall be attached to a Point Entity (Type 116) or Connect Point Entity (Type 132). Either the Point or Connect Point may be independent or dependent (e.g., padstack)

Translation Usage Notes:
General:
Output:
Input:



5.3.2.8. Join

Description:

The *JOIN* object logically represents any physical conductive material (e.g., conductive filled area, conductive path, or via) that is electrically common within a Net.

Requirements/Restrictions:

1. A *JOIN* object shall be part of (pointed to from) a *NET.*
2. The *LEP SEMANTIC PROPERTY* which is referenced from each *JOIN* object shall be specified as follows:

LEP SEMANTIC PROPERTY "Name" values

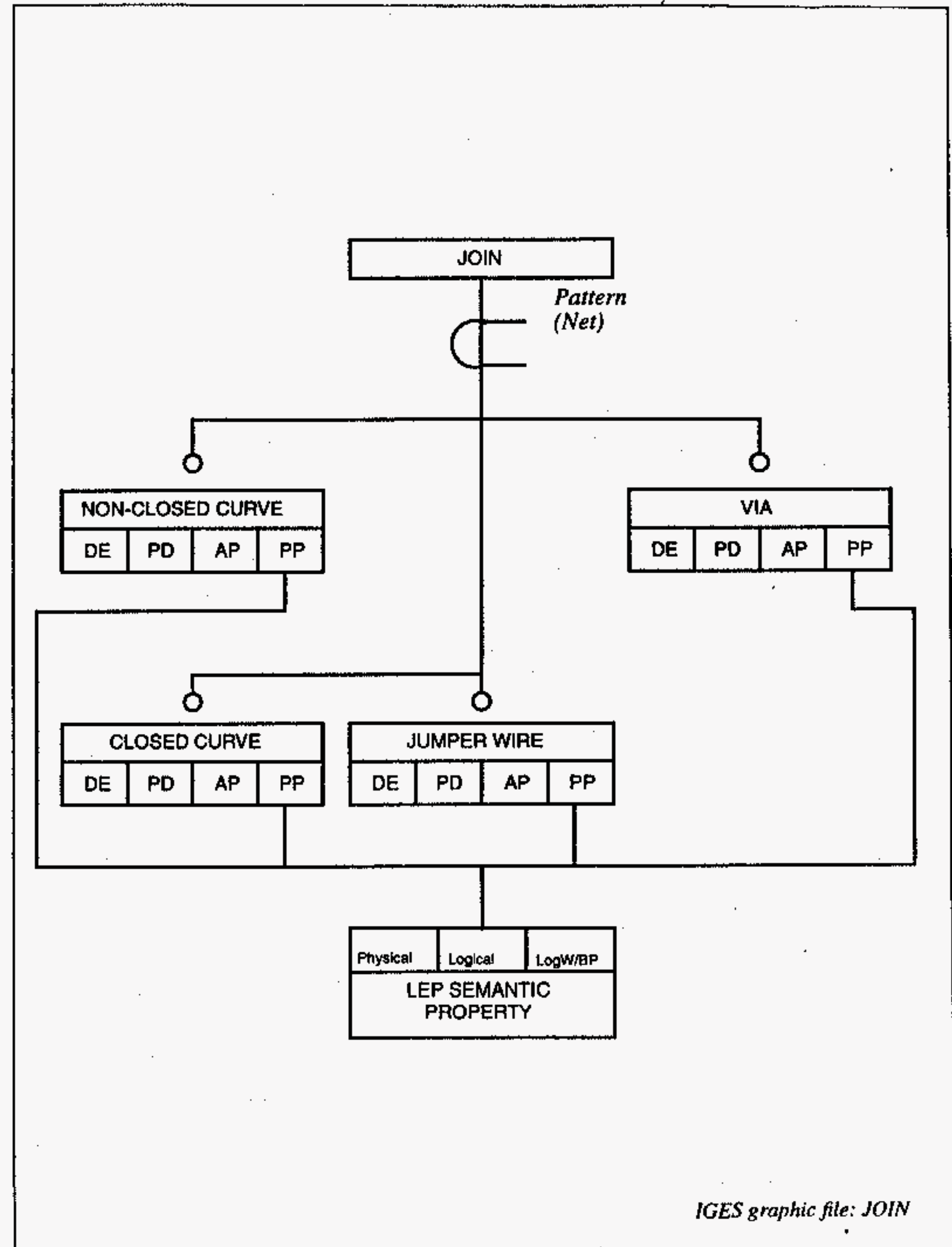
LEP object	LEP Sem. Prop. Name
Non-Closed Curve	Path
Closed Curve	Area
Jumper Wire	Jumper Wire
Via	Via

Translation Usage Notes:

General:

Output:

Input:
Each LEP *JOIN* object shall include an *LEP SEMANTIC PROPERTY* as defined in requirement 2 above.



IGES graphic file: JOIN

5.3.2.9. Jumper Wire

Description:

The *JUMPER WIRE* object (often called a “haywire,”) is a conductive wire that is used when necessary to electrically connect two or more conductive paths. It is typically used to join two separate substrate nets into one design net.

A *WIREBOND* is to be encoded as a *JUMPER WIRE* object.

Requirements/Restrictions:

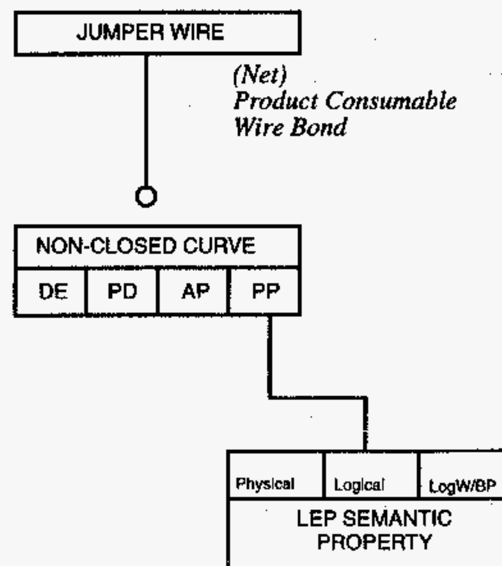
1. The non-closed curve object shall be tagged with an *LEP SEMANTIC PROPERTY* specified as follows:
   jumper_wire, or
   wire_bond.

Translation Usage Notes:

General:

Output:

Input:



IGES graphic file: JUMPWIRE

5.3.2.10. Keepin/Keepout

Description:

A closed-geometry-defined region of an LEP identified as having a design placement restriction.

Requirements/Restrictions:

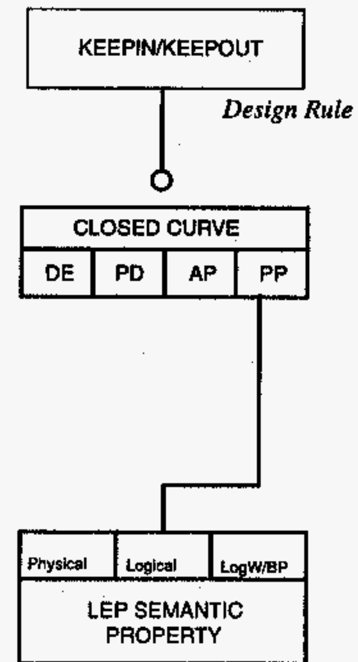
Translation Usage Notes:

General:

See *COMPONENT PLACEMENT KEEPIN/KEEPOUT, ROUTING KEEP/INKEEPOUT, TRACE KEEPIN/KEEPOUT* and *VIA KEEPIN/KEEPOUT* for the appropriate *GENERIC DATA PROPERTY* which applies.

Output:

Input:



5.3.2.11. Land

Description:

The *LAND* object records and depicts the geometry of a conductive pattern which is used for electrical connection or package *PIN* attachment.

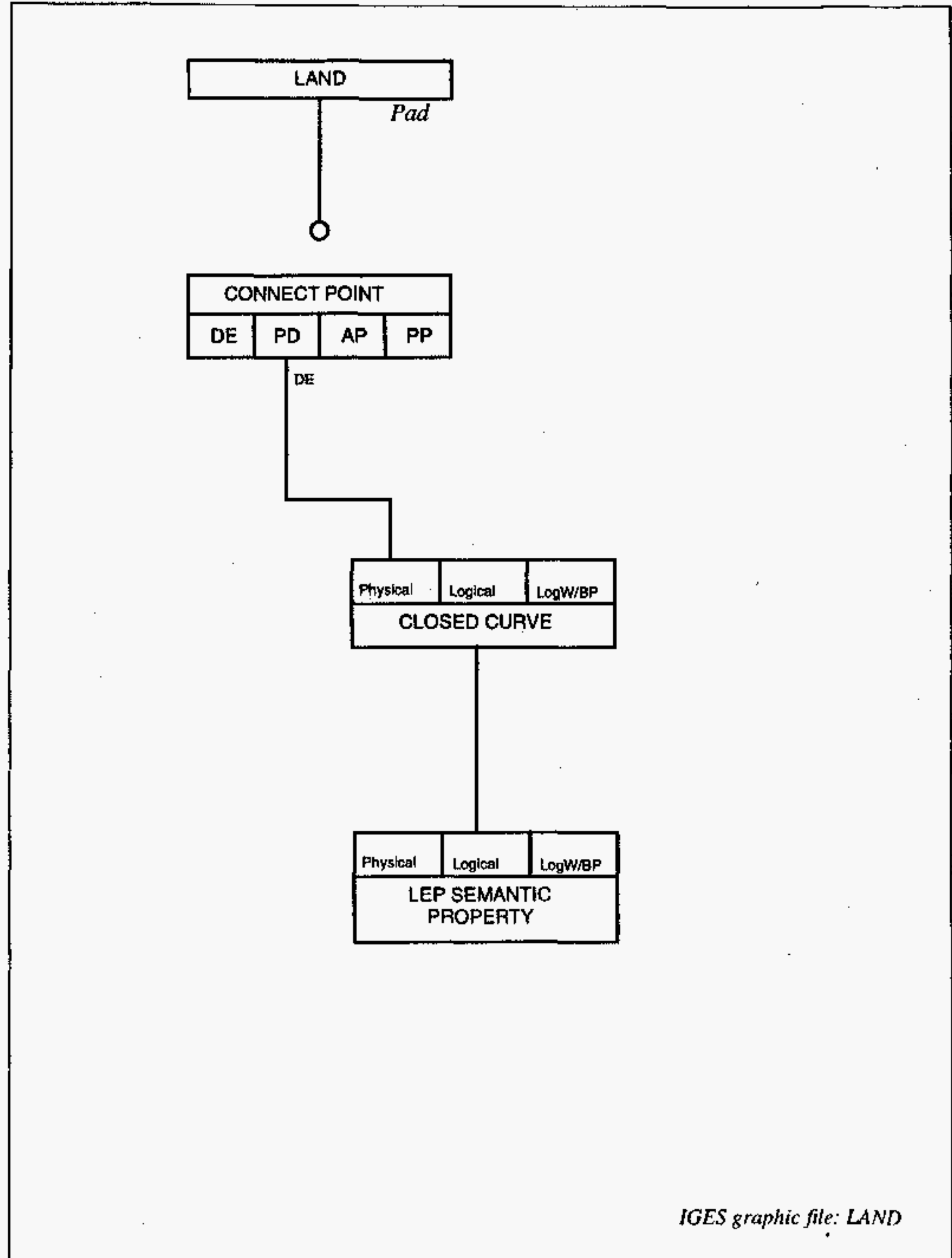
Requirements/Restrictions:

1. The *LEP SEMANTIC PROPERTY* which is referenced from the *CLOSED CURVE* shall specify
   pad_tech_type

Translation Usage Notes:

General:

When a *LAND* object is subordinate to a *PADSTACK DEFINITION*, the IGES file may have distinct land geometry for each possible *LAND* subtype. When a *LAND* object is subordinate to a *PADSTACK INSTANCE*, the IGES file shall have only one land geometry (anti, regular or thermal) for a particular padstack. The IGES file may have only one *LAND* object for each LEP functional layer that the particular padstack pierces.



IGES graphic file: LAND

5.3.2.12. Layer Outline

Description:

The *LAYER OUTLINE* object depicts the outline (boundary) for a specific physical layer of the LEP (i.e., substrate, power plane, etc.). At least one LAYER OUTLINE is included in an LEP file to assure that the design area is delimited; this object may be omitted from or included in photoplot tooling as required.

Requirements/Restrictions:

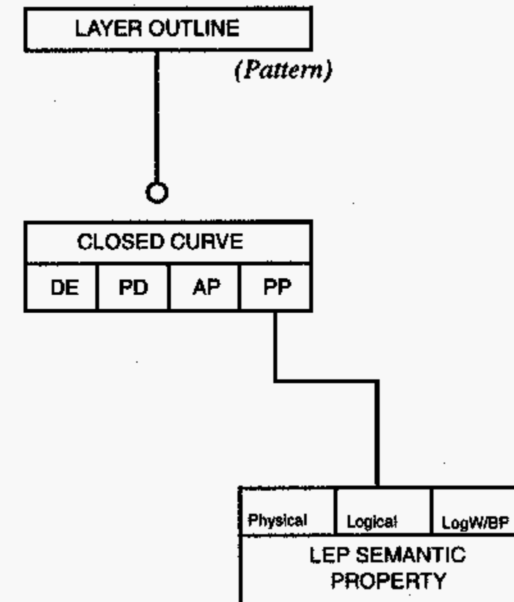
1. The *LEP SEMANTIC PROPERTY* which is referenced from the *CLOSED CURVE* shall specify
   lep_physical_outline

Translation Usage Notes:

General:

Output:

Input:



IGES graphic file: LAYEROUT.IGS

5.3.2.13. LEP Definition

Description:

The *LEP DEFINITION OBJECT* is a *NETWORK SUBFIGURE DEFINITION* object, which relates all of the constituent objects that are required to define the LEP.

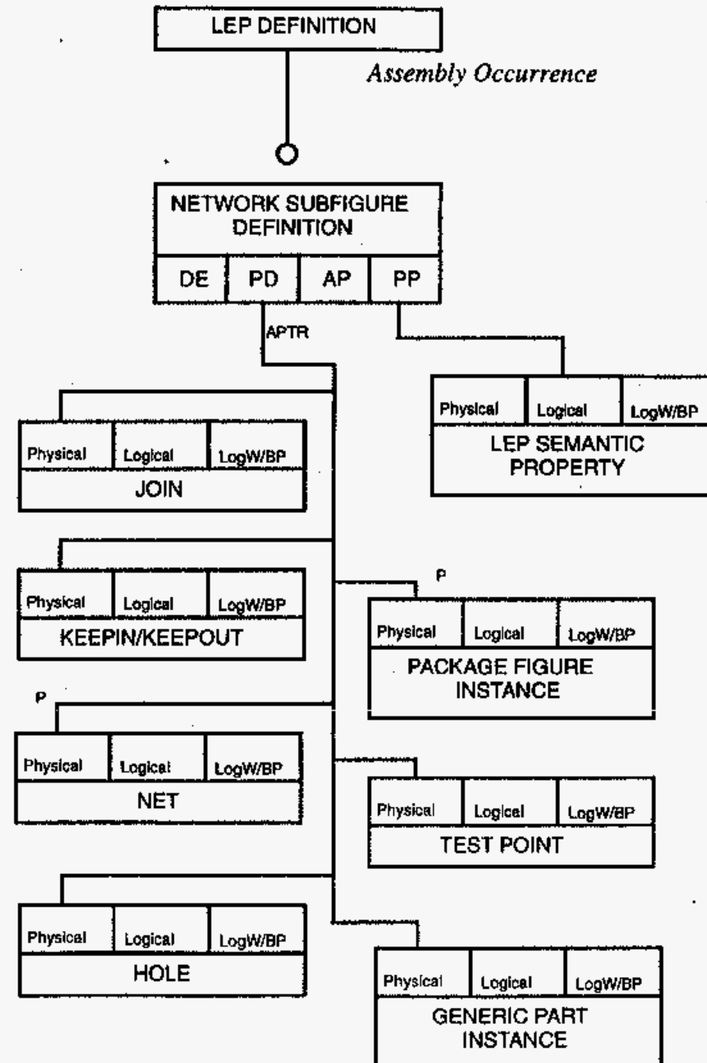
Requirements/Restrictions:

1. The LEP SEMANTIC PROPERTY, which is referenced from the *NETWORK SUBFIGURE DEFINITION* object, shall specify
   date and
   (schematic_number, or
   assembly_number).
2. The NAME field in the Network Subfigure Definition Entity (Type 320) shall be unique (for all LEP Definitions) within the scope of the IGES File.

Electrical and LEP Manufacturing Attribute List (ALT=5)

20 Electrostatic Discharge Rating
22 LEP Design Thickness

Translation Usage Notes:
General:
Output:
Input:



5.3.2.14. LEP Instance

Description:

The LEP Instance object is an instantiation of an LEP Definition.

Requirements/Restrictions:

1. The *GENERIC DATA PROPERTY*, which is referenced from the Network Subfigure Instance object, shall specify

   top_level_assembly_instance.

2. The following Attribute Table objects may be referenced from the LEP Instance:

Electrical Attribute List

6024 Generic Design Rule

Electrical & LEP Manufacturing Attribute List (ALT=5)

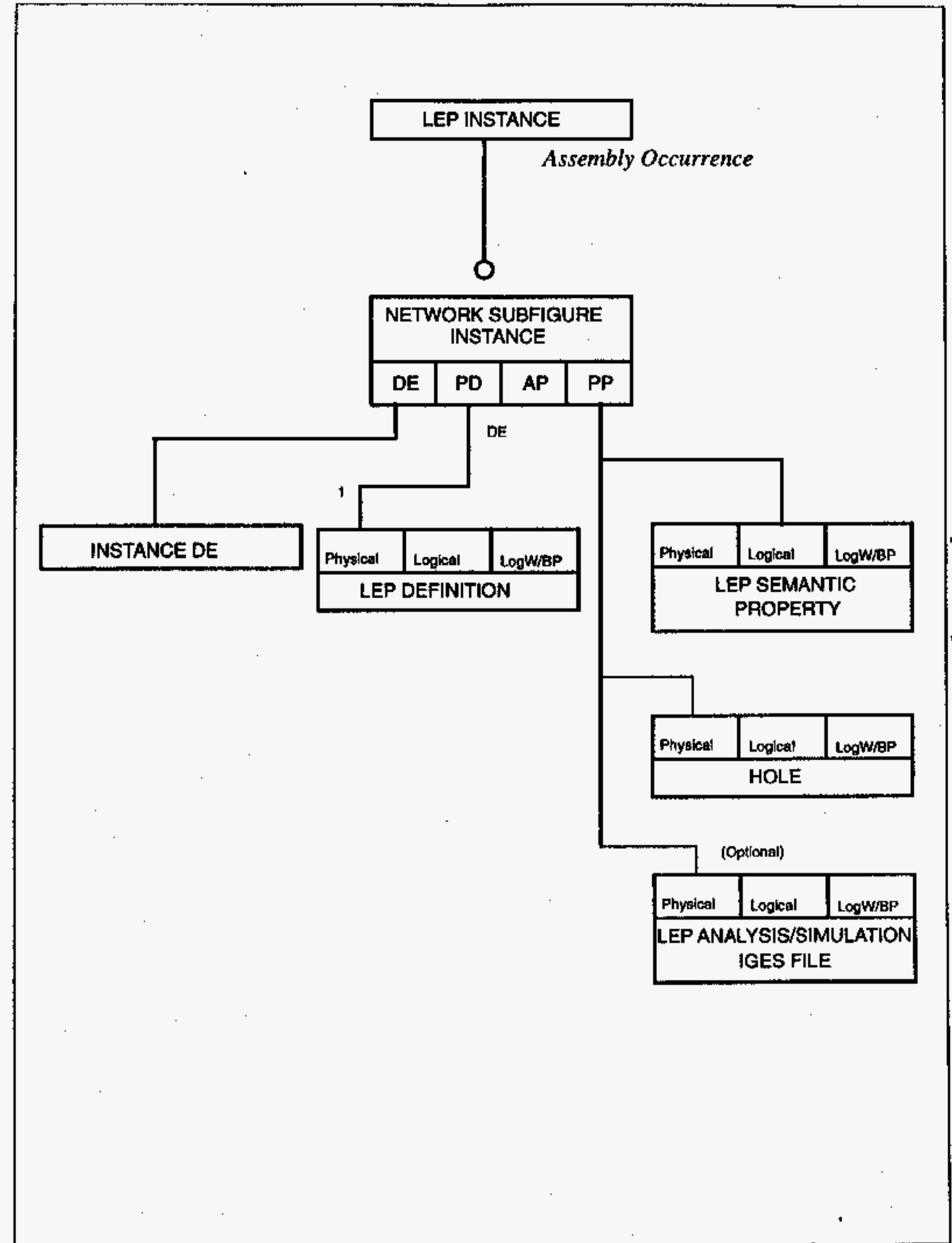
1 Component Physical Orientation
3 Component Physical Thickness
20 Electrostatic Discharge Rating
22 LEP Design Thickness

Translation Usage Notes:

General:

Output:

Input:



5.3.2.15. Level to LEP Layer Map

Description:

The *LEVEL TO LEP LAYER MAP* object maps directly to the IGES Level To LEP Layer Map Property (Type 406, Form 24). Please refer to IGES for additional information.

Requirements/Restrictions:

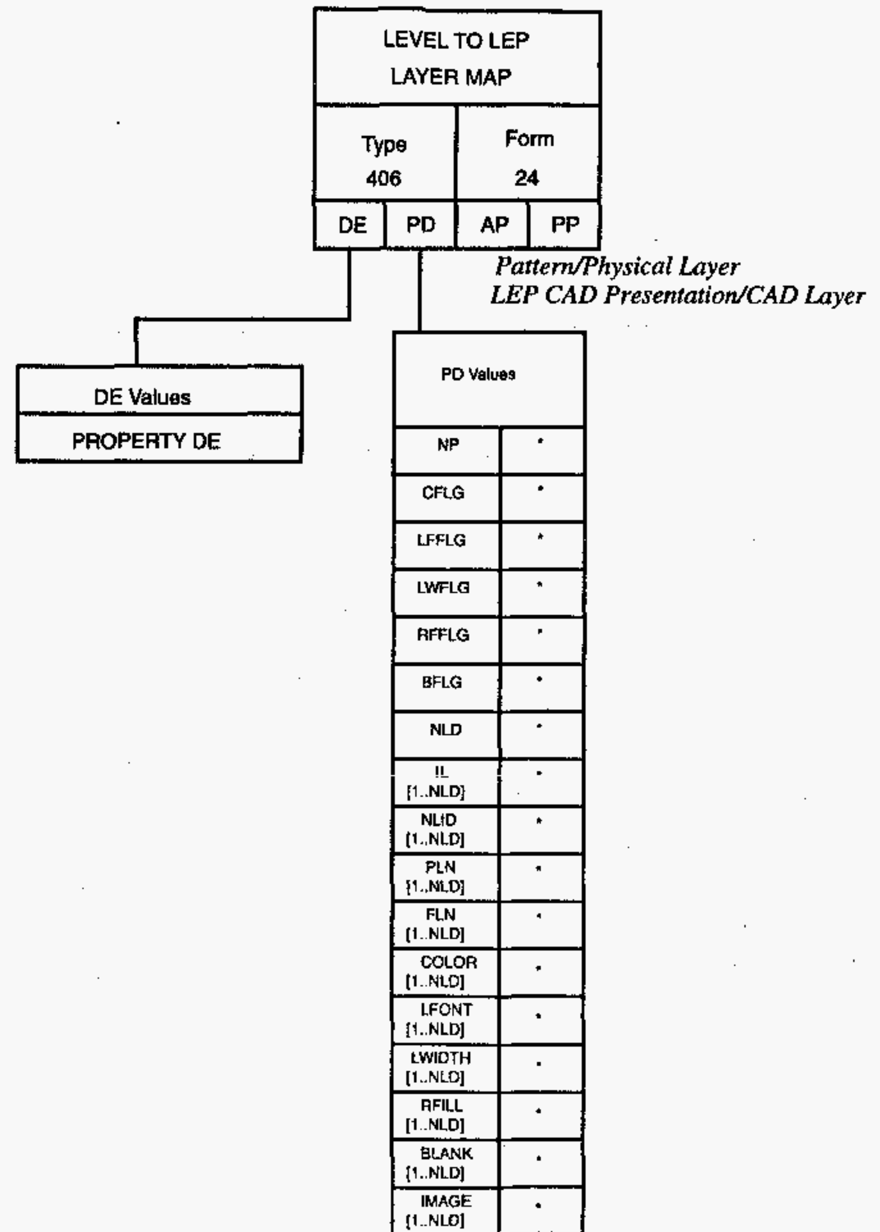
1. The Level To LEP Layer Map Property Entity (Type 406, Form 24) is always independent (DE Attribute Subordination equals 0) within the IGES file, thus it pertains to all other entities in the file.

Translation Usage Notes:

General:
Each IGES level (DE-5 or Type 406, Form 3) may have a level function property such as {trace_top, trace_bot, trace_2, trace_n, silkscreen_top ...}

Output:

Input:



IGES graphic file: LLPLACE

5.3.2.16. Net

Description:

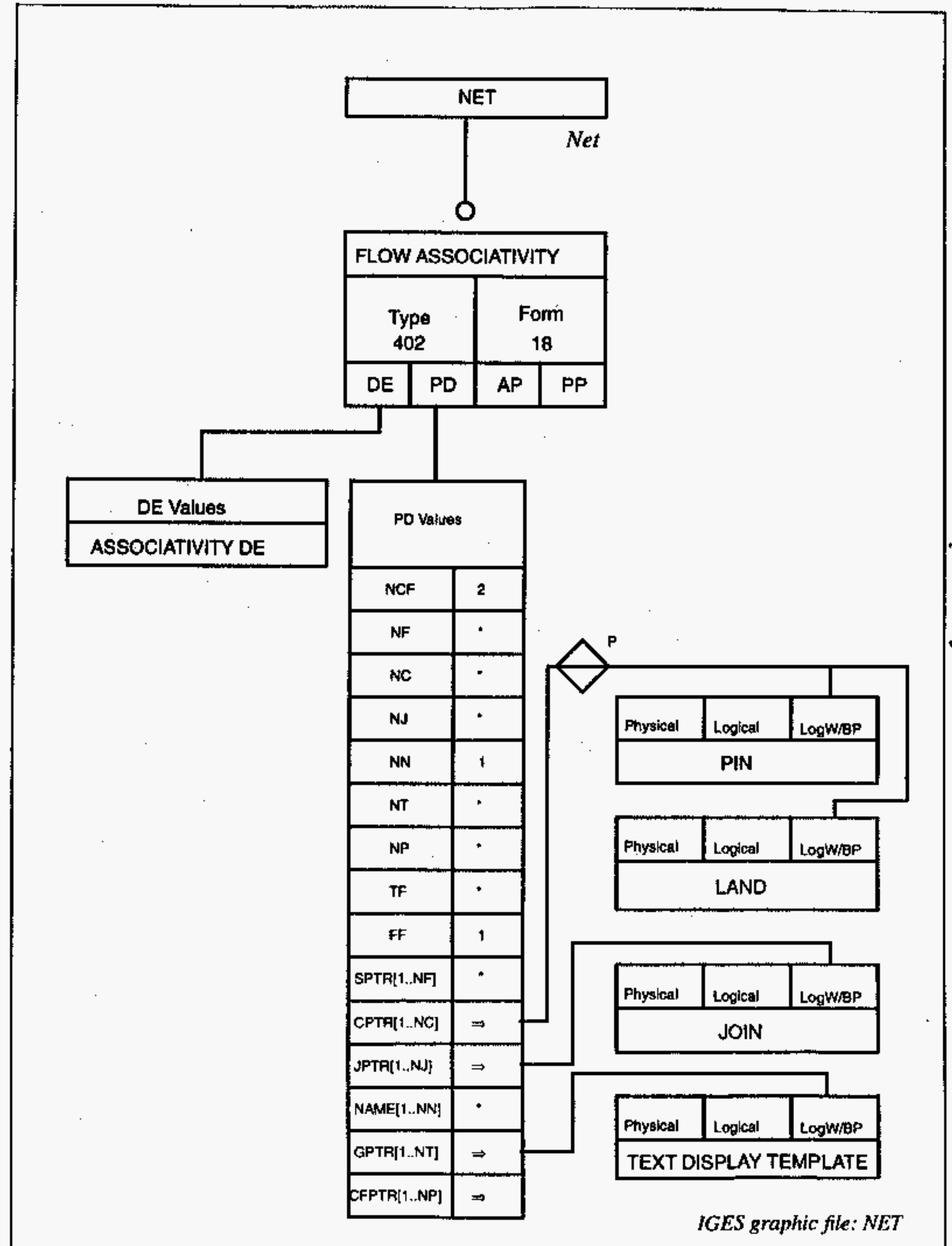
The *NET* object associates all of the electrically common *PIN*s and physical *JOIN* geometry (i.e., conductive filled areas, conductive paths and vias). The *NET* is intended to represent the LEP design net. If you are concerned with the substrate net you may analyze the structure of the *JOIN*s to determine which Pins are electrically connected at the substrate level (i.e., not through the use of *JUMPER WIRE*s).

Requirements/Restrictions:

1. The NAME field in the Flow Associativity Entity (Type 402, Form 18) shall be unique (For all *NET*s) within the scope of the LEP Definition.
2. Pins may be referenced by a *PIN* object.
3. All objects that are subordinate to the Flow Associativity Entity (Type 402, Form 18)—with the exception of Pins—shall be physically dependent on the same parent as the Flow Associativity.
4. There shall be 2 or more objects subordinate to the *NET*, which are either *PIN* objects, *LAND* objects, *VIA* objects, or *JUMPER* objects.

Translation Usage Notes:

General:



IGES graphic file: NET

The Text Display Template Entity (Type 312)shall be used to display the net name (PD NAME) or identification.

Output:

Input:

5.3.2.17. Package Figure Definition

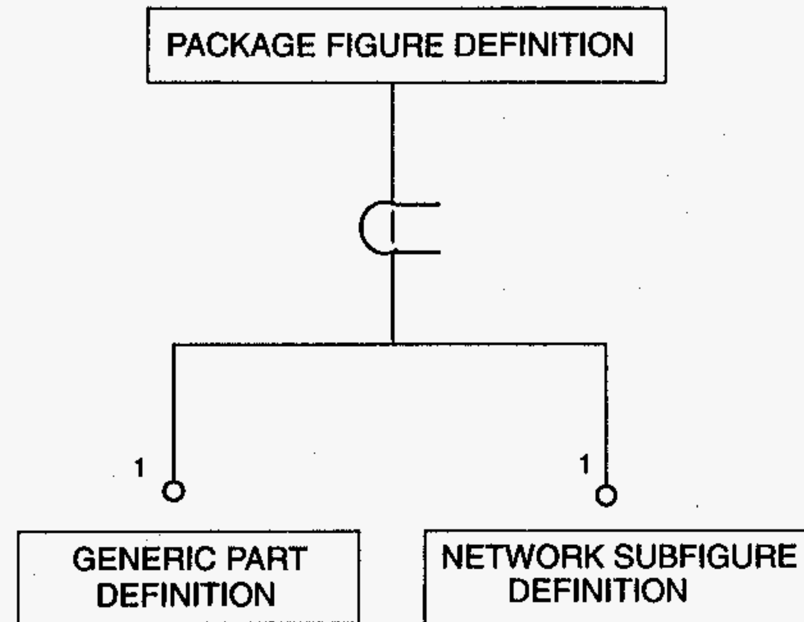
Description:

The *PACKAGE FIGURE DEFINITION* object is a Subfigure Definition Entity (Type 308) or a Network Subfigure Definition Entity (Type 320), which specifies all of the constituent objects required to define, as a minimum, the "footprint" of a package figure which is assembled to the LEP.

Requirements/Restrictions:

1. The *GENERIC PART DEFINITION* object shall be used to represent the definition of a non-electrical component (an attached part).
2. The NETWORK SUBFIGURE DEFINITION object shall be used to represent the definition of an electrical component (an attached electrical part).
3. The *GENERIC DATA PROPERTY,* which is referenced from the *PACKAGE FIGURE DEFINITION* object, shall specify
   - component_std_name
   - component_version
   - component_pkg_type
   - component_tech_type
   - component_layout_surface
   - component_insertion_origin
   - component_placement_feeder_location (*)
   - component_insertable (*)
   - component_height
   - component_max_size
   - component_outline_overhang (*).

* When applicable.



Translation Usage Notes:

General:
The part representation may be a symbolic figure, a minimal outline, a 2-dimensional detail, or a 3-dimensional detail as required for the application.

Output:
Input:

5.3.2.18. Package Figure Instance

Description:

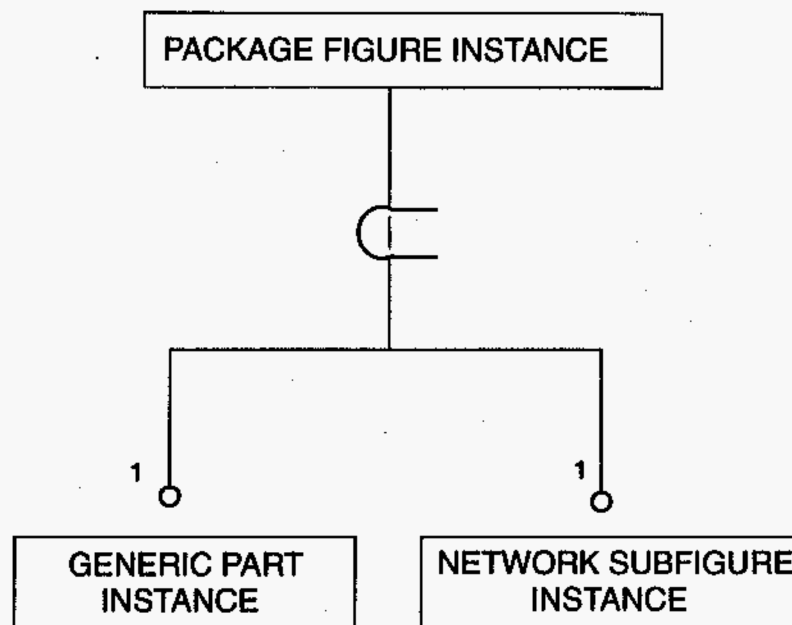
The *PACKAGE FIGURE INSTANCE* object is a Subfigure Instance Entity (Type 408) or a Network Subfigure Instance Entity (Type 420), which specifies all of the constituent objects required to define, as a minimum, the "footprint" of a package figure which is assembled to the LEP.

Requirements/Restrictions:

1. The *GENERIC PART INSTANCE* object shall be used to instance the definition of a non-electrical component (an attached part).
2. The *NETWORK SUBFIGURE INSTANCE* object shall be used to instance the definition of an electrical component (an attached electrical part).
3. The *LEP SEMANTIC PROPERTY,* which is referenced from either sub-type of the *PACKAGE FIGURE INSTANCE* object, shall specify
   abs_voltage_max
   component_instance_side
   component_preplace (*)
   component_via (*)
   standalone_via (*).

* When applicable.

Translation Usage Notes:
General:
Output:
Input:



5.3.2.19. Padstack Definition

Description:

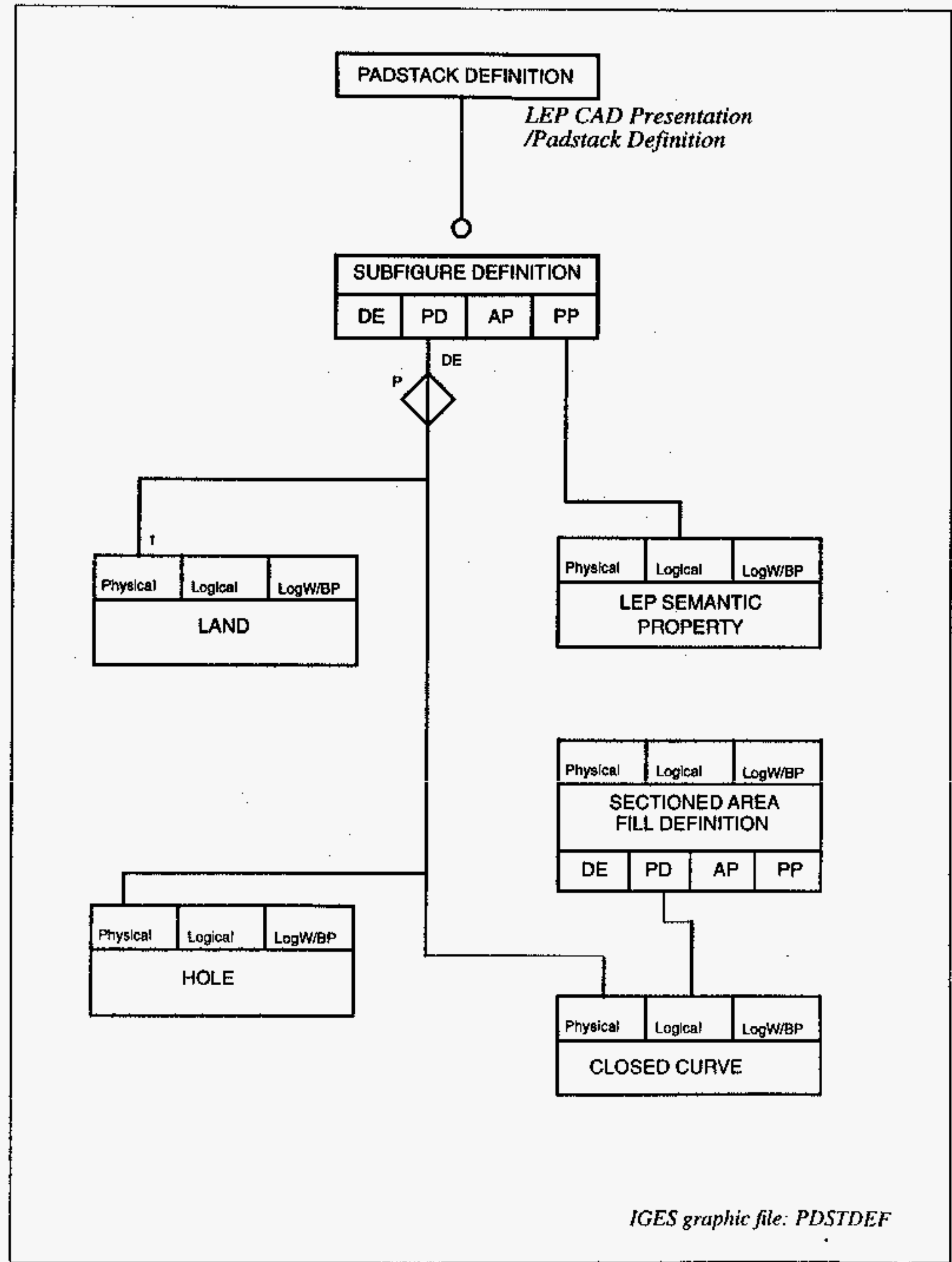
The *PADSTACK DEFINITION* object is a Subfigure Definition Entity (Type 308), which documents and depicts all of the constituent objects, that are required to list the characteristics of the volume in which a package figure is electrically connected to the LEP.

The characteristics of each layer that is associated with the padstack shall be defined within the *PADSTACK DEFINITION*.

Requirements/Restrictions:

1. The *LEP SEMANTIC PROPERTY*, which is referenced from the *PADSTACK DEFINITION* object, shall specify
   physical_padstack_definition,
   component_pkg_type (= pad).

Translation Usage Notes:
General:
The DE Level attribute of the *HOLE* shall be used to specify which levels (physical layers) the padstack pierces. In order to affect more than one level, a *LEVEL DEFINITION* object shall be used.
Output:
Input:



IGES graphic file: PDSTDEF

5.3.2.20. Padstack Instance

Description:

The *PADSTACK INSTANCE* object is an instantiation of a *PADSTACK DEFINITION*.

Requirements/Restrictions:

1. The *LEP SEMANTIC PROPERTY*, which is referenced from the *GENERIC PART DEFINITION* object, shall specify
   (allowable_test_point or
   not_allowable_test_point),
   component_default_padstack,
   pad_tech_type (= smt), or
   pad_tech_type (= thru_hole_anti), or
   pad_tech_type (= thru_hole_regular), or
   pad_tech_type (= thru_hole_thermal).

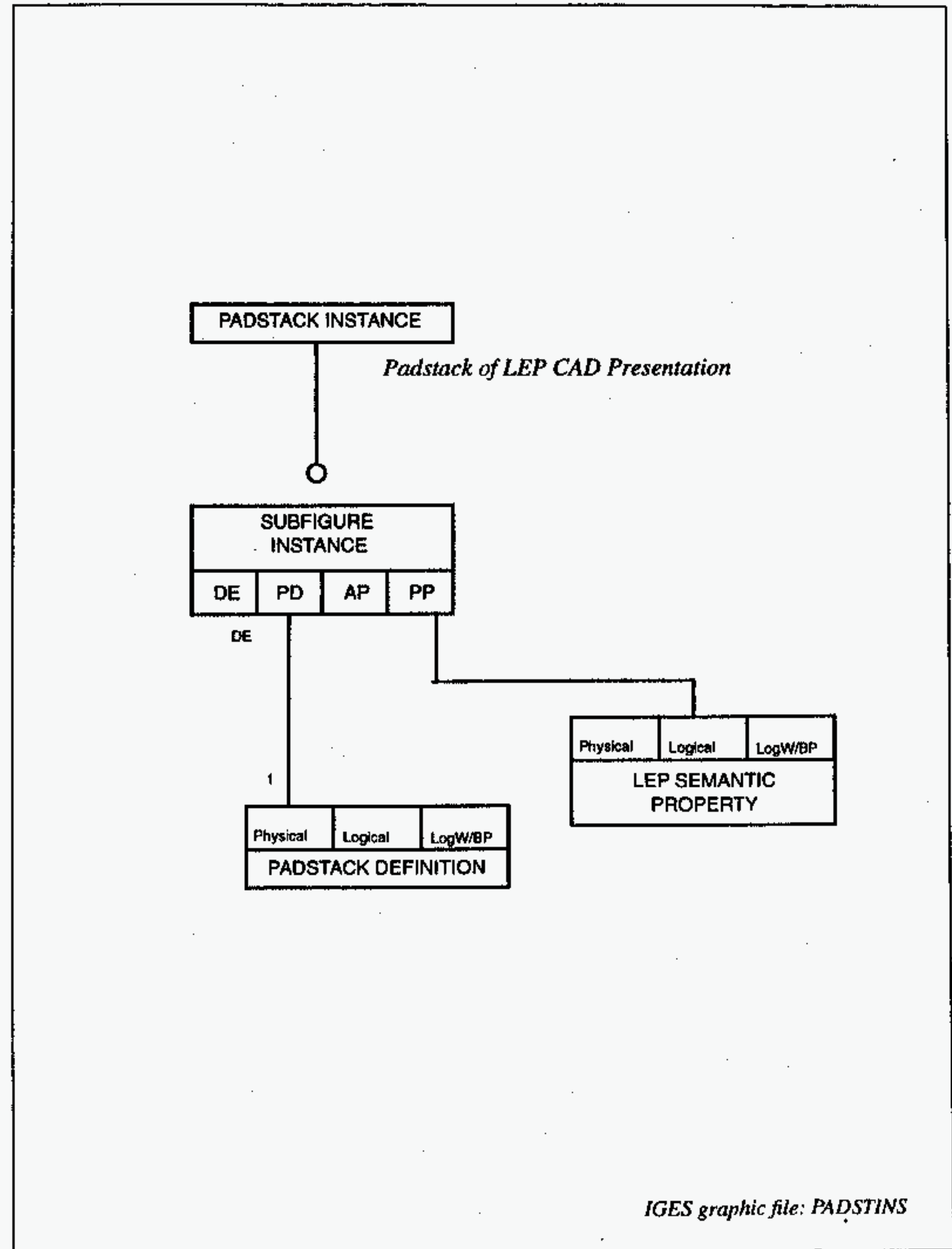
Translation Usage Notes:

General:

The DE Level attribute is used to indicate which levels (physical layers) the padstack affects. In order to affect more than one level, a *LEVEL DEFINITION* object shall be used.

Output:

Input:



IGES graphic file: PADSTINS

5.3.2.21. Panel Definition

Description:

The *PANEL DEFINITION* object is a Subfigure Definition Entity (Type 308), which relates all of the constituent objects that are required to depict a manufacturing panel.

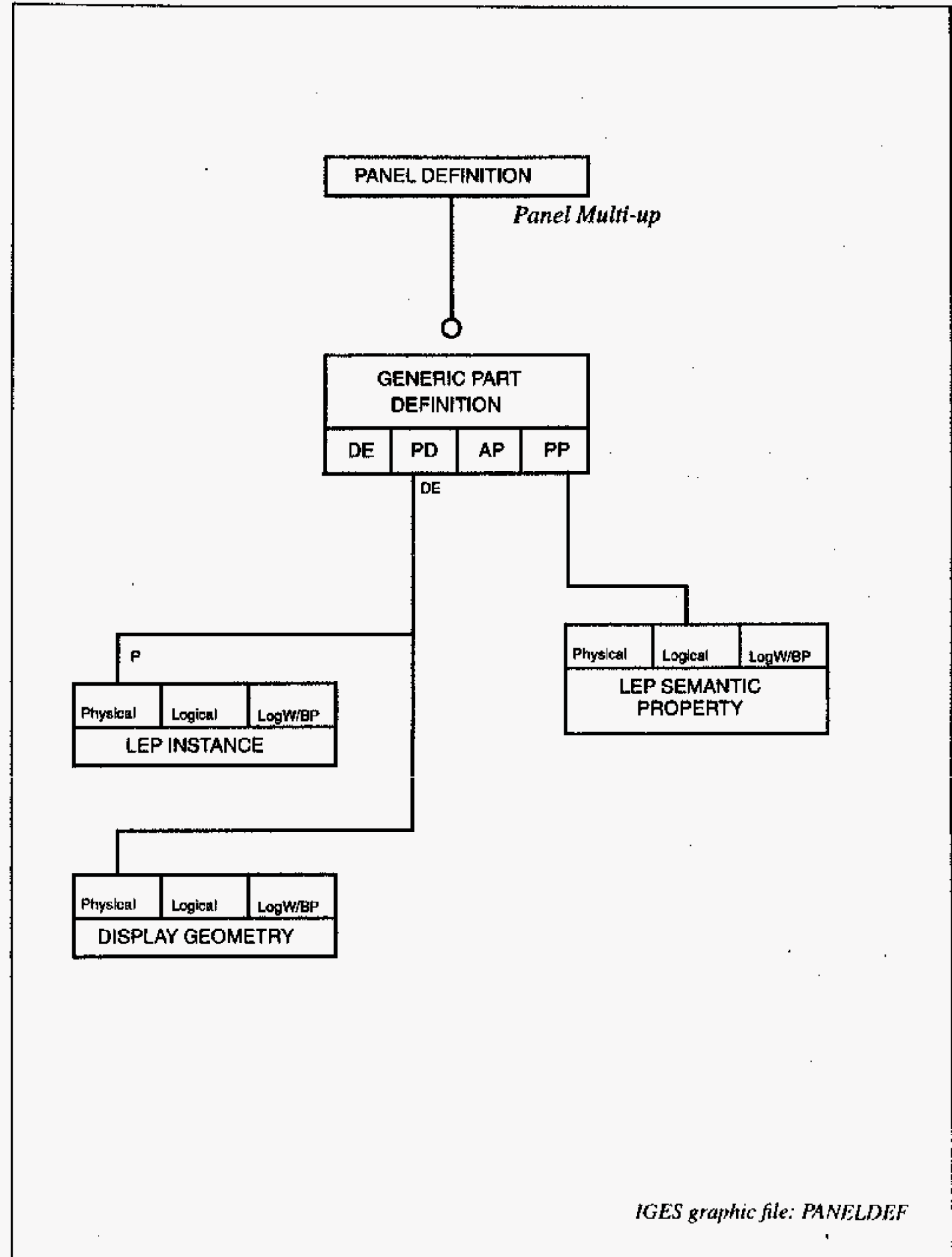
Requirements/Restrictions:

1. The *LEP SEMANTIC PROPERTY*, which is referenced from the *GENERIC PART DEFINITION* object, shall specify
   panel.
2. The NAME field in the *SUBFIGURE DEFINITION* shall be unique (for all *PANEL DEFINITIONs*) within the scope of the IGES File.

Translation Usage Notes:

General:
The *DISPLAY GEOMETRY* is included only as needed to describe the Panel. *DISPLAY GEOMETRY* may also reference an LEP SEMANTIC PROPERTY with a value of fiducial.
Output:

Input:



IGES graphic file: PANELDEF

5.3.2.22. Panel Instance

Description:

The *PANEL INSTANCE* object is an instantiation of a *PANEL DEFINITION*.

Requirements/Restrictions:

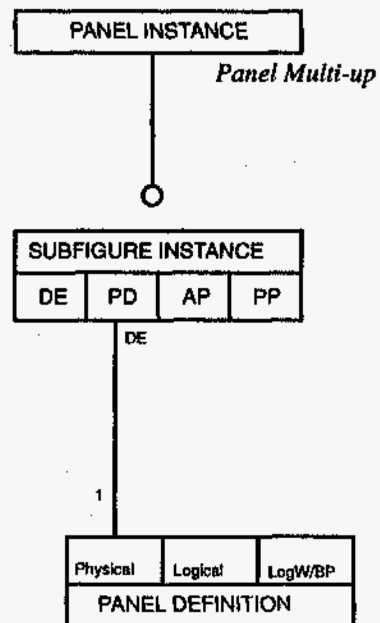
1. The *PANEL INSTANCE* may be implemented using a Subfigure Instance Entity (Type 408).

Translation Usage Notes:

General:

Output:

Input:



IGES graphic file: PANELINS

5.3.2.23. Part Placement Boundary

Description:

The *PART PLACEMENT BOUNDARY* object is a closed curve, which shows the area on the LEP that a Package Figure Definition encompasses. The Part Placement Boundary encompasses a volume formed by an XY area plus a lower and upper height for the area.

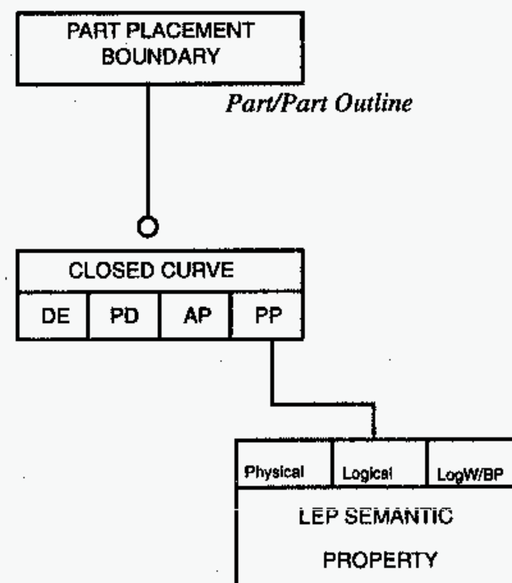
Requirements/Restrictions:

1. The *CLOSED CURVE* shall only be an Arc, a Simple Planar Curve Entity (Type 106, Form 63), or a Subfigure Definition Entity (Type 308) which is coplanar, closed, and non-crossing.
2. The *LEP SEMANTIC PROPERTY*, which is referenced from the *GENERIC PART DEFINITION* object, shall specify
   component_max_size,
   component_keep_outline.

Translation Usage Notes:
This object may exist on any level.
General:

Output:

Input:
LEP 'Part Placement_Outline' Property is an IGES LEP SEMANTIC PROPERTY.



5.3.2.24. Pin

Description:

The *PIN* object documents the location, and functional characteristics of the electrical connection of a package pin to the LEP.

Requirements/Restrictions:

1. There is no subordination associated with the Subfigure Instance (PSFI) pointer. It is an informational reference only.
2. The Connect Point Identifier (CPID) field in the Connect Point shall be unique within the scope of the Package Figure Definition or Package Figure Instance.

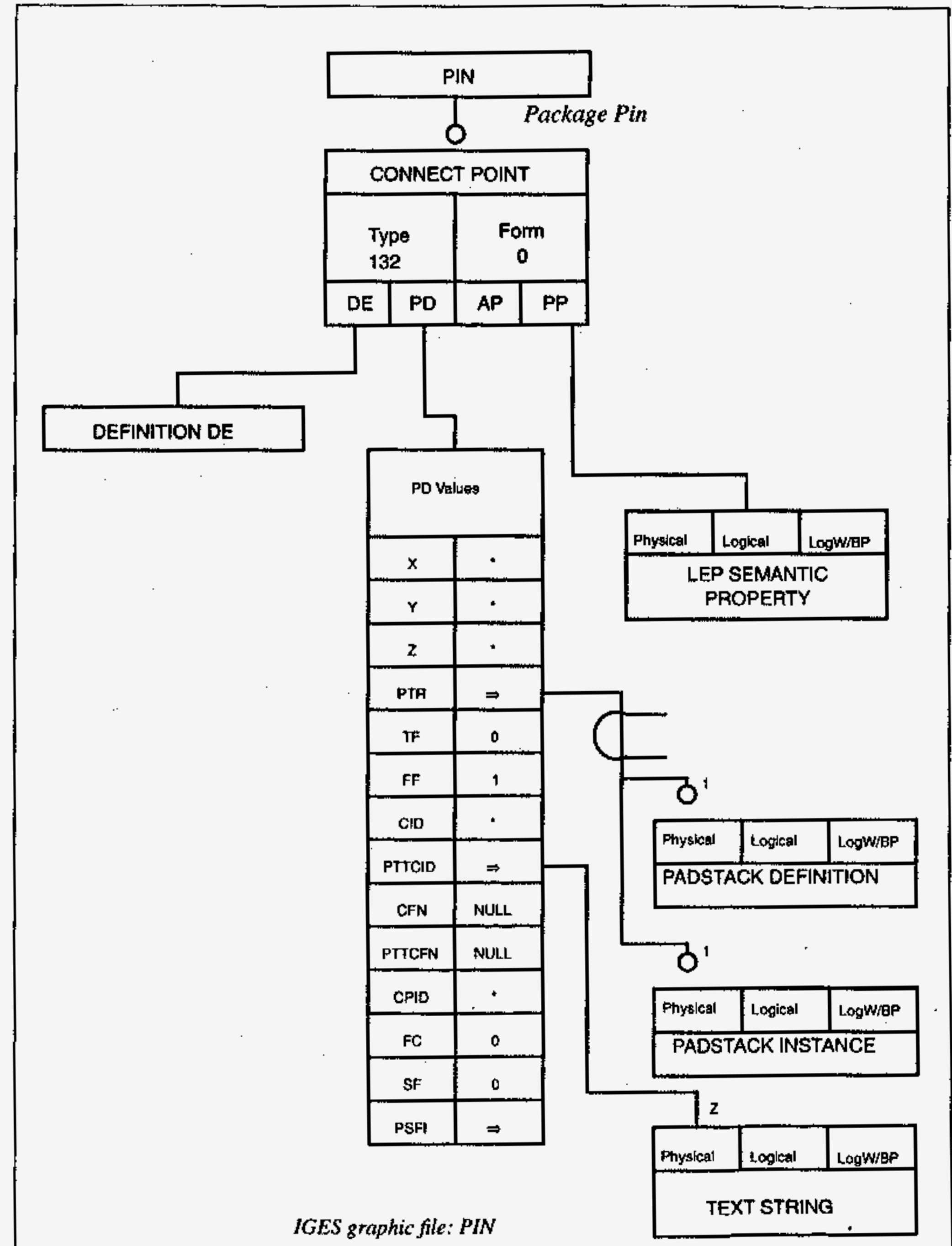
Translation Usage Notes:

General:
The value for the *LEP SEMANTIC PROPERTY* for the *PIN* designated as "pin 1" shall be:

pin1

Output:

Input:



IGES graphic file: PIN

5.3.2.25. Routing Keepin/Keepout

Description:

The *ROUTING KEEPIN* object is a *CLOSED CURVE*, which represents an outline which totally encloses all routing objects (e.g., conductive filled areas, conductive paths, and vias).

Requirements/Restrictions:

1. The *LEP SEMANTIC PROPERTY*, which is referenced from the *GCLOSED CURVE* object, shall specify

   trace_keep_outline

Translation Usage Notes:

General:

Output:

Input:

see *LEP KEEPIN/KEEPOUT* (Section 5.3.2.10.).

5.3.2.26. Test Point

Description:

The *TEST POINT* object indicates a location which is designated for probing during a specific electrical test procedure (e.g., in-circuit, bare board, functional).

Requirements/Restrictions:

1. An *LEP SEMANTIC PROPERTY* shall specify
   allowable_test_point
   which shall be referenced from the object representing the *TEST POINT.*

Translation Usage Notes:

General:

Output:

Input:

see *LEPSEMANTIC PROPERTY* (Section 5.3.5.).

5.3.2.27. Trace Keepin/Keepout

Description:

The *TRACE KEEPIN/KEEPOUT* object is a closed curve, which represents an outline in which all or no traces (e.g., conductive areas or conductive paths) shall be placed.

Requirements/Restrictions:

Translation Usage Notes:

General:
The values for the *LEP SEMANTIC PROPERTY* shall be:

trace_keep_outline and
(keepin or
keepout)

Output:

Input:

see *KEEPIN/KEEPOUT* (Section 5.3.2.10.).

5.3.2.28. Via

Description:

The *VIA* object represents a special type of Join. It is used to electrically connect conductive material between the physical layers of the LEP.

Requirements/Restrictions:

1. The *GENERIC DATA PROPERTY*, if referenced from the *PIN* object, shall specify
   component_via.
2. If the Via is subordinate to a definition that is intended to be instantiated on the LEP (i.e., Package Figure Definition), it shall reference a Padstack Definition. If the Via is subordinate to a LEP Definition, it shall reference a Padstack Instance.

Translation Usage Notes:

General:

Output:

Input:

see *PADSTACK OBJECTS* (5.3.2.19. and 5.3.2.20.).

5.3.2.29. Via Keepin/Keepout

Description:

The *VIA KEEPIN/KEEPOUT* object is a *CLOSED CURVE*, which represents an outline in which all or no Vias shall be placed.

Requirements/Restrictions:

Translation Usage Notes:

General:
The values for the LEP SEMANTIC Property shall be as follows:

via_keep_outline and
(keepin or
keepout)

Output:

Input:

see *KEEPIN/KEEPOUT* (Section 5.3.2.10.).

5.3.2.30. Wire Bond

Description:

The *WIRE BOND* object is a special type of *JOIN*. It is used to electrically connect a pre-packaged IC or HMA to the surface of an MCM. A Wire Bond is typically soldered, thermally bonded, or compression bonded to a special type of Padstack (wire bond pad).

Requirements/Restrictions:

1. The *LEP SEMANTIC PROPERTY*, which is referenced from the Non-Closed Curve object, shall specify
   wire_wrap
   gauge
   wire_color
   twisted_pair
   wrap_class (optional).

Translation Usage Notes:

General:

Output:

Input:

see *JUMPER WIRE* (Section 5.3.2.8.).

BLANK PAGE

5.3.3. Display Geometry Object Models

These objects are used to depict shapes of physical objects in the layered electrical product and are also common to files describing mechanical products. (see ARM entity *Artwork Geometry*)

5.3.3.1. Arc

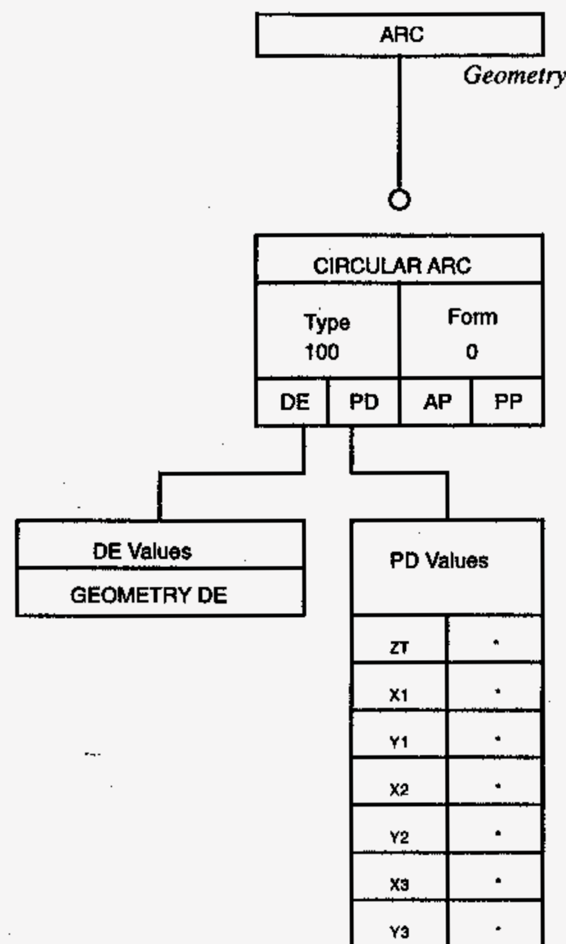
Description:

The *ARC* object represents a non-closed circular arc.

Requirements/Restrictions:

1. The start and end point of the IGES Circular Arc Entity (Type 100) shall not be coincident (with respect to the Global Section Minimum User-Intended Resolution).
2. The radius shall be greater than or equal to the Minimum User-Intended Resolution.

Translation Usage Notes:
General:
Output:
Input:



IGES graphic file: ARC

5.3.3.2. Circle

Description:

The *CIRCLE* object represents a closed circular arc.

Requirements/Restrictions:

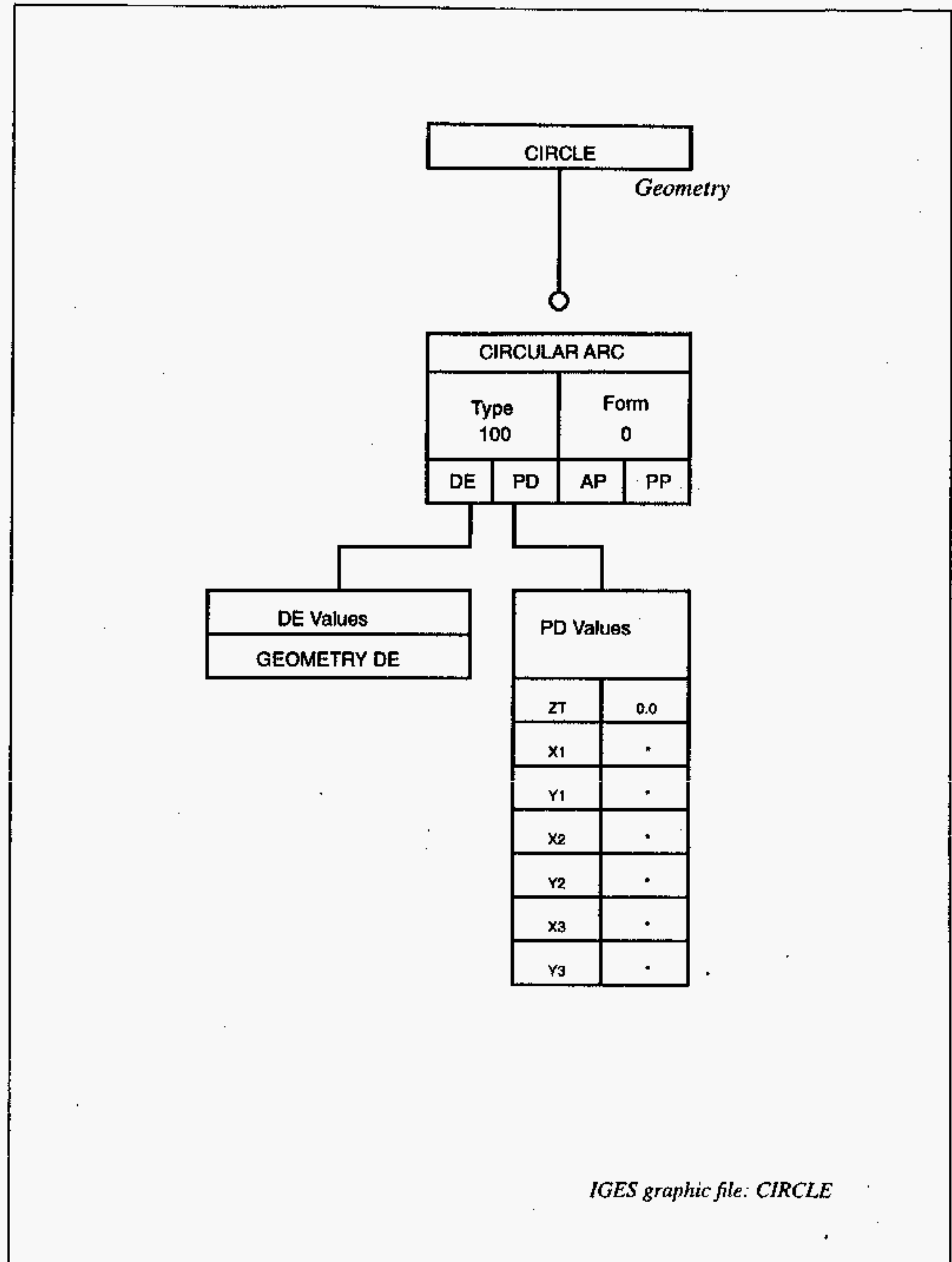
1. The start and end point of the IGES Circular Arc Entity (Type 100) shall be coincident (with respect to the Global Section Minimum User-Intended Resolution).
2. The radius shall be greater than or equal to the Minimum User-Intended Resolution.

Translation Usage Notes:

General:

Output:

Input:



IGES graphic file: CIRCLE

5.3.3.3. Closed Curve

Description:

The *CLOSED CURVE* object is any one of the pre-defined curves which is capable of defining a closed curve.

Requirements/Restrictions:

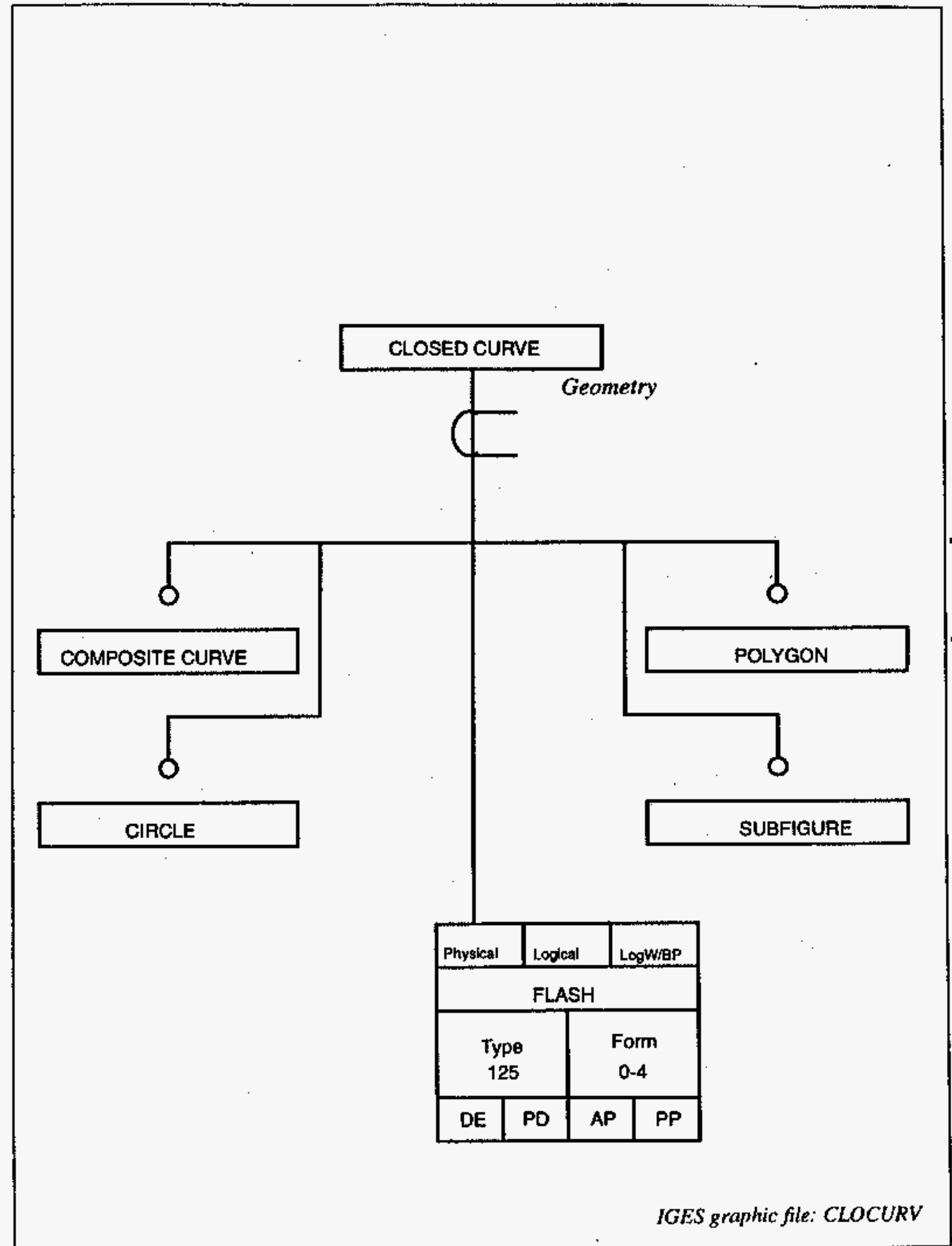
1. A *CLOSED CURVE* shall not self intersect or become tangent with itself at any point other than the start point which shall be coincident with the end point.

Translation Usage Notes:

General:

Output:

Input:



IGES graphic file: CLOCURV

5.3.3.4. Composite Curve

Description:

The Composite Curve object is a contiguous curve made up of a set of two or more subordinate curves.

Requirements/Restrictions:

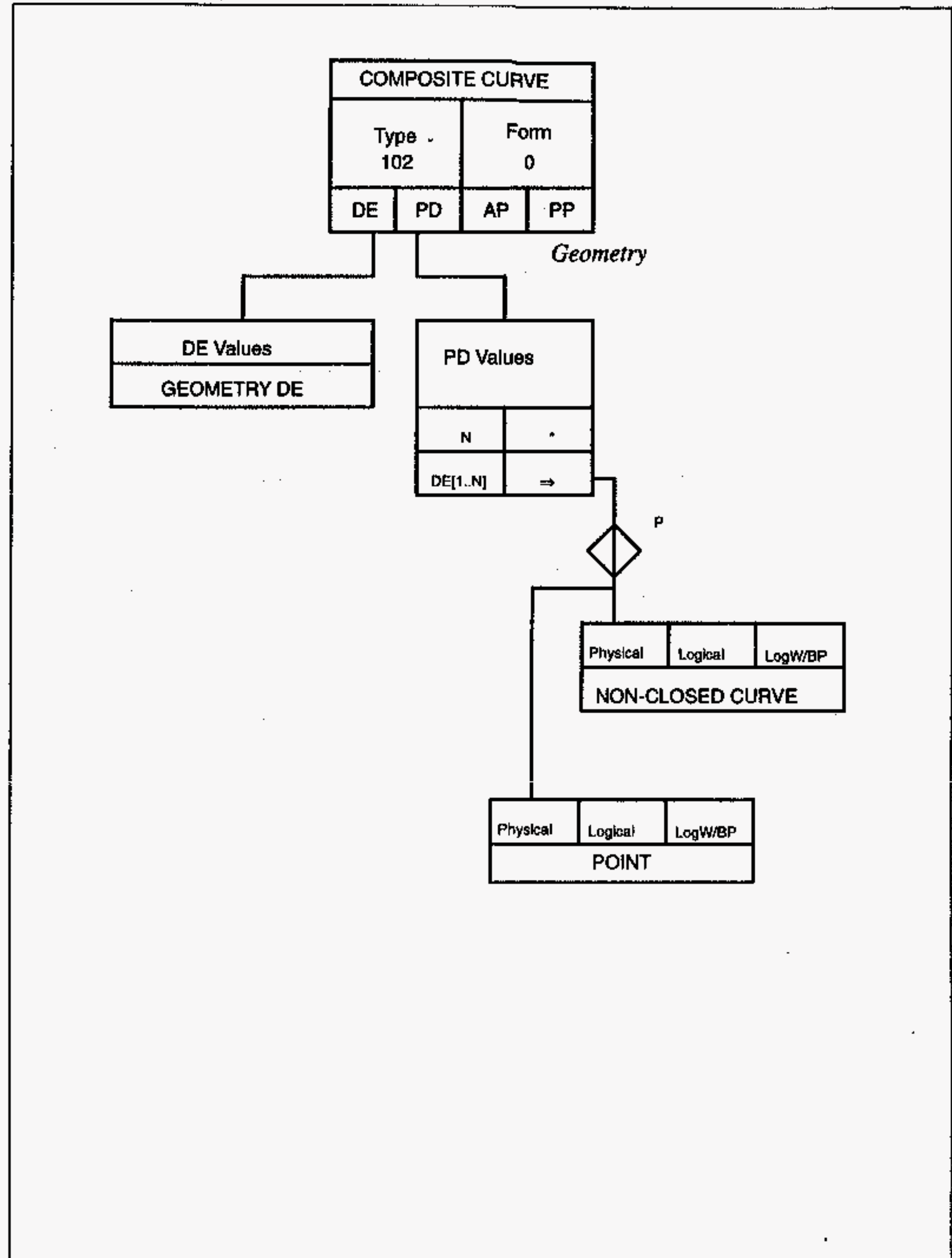
1. There shall be two or more subordinate curves.
2. The Composite Curve shall not reference another Composite Curve as a subordinate entity.
3. The Composite Curve may or may not be closed.
4. Two Points or Connect Points shall not appear consecutively in the defining list (see IGES section 4.4).

Translation Usage Notes:

General:

Output:

Input:



5.3.3.5. Geometry Figure Definition

Description:

The *GEOMETRY FIGURE DEFINITION* object is a Subfigure Definition Entity (Type 308), which contains only graphic display geometry objects.

Requirements/Restrictions:

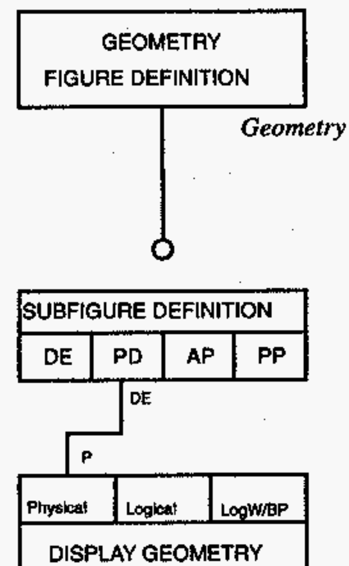
1. The LEP Object Type/Sub-Type property, which is referenced from the Subfigure Definition object, shall specify (*otype*=Geometry_Figure, *stype*=*).
2. The NAME field in the Subfigure Definition shall be unique (for all Geometry Figure Definitions) within the scope of the IGES File.

Translation Usage Notes:

General:

Output:

Input:



IGES graphic file: GEOFIGD

5.3.3.6. Geometry Figure Instance

Description:

The *GEOMETRY FIGURE INSTANCE* Object is either a Simple Geometry Figure Instance or a Modified Geometry Figure Instance.

Requirements/Restrictions:

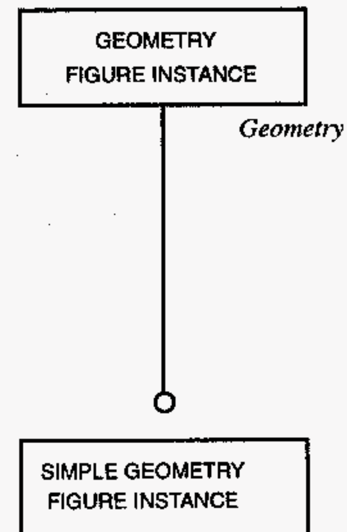
1. The LEP Object Type/Sub-Type property, which is referenced from the Subfigure Instance object, shall specify (*otype*=Geometry_Figure, *stype*=*).

Translation Usage Notes:

General:

Output:

Input:



IGES graphic file: GEOFIGI

5.3.3.7. Line

Description:

The *LINE* object represents a line segment.

Requirements/Restrictions:

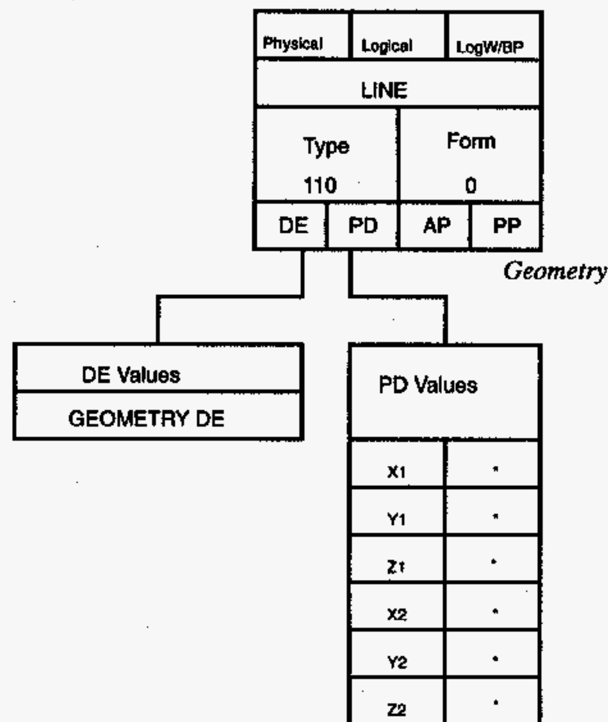
1. The start and end point of the IGES Line Entity (Type 110) must not be coincident (with respect to the Global Section Minimum User-Intended Resolution).

Translation Usage Notes:

General:

Output:

Input:



IGES graphic file: LINE

5.3.3.8. Multi-Line

Description:

The *MULTI-LINE* object represents more than two points connected by a series of line segments. Multi-Lines are not closed curves.

Requirements/Restrictions:

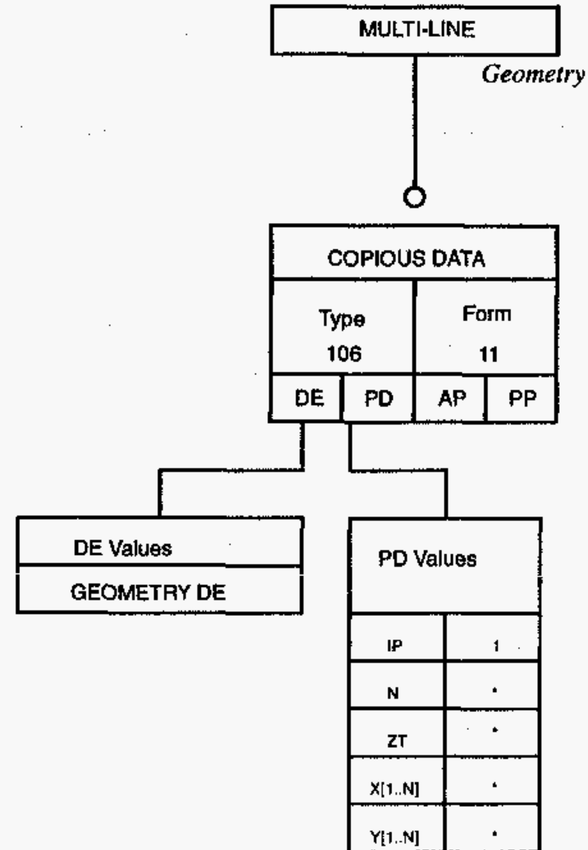
1. The *MULTI-LINE* may or may not be closed.
2. The *MULTI-LINE* shall contain more than two points.

Translation Usage Notes:

General:

Output:

Input:



IGES graphic file: MLINE

5.3.3.9. Non-Closed Curve

Description:

The *NON-CLOSED CURVE* object is any one of the predefined curves which is capable of defining an open curve.

Requirements/Restrictions:

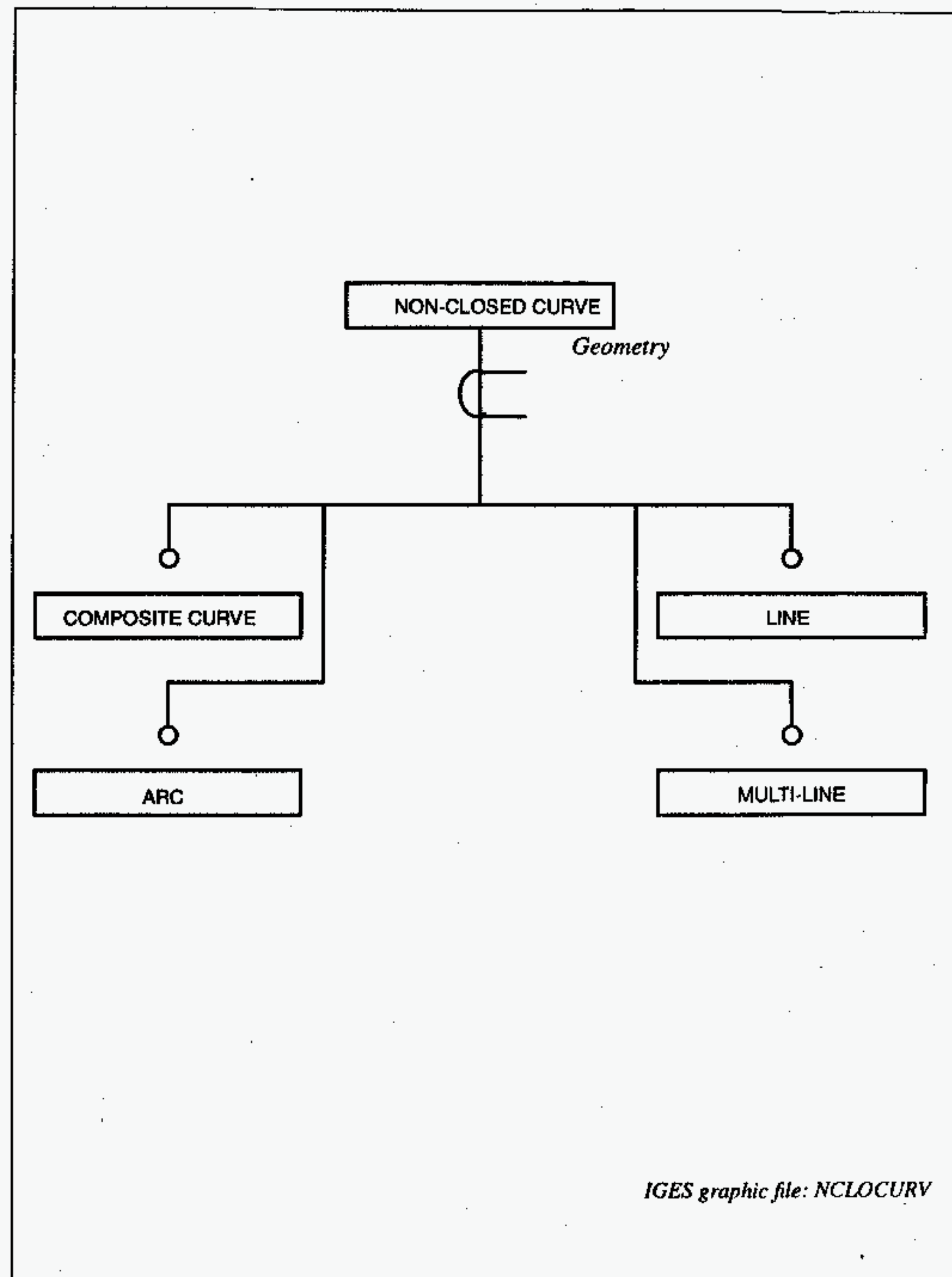
1. A *NON-CLOSED CURVE* shall not self intersect or become tangent with itself at any point, nor shall the start point be coincident with the end point.

Translation Usage Notes:

General:

Output:

Input:



5.3.3.10. Polygon

Description:

The *POLYGON* object is a closed multi-line that does not self intersect nor become tangent with itself at any point, other than the start and end point. Because the polygon is a *CLOSED CURVE*, the start point and end point are coincident.

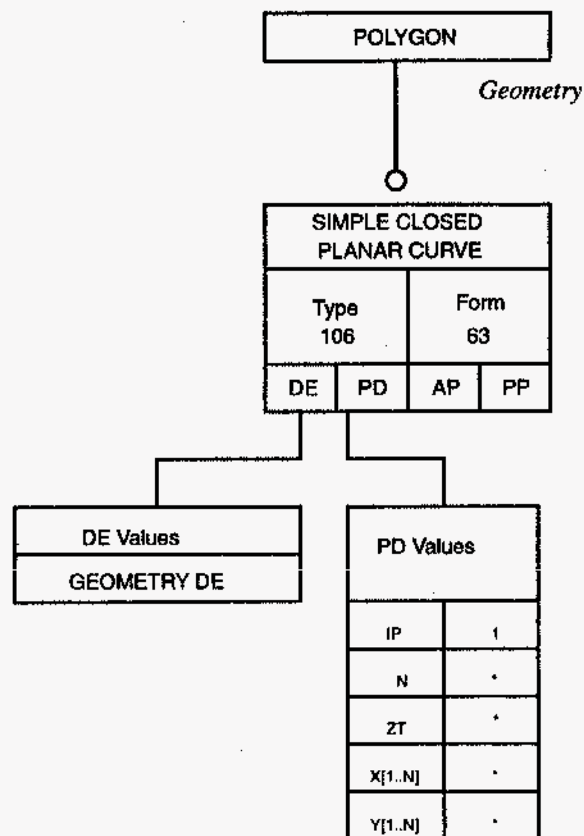
Requirements/Restrictions:

Translation Usage Notes:

General:

Output:

Input:



IGES graphic file: PGON

5.3.3.11. Point

Description:

The *POINT* object represents a 2-D geometric construction point.

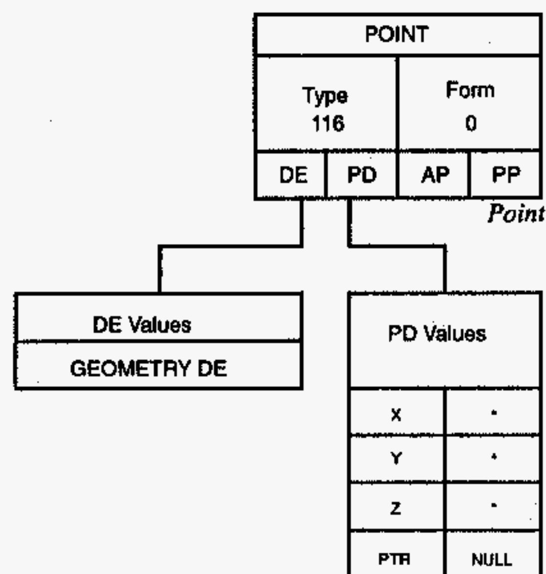
Requirements/Restrictions:

Translation Usage Notes:

General:

Output:

Input:



IGES graphic file: POINT

5.3.3.12. Predefined Planar Shape

Description:

The *PREDEFINED PLANAR SHAPE* object is a closed curve, which represents one of a set of predefined shapes, where the specific shape is defined by three floating point parameters. (This object is defined as the Flash Entity (Type 125); see IGES section 4.19).

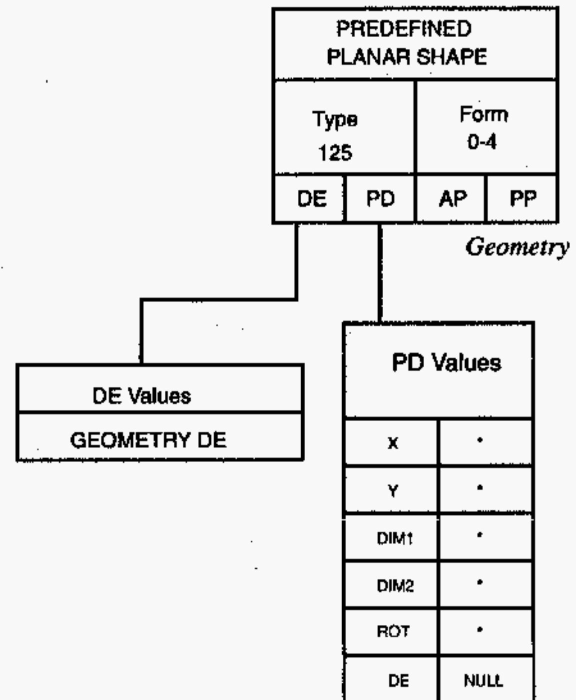
Requirements/Restrictions:
Only Forms 0-4 are valid in this AP

Translation Usage Notes:

General:

Output:

Input:



IGES graphic file: PREPLANS

5.3.4. Miscellaneous Object Models

The following objects are not peculiar to the LEP object, but are needed for its description.

5.3.4.1. Generic Data Property

Description:

The *GENERIC DATA PROPERTY* object maps directly to the IGES Generic Data Property Entity (Type 406, Form 27). Please refer to IGES for additional information.

Requirements/Restrictions:

1. Certain objects in this AP require specific values for a Generic Data Property (Type 406, Form 27) Name Value as defined in the following list.

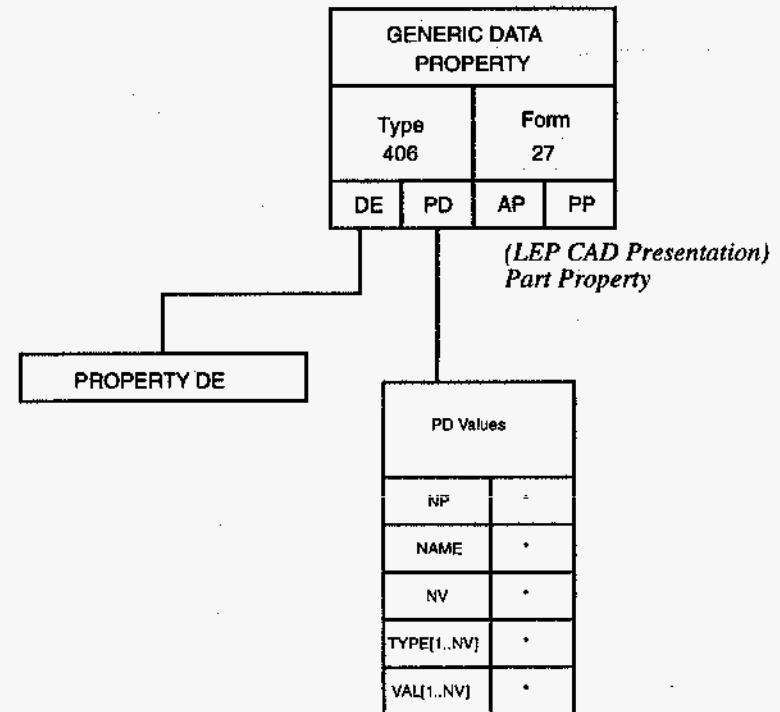
Translation Usage Notes:

For general (i.e., user extendable) use of this IGES Property see Section 5.3.5 LEP SEMANTIC PROPERTY. Where possible, select the PD Name value from the pre-defined list beginning on the next page, allowing the receiving system to interpret the meaning of the Name string consistent with this AP.

General:

Output:

Input:



IGES graphic file: GENDPROP

Predefined values for the Generic Property "Name" parameter:

abs_voltage_max The property (of the *PACKAGE SYMBOL INSTANCE*) which provides simulation and analysis programs an upper limit on the expected EMF to be applied to the component.

allowable_test_point The logical property (of a Connect Point Entity (Type 132)) indicating it MAY be probed by the tester.

area The logical property of a *JOIN* object indicating that the surface bounded is conductive and is associated with a *NET*.

assembly_number The property (of the *LEP PHYSICAL LAYOUT IGES FILE*) which provides the text string name of the LEP assembly drawing file (assigned by the enterprise that created the LEP design).

back_pad_polarization The logical property (of a Connect Point Entity (Type 132)) indicating it is in direct contact with the substrate and specifies a value of +, -, key, gnd, or pin(n).

component_default_padstack The logical property (of a *PADSTACK DEFINITION*) to indicate that nothing special has been added or removed from the Padstack selected to be generic.

component_height The length measure (of the *PACKAGE SYMBOL DEFINITION*) which provides insertion processes a z dimension measured from the nearer surface of the LEP below which the component shall reside after insertion (or an indication no value is provided).

component_insertable The logical property (of the *PACKAGE SYMBOL DEFINITION*) which provides insertion processes an indication as to whether the component lends itself to processing or not.

component_insertion_force A real number providing the maximum insertion force to be applied.

component_insertion_height A length measure providing the maximum difference in the z direction between the nearer surface of the LEP and the farthest point, edge, or surface of the component.

component_insertion_origin The length measure (of the *PACKAGE SYMBOL DEFINITION*) which provides insertion processes the real X and Y offset from the definition space origin to the origin used by the inserter.

component_instance_side The text string (of the *PACKAGE SYMBOL INSTANCE*) which provides other processes a text string indicating on which side (top or bottom) of the LEP the component will be found.

component_keep_outline The logical property of a *CLOSED CURVE* in the Z = 0.0 plane flagging it as an X, Y region of inclusion or exclusion of LEP Components.

component_layout_surface The text string(of the *PACKAGE SYMBOL DEFINITION*) which indicates which LEP surface the component defined occupies. { front/back/top/bottom/both/unknown }

component_outline_overhang The property (of the *CLOSED CURVE*) depicting the outline of a connector or other component which extends beyond the edge of the LEP outline.

component_max_size The property (of the *PACKAGE SYMBOL DEFINITION*) which provides insertion processes three maximum dimensions inside which the component can be contained.

component_physical_thickness The property (of the *PACKAGE SYMBOL DEFINITION*) which provides insertion processes with the depth space in which the component body is contained.

component_pkg_type The textual property (of the *PACKAGE SYMBOL DEFINITION*) which provides the package type name of the component defined.

component_placement_angle An angular measure providing the insertion machinery the rotation to be applied between grasp and insertion relative to the default component rotation.

component_placement_body A string providing a text code corresponding to the package type of the component to be inserted.

component_placement_center The length measures providing the insertion machinery the X, Y, and Z offset from the component origin to the origin presumed by the inserter.

component_placement_form_code An integer providing the insertion machinery an integer code to indicate the forming needed for the referencing component.

component_placement_form_code_description A string providing a text description of the forming needed for the referencing component.

component_placement_lead_diameter A length measure providing the insertion machinery the nominal diameter of the pin to be inserted.

component_placement_yn A logical property indicating whether the referencing entity models a component which is to be inserted or not.

component_placement_zspan A string providing a text code corresponding to the insertion machine depth stop.

component_preplace The property (of the *PACKAGE SYMBOL INSTANCE*) which provides placement systems a predefined location and/or mounting side for that particular component instance.

component_std_name The textual property (of the *PACKAGE SYMBOL DEFINITION*) which provides the name of the component defined.

component_tech_type The textual property (of the *PACKAGE SYMBOL DEFINITION*) which provides the name of the technology type (SMT or through-hole) of the component defined.

component_version The textual property (of the *PACKAGE SYMBOL DEFINITION*) which provides the (alphanumeric) version identifier of the component defined.

component_via The marking of the *VIA* instance to indicate it will be built around a component *PIN*.

conductive_trace_function The descriptive text property of an open curve defining a conductive *JOIN* on a particular layer or set of layers of an LEP intended to connect Pins, Pads and Vias to a power source, a ground return or a particular signal.

date The textual property (of an LEP) providing a value of the form mm/dd/yy plus a second value from the set {file_creation, last_revision, revision_effective }

fiducial The compound textual property (of an LEP *CLOSED CURVE*) to indicate its use as a positional reference. The first value indicates the nature of the owning entity. {component/cluster/board/panel} The second value indicates whether the locating symbol appears on the top or the bottom surface of the LEP. { top/bottom }

jumper_wire The logical property of a *JOIN* object indicating that the object is conductive and is associated with a *NET.*

junction_max_t The temperature measure (of the *PACKAGE SYMBOL INSTANCE*) which provides simulation and analysis programs a value of the maximum temperature rating of the semi-conductor junction in the component.

lep_physical_outline The logical property (of an LEP *CLOSED CURVE*) which indicates the curve represents the physical outline of the LEP substrate.

no_connection The logical property (of a Connect Point Entity (Type 132) indicating it shall NOT be pointed to by the Flow Associativity Entity (Type 402, Form 18).

not_allowable_test_point The logical property (of a Connect Point Entity (Type 132) indicating it shall NOT be probed by the tester.

pad_tech_type The textual property (of the Connect Point Entity (Type 132) modeling the *PAD*, *PIN*, or socket acting as Port to the LEP or one of its components) which indicates the type of technology (and for thru_holes the design of the pad) used in implementing the port (or the no_connection).

panel The logical property of a *PANEL DEFINITION* object indicating that the object is an LEP manufacturing item created to facilitate the fabrication of one or more PWBs.

panel_offset A pair of length measures providing the X, Y offset from the *LEP PHYSICAL LAYOUT IGES FILE* to the origin of the fabrication panel.

pathThe logical property of a *JOIN* object indicating that the non-closed curve is conductive and is associated with a *NET*.

photoplot_aperture The textual property of a CLOSED CURVE which is a LAND object with values of type string; supplied when it is desired to record the information about the particular tool (i.e., photoplotter wheel) and aperture index of the tool identified to produce the required shape on the photoplot.

physical_outline The required property of a *CLOSED CURVE* surrounding the silhouette viewed in a plane parallel to a layer of the LEP.

physical_padstack_definition The logical property (of the set of entities depicting the sequence of *PAD*s at a particular X, Y location of an LEP) which tags the parent entity as the definition. Some of the *PAD*s are to be connected to each other and to any conductor penetrating the hole at the X, Y location. Other *PAD*s may be intentionally isolated electrically from any conductor penetrating the hole at the X, Y location.

pin1 The logical property (of the Connect Point Entity (Type 132) modeling the *PAD*, *PIN*, or socket acting as Port to the LEP or one of its components) which tags the port as "Pin 1" of the device.

placement_outline The orientational and positional property of a *CLOSED CURVE* indicating the intended position of a particular component on a physical layer of the LEP.

planar_shape_function The textual property of a *CLOSED CURVE* defining a conductive region of a particular layer or set of layers of an LEP intended to connect *PINS*, *PAD*s and *VIA*s to a power source, a ground return or a particular signal.

pwb_number The textual property (of the *LEP PHYSICAL LAYOUT IGES FILE* defining the substrate) which provides the name of the LEP design drawing file (assigned by the enterprise that created the LEP design).

registration_mark The logical property (of LEP GEOMETRY) indicating that the geometry purpose is the alignment of layers of an LEP or its artwork or photo-tool.

schematic_number The textual property (of the *LEP TECHNICAL ILLUSTRATION IGES FILE*) which provides the name of the schematic drawing file (assigned by the enterprise that created the LEP design),

selected_test_point The logical property [of a Connect Point Entity (Type 132)] indicating it has been targeted to be probed by the tester.

silkscreen_bot The property indicating the referencing Text and/or graphics are to be applied to the Bottom Layer of the LEP.

silkscreen_top The property indicating the referencing Text and/or graphics are to be applied to the Top Layer of the LEP.

standalone_via The tagging of the *VIA* instance to indicate it will not be built around a component Pin.

station_action The textual property (of the *SUBFIGURE INSTANCE* representing the work station where some processing is to occur) which indicates what kind of processing is to be done.

station_number A string providing the I. D. of the work station where some processing is to occur.

test_point_pwb_id The textual property (of a stand-alone *VIA*) to provide a unique name to facilitate test descriptions.

test_point_nail_id The property (of a tester probe) to uniquely identify the probe to be put in contact with the referencing LEP feature.

therm_cond The textual property (of the Component Instance) which provides simulation and analysis programs a value of the nominal thermal conductivity coefficient (see IGES Tabular Data Property Entity (Type 406, Form 11), PTYPE=19) of the component case material.

therm_jc The property (of the Component Instance) which provides simulation and analysis programs a value of the nominal thermal resistance between the semiconductor junction and the outer case surface in watts per degree Centigrade.

therm_r The property (of the *PACKAGE SYMBOL INSTANCE*) which provides simulation and analysis programs a value of the nominal thermal resistance from case to ambient in watts per degree Centigrade.

thermal_outline The graphic property of a *CLOSED CURVE* indicating the extent of thermal emission or sensitivity of the component(s) mounted in the vicinity.

thru_hole The property (of a HOLE object) which indicates penetration of all physical layers of the LEP.

tolerance The permitted range of the value referencing referencing this property.

top_level_assembly_instance The logical property (of the LEP entity) tagging it as the highest level instance within the scope of the model.

trace_bot The logical property of a *JOIN* indicating it appears on the Bottom layer of the LEP.

trace_keep_outline The property (of a *CLOSED CURVE* in the Z = 0.0 plane) defining an X, Y region of inclusion or exclusion of LEP Traces.

trace_n The property of a *JOIN* indicating it appears on layer number n of the LEP.

trace_top The logical property of a *JOIN* indicating it appears on the Top layer of the LEP.

trace_2 The property of a *JOIN* indicating it appears on layer number 2 of the LEP.

via The logical property of a *JOIN* object indicating that a hole is conductive and is associated with a *NET*.

via_keep_outline The property (of a *CLOSED CURVE* in the Z = 0.0 plane) defining an X, Y region of inclusion or exclusion of LEP *VIA*s.

wire_wrap The logical property (of the *PIN*) indicating the connection is wirewrapped.

gauge The numeric property (of the *JOIN*) indicating the wire gauge (usually 26 or 30; occasionally others)

wire_bond The logical property of a *JOIN* object indicating that the object is conductive and is associated with a *NET*.

wire_color color of wiring for maintenance and ease of assembly

twisted_pair The logical property (of the *JOIN*) indicating multiple wires are twisted together for better signal integrity; signal names are usually assigned to sort together (e.g., reset_D+ and reset_D-) to designate pairing.

wrap_class The (optional) textual property (of the *JOIN*); values are "A," "B," "MODIFIED-B", indicating how many wraps around the post, and the "tightness" of wrap.

5.3.4.2. Hierarchy Property

Description:

The *HIERARCHY* object maps directly to the IGES Hierarchy Property Entity (Type 406, Form 10). Please refer to IGES for additional information.

Requirements/Restrictions:

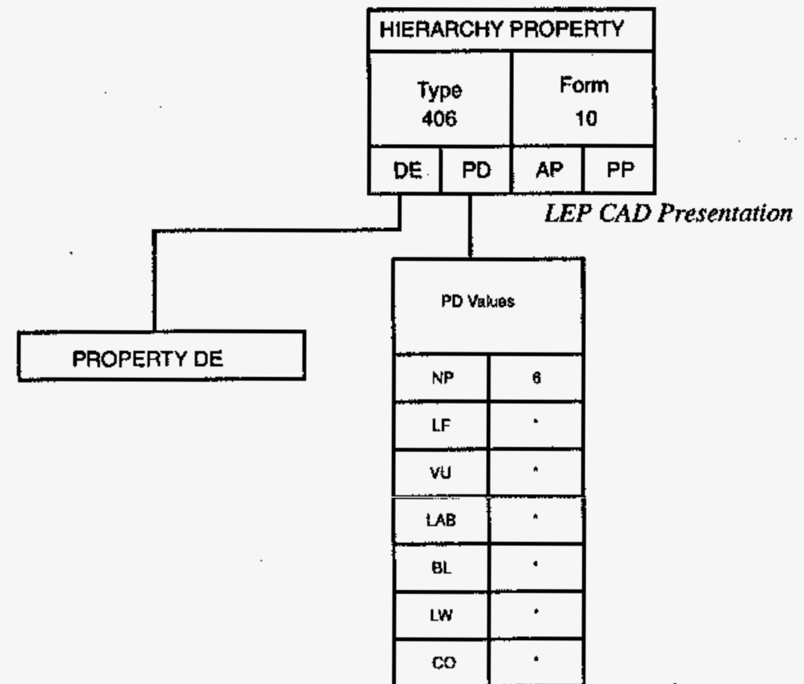
1. The *HIERARCHY* object may be referenced from the Property Pointer list of any object that references physically subordinate children.

Translation Usage Notes:

General:

Output:

Input:



IGES graphic file: HIERPROP

5.3.4.3. Name

Description:

The *NAME* object maps directly to the IGES Name Property Entity (Type 406, Form 15). Please refer to IGES for additional information.

Requirements/Restrictions:

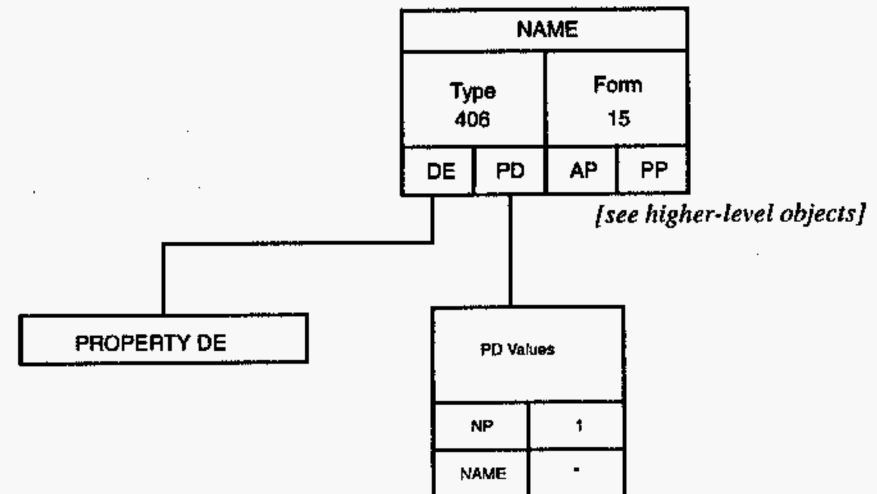
1. The *NAME* object may be referenced by any other object that does not have an explicit name field in its parameter data.
2. The Name shall be unique within the scope of the file in which it appears.

Translation Usage Notes:

General:

Output:

Input:



IGES graphic file: NAME

5.3.4.4. Network Subfigure Definition

Description:

The *NETWORK SUBFIGURE DEFINITION* object maps directly to the IGES Network Subfigure Definition Entity (Type 320, Form 0). Please refer to IGES for additional information.

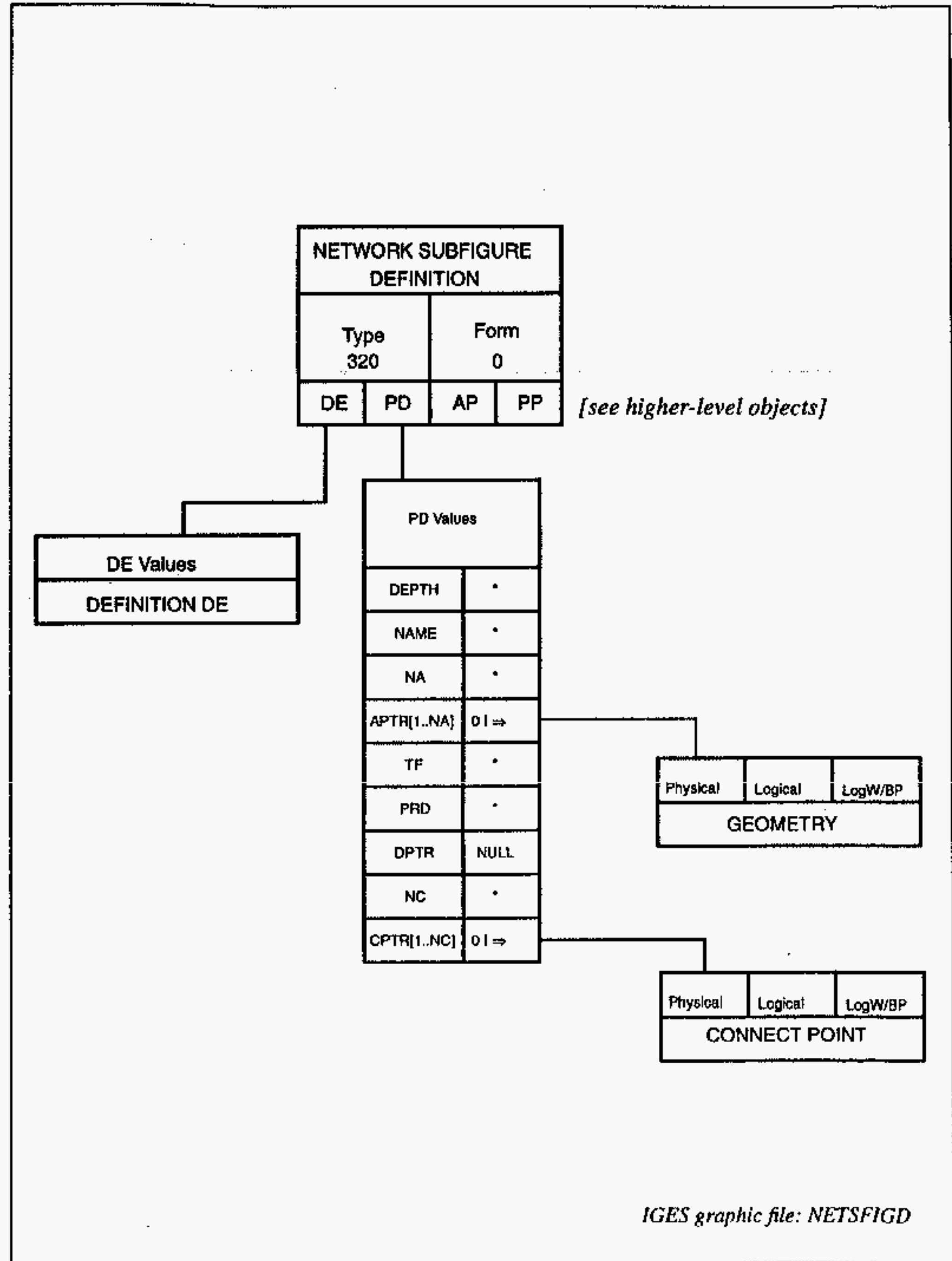
Requirements/Restrictions:

1. The NAME field in the Network Subfigure Definition Entity (Type 320) shall be unique (for all *NETWORK SUBFIGURE DEFINITIONs* of the same object type) within the scope of the IGES File.
2. The *LEP SEMANTIC PROPERTY* object referenced from the *CONNECT POINT*, if appropriate, shall specify
   back_pad_polarization.

Translation Usage Notes:

General:
The values for the primary reference designator (PRD) and the pointer to the display of the primary reference designator (DPTR) are defaulted or set to null.
Output:

Input:



5.3.4.5. Network Subfigure Instance

Description:

The *NETWORK SUBFIGURE INSTANCE* object maps directly to the IGES Network Subfigure Instance Entity (Type 420, Form 0). Please refer to IGES for additional information.

Requirements/Restrictions:

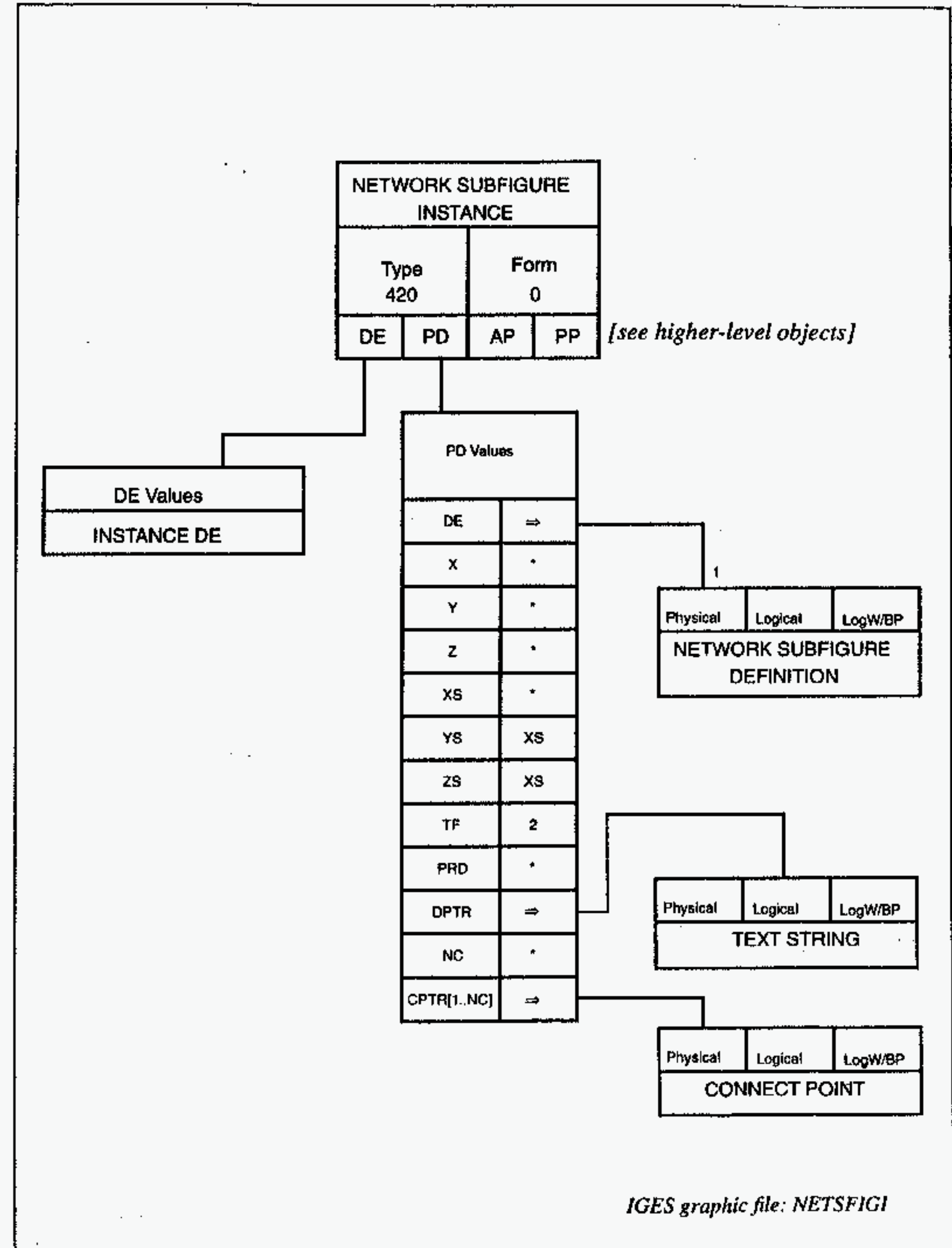
1. The reference designator shall be entered in the PD PRD (A *TEXT STRING* object shall not be used for the reference designator).
2. The pointer (PD DPTR) shall be to the DE of the Text Display Template Entity (Type 312) for display of the primary reference designator(PD PRD).

Translation Usage Notes:

General:

Output:

Input:



IGES graphic file: NETSFIGI

5.3.4.6. Object Locator

Description:

The *OBJECT LOCATOR* object is any lesser object which documents location.

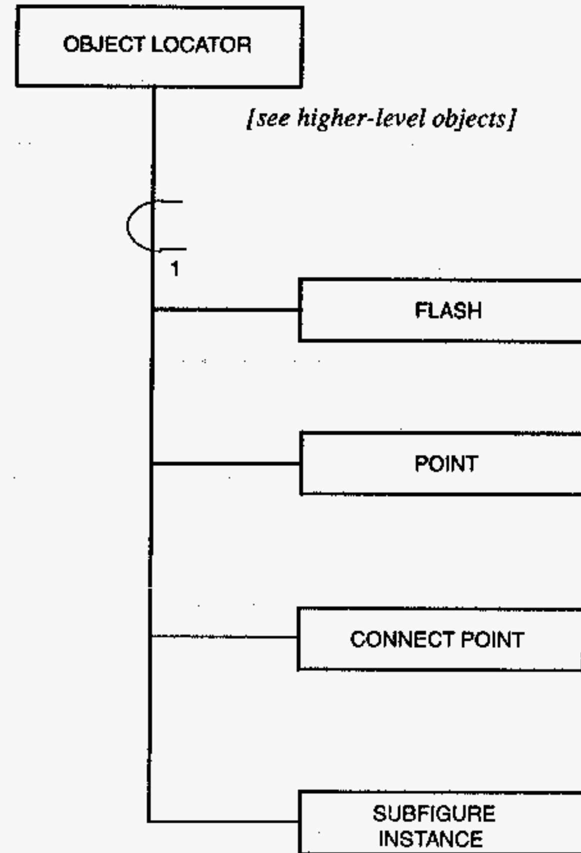
Requirements/Restrictions:

Translation Usage Notes:

General:

Output:

Input:



IGES graphic file: OBJLOC

5.3.4.7. Part Number Property

Description:

The *PART NUMBER PROPERTY* object maps directly to the IGES Part Number Property (Type 406, Form 9). Please refer to IGES for additional information.

Requirements/Restrictions:

1. Shall attach to a Subfigure Instance Entity (Type 408) or a Network Subfigure Instance Entity (Type 420).

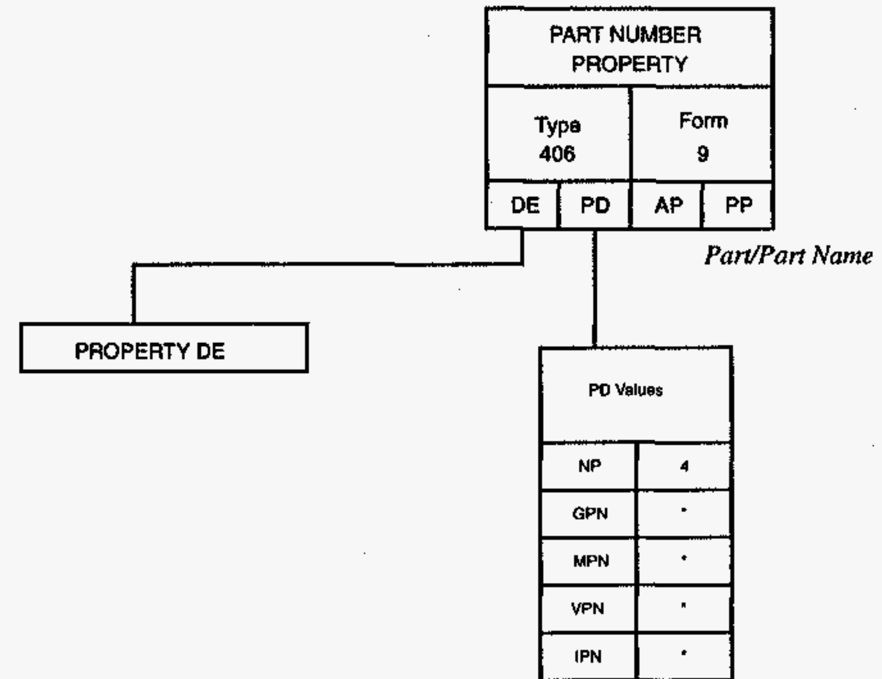
Translation Usage Notes:

One or more of four strings may be supplied as the generic part number (GPN), the Military Standard part number (MPN), the vendor part number or name (VPN), and/or the internal part number (IPN).

General:

Output:

Input:



5.3.4.8. Sectioned Area Fill Definition

Description:

The *SECTIONED AREA FILL DEFINITION* object enables a *CLOSED CURVE* entity to be displayed with specific fill characteristics of solid or blank fill.

Requirements/Restrictions:

The *SECTIONED AREA FILL DEFINITION* may only be referenced by the following *DISPLAY GEOMETRY* objects which represent closed planar boundaries.

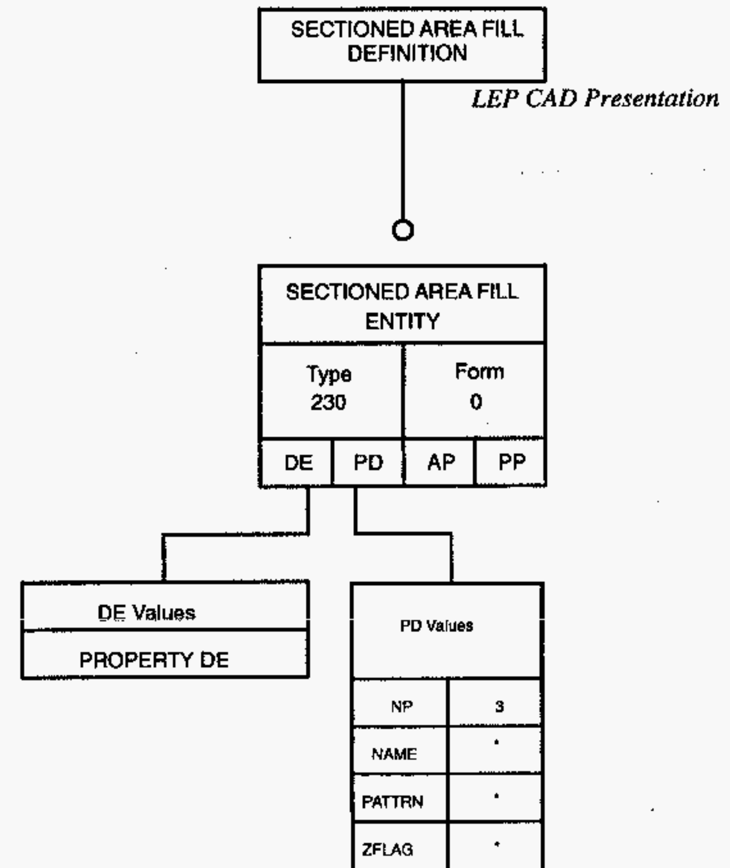
1. The following *DISPLAY GEOMETRY* objects may reference the Sectioned Area Fill Definition.
   - A) *CLOSED CURVE*
   - B) *PREDEFINED PLANAR SHAPE*
   - C) *POLYGON*
2. The fill pattern shall be code 0 or code 19.

Translation Usage Notes:

General:

Output:

Input:



5.3.4.9. Subfigure Definition

Description:

The *SUBFIGURE DEFINITION* object maps directly to the IGES Subfigure Definition Entity (Type 308, Form 0). Please refer to IGES for additional information.

Requirements/Restrictions:

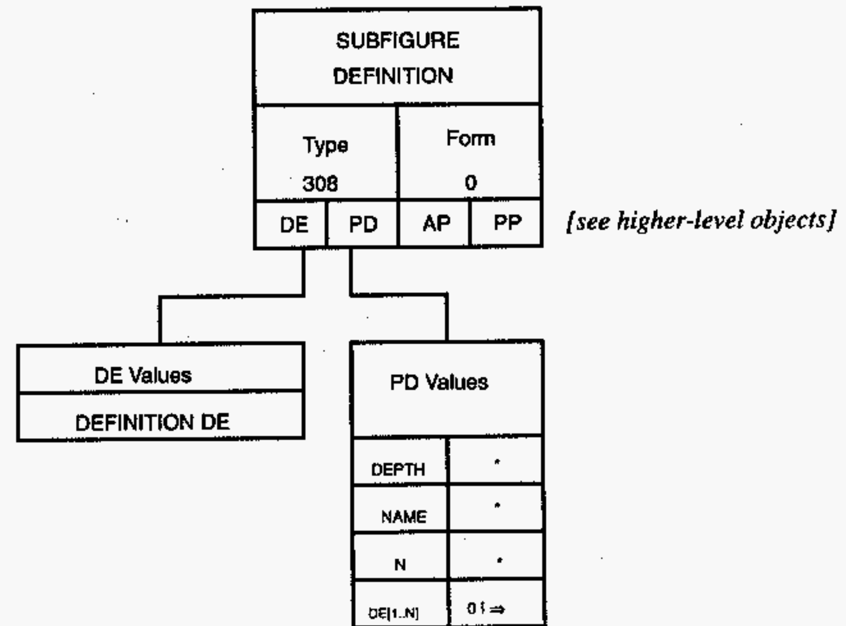
1. The NAME field in the Subfigure Definition shall be unique (for all *SUBFIGURE DEFINITIONs* of the same object type) within the scope of the IGES File.

Translation Usage Notes:

General:

Output:

Input:



5.3.4.10. Subfigure Instance

Description:

The *SUBFIGURE INSTANCE* object maps directly to the IGES Subfigure Instance Entity (Type 408, Form 0). Please refer to IGES for additional information.

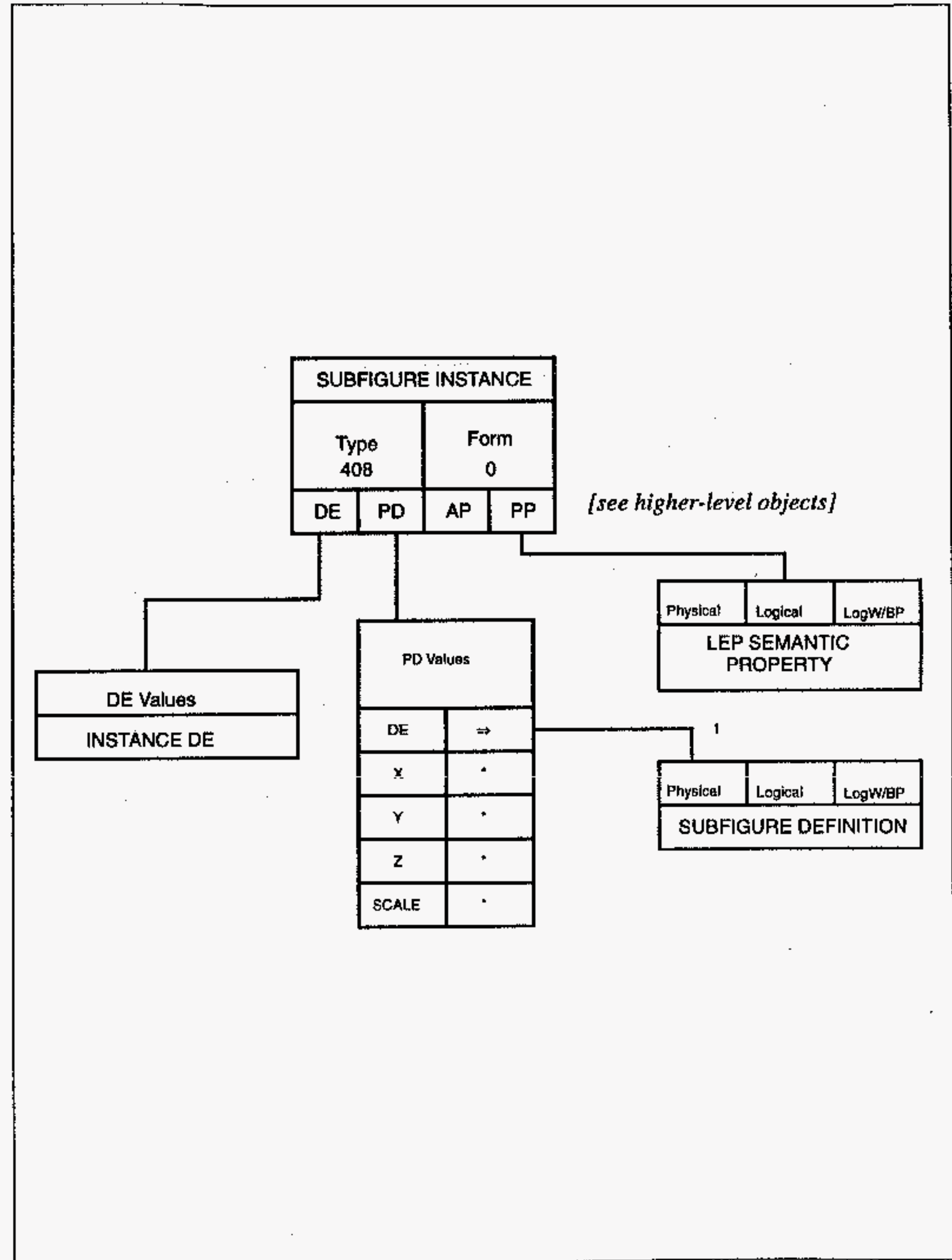
Requirements/Restrictions:

Translation Usage Notes:

General:
A suitable *LEP SEMANTIC PROPERTY* may indicate the kind of object referencing the *SUBFIGURE INSTANCE*.

Output:

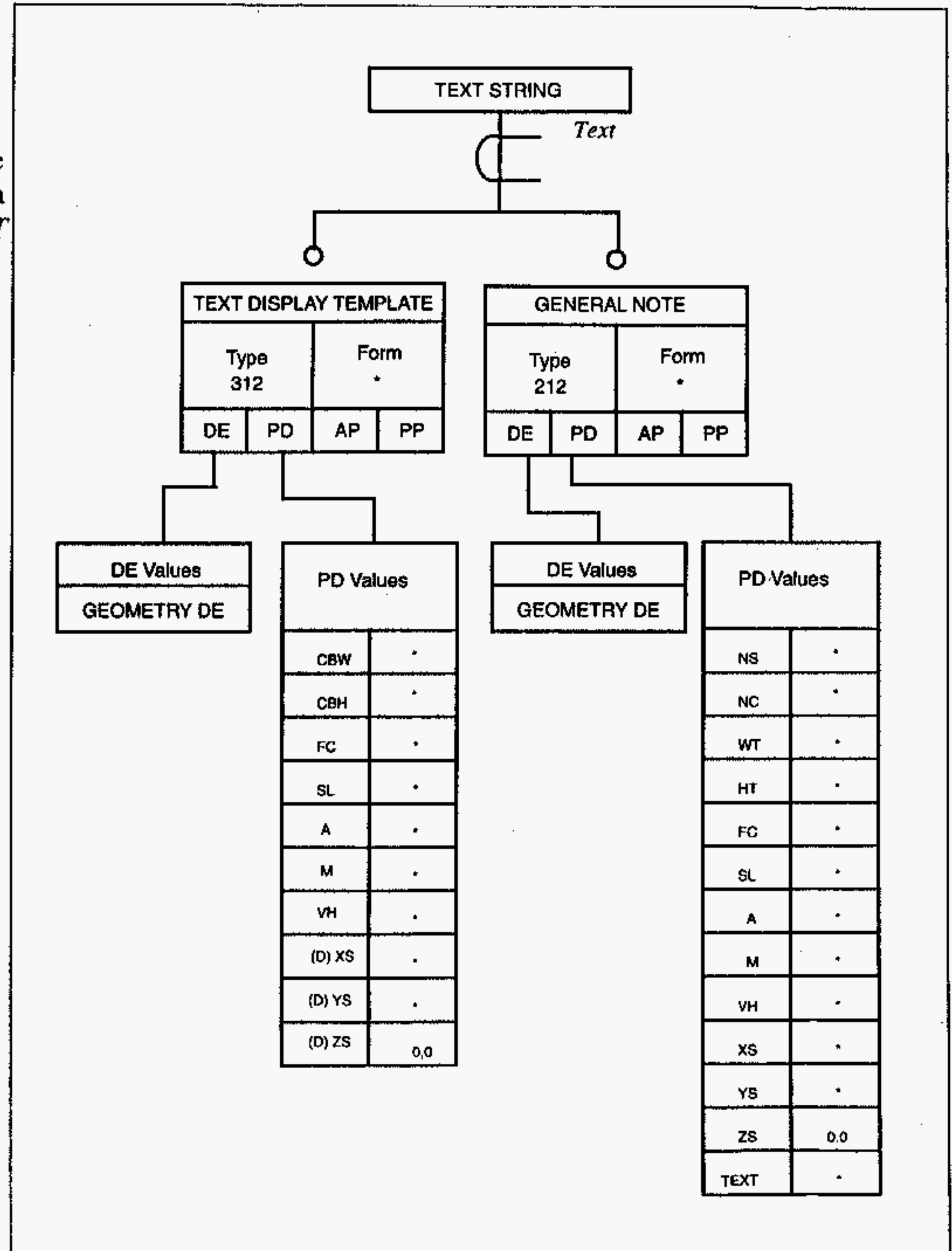
Input:



5.3.4.11. Text String

Description:
The *TEXT STRING* object is a General Note Entity (Type 212) containing one or more related character strings or a Text Display Template (Type 312) which displays a *TEXT STRING* derived from a particular parameter value of the referencing entity. This enables internal LEP annotation and provision of human interpretable semantics.
Requirements/Restrictions:

1. The General Note shall be Form 0.
2. The General Note FC field shall be one of {1, 1001, 1002, or 1003}.
3. The General Note shall not be used to display
   - The primary reference designator of a Network Subfigure Instance Entity (Type 420),
   - The pin name of a Connect Point Entity (Type 132),
   - The flow name of a Flow Associativity Entity (Type 402, Form 18).
4. The Incremental Text Display Template Entity (Type 312, Form 1) shall be used to display
   - The primary reference designator of a Network Subfigure Instance Entity (Type 420).
   - The pin name of a Connect Point Entity (Type 132) whenever such display is desired or required.
5. The Absolute Text Display Template (Form 0) shall be used to display the flow name of a Flow Associativity Entity (Type 402, Form 18) whenever such display is desired or required.



Translation Usage Notes:

General:
The General Note Entity (Type 212, Forms , 1,6, 7, or 8) is for Technical Illustration files.

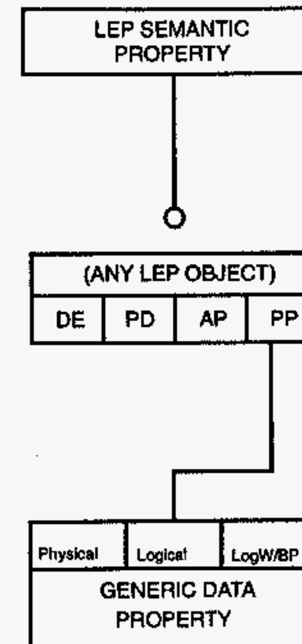
Output:

Input:

5.3.5. LEP Semantic Property

The *LEP SEMANTIC PROPERTY* object maps directly to the IGES Generic Data Property Entity (Type 406, Form 27). Please refer to the IGES for additional information. The following LEP object types and sub-types reference the LEP Property object:

5.3.1.2.	LEP Physical IGES File
5.3.2.1.	Component Keepin/Keepout
5.3.2.2.	Component Thermal Outline
5.3.2.3.	Drawing Definition
5.3.2.4.	Drawing Instance
5.3.2.5.	Generic Part Definition
5.3.2.6.	Generic Part Instance
5.3.2.7.	Hole
5.3.2.8.	Join
5.3.2.9.	Jumper Wire
5.3.2.10.	Keepin/Keepout
5.3.2.12.	Layer Outline
5.3.2.13.	LEP Definition
5.3.2.14.	LEP Instance
5.3.2.17.	Package Figure Definition
5.3.2.18.	Package Figure Instance
5.3.2.19.	Padstack Definition
5.3.2.20.	Padstack Instance
5.3.2.21.	Panel Definition
5.3.2.23.	Part Placement Boundary
5.3.2.24.	Pin
5.3.2.28.	Via
5.3.2.29.	Via Keepin/Keepout
5.3.2.30.	Wire Bond



5.3.6. DE Referenced Object Models

These four definitions plus Transformation Matrix may be applied to many of the other models.

5.3.6.1. Level Definition

Description:

The *LEVEL DEFINITION* object enables an entity to exist on one or more levels.

Requirements/Restrictions:

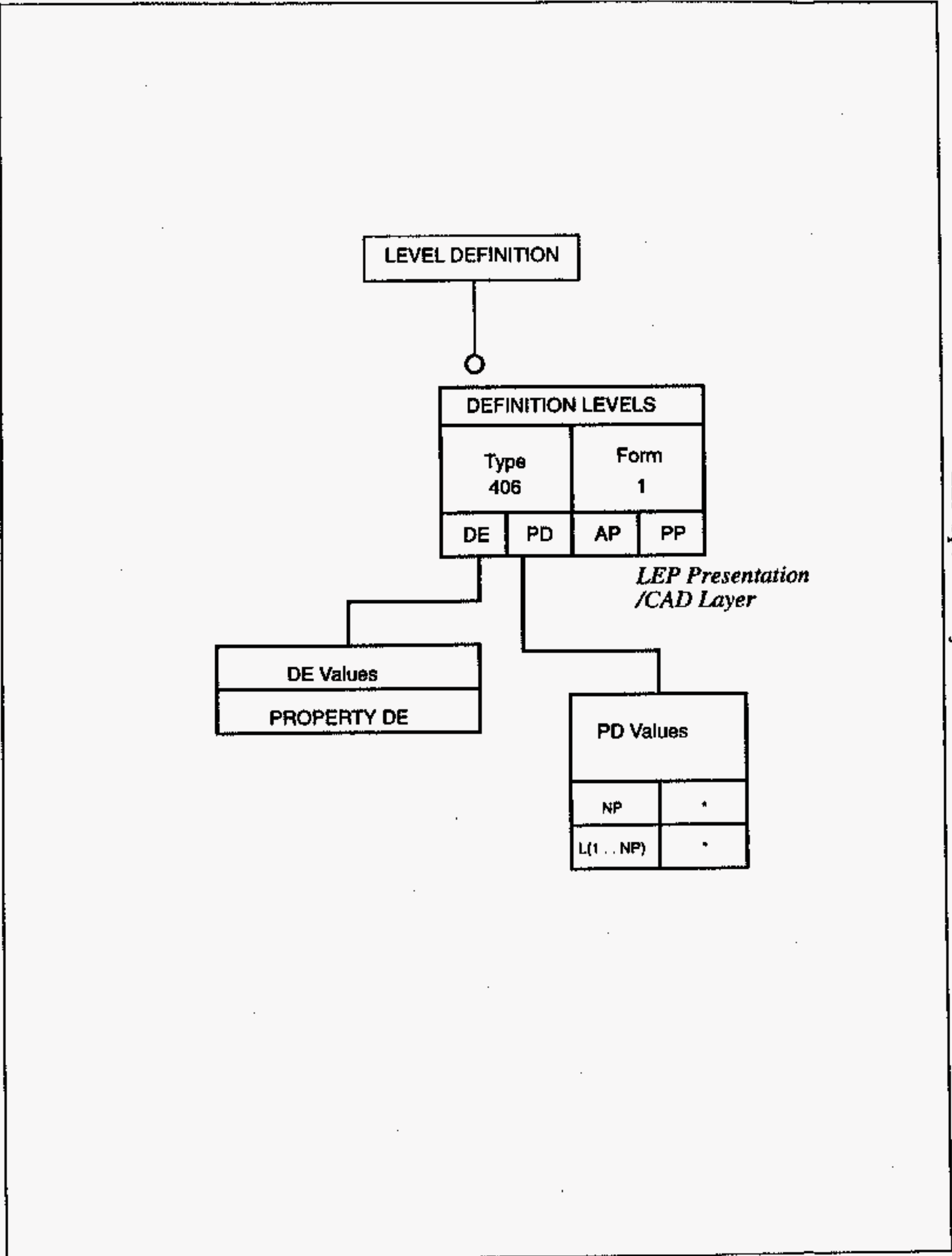
1. For the entity to be defined on a single level, it's DE field 5 shall contain that Level number.
2. For the entity to be defined on two or more levels, its DE field 5 shall contain a negated pointer to the appropriate Definition Levels Property Entity (Type 406, Form 1).

Translation Usage Notes:

General:

Output:

Input:



5.3.6.2. Line Width Definition

Description:

The *LINE WIDTH DEFINITION* object enables an entity to have specific line width characteristics which include width, justification, and specific end conditions.

Requirements/Restrictions:

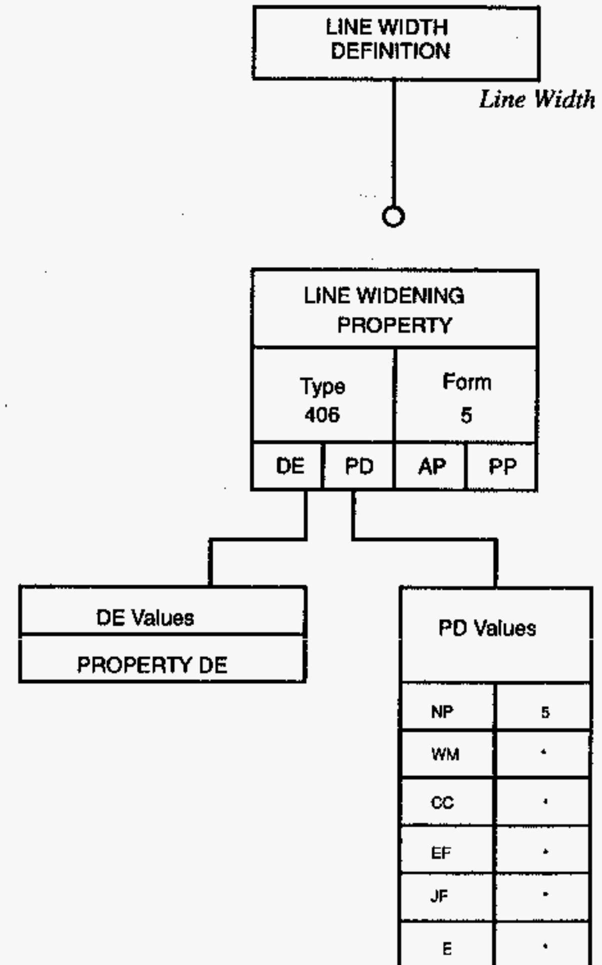
1. The *LINE WIDTH DEFINITION* shall only be invoked by *DISPLAY GEOMETRY* objects.

Translation Usage Notes:

General:

Output:

Input:



5.3.6.3. Transformation Matrix

Description:

The *TRANSFORMATION MATRIX* object is either a Transformation Matrix (Normal) or a Transformation Matrix (Mirrored) Entity (Type 124).

Requirements/Restrictions:

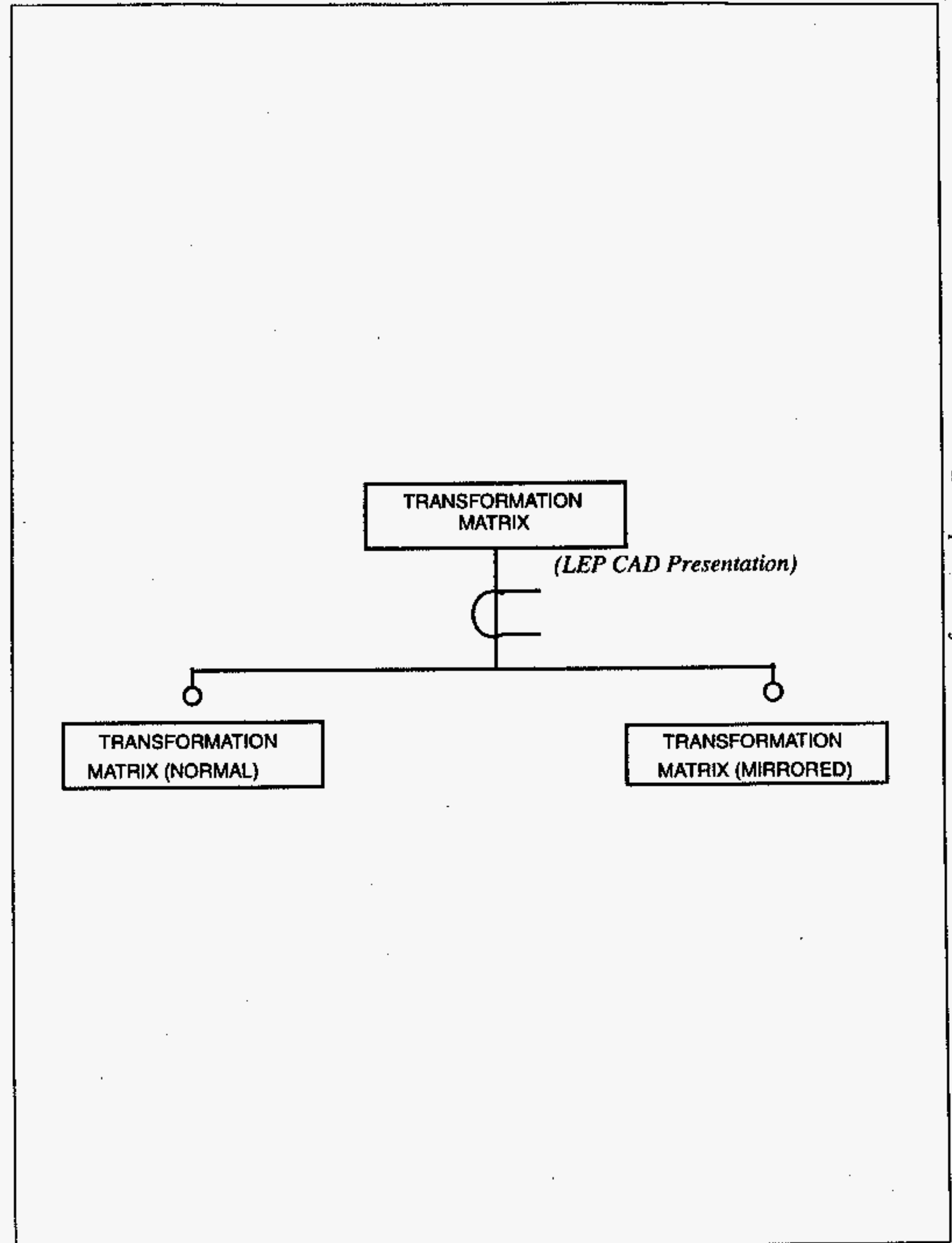
1. The determinant of the matrix shall be equal to plus or minus one.

Translation Usage Notes:

General:

Output:

Input:



5.3.6.4. Transformation Matrix (Mirrored)

Description:

The *TRANSFORMATION MATRIX (MIRRORED)* object calls for an X,Y, Z rotation and an X,Y, Z offset in a left-handed coordinate system.

Requirements/Restrictions:

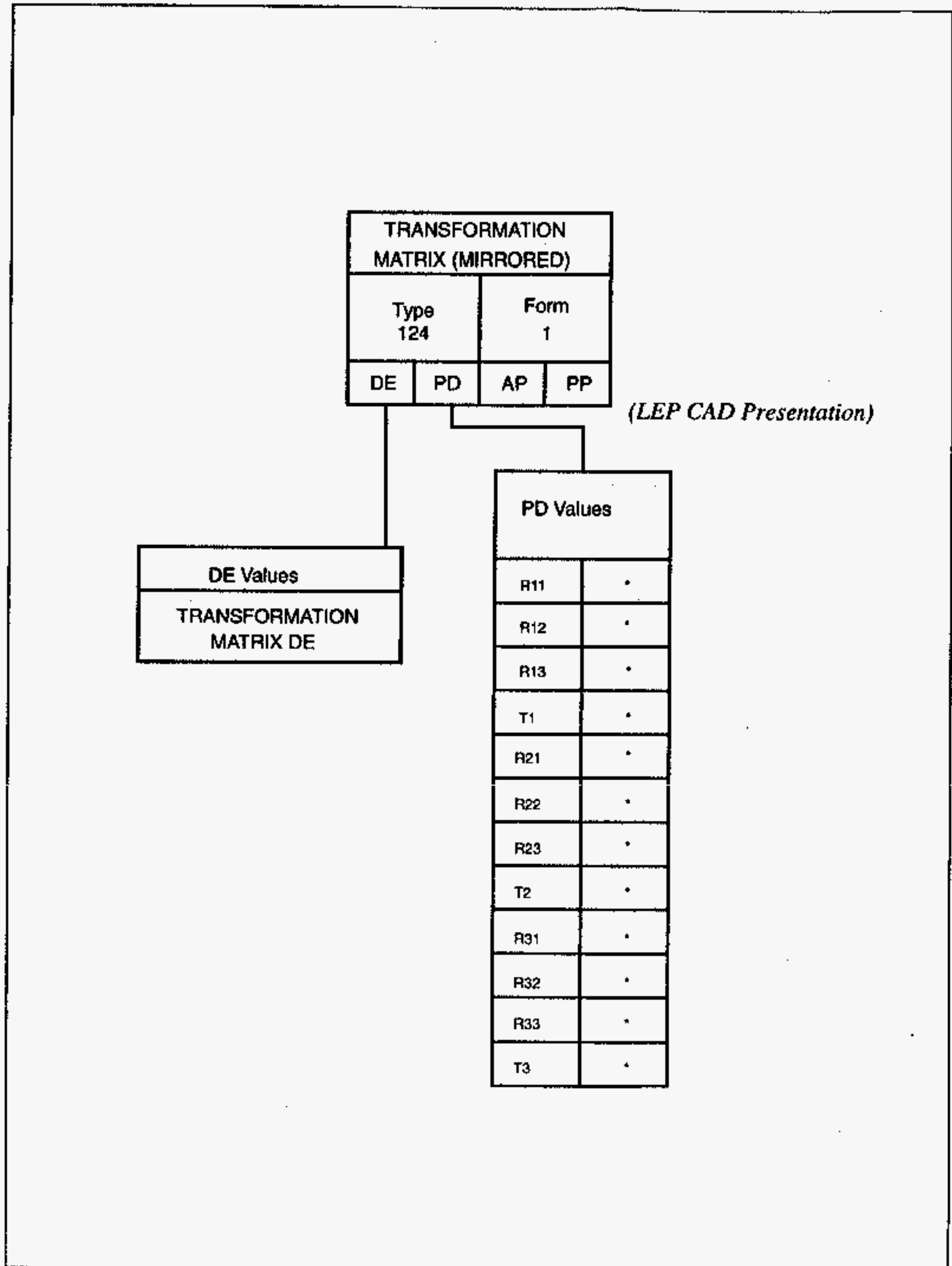
1. The determinant of the matrix shall be equal to plus or minus one.

Translation Usage Notes:

General:

Output:

Input:



5.3.6.5. Transformation Matrix (Normal)

Description:

The *TRANSFORMATION MATRIX (NORMAL)* object calls for an X,Y, Z rotation (and an X,Y, Z offset) in a right-handed coordinate system.

Requirements/Restrictions:

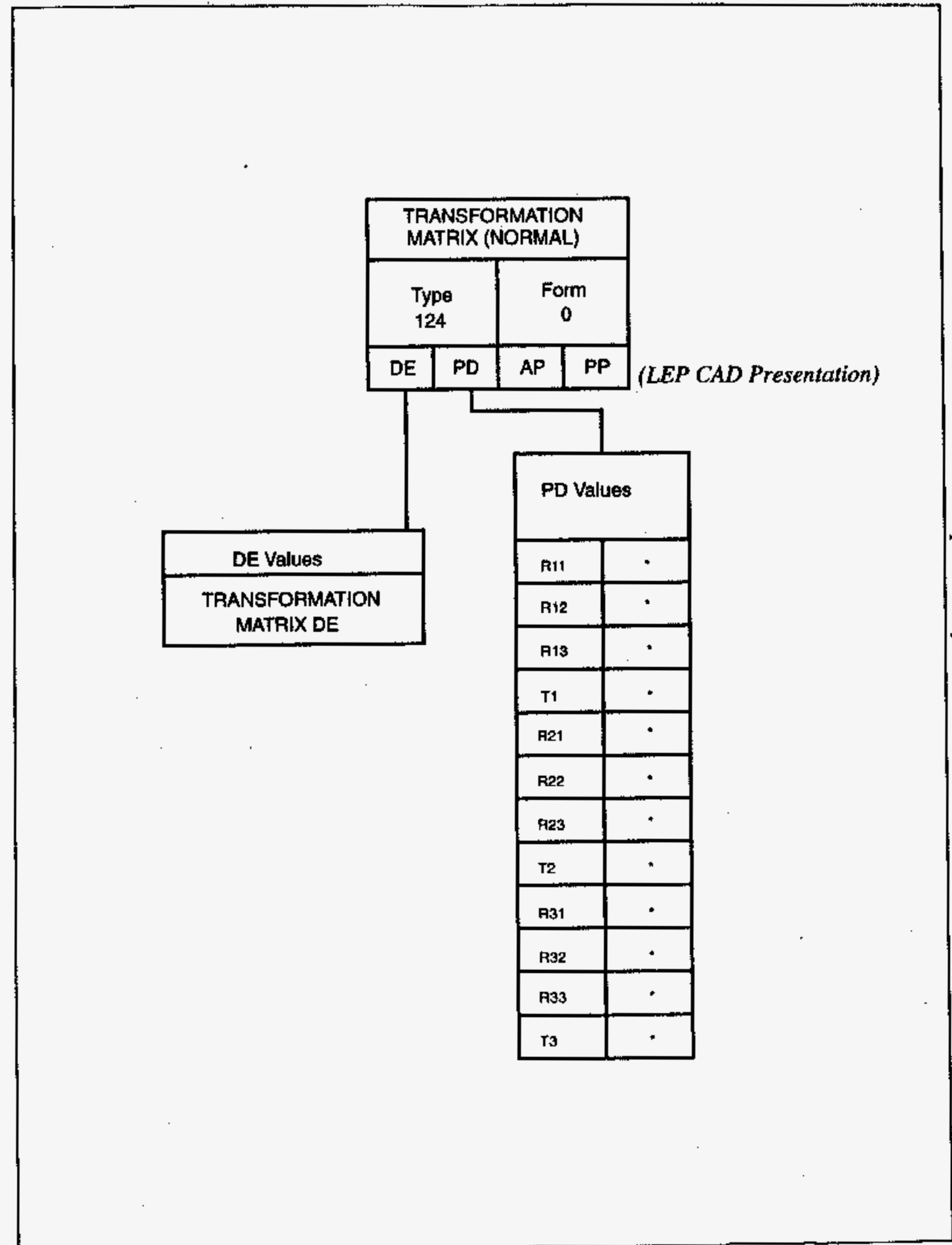
1. The determinant of the matrix shall be equal to plus or minus one.

Translation Usage Notes:

General:

Output:

Input:



BLANK PAGE

5.3.7. DE Section Object Models

Default values for the DE section are provided for seven classes of objects.

5.3.7.1. Associativity DE

Description:

The *ASSOCIATIVITY DE* object is an IGES Directory Entry Section template to be used by all associativity entities.

Requirements/Restrictions:

Translation Usage Notes:

General:

Output:

Input:

DE Values		
Entity Type		402
Parameters		--
Structure		N/A
Line Font		N/A
Level		N/A
View		N/A
TM		N/A
LDis Assoc		N/A
Status Number	Blank	N/A
	Subord	*
	Entity Use	3
	Hierarchy	N/A
Sequence		--
Entity Type		402
Line Weight		N/A
Color		N/A
Parm Count		--
Form number		*
Reserved		--
Reserved		--
Label		N/A
Subscript		N/A
Sequence		--

5.3.7.2. Definition DE

Description:

The *DEFINITION DE* object is an IGES Directory Entry Section template to be used by all definition entities.

Requirements/Restrictions:

1. The Hierarchy Status shall indicate independence.

Translation Usage Notes:

General:

Output:

Input:

DE Values		
Entity Type		*
Parameters		--
Structure		N/A
Line Font		N/A
Level		N/A
View		N/A
TM		N/A
LDis Assoc		N/A
Status Number	Blank	N/A
	Subord	*
	Entity Use	02
	Hierarchy	01
Sequence		--
Entity Type		*
Line Weight		N/A
Color		VALUE
Parm Count		--
Form number		*
Reserved		--
Reserved		--
Label		N/A
Subscript		N/A
Sequence		--

5.3.7.3. Geometry DE

Description:

The *GEOMETRY DE* object is an IGES Directory Entry Section template to be used by all display geometry entities.

Requirements/Restrictions:

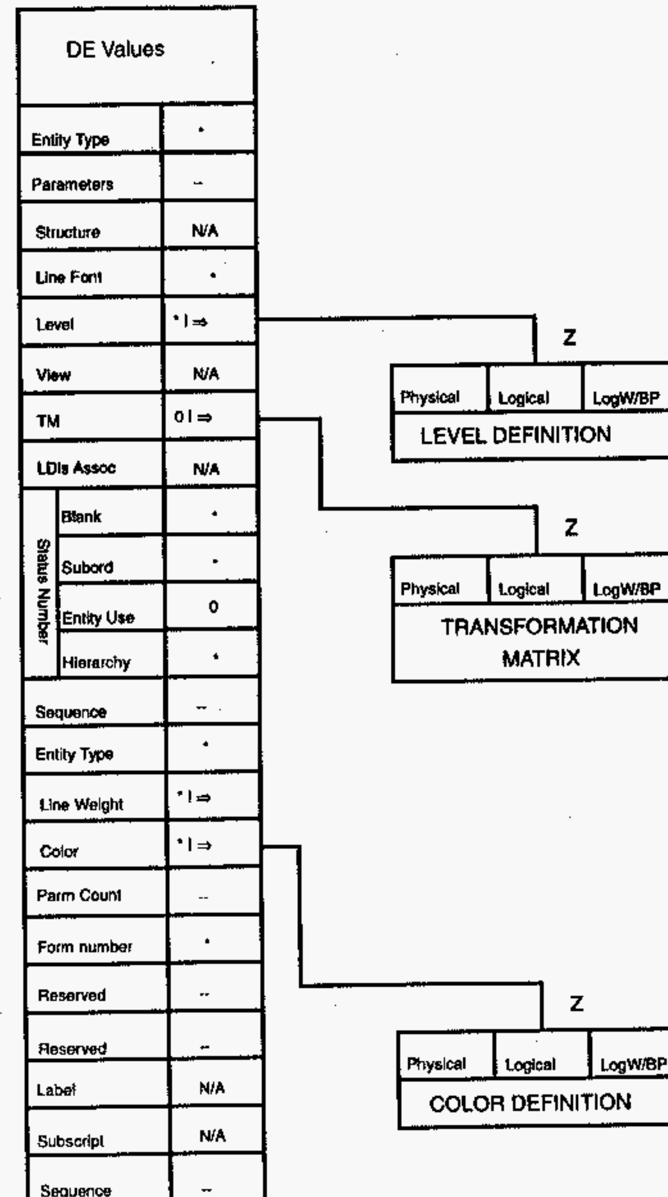
1. The *REGION FILL DEFINITION* only applies to the *CLOSED CURVE* and the *PREDEFINED PLANAR SHAPE* objects.
2. The Line Width Definition only applies to geometry and Text String objects.
3. The Color Definition Entity (Type 314) is out of scope except for technical illustrations and documentation

Translation Usage Notes:

General:

Output:

Input:



5.3.7.4. Instance DE

Description:

The *INSTANCE DE* object is an IGES Directory Entry Section template to be used by all instance entities.

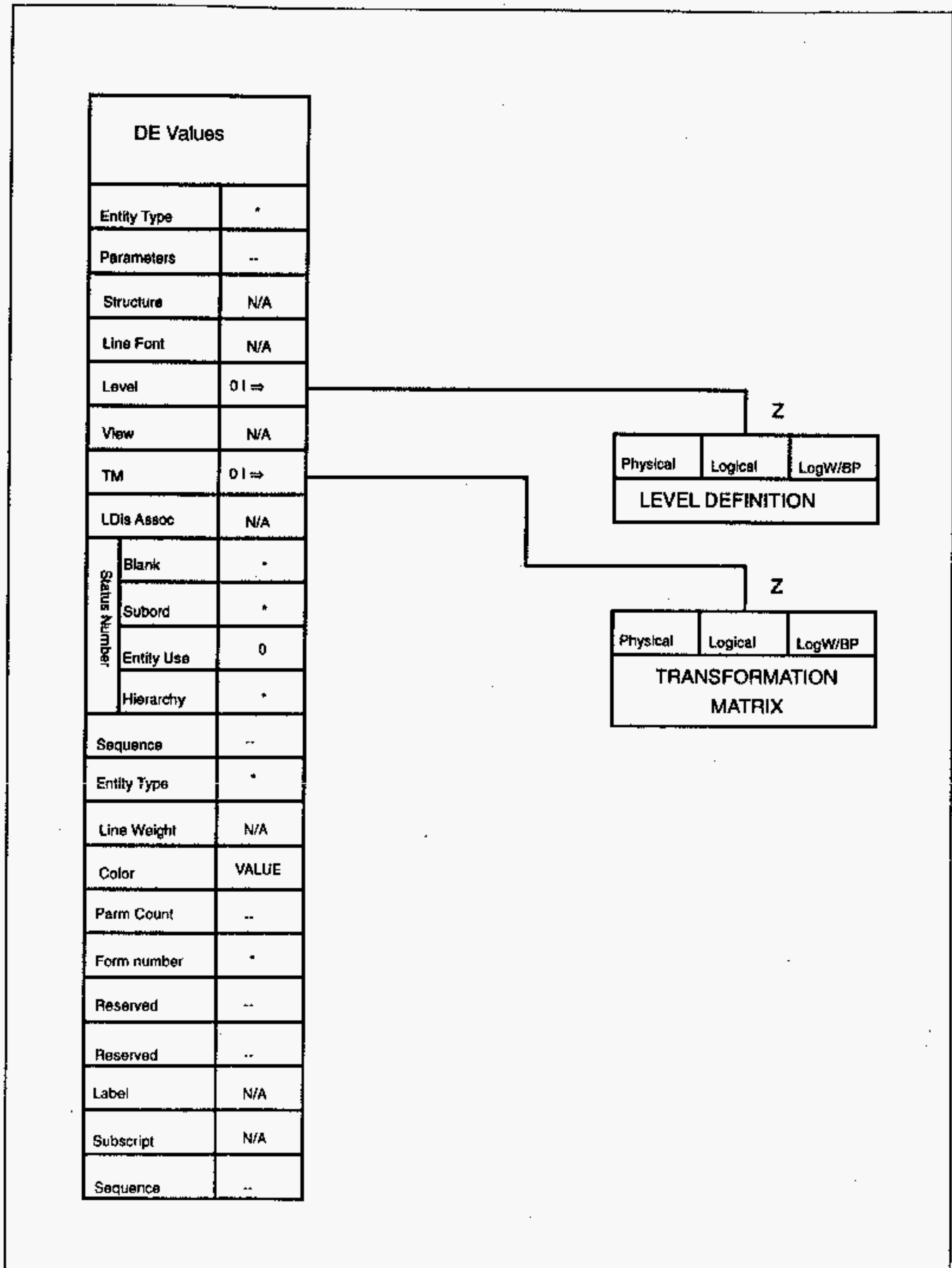
Requirements/Restrictions:

Translation Usage Notes:

General:

Output:

Input:



5.3.7.5. Property DE

Description:

The *PROPERTY DE* object is an IGES Directory Entry Section template to be used by all property and attribute entities.

Requirements/Restrictions:

1. For the Level to *LEP LAYER MAP PROPERTY*, SUBORD = 0

Translation Usage Notes:

General:

Output:

Input:

DE Values		
Entity Type		*
Parameters		–
Structure		N/A
Line Font		N/A
Level		N/A
View		N/A
TM		N/A
LDis Assoc		N/A
Status Number	Blank	N/A
	Subord	00
	Entity Use	3
	Hierarchy	N/A
Sequence		--
Entity Type		*
Line Weight		N/A
Color		VALUE
Parm Count		--
Form number		*
Reserved		--
Reserved		--
Label		N/A
Subscript		N/A
Sequence		--

5.3.7.6. Transformation Matrix DE

Description:

The *TRANSFORMATION MATRIX DE* object is an IGES Directory Entry Section template to be used by all Transformation Matrix entities.

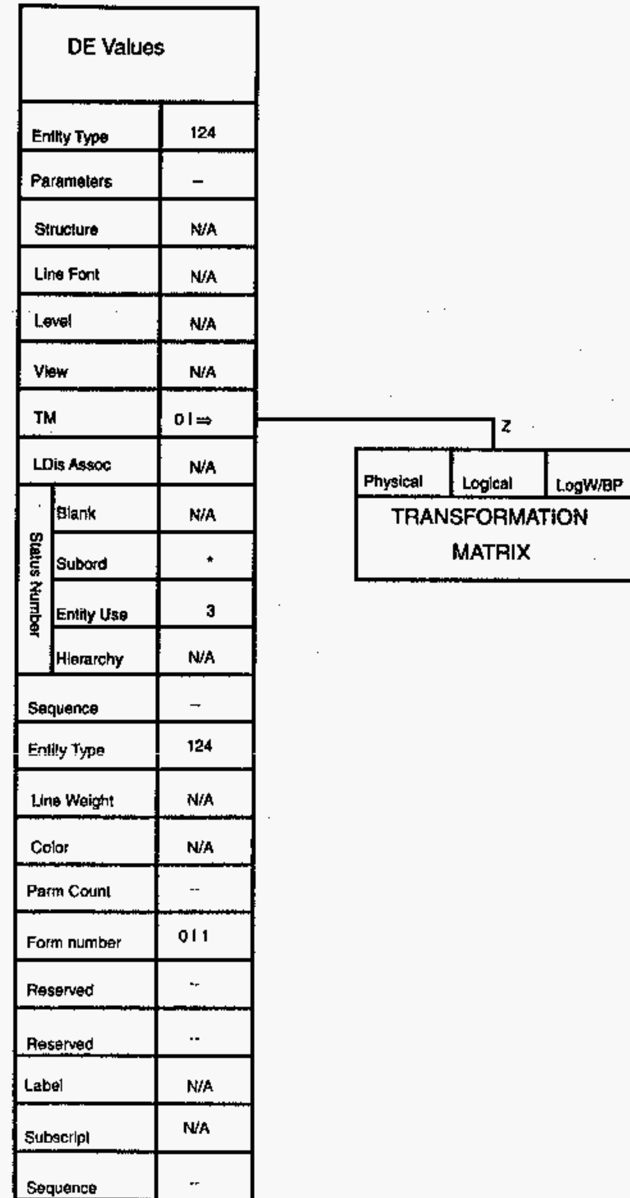
Requirements/Restrictions:

Translation Usage Notes:
A Transformation Matrix can reference another Transformation Matrix.

General:

Output:

Input:



Blank Page

(END OF SECTION 5)

6. IMPLEMENTATION AND CONFORMANCE TESTING GUIDELINES

The successful exchange of information using an IGES AP requires the participating organizations to establish information configuration control and software configuration control procedures for their product data creation and exchange systems. It must be understood that the use of IGES AP's will in many cases require organizations to revise their policies and procedures for the creation, exchange, and archival storage of product data. The successful use of an IGES application protocol also requires that the participating IGES processors conform to the AP specification. The purpose of conformance testing is to increase the confidence that different implementations of the AP will be able to exchange information successfully. This AP requires that the functionality of the LEP constructs be preserved in the translation into and out of the IGES format. Therefore, the CAD system for which the processors are being tested must provide this minimum level of functionality for modeling electronic product design and fabrication support systems. This section will define both testing requirements and guidelines for implementation of the functionality defined in the AIM section.

This Section often differentiates the file "which has been written" from the file "which has been read into" a system. The former is termed the "preprocessor" file by IGES; the former is also referred to as the "sending" system file herein. Terms for the latter file-types are "postprocessor" and "receiving" system.

6.1. PROCESSOR CONFORMANCE REQUIREMENTS

In order to conform to this AIM a CAE system shall be capable of reading and writing files that conform to the entire *LEP PHYSICAL LAYOUT* IGES File (see 5.3.1.2) object. It is appropriate to support all objects (including those objects needed for *LEP PARTS LIBRARY IGES FILES*, *TECHNICAL ILLUSTRATION IGES FILES* , and *LEP SIMULATION/ANALYSIS IGES FILES* which are not required for LEP PHYSICAL LAYOUT IGES FILES), but it is not necessary in terms of conformance.

The conformance requirements for implementations of this AP are enumerated as follows: 1.) All IGES files created by an AP compliant preprocessor shall conform to the IGES specification, Reference 4, and to the constructs specified in this AP. An AP compliant preprocessor shall convert LEP information of the ARM into the IGES constructs specified in the AIM, with all the required attributes and values. The functionality defined for each construct of the ARM shall be preserved. 2.) An AP compliant postprocessor shall interpret files that conform to the IGES constructs specified in this AP. It shall also convert each construct of the AIM into native constructs which match the geometry, attributes, and relationships of the LEP constructs specified in the ARM. The functionality of the LEP constructs shall be preserved. Any visual presentation produced by the postprocessor shall accurately and correctly represent the IGES constructs contained in the file and specified by this AP.

Development and use of the IGES AP Abstract Test Suite is divided into several Test Groups (TG). Each test group contains discrete Test Purposes (TP). A TP defines the objective of an abstract test case. For a preprocessor, the TP begins with "to test the generation of a(n)...". For a post processor, the TP begins with "to test the interpretation of a(n)...". An abstract test case is

required for the preprocessor and postprocessor. An abstract test case is derived from a TP and is written in a formal language. When parameter values are provided for the constructs in the abstract case, it can be used to generate an executable test case. An abstract test case contains: TP; test case identifier; reference to specific parts of the AP; definition of constructs required to exercise the TP; statements indicating the construction sequence; and verdict criteria. Abstract test cases are documented in non-system specific procedures and are used to produce comparable results from the conformance testing of multiple implementations.

An executable test case is derived from an abstract test case and is in a form which allows it to be run on the implementation under test. An executable test case contains some or all of the following; TP; test case indentifier, reference to specific parts of the AP; constructs required to exercise the TP together with their associated parameter values; a test script defining the construct sequence; verdict criteria; an IGES file for postprocessor conformance testing; and a pictorial representation of the populated constructs.

6.1.1. Testing Considerations

In any situation where there is more than a modicum of information to represent, more complex structures are made from combining less complex structures in a very well defined and specific manner. The depth of such nesting is inconsequential, the relationships between the separate parts at a given nesting level, and the discrete assemblies at adjacent levels, are of prime importance. The IGES Electrical Application Committee defines the minimum level of intelligence as being able to support the concept of connectivity between two or more individual connection points on one or more individual components using a unique path that connects only these specified objects. Further, each unique path has a name, as does each pin on each component. This functionality is defined in IGES 3.6.3 and is crucial for CALS (MIL-D-28000, as a replacement for Class III).

Each of these constructs is an organized collection of simpler[1] items, the highest construct may be a fully populated PCB, the lowest item is an individual entity. For example, several individual Line entities may be collected together using a signal list associativity[2]. The signal list associativity may have a name of "reset." The Lines it points to may each have their own properties, such as Line Widening (trace thickness). Each Line may be on a different level, with the thickness of the lines on the inner levels being different from those on the outer levels. These single individual Lines, with their own properties become one item at the next level of abstraction. A PCB via keepout is simply a collection of contiguous lines or closed curves gathered together with an associativity which has a property of "board via keepout."

6.1.2. Conformance Considerations

The Layered Electrical Product Application Protocol defines how different functional constructs are represented. This eliminates the confusion and ambiguity that may result when there is more than one way to do something. However, files may conform to the AP, yet not include every entity in the AP. Conformance simply says that the information in the conformant file is represented using the AP defined entities, in the AP defined manner, so as to get the most information possible exchanged.

1. Simple in terms of assembling the final product; recognizing that a bare chip integrated circuit, often the "starting point" of a product, may be the most complex LEP assembly component in terms of design.
2. The name of the actual IGES entity here called "signal list" is an IGES "Flow Associativity." The concept of connectivity is shared by both the electrical application, and the piping application.

For example, a CALS Class III IGES file may well conform to the AP even though it does not contain a thermal outline for each component. It does not conform to the AP if all of the components are represented with simple Subfigure Definition and Instance entities, even if there is a thermal outline included in the subfigure definition. Yet a conforming file may well contain simple Subfigures, indeed, the padstacks may be represented with them.

Determining conformance is not as easy as just looking for the existence or absence of certain entities. Even assuming that all of the right entities are there, they may not be hooked up correctly, and simply noticing that a key entity, such as the signal list (called Flow Associativity in IGES) is missing doesn't mean it is not compliant.

All of the extraction algorithms presented are only valid for LEP AP files. They assume that each component has a reference designator as part of it's make up, or that a signal may be followed through the paths on the board, to each separate connect point (which has a pin identifier), which is part of a unique component instance.

While a MIL-D-28000 Class I picture looks just like a Class III picture, the lack of intelligence of the Class I file precludes most process automation and machine controlled functions. It is just a picture, and as such, only worth about 1,000 words.

Due to the complexity of this AP, it is not feasible to conduct exhaustive testing of processors for all possible combinations of AP constructs. The testing approach described in Section 4.1 of Reference 17, called "falsification testing," is the approach followed in this AP. The conformance testing requirements described in this section cover all constructs of the ARM. The enumerated test groups (sec 6.2) are not exhaustive (i.e., they do not cover all possible combinations of ARM and AIM constructs) although the commonly encountered information required for transfer indicated in Figure 6-1 is present.

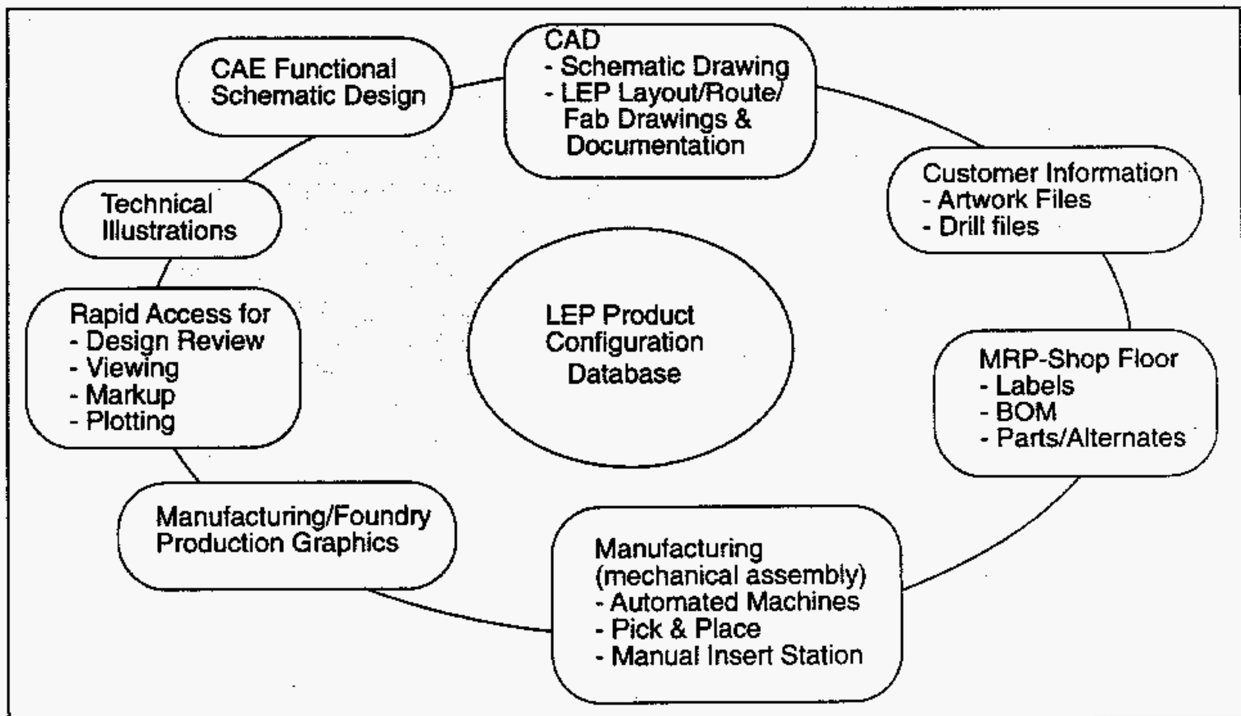


Figure 6-1, LEP Information Interface Requirements

6.2. Test Purposes and Test Groups

to test the generation of a(n)

to test the interpretation of a(n)

LEP Product Model Test Groups:

Test Group 1: Geometry, open and closed.

Test Group 2: Electrical and non-electrical symbols.

Test Group 3: Padstacks.

Test Group 4: Pads, Through-Hole pins, Vias and Testable Points.

Test Group 5: Electrical Component Definition.

Test Group 6: Electrical Component Instance.

Test Group 7: Stand-alone Vias, Pads.

Test Group 8: Netlist.

Test Group 9: LEP PWB.

Test Group 10: Photolithograhy.

Test Group 11: LEP Fabrication.

Test Group 12: Machine Controlled Processes.

Test Group 13: Production Test.

LEP Special Uses Test Groups:

Test Group 14: Non-electrical Part.

Test Group 15: Panel.

Test Group 16: Production Engineering.

Test Group 17: Technical Publication.

Test Group 18: Dimensions.

Test Group 1: Geometry, open and closed.

Open Geometry is used for ordinary graphics, and electrical Traces. Closed Geometry is used for both Outlines and filled areas.

Test Purposes:

1. Point with no display symbol.
2. Two Point Line, 1.0 inch long.
3. Two Traces, each being 1.0 inch long and 0.010 inches wide and having two Points. The two Traces shall each have one end Point in common with the other. Each Trace shall be on a different Level.
4. Four line segments that define a closed, coplanar square shape (PLEASE NOTE: this is also known as a “Closed Curve” or an “Outline”).

5. Grouping of four line segments and four 90 degree Arcs that define a square closed area with 0.25 inch radius rounded corners such that they are treated as a single entity (selected, moved, deleted).
6. Simple filled area. Grouping of four line segments that define a closed area such that they are treated as a single entity and filled with a solid fill.
7. Complex filled area. A filled area in the shape of an upper case character "T". Each area shall be longer than 2.0 inches, and wider than 1.0 inch. All corners shall be rounded with 0.25 inch radius Arcs.

Test Group 2: Electrical and non-electrical symbols.

Test Purposes:

1. Simple filled area 0.120 inch extents; a circle, a square, a diamond.
2. Complex filled area 0.120 inch extents; a bow-tie, a bulls-eye.
3. Simple filled area 0.240 inch extents; a circle, a square, a diamond.
4. Fiducial.

Test Group 3: Padstacks.

Test Purposes:

1. Two 0.060 diameter circular conductive areas on two distinct Levels.
2. Two 0.048 diameter circular conductive areas on two other distinct Levels.
3. A closed square Outline with the Geometry being 0.010 inches wide, and 0.100 inch center to center spacing.
4. A 0.046 diameter drilled Hole, passing through the entire LEP, plated to 0.036 diameter.
5. A display symbol, that provides visual identification of the drilled Hole. This symbol shall be on yet another distinct Level.
6. The grouping of items 1-5 such that they appear as 1, addressable.... (PLEASE NOTE: this group is called a "Padstack").

Test Group 4: Pads, Through-Hole Pins, Vias and Test Points.

Test Purposes:

1. 0.060 square surface mount Pad.
2. 0.060 square surface mount Pad that may be used as a Test Point.
3. Padstack that is a Component Pin.
4. Padstack that is a Component Pin that may be used as a Test Point.
5. Padstack that is a Via that may be used as a Test Point.

Test Group 5: Defined Electrical Component.

Test Purposes:

1. Closed Curve defining the 2D physical Outline of the Component.
2. Closed Curve defining the 2D thermal Outline of the Component.
3. Closed Curve defining the 2D LEP PWB graphics Outline of the Component. (i.e. the Outline on the board; the "silk screen" graphics).
4. A Component of a specific package type (dip, sip, axial,...).
5. An SMT Component.
6. A Through-Hole Component.
7. The height of the Component.
8. A restriction confining placement to one side only.
9. A restriction that the Component may not be auto-inserted or placed.
10. An association of a Component Pin/Pad to Component.
11. The Padstack referenced by Pin 1.
12. The Padstack referenced by other Pins.

(PLEASE NOTE: may be more than 2 types- quad has 4).

13. All Pins/Pads that are part of this Component.
14. The x,y location (relative to Component origin) of pin1.
15. The x,y location (relative to Component origin) of other Pins.
16. The alphanumeric tag of each Pin/Pad occurrence.
17. A restriction that a specific Pin/Pad may not be used as a Test Point. (while the Pin/Pad itself may be used as a Test Point, it may be disallowed when packaged due to problems like clearance or neighboring components.

Test Group 6: Placed Electrical Component.

Test Purposes:

1. Component with a unique reference designator.
2. The x,y location of a Component.
3. The part number of a Component.
4. Association of the placed Component to the defined Component.
5. Component which must be inserted manually.
6. Component on the bottom side of the LEP PWB.
7. Component on the top side of the LEP PWB.
8. An association of a Pin/Pad to its parent Component.

Test Group 7: Stand-alone Vias, Pads.

Test Purposes:

1. Through-pin as a stand-alone Via not related to a Component.
2. The x,y location of a stand-alone Via.
3. The alphanumeric identifier of a stand-alone Via occurrence.
4. The signal connected to a stand-alone Via occurrence.
5. Restriction that the Via may not be used as a Test Point.
6. Pad as a stand-alone object not related to a Component.
7. The x,y location of a stand-alone Pad occurrence.
8. Alphanumeric identifier of a stand-alone Pad occurrence.
9. Signal connected to a stand-alone Pad occurrence.
10. Restriction that the Pad occurrence may not be used as a Test Point.

Test Group 8: Netlist.

Test Purposes:

1. Association of all Pins that share a specific signal.
2. Association of all Pads that share a specific signal.
3. Association of all Vias that share a specific signal.
4. Association of all Test Points that share a specific signal.
5. Association of all interconnect Geometry, such as Traces, jumpers, and filled planes that share a specific signal.
6. Signal name.
7. Association of all objects (Pins, Pads, Vias, Test Points, and interconnect Geometry) to a signal name.

Test Group 9: LEP PWB.

Test purposes:

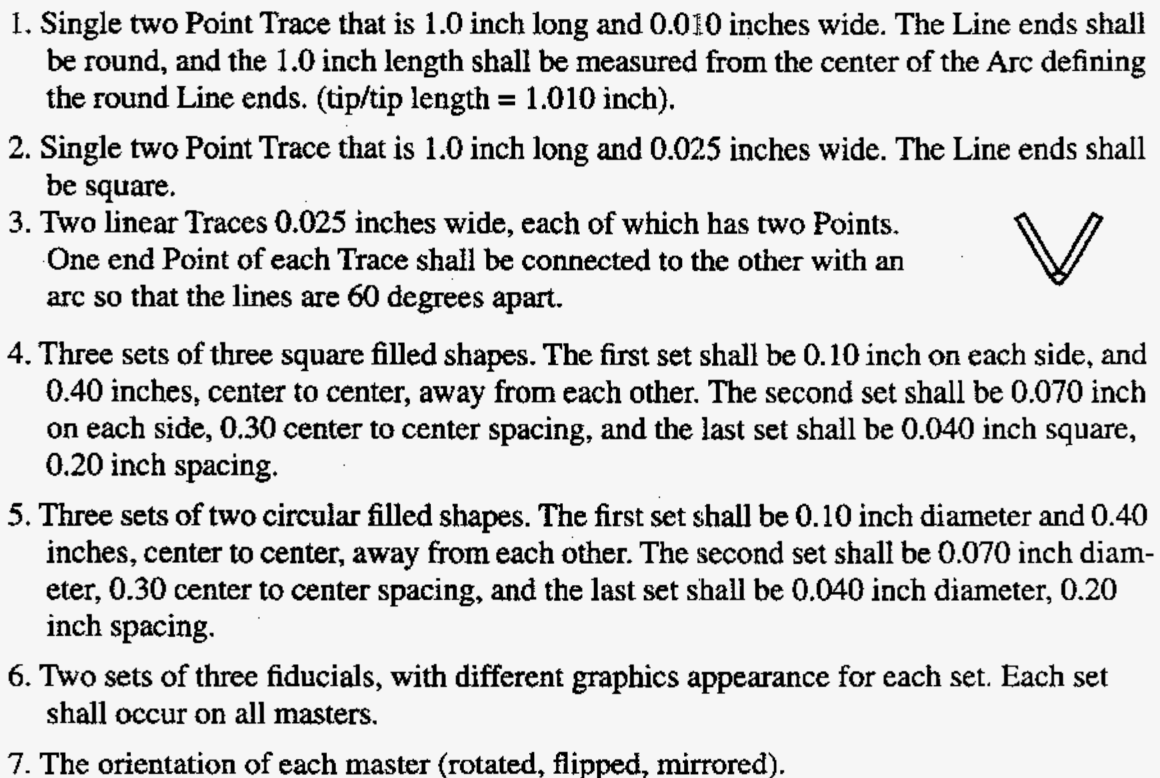
1. The physical Outline of the PWB.
2. The origin of the PWB.
3. The Outline(s) defining Component Keepin/Keepout regions.
4. The Outline(s) defining Trace Keepin/Keepout regions.
5. The Outline(s) defining Via Keepin/Keepout regions.
6. The Outline(s) defining conductive planes.
7. Default Via.
8. Default Padstack.

9. Padstack that is a Via, and it may be used as an allowable Test Point.
10. SMT Pad that may be used as an allowable Test Point.
11. Alphanumeric identifier such as "Assy 123-456-7890.rev2.1"
12. Three master registration fiducials, one of which is at the origin of the LEP PWB. The symbols shall be unique in appearance, and associated with each other.
13. Three fabrication fiducials, one of which is at the origin of the LEP PWB. The symbols shall be unique in appearance with respect to all other depictions in the file. The symbols shall be associated with each other.

Test Group 10: Photolithograhy.

Test Purposes:

1. Single two Point Trace that is 1.0 inch long and 0.010 inches wide. The Line ends shall be round, and the 1.0 inch length shall be measured from the center of the Arc defining the round Line ends. (tip/tip length = 1.010 inch).
2. Single two Point Trace that is 1.0 inch long and 0.025 inches wide. The Line ends shall be square.
3. Two linear Traces 0.025 inches wide, each of which has two Points. One end Point of each Trace shall be connected to the other with an arc so that the lines are 60 degrees apart.



4. Three sets of three square filled shapes. The first set shall be 0.10 inch on each side, and 0.40 inches, center to center, away from each other. The second set shall be 0.070 inch on each side, 0.30 center to center spacing, and the last set shall be 0.040 inch square, 0.20 inch spacing.
5. Three sets of two circular filled shapes. The first set shall be 0.10 inch diameter and 0.40 inches, center to center, away from each other. The second set shall be 0.070 inch diameter, 0.30 center to center spacing, and the last set shall be 0.040 inch diameter, 0.20 inch spacing.
6. Two sets of three fiducials, with different graphics appearance for each set. Each set shall occur on all masters.
7. The orientation of each master (rotated, flipped, mirrored).

Test Group 11: LEP Fabrication.

Test purposes:

1. Buried Via.
2. Unplated tool Hole with a diameter of 0.125 inch.
3. A complex cutout. For example, one that looks like a keyhole. The diameter of the circular part of the cutout shall be 0.400 inch. The width of the straight line portion shall be 0.200 inch. The total length shall be 1.0 inch.
4. The order of each master in the lay-up sequence.
5. Fiducials used for position of stratum registration.

Test Group 12: Machine Controlled Processes.

Component pick/place; insertion; verification.

Test purposes:

1. Identifier at the maximum coordinate along each side.
2. All Components that are between 1.0 and 3.0 inches along one edge of the LEP PWB, and 0.0 and maximum board size along the other edge.
3. The x,y location of the Component center.
4. The x,y physical size of the Component (x/y min/max).
5. The rotation of the placed Component with respect to the Component definition.
6. The lead span of a leaded, auto-insertable Component.
7. The lead diameter of a leaded, auto-insertable Component.
8. Insertion force required for a leaded, auto-insertable Component.
9. The x,y location of pin1 (used in verification).

Test Group 13: Production Test; PWB.

Note that each signal should have at least one Test Point, preferably an assigned Test Point.

Test purposes:

1. Padstack that is a Component Pin, and is an allowable Test Point.
2. Padstack that is a Component Pin, and is an assigned Test Point.
3. Padstack that is a Via, and is an allowable Test Point.
4. Padstack that is a Via, and is an assigned Test Point.
5. Pad that is an allowable Test Point.
6. Pad that is an assigned Test Point.
7. A unique identifier for each Test Point.
8 The x, y location, relative to the origin of the LEP PWB, of each assigned Test Point.
9. The signal name at each assigned Test Point.
10. A tester assigned identifier for each assigned Test Point.
11. Electrical properties, such as voltage, amperage, frequency.

Test Group 14: Non-electrical part.

Test purposes:

1. The x,y location of a Component.
2. The part number of a Component.

3. Association of the placed Component to the defined Component.
4. Component which must be inserted manually.
5. Component on the bottom side of the LEP PWB.
6. Component on the top side of the LEP PWB.

Test Group 15: Panel; PWB.

Test purposes:

1. For the next graphical display.
2. Color, linefont, display thickness.
3. A symbol for each different drill size [reference to MIL-D-28000 Class IV].

Test Group 16: Production Engineering (process sheets).

Test purposes:

1. Identify components, component outlines, product outline.
2. Appearance (see IGES Drafting AP).

Test Group 17: Technical Illustrations.

1. Appearance (see IGES Drafting AP).

Test Group 18: Dimensions.

Test purposes:

1. Dimensions, views (see IGES Drafting AP).
2. Appearance (see IGES Drafting AP).
3. Font, color, DE Line weight.

6.3. LEP Constructs

The layered electrical product is a complex assembly made up of both graphic and non-graphic information, interrelated through parent/child relations and associations. This Section provides the additional detail about IGES entities and structures for given use applications. This information supplements and enhances the object descriptions in Section 5.3.

6.3.1. LEP Constructs Supporting Signal Conductivity

Traces: Traces (*JOIN* objects)use simple geometry entities such as lines and arcs. They interconnect electronic Components and carry Signals. Physically, they have a specific width. Portions of the resulting Trace may be deposited on more than one layer. Traces usually pass between the different layers by means of *VIAS* and through-pins.

Jumpers: Jumpers are a physical interconnect which may be implemented as a component or a trace. The jumper and the other *JOINS* it connects shall bear a single signal name.

Planes: Planes, such as ground and power planes, outline the appropriate (filled) areas.

Vias: *VIAS* are holes made between two or more layers on the LEP. They are plated so they can carry the Signal. The holes are made by different means such as drilling, milling or cutting. This variability makes it hard to select specific entities which relate to some aspects of Vias. However one makes the hole, the functions of the Via are still the same. Vias may be through, blind, or buried. Vias are often used as Test Points.

Padstacks (through and SMT): Padstacks provide the electrical interface between the LEP PWB and the Component. Padstacks may be either through-hole or surface mount. Each Padstack shall have a LEP defined property to identify which type it is.

6.3.2. LEP Constructs Supporting a Physical Representation

A physical LEP shall contain at least one simple closed planar geometry entity depicting the LEP outline. This LEP *CLOSED CURVE* may be a *KEEPIN/KEEPOUT*, tool path, or any other functional type of closed geometry entity.

Several applications require a PDQ LEP representation. Computer controlled equipment may then extract needed data without manual intervention. For example, Component insertion equipment can put a Component at the X, Y coordinates specified by the Network Subfigure Instance Entity (Type 420) locating the Component.

Operations which modify the physical representation of the LEP, such as scaling or rotation, will have adverse affects on some application systems. Constructs which are beyond the scope of an automated operation (such as dimensions and view/draw) could stall or disable the processing system.

6.3.3. LEP Constructs Supporting a Logical Representation

A logical LEP element (e.g., a gate or a driver) shall be represented by a Network Subfigure Definition Entity (Type 320). The Definition may or may not reference a *CLOSED CURVE* depicting a schematic symbol. In either case, the Network Subfigure Definition Entity (Type 320) shall list the ports corresponding to physical connect points. The corresponding Connect Points (Type 132) referenced by the Network Subfigure Instance Entity (Type 420) may participate in logical flow associativities. Physical locations are not needed. If present, they locate symbols and port representations in the "drawing space" of the schematic.

6.4. Entity Usage in a PDQ LEP Representation

IGES allows many different ways to add semantics to a file. Among these ways are using properties, associativities, groupings, and even attributes. One way uses the level on which an entity is found. Another uses the color of the entity. These may seem like clever schemes which make it easy for users or vendors to support the semantic convention. However, it actually leads to confusion, frustration, and reduced semantic information exchange when the convention is not both well documented and universally supported. This Section provides the additional detail about IGES entities and structures for given use applications. This information supplements and enhances the object descriptions in Section 5.3.

The Application Protocol for a PDQ LEP will focus on the use of IGES Property Entities

(Type 406) to add more semantic information to the LEP.

This section specifies what entities may appear in conformant files in a PDQ LEP representation. This section also specifies what entities are required in a PDQ LEP. Some of the required entities are Property Entities (type 406). It is not difficult for a pre-processor to include these Properties when writing an IGES file. This approach does assume though, that the data for the Properties do exist in the native database of the sending system. For example, the component_height property is required for several applications. However, if the value is not contained in the part data within the originating CAD system, the component_height property cannot be generated in a conformant way. This may lead to rejection of an IGES file for non-conformance when the fault was not with the IGES preprocessor.

Please remember that all rules and requirements which apply to all IGES entities used still apply to those conformant to the LEP AP. For example, a zero length text string is non-conformant and the maximum pre-defined Color Number is eight.

6.4.1. DE Fields are NOT Expected to Convey Meaning

DE fields shall not convey any semantic information beyond their specific use.

DE fields 1,2,10,11,14,15 and 20 are artifacts of the physical format of the IGES file, and shall not provide semantic information.

Table 6-1. Overview of DE Settings for PDQ

Note	DE Field Name	DE #	Acceptable Values
a	Structure	3	0
b	Line Font	4	1
c	Level	5	0 to 1024
d	View	6	0 or pointer
e	Transformation Matrix	7	#, or pointer
f	Status; Blanked	9.1	*ee
g	Status; Subordinate	9.2	*ee
h	Status; Entity Use	9.3	*ee
i	Status; Hierarchy	9.4	*ee
j	Line Weight	12	per IGES
k	Color	13	0 to 8
l	Reserved (blank)	16	“ ”
l	Reserved (blank)	17	“ ”

Table 6-1. Overview of DE Settings for PDQ

Note	DE Field Name	DE #	Acceptable Values
m	Label	18	" "
m	Entity Subscript	19	0 or " "
n	Label Display	8	0 or " "

(a) Structure (DE 3): DE field 3 has no relevance to any portion of the LEP AP and shall be set to zero.

(b) Line Font (DE 4) No functionality shall be provided through Line Font. Line Font shall be 1 for all entities of a PDQ LEP.

(c) Levels (DE 5): It is a common practice in the design of a layered electrical product to use the system's "layers" to specify functionality. For example, LEP signal Traces may be assigned layers 1 through 24; layer 31 has top surface reference designators, and so on. While the practice is common, the layering convention is not. In fact, there is often a lack of consistency in layering conventions within the same design group. Thus, the layer a particular entity is on, shall convey no functionality in and of itself. Yet, layering must be understood and processed.

Therefore, a Level Function Property (type 406 form 3) shall be applied to each level. The Level Function stands alone, the level it applies to need not be in the file.

(d) Views (DE 6) The value shall be zero for all entities which are part of the PDQ PWB.

(e) Transformation Matrix (DE 7)

(f) Status; Blanked (DE 9.1)

(g) Status; Subord (DE 9.2)

(h) Status; Entity Use (DE 9.3)

(i) Status; Hierarchy (DE 9.4)

(**e) These switches vary, refer to specific entity for correct settings.

(j) Line Weight (DE 12) Line Weight shall imply no functionality. Also, the Line Widening Property (type 406 form 5) shall represent Trace thickness.

(k) Color (DE 13): Color shall imply no functionality. One of the eight non-zero Color Numbers shall appear in the proper DE field 13 for every entity having a color attribute. The Color Number shall appear in field 13 of the entity itself.

(l) DE fields 16 and 17 have no relevance to any portion of the LEP AP and shall ignored. Both fields shall contain blank characters only.

(m) DE field 18 and 19 taken together uniquely identify the entity within the range of the IGES file. The Entity Label may provide a name to distinguish each particular kind of usage of a given Entity Type from others having the same Entity Type Number and Form number. This AP does not require usage of this construct.

(n) Label Display (DE 8) This field shall contain either a zero or blanks.

6.4.2. Entities Which MAY Appear in a PDQ LEP Representation

6.4.2.1. LEP Usage of Geometry

Geometry entities may represent many distinct kinds of LEP objects and features. Examples include a LEP outline, a surface mount Pad, and a strangely shaped power plane. In each case, the geometric shape depicts the object or feature graphically. In some cases, the same IGES entity

types may have been used for each object. Thus, no semantic information is provided by most Geometry (type 1xx) entities.

Geometry entities may also represent non-electrical objects and features such as tooling holes, *FIDUCIALS*, LEP outline(s), and heat sinks. Electrical parts such as Vias, Components, edge connectors, paste resistors, SMT Pads, testpoints, Padstacks, Traces, jumpers, and signal planes are represented in like manner.

Most of the geometry (but not all) will be on the X-Y plane, with Z being 0.0. Even the geometry representing signals on top, inner, and bottom layers will have z = 0.0. The Z depth of each layer is determined by the substrate thickness and the sequence of layer assembly.

The bulk of the LEP Application needs for geometry do not require exotic Geometry entities. So, a reduced set of Geometry entities permitted in an LEP AP conformant file has been defined. This reduced set, the "LEP Geometry Entities," is defined below.

There are two uses of geometry in the LEP AP. One use is to define enclosed areas, the other is to define a non-closed linear path as defined in Section 5.3.3. and as detailed below.

Enclosed areas may be defined with a single circle, a multi-segment line, or a combination of geometries which are grouped together. A single Geometry Entity shall represent or associate the constituent Geometry Entities that enclose the given area. The Circular Arc Entity (Type 100) and Copious Data Enclosed Planar Curve Entity (Type 106, Form 63) are examples of single entity enclosed areas. Multi-entity paths which form an enclosed area and are subordinate to an occurrence of a Composite Curve Entity (Type 102) typify a multi-entity enclosed area.

Levels may imply grouping, but shall not imply function. The Level Function Property Entity (Type 406, Form 3) shall assign function.

The following "LEP Geometry Entities" (as restricted in this section) may appear in LEP AP conformant files:

Circular Arc Entity (Type 100)
Copious Data Entity (Type 106, Forms 11 and 63)
Line Entity (Type 110)
Point Entity (Type 116)
Flash Entity (Type 125)
Connect Point Entity (Type 132)
Composite Curve Entity (Type 102)
Plane Entity (Type 108)
Transformation Matrix Entity (Type 124)

(The determinant of the Transformation Matrix shall be +/- 1.0)

(no other scaling shall appear in conformant files.))

Filled areas shall use the Sectioned Area Entity (Type 230). Only the (full circle) Circular Arc Entity (Type 100), Composite Curve Entity (Type 102), Flash Entity (Type 125) and Simple Closed Planar Curve Entity (Type 106, Form 63) shall represent areas which are shown filled.

The Flash Entity (Type 125) shall not reference a Transformation Matrix.

6.4.2.2. LEP Usage of Annotation

The following annotation entities may appear in conformant files:

General Note Entity (Type 212, Form 0)
(Type 212, Forms 1,6,7,8) as required (see 6.6.3.5.)

Sectioned Area (fill area) (Type 230, Form 0) fill patterns 0,19 only

The Sectioned Area Entity (Type 230) shall depict filled areas such as irregularly shaped signal planes, ground planes or power planes. Only the (full) Circular Arc Entity (Type 100), Composite Curve Entity (Type 102), and Simple Closed Planar Curve Entity (Type 106, Form 63) shall be filled by the Sectioned Area Entity. Only the unfilled Fill Pattern (0) and the solid filled Fill Pattern (19) of the Sectioned Area Entity (Type 230) shall appear in conformant files.

No other annotation entities shall appear in conformant files.

Table 6-2. DE Settings of Annotation Entities in a PDQ IGES File

Annotation Entity - DE Field		General Note	Sectioned Area
Line Font	(4)	0, 1	0, 1
Level	(5)	0 to 1024	0 to 1024
View	(6)	0	0
Transformation Matrix	(7)	0, pointer	0, pointer
Status; Blanked	(9.1)	00	00
Status; Subordinate	(9.2)	01	01
Status; Entity Use	(9.3)	01	01
Status; Hierarchy	(9.4)	01	01
Line Weight	(12)	per IGES	per IGES
Color*	(13)	0 to 8	0 to 8

*Support of this entity (the Type 314) is not required for preprocessors by this AP, but support is required by IGES for post- processors. Please refer to the discussion in section 6.4.1. and 6.6.4.2.

6.4.2.3. LEP Usage of Structure (Excluding Properties)

The following entities may appear in conformant files:

Subfigure Definition Entity (Type 308)
Subfigure Instance Entity (Type 408)
Network Subfigure Definition Entity (Type 320)
Network Subfigure Instance Entity (Type 420)
Text Display Template Entity (Type 312)

6.4.2.4. LEP Usage of Existing Specific Property Entities

One of the easiest and most widely supported methods for adding more semantic information to an entity or group of entities is the Property Entity (Type 406). A property is a well defined mechanism in IGES. One entity may have several Properties. One Property may apply to many entities. An entity may exist with no Properties, and a Property may be in a file but not referenced by any entity. The second group of additional pointers (see IGES section 2.2.4.4.2) provide the mechanism for referencing Properties.

The conformant LEP representation will use Properties in adding both graphic and non-graphic information in the file. Properties are of Type 406. The conformant LEP representation will use existing Properties whenever possible. The Generic Data Properties detailed in this AP shall be used in most other cases. This means that other methods of adding semantic information (such as the Attribute Table) will not be conformant.

Definition Level Property Entity (Type 406, Form 1) allows an entity (or entities) to exist on more than one level. The DE Level field of the target entity (or entities) contains a pointer to the Definition Levels Property. The Definition Levels Property will list all the levels on which the target entities exist. For example, the DE field of a Drilled Hole Property may contain a pointer to the Definition Levels Property which lists all the levels penetrated by the drilled hole.

Region Restriction Property Entity (Type 406, Form 2) may identify keepin/keepout areas for Traces, Components and Vias.

Line Widening Property Entity (Type 406, Form 5) may specify line geometry thickness (e.g. to depict Trace width).

Drilled Hole Property Entity (Type 406, Form 6) may describe a drilled hole size, plating, and affected levels.

Part Number Property Entity (Type 406, Form 9) assigns up to four different unique identifiers for referencing a Component. The property provides for a generic part number, a military standard part number, a vendor part number, and an internal part number. Components (such as those represented by a Network Subfigure Instance) are the primary users of this Property.

Tabular Data Property Entity (Type 406, Form 11) used when a file requires annotation for simulation or analysis (SPICE, FEM, etc.).

Name Property Entity (Type 406, Form 15). The name Property may exist in the file, but shall not convey LEP semantic information. Specifically, it may name a level, but it shall NOT imply a level function.

Level to LEP Layer Map Property (Type 406, Form 24) defines a correspondence among four items: 1) An exchange file level number, 2) A native level identifier, 3) A physical LEP layer number, and 4) A pre-defined functional level identification.

6.4.2.5. LEP Usage of the (Augmented) Generic Data Property Entity

Several new Properties are introduced in order to simplify and extend the capability of IGES for the exchange of LEP semantic information. These Properties will all share a common form number. Thus, they must use pattern matching to determine the name, and therefore the function, of the Property. While this method increases processing time, it does allow a file to be complete and self contained. See Section 5.3.4.1. for definitions and Section 6.6.4.3. for usage tables of these

new Generic Properties.

6.4.2.6. LEP Usage of Simulation/Analysis Annotation Entities

Several entities provide attributes appropriate for circuit simulation, analysis, and other (e.g., finite element) modeling when it is desired to so annotate the LEP representation. While these annotations are not usually found in a general product definition file, these entities are useful during design and development life cycle phases. See Section 5.3.1.4. for the IGES file structure using these entities, and Section 6.6.5. for their usage guidelines.

6.4.3. Entities Which Shall NOT be Used in a PDQ LEP Representation

The following entities are those which have similar functionality to recommended entities. They shall not be used in LEP AP conformant files.

Reference designator Property Entity (Type 406, Form 7)
Reference designator is assigned in Network Subfigure Instance (type 420)

Pin Number Property Entity (Type 406, Form 8)
Pin number is assigned in Connect Point Entity (Type 132)

6.5. LEP Entity Usage for Core Applications

Some physical representations are common to many different application areas. This section will discuss them. This Section provides the additional detail about IGES entities and structures for given use applications. This information supplements and enhances the object descriptions in Section 5.3.

The LEP application includes engineering and design tasks such as schematic entry, netlist generation, auto-placement, auto-route, and Component engineering.

This LEP application DOES require a PDQ data repository.

All constructs specified in 6.5.1 apply when an LEP product is fully contained in a single file. When the LEP is described with a hierarchy of designs—each design in a separate file—the PWB component of the assembly will contain those features required for the PWB only as described in 6.5.4. When the hierarchy of designs is used, features such as padstacks, traces, drilled holes, and fiducials are found in the PWB file, and the LEP assembly file includes the identification of the PWB design required. NOTE: The use of "PWB" is intended to include the substrate for a MCM or other LEP product-types.

6.5.1. LEP Entity Usage for Padstack Representation

A Padstack shall be represented by an ordinary Subfigure Definition Entity (Type 308) as specified in 5.3.2.19.

The Padstack shall contain, at a minimum, one simple Closed Curve which represents a conductive area. That conductive area joins two signal carriers into one. This conductive area shall exist on at least one Level in the IGES file. That level shall be set in the Level field of the geometry entity. [The desired setting may require the use of a Hierarchy Property Entity (Type 406, Form 10).] More than one layer may share the same conductive area by application of the Definition Levels Property Entity (Type 406, Form 1).

Different geometric shapes which perform different functions (such as isolating a through-pin from a signal plane) may appear in the Padstack.

6.5.1.1. LEP Entity Usage for Physical Padstack Definition

Entities required:

A Subfigure Definition Entity (Type 308) shall define each Physical Padstack. Each such definition shall have an *LEP SEMANTIC PROPERTY* named component_default_padstack.

A *CLOSED CURVE* shall enclose the area of each Land in the Physical Padstack. See Geometry below for more details.

A Sectioned Area Entity (Type 230) shall fill the *CLOSED CURVE* which depicts a *LAND*. Only the Fill Patterns specified by code 0 (blank fill) and code 19 (solid fill) of the Sectioned Area Entity (Type 230) shall appear in conformant files.

Geometry -

At least one *CLOSED CURVE* shall define the outline of the area which represents the conductive Land in a Padstack. One of four LEP Geometry Entities may depict each Land. Those four are: the (full) Circular Arc Entity (Type 100), the Composite Curve Entity (Type 102), the Flash Entity (Type 125), or the Simple Closed Planar Curve Entity (Type 106, Form 63).

Annotation -

The *CLOSED CURVES*(s) enclosing the conductive area of the Land(s) in the Physical Padstack shall (each) reference a Sectioned Area Entity (Type 230). That Sectioned Area shall reference an *LEP SEMANTIC PROPERTY* named component_pkg_type (=pad). No other Annotation Entities shall appear as part of a Physical Padstack representation.

Structure -

The Subfigure Definition Entity (Type 308) shall depict the Physical Padstack. The Transformation Matrix field of the Subfigure Definition shall be zero. The Subfigure Definition shall reference each *CLOSED CURVE* described above as annotation. Each Closed Curve shall reference a Sectioned Area Entity (Type 230) with a Fill Pattern of 0 or 19.

Associativities -

Conformance to the LEP AP does not require any Associativities in the definition of the Physical Padstack.

Properties -

The Subfigure Definition Entity (Type 308) defining the Padstack shall reference the following properties:

LEP SEMANTIC PROPERTY) named physical_padstack_definition

LEP SEMANTIC PROPERTY named component_pkg_type (=pad).

6.5.1.1.1. Entities Required to Add the Representation of Drilled Holes into a Padstack.

Entities required:

A Point Entity (Type 116) shall locate the Hole. Each such Point Entity shall reference a Drilled Hole (Type 406, Form 6) Property.

Geometry -

A Point Entity shall provide the x, y location of the Hole(s) within the Padstack definition space.

Annotation -

Conformance to the LEP AP does not require any Annotation Entities in the representation of the Padstack.

Structure -

The Padstack Subfigure (type 308) shall reference one Point Entity (Type 116) for each hole. The Transformation Matrix field of the Point Entity (type 116) shall be 0.0. The Point Entity (Type 116) shall reference the Drilled Hole Property (Type 406, Form 6).

If the Point Entity (Type 116) references another Subfigure Definition Entity (Type 308) for display purposes, that definition shall exist in the same physical IGES file as the Point. More than one Padstack may use the same display definition.

Associativities -

Conformance to the LEP AP does not require any Associativities in the representation of the Point.

Properties -

The Point Entity (Type 132) locating the Padstack Hole shall reference the following Property:

The Drilled Hole (Type 406, Form 6) Property.

The Drilled Hole Property shall specify the Drill diameter size, the Finish diameter size, and the Plating indication flag of a plated Component Pin or Via hole. The DE Level field of the Drilled Hole Property Entity (Type 406, Form 6) shall be zero for a through-hole which passes through the entire LEP. The DE Level field of the Drilled Hole Property Entity (Type 406, Form 6) shall reference a Definition Levels Property Entity (Type 406, Form 1) for a Hole which does NOT go all the way through the PWB. That Definition Levels Property shall indicate which levels are penetrated.

The Point Entity (Type 132) locating the Padstack Hole may reference either or neither of the following two Properties:

The test_point *LEP SEMANTIC PROPERTY* named not_allowable_test_point or

the test_point *LEP SEMANTIC PROPERTY* named allowable_test_point.

If neither is referenced, the not_allowable_test_point semantic applies.

6.5.1.1.2. Entities Required to Add Display Appearance of Holes in a Padstack.

Entities required:

A Point Entity (Type 116) locating the Padstack Hole may reference a Subfigure Definition Entity (type 308) to display the presence of the *HOLE*. If the Point does not reference a Subfigure Definition in the file, the receiving system may display the default system icon for Point graphics

instead.

Geometry -

A Subfigure Definition Entity (Type 308) may depict the presence of the *HOLE* as illustrated in Figure 6-2. One *CLOSED CURVE* may define the outline of the area which represents the *HOLE* in a *PADSTACK*.

Annotation -

Conformance to the LEP AP does not require any Annotation Entities in the representation of the *HOLE*.

Structure -

The Point Entity (Type 116) may reference one Subfigure Definition (Type 308). The Transformation Matrix field of the Point Entity (Type 116) shall be zero.

Table 6-3. IGES Data Fragments Showing Padstack Structures

Entity Type	Form	DE Status	Parameter Data
308	0	010201	308,0,11Hpad_060_038,3,3,5,7,0,1,9;
100	0	010001	100,0.0,0.0,0.0,0.09,0.0,0.03,0.0;
230	0	010101	230,03,19,0.0,0.0,0.0,0.005;
116	0	010001	116,0.0,0.0,0.0,0,1,11;
406	27	020301	406,2,3,8HPadstack,3,17Hthru_hole_regular;
406	6	020301	406,5,0.046,0.038,1;
408	0	000001	408,1,0.2,0.2,0.0,1.0;

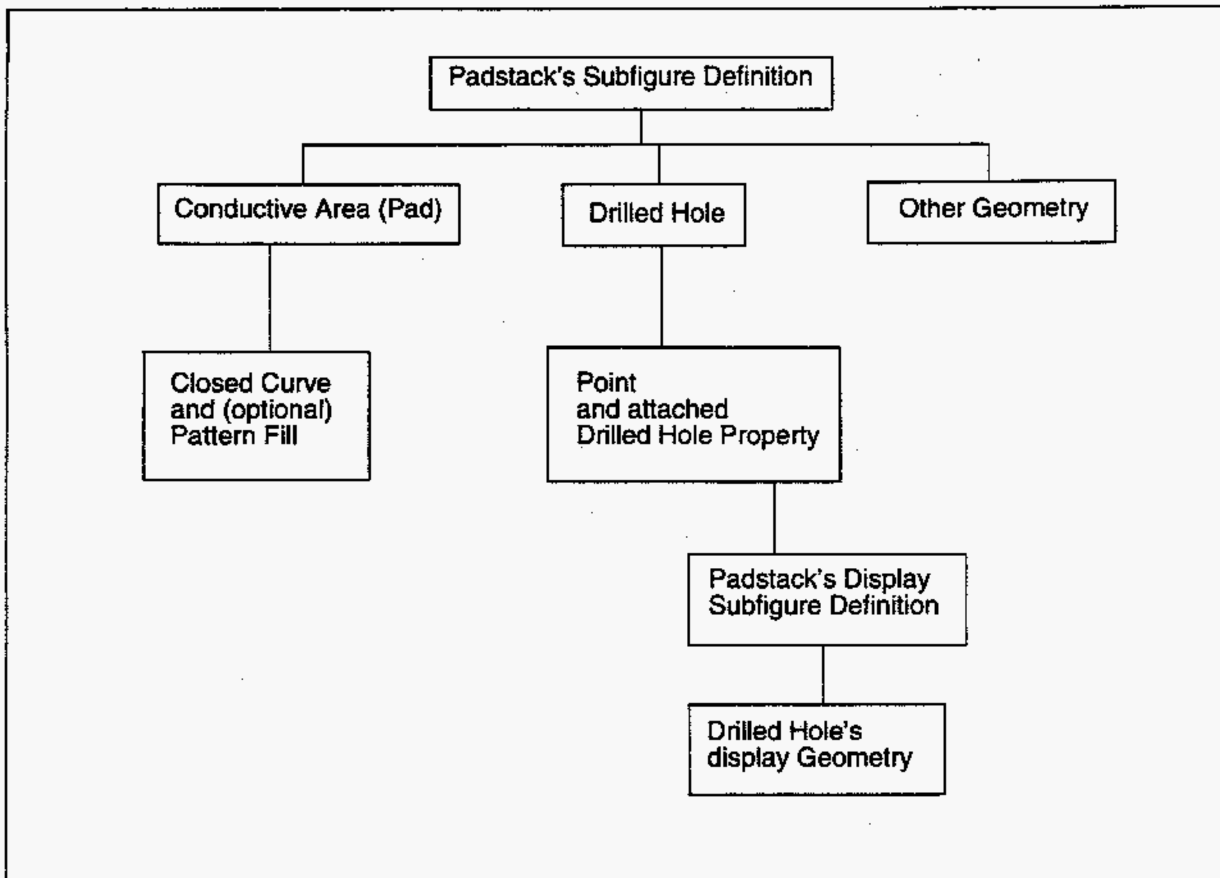


Figure 6-2. General Structure of Objects (See Section 5.2) Found in a *PADSTACK*

Associativities -

Conformance to the LEP AP does not require any Associativities in the representation of the Point Entity (Type 116).

Properties -

The Subfigure Definition Entity (Type 308) depicting the Padstack Hole shall reference the following Property:

LEP SEMANTIC PROPERTY named physical_outline.

6.5.2. LEP Entity Usage for Physical Component Representation

There are two parts to the representation of Physical Components in an IGES file. The first part is the definition. The second part is the instance. Multiple occurrences of similar Physical Components may share a single Definition. Connect Points in each Instance need to correspond exactly to their counterparts in the Definition. Circuit nodes, or Signals are represented by Flow Associativity Entities (Type 402, Form 18).

6.5.2.1. LEP Entity Usage for Physical Component Definition

The Physical Component itself is simple to represent. It requires a Network Subfigure Instance Entity (Type 420). Although it may be shared with other Instances, a Network Subfigure Definition Entity (Type 320) is also needed. One Connect Point Entity (Type 132) for each *PIN* or Pad is also needed. The Connect Points of the Instance may each reference a Flow Associativity Entity (Type 402, Form 18). The Flow Associativity groups all the pertinent objects subjected to the same electrical signal (*NET*).

Entities required:

A Network Subfigure Definition Entity (Type 320) shall define each Physical Component in the AP conformant files. Each Network Subfigure Definition (Type 320) shall reference as many LEP Geometry Entities as needed to depict the Component. Each Network Subfigure Definition Entity (Type 320) may reference many of the properties listed below. Conformance to the LEP AP does not require any Text Display Template Entities (Type 312) in the representation of the Physical Component. At least one Connect Point Entity (Type 320) shall provide a means of connecting signals to Pin 1 of the Component.

Geometry -

One Closed Curve shall define the outline of the Physical Component in the X, Y plane of the LEP. Each Closed Curve shall have one LEP Property attached to it which defines its function. Examples of LEP functions include: placement_outline, thermal_outline, physical_outline, and footprint. Please refer to Tables 6-10 through 12 for a complete list.

Any LEP Geometry Entity may appear as part of the Component Geometry. All LEP Geometry shall lie entirely in the Z = 0.0 plane.

Annotation -

Conformance to the LEP AP does not require any Annotation Entities in the representation of the Physical Component Definition.

Structure -

The Network Subfigure Definition Entity (Type 320) shall reference one or more LEP Geometry Entities. Each Geometry Entity may reference some properties. The Network Subfigure Definition Entity (Type 320) shall reference at least one property. The Network Subfigure Definition Entity (Type 320) Type Flag shall indicate (TF = 2) because the type is physical. The Network Subfigure Definition Entity (Type 320) shall reference one (for pin 1) or more Connect Point (Type 132) entities. The Transformation Matrix field of the Network Subfigure Definition Entity (Type 320) shall be zero.

Associativities -

Conformance to the LEP AP does not require any Associativities in the representation of the Component Definition.

Properties -

Mandatory Properties:

The Network Subfigure Definition Entity (Type 320) defining the Physical Component shall reference the following properties:

The *LEP SEMANTIC PROPERTY* named component_std_name.

The *LEP SEMANTIC PROPERTY* named component_version.

Optional Properties:

The Physical Component Definition (type 320) may reference the following properties:

The *LEP SEMANTIC PROPERTY* named component_pkg_type.

The *LEP SEMANTIC PROPERTY* named component_tech_type.

The *LEP SEMANTIC PROPERTY* named component_layout_surface.

The *LEP SEMANTIC PROPERTY* named component_insertion_origin.

The *LEP SEMANTIC PROPERTY* named component_insertable.

The *LEP SEMANTIC PROPERTY* named component_height.

The *LEP SEMANTIC PROPERTY* named component_max_size.

The *LEP SEMANTIC PROPERTY* named component_outline_overhang.

6.5.2.2. LEP Entity Usage for Physical Component Instance

Entities required:

The Network Subfigure Instance Entity (Type 420) locates the Component occurrence and provides a way to connect to its ports. A Connect Point Entity (Type 132) models each port (Pin or Pad). The Reference Designator shall be provided in the PRD parameter of the Network Subfigure Instance Entity (Type 420). The Part Number(s) shall be provided in the Part Number Property Entity (Type 406, Form 9). Other properties provide such information as whether the Component is attached to the top or the bottom of the LEP and how many 90 degree rotations perpendicular to the surface of the LEP are required of insertion processes.

Geometry -

A Connect Point Entity (Type 132) having an *LEP SEMANTIC PROPERTY* named pad_tech_type shall represent each Component Pin or each Component Pad in LEP AP conformant files. That property shall specify whether the Pad or Pin is of type surface mount (SMT) or through-hole. One of the Connect Points shall also reference an *LEP SEMANTIC PROPERTY*) named pin1.

One Connect Point Entity (Type 132) shall appear in the Network Subfigure Instance Entity (Type 420) for each Pin in the referenced Network Subfigure Definition, whether the Pin is actually used or not.

Annotation -

Conformance to the LEP AP does not require any Annotation Entities in the representation of the Physical Component Instance.

Structure -

The Network Subfigure Instance Entity (Type 420) shall reference one and only one Network Subfigure Definition Entity (Type 320). The Instance shall also specify the location of the Component with respect to the origin of the LEP.

The Network Subfigure Instance Entity (Type 420) shall reference one or more Connect Point Entities (Type 132).

Associativities -

The Connect Points of the Instance may each reference a Flow Associativity Entity (Type 402, Form 18). The Flow Associativity groups all the pertinent objects subjected to the same electrical signal.

Properties -

Mandatory Properties:

The Network Subfigure Instance Entity (Type 420) modeling the Component shall reference the following properties:

Part Number Property Entity (Type 406, Form 9)

LEP SEMANTIC PROPERTY named component_instance_side

LEP SEMANTIC PROPERTY named component_instance_rotation.

Each Connect Point Entity (Type 132) modeling a Component Pad or Pin shall reference the following LEP property:

The *LEP SEMANTIC PROPERTY* named pad_tech_type. Furthermore, the Connect Point Entity Entity (Type 132) modeling the Component Pad or Pin bearing the "pin1" designation shall reference the following LEP property:

The *LEP SEMANTIC PROPERTY* named pin1

Optional Properties:

The Network Subfigure Instance Entity (Type 420) may reference any of the following seven LEP *LEP SEMANTIC PROPERTY*:

component_preplace
abs_voltage_max
tolerance
therm_cond
therm_jc
therm_r
junction_max_t

6.5.2.3. LEP Entity Usage for Physical Connect Point

Entities required:

Connect Point Entities (Type 132) shall locate the place where a connection may occur between the Component and the LEP. One Connect Point Entity (Type 132) shall appear in the Network Subfigure Definition for each Pin in the corresponding Physical Component.

Geometry -

See Entities required above and Structure below. Any LEP Geometry Entity may appear as part of the Physical Connect Point geometry. All LEP Geometry shall lie entirely in the Z = 0.0 plane.

Annotation -

Conformance to the LEP AP does not require any Annotation Entities in the representation of the Physical Connect Point.

Structure -

The Connect Point Entity (Type 132) may reference one or more LEP Geometry Entities.

Each Geometry Entity may reference some properties. The CID parameter of the Connect Point shall provide the Pin Number. The Transformation Matrix field of the Connect Point shall be 0.0.

Associativities -

Each Connect Point Entity may reference a Flow Associativity Entity (Type 402, Form 18). The Flow Associativity groups all the pertinent objects subjected to the same electrical signal.

Properties -

Each Connect Point Entity modeling a Physical Connect Point may reference the following Properties:

Definition Levels (Type 406, Form 1) Property.

The *LEP SEMANTIC PROPERTY* named pad_tech_type (smt/through-hole_anti/thru_hole_regular/thru_hole_thermal).

The *LEP SEMANTIC PROPERTY* named pin1.

6.5.2.4. LEP Entity Usage for Physical Via

Entities required:

The Connect Point Entity (Type 132) models the Via. The Singular Subfigure Instance Entity (Type 408) may depict the shape of the Via. The Connect Point Entity may reference at most one Singular Subfigure Instance Entity in LEP AP conformant files.

Geometry -

The Connect Point Entity (Type 132) locates the Via within the LEP. A Singular Subfigure Instance Entity (Type 408) having a *LEP SEMANTIC PROPERTY* named stand-alone_via may tag each Via. Each such Subfigure Instance may reference any LEP Geometry Entity.

Annotation -

Conformance to the LEP AP does not require any Annotation Entities in the depiction of the Via.

Structure -

The Connect Point Entity (Type 132) modeling a Via shall not reference an owning entity.

The Connect Point Entity (Type 132) modeling a Via may reference one Singular Subfigure Instance Entity (Type 408) to graphically depict the presence of a Via at the indicated location.

If the Connect Point Entity (Type 132) does not reference a Subfigure Definition in the LEP AP conformant file, the receiving system may display its default icon for Via graphics.

Associativities -

The Connect Point Entity Entity (Type 132) modeling a Via shall reference the proper Flow Associativity Entity (Type 402, Form 18) to connect the Via to the signal join.

Properties -

Mandatory Properties:

The Connect Point Entity (Type 132) modeling a Via shall reference the following LEP properties:

The *LEP SEMANTIC PROPERTY* named stand-alone_via.

Either the *LEP SEMANTIC PROPERTY* named allowable_test_point or the *LEP SEMANTIC PROPERTY* named not_allowable_test_point.

Drilled Hole (Type 406, Form 6) Property.

The PWB Drilled Hole Property shall specify the Drill diameter size, the finish diameter size, and the Function Code (5) of a plated Via *HOLE*. The DE Level field of the Drilled Hole Property shall be zero for a through-hole which passes through the entire LEP. The DE Level field of the PWB Drilled Hole Property shall reference a Definition Levels Property Entity (Type 406, Form 1) for a *HOLE* which does NOT go all the way through the PWB. That Definition Levels Property (Type 406, Form 1) shall specify which levels are penetrated.

Optional Property:

Each Connect Point Entity (Type 132) modeling a Via may reference a Definition Levels Property Entity (Type 406, Form 1).

Table 6-4. DE Settings of Via Entities in a PDQ IGES File

Component Entity - DE Field		Network Subfigure Definition	Network Subfigure Instance	Connect Point
Line Font	(4)	0, 1	0, 1	0, 1
Level	(5)	0 to 1024	0 to 1024	0 to 1024
View	(6)	0	0	0
Transformation Matrix	(7)	0	0 or pointer	0 or pointer
Status; Blanked	(9.1)	00	00	00
Status; Subordinate	(9.2)	01	01	01
Status; Entity Use	(9.3)	02	00	00
Status; Hierarchy	(9.4)	01	01	01
Line Weight	(12)	per IGES	per IGES	per IGES
Color	(13)	0 to 8	0 to 8	0 to 8

Note 1: The preferred method of displaying text found in the 320,420,132, and 408 is by using the text display template. However, a preprocessor may display text using the General Note Entity (Type 212) instead. The only concern with using this method is that it is possible to get the PDQ intelligence and the General Note Entity (Type 212) text out of agreement.

6.5.3. LEP Entity Usage for Physical Mechanical Part Representation

Entities required:

The Singular Subfigure Instance and Subfigure Definition Entities model the Mechanical Part in LEP AP conformant files. The Singular Subfigure Instance Entity (Type 408) shall locate and depict the Mechanical Part. It shall reference one Subfigure Definition (which may also be referenced by other Instances.) The Subfigure Definition Entity (Type 308) shall reference one LEP Closed Curve depicting the X, Y outline of the Mechanical Part. The LEP Closed Curve may reference one or more other LEP Geometry Entities. Various properties shall add the needed semantics.

Geometry -

A Closed Curve having a *LEP SEMANTIC PROPERTY* "physical_outline" Property shall depict each Mechanical Part.

Annotation -

Conformance to the LEP AP does not require any Annotation Entities in the representation of the Mechanical Part.

Structure -

See the discussion under Required Entities above.

Associativities -

Conformance to the LEP AP does not require any Associativities in the representation of the Mechanical Part.

Properties -

Mandatory Properties:

The Subfigure Definition Entity (Type 308) depicting the Mechanical Part shall reference the following property:

The *LEP SEMANTIC PROPERTY* named physical_outline.

The Singular Subfigure Instance Entity (Type 408) locating the Mechanical Part shall reference the following three properties:

Part Number Property Entity (Type 406, Form 9),

LEP SEMANTIC PROPERTY named component_instance_side, and

LEP SEMANTIC PROPERTY named component_instance_rotation.

Optional Properties:

The Subfigure Definition Entity (Type 308) depicting the Mechanical Part may reference the following properties:

The *LEP SEMANTIC PROPERTY* named component_std_name.

The *LEP SEMANTIC PROPERTY* named component_version.

The *LEP SEMANTIC PROPERTY* named component_pkg_type.

The *LEP SEMANTIC PROPERTY* named component_tech_type.

The *LEP SEMANTIC PROPERTY* named component_layout_surface.

The *LEP SEMANTIC PROPERTY* named component_height.

The *LEP SEMANTIC PROPERTY* named component_max_size.

The *LEP SEMANTIC PROPERTY* named component_outline_overhang.

In addition, the Subfigure Definition depicting the Mechanical Part may reference the following LEP properties:

Region Restriction Property Entity (Type 406, Form 2), and

Drilled Hole Property Entity (Type 406, Form 6).

The Singular Subfigure Instance Entity (Type 408) locating the Mechanical Part may reference any of the following three *LEP SEMANTIC PROPERTIES*:

component_preplace,

therm_cond, and

therm_r

6.5.4. LEP Entity Usage for Physical PWB Representation

The PWB from which a PCA is assembled may be an integral part of the assembly file/drawing or may be a separate file/drawing that is called out (in the drawing hierarchy) by the assembly drawing. This Section is applicable to the latter case, and the traces, vias, drilled holes and other features are applicable to the PWB. All components, Traces, Vias, (see Section 5.3.2.) and other entities are subordinate to the LEP PWB.

6.5.4.1. LEP Physical PWB Definition

Entities required:

A Network Subfigure Definition Entity (Type 320) shall depict the Physical PWB. Each Network Subfigure Definition shall reference as many LEP Geometry Entities as needed to depict the PWB. Each Network Subfigure Definition may reference many of the properties listed below. Conformance to the LEP AP does not require any Text Display Template Entities (Type 312) in the representation of the PWB. At least one Connect Point Entity (Type 320) shall provide a means of connecting signals to "pin1" of the PWB.

Geometry -

Any LEP Geometry Entity may appear in conformant files. All geometry shall lie entirely in the Z = 0.0 plane. There shall be one Closed Curve defining the LEP outline. The Closed Curve shall reference the "lep_physical_outline" property. LEP properties shall assign functionality to any additional curves. Any of the LEP Geometry Entities may represent deposited Traces.

Annotation -

The LEP Physical PWB Definition (Type 320) may reference a General Note Entity (Type 212, Form 0).

Structure -

Network Subfigure Definition Entity (Type 320) depicts the LEP itself. Additional Network Sub-

figure Definitions and Instances may represent electrical Components, and Vias. Subfigure Definitions and Singular Subfigure Instances may represent Padstacks and mechanical parts.

Associativities -

Conformance to the LEP AP does not require any Associativities in the representation of the Definition.

Properties -

Mandatory Properties:

The Closed Curve depicting the LEP Outline shall reference the following properties:

The *LEP SEMANTIC PROPERTY* named date.

The *LEP SEMANTIC PROPERTY* named lep_physical_outline.

The *LEP SEMANTIC PROPERTY* named schematic_number.

The *LEP SEMANTIC PROPERTY* named pwb_number.

The *LEP SEMANTIC PROPERTY* named assembly_number.

Each collector of Geometry representing conductive Joins in the form of Traces shall reference the following property.

Line Widening Property Entity (Type 406, Form 5)

Level Function Property Entities (Type 406, Form 1) shall ascribe functionality (such as silk-screen layer) to various IGES levels.

Optional Properties:

The Network Subfigure Definition Entity (Type 320) modeling the LEP Physical PWB as a Component may reference the properties listed in Table 6-11.

6.5.4.2. LEP Physical PWB Instance

Entities required:

The Network Subfigure Instance Entity (Type 420) locates the Physical PWB Instance occurrence and provides a way to connect to its ports. A Connect Point Entity (Type 132) models each port (e.g., Pin or Pad on an edge connector).

Geometry -

A Connect Point Entity (Type 132) having a *LEP SEMANTIC PROPERTY* named pad_tech_type shall represent each Component Pin or each Component Pad on the Physical PWB. That property shall specify whether the Pad or Pin is of type edge_finger_pad, connector_socket, or connector_pin.

One Connect Point shall appear in the Network Subfigure Instance for each Pad, Pin or Socket in the referenced Network Subfigure Definition, even if the Pad, Pin or Socket is not used.

Annotation -

Conformance to the LEP AP does not require any Annotation Entities in the representation of the Physical PWB Instance. Stand-alone annotation may denote information on the LEP such as part_number, silk-screen, reference designator, or Pin number.

Structure -

The Network Subfigure Instance Entity (Type 420) shall reference one and only one Network Subfigure Definition Entity (Type 320). The Instance shall also specify the location of the Physical LEP with respect to the origin of Model Space.

The Network Subfigure Instance shall reference one or more Connect Point Entities (Type 132).

Associativities -

The LEP Physical PWB Definition (Type 320) shall reference one Flow Associativity Entity (Type 402, Form 18) for each Join on the PWB. The Flow Associativity shall reference each Trace, filled area, Via, and Component Pin that is a member of the Join carrying the corresponding signal.

Properties -

Mandatory Properties:

The Network Subfigure Instance Entity (Type 420) representing the LEP Physical PWB Instance shall reference the following properties:

Part Number Property Entity (Type 406, Form 9).

LEP SEMANTIC PROPERTY named top_level_assembly_instance.

Optional Properties:

The Network Subfigure Instance Entity (Type 420) may reference any of the following six *LEP SEMANTIC PROPERTIES*:

abs_voltage_max
tolerance
therm_cond
therm_jc
therm_r
junction_max_t

Table 6-5. DE Settings of LEP Instance Entities in a PDQ IGES File

LEP Entity - DE Field		Network Subfigure Definition	Network Subfigure Instance
Line Font	(4)	0, 1	0, 1
Level	(5)	0 to 1024	0 to 1024
View	(6)	0	0
Transformation Matrix	(7)	0	0, pointer
Status; Blanked	(9.1)	00	00
Status; Subordinate	(9.2)	01	00

Table 6-5. DE Settings of LEP Instance Entities in a PDQ IGES File

LEP Entity - DE Field		Network Subfigure Definition	Network Subfigure Instance
Status; Entity Use	(9.3)	02	00
Status; Hierarchy	(9.4)	01	01
Line Weight	(12)	per IGES	per IGES
Color	(13)	0 to 8	0 to 8

The PDQ LEP representation in the IGES file shall have a Model space scale (Global index 13) of 1.0. A Transformation Matrix (needed for rotation) shall have no translation component.

6.5.4.3. LEP Constructs Supporting Signal Conductivity

Traces Traces use simple geometry objects such as *LINES* and *ARCS*. They interconnect electronic components and carry signals. Physically, they have a specific width and may pass between the different product layers by means of vias and through-pins.

Jumpers A physical interconnect which may be implemented as a component or a trace, but which will have only one signal name.

Planes Planes, such as ground and power planes, outline the appropriate (filled) area.

Vias Vias are holes or voids made between two or more layers on the LEP. They are plated so they can carry the signal. The holes are made by different means such as drilling, milling or cutting. This variability makes it hard to select specific entities which relate to some aspects of vias. However one makes the hole or void, the functions of the via are still the same. Vias may be through, blind, or buried. Vias are often used as test points.

Pad stacks (through and SMT) Pad stacks provide the electrical interface between the LEP substrate and the component. Pad stacks may be either through_hole or surface mount.

6.5.4.4. LEP Constructs Supporting Components

Network Subfigure Definition The Network Subfigure Definition (Type 320) defines an electrical component. It has specific points where connections between it and a signal carrier can be made. These points are component terminals of the LEP. The Connect Point Entity (Type 132) models the component terminal. The component terminals may be pins, pads, wire bonds, or any other means of connecting a component to a signal. In addition to Connect Points (Type 132), the Network Subfigure Definition (Type 320) may contain geometry defining functionality, such as thermal outline, footprint outline, and placement outline.

The Network Subfigure Definition (Type 320) may also contain properties, things like technology type (through-hole or surface mount), board side placement rules, value, tolerance, version number, and any special instructions. A Network Subfigure Instance (Type 420) places the electrical component defined by the Network Subfigure Definition (Type 320). The LEP AP requires each instance to have a corresponding definition within the same file. More than one Network Subfigure Instance (Type 420) in the file may reference the same Network Subfigure Definition (Type 320). Each Network Subfigure Instance (Type 420) is unique, it is placed at a specific location,

and it has a unique reference designator. The instance uses the geometry and the properties defined in the definition. The Network Subfigure Instance (Type 420) has one Connect Point (Type 132) for each pad/pin. Each has a counterpart in the Network Subfigure Definition (Type 320). The counterpart Connect Points (Type 132) must correspond because different signals connect to the same generic definition *PIN* at each instance. For example, both R1 and R2 may reference the Network Subfigure Definition (Type 320) having the name (axial.100). Distinct Connect Points (Type 132) in the instances are needed to connect R1 pin 1 to sig_1 while R2 pin 1 connects to ground.

Connect Point A Connect Point Entity (Type 132) plays two roles depending on whether it is referenced by a Network Subfigure Instance Entity (Type 320) or not. A Connect Point Entity (Type 132) not referenced by a Network Subfigure Instance Entity (Type 320) defines a specific and unique point of connection between two or more signal carriers. A Connect Point Entity (Type 132) referenced by a Network Subfigure Definition Entity (Type 320) provides a template for a Connect Point in Network Subfigure Instance Entities (Type 320) and does not participate in connectivity representations. The Connect Point Entity (Type 132) provides a physical location, a function flag, and an identifier. It references an ordinary Subfigure Definition Entity (Type 308) which contains information such as pad size and shape, drill hole (if applicable), and display geometry. The most common examples of Connect Points are vias, test points, component pins, or SMT dedicated through-pins. The function flag (PD FF) distinguishes between Connect Point Entity (Type 132) occurrences in logical representations and those in physical representations.

6.5.4.5. LEP Constructs Supporting Connectivity

Connectivity is represented by appropriate use of the Network Subfigure Definition (Type 320); the Network Subfigure Instance (Type 420); the Connect Point (Type 132); and the Flow Associativity (Type 402, Form 18). The Flow Associativity (Type 402, Form 18) may include connecting geometry.

The Flow Associativity provides the signal name and pointers to all features associated with that signal. Such features include pins, pads, stand-alone vias, traces and planes. There is one Flow Associativity (Type 402, Form 18) for each signal in the file. Note that it is possible to include both the logical and physical product data in one file using the Flow "Type" flag or entity level assignments to discriminate between the two product representations. Though permitted by IGES, having both logical connectivity and physical connectivity in the same file is not conformant to the LEP AP.

"Backpointers" pointing from each of the features back to the Flow Associativity are not required for LEP AP conformance.

6.5.4.6. LEP Constructs Supporting a Physical Representation

A physical LEP is represented by simple closed planar *GEOMETRY* depicting its outer perimeter. An LEP may have more than one defined outline. There may be keepin/keepout outlines for component placement, traces, and vias, or tool path outlines, and other objects.

All entities on the LEP are subordinate to it. This includes: all components such as Network Subfigure Instances (Type 420) and as Network Subfigure Definitions (Type 420), traces (*JOIN*), flow associativities (*NET*), padstacks (*PADSTACK DEFINITION* and *PADSTACK INSTANCE*), vias (*VIA*), tooling holes, fiducials, through holes, mounting holes, mounting hardware, and other objects. This means there must be a top level Network Subfigure Instance (Type 420) of the LEP

which calls the top level Network Subfigure Definition (Type 320) of the LEP. If the top level instance is not present, nothing will be displayed.

What then is not subordinate to the LEP? Dimensions are not. Any assembly information, views/draws, scales, grids, and certain kinds of details and cross-sections are not subordinate to the LEP representation.

Several applications require a PDQ LEP representation. This refers to one in which Product Data may be Queried. Computer controlled equipment can then extract needed data without manual intervention. For example, component insertion equipment can put a component at the X, Y location specified by the component's Network Subfigure Instance. Operations which modify the physical representation of the LEP, such as scaling or rotation, will have adverse affects on some application systems. Constructs which are beyond the scope of an automated operation, such as dimensions and view/draw could stall or disable some processing systems.

6.5.4.7. LEP Physical Flow Associativity (Signal/netlist)

Each Flow Associativity Entity (Type 402, Form 18) models a distinct Signal in a Netlist. Each Signal is insulated from all others. The Netlist is a report of all the signals in a particular (sub)assembly of a product. Each LEP Connect Point Entity (Type 132) represents a port of a symbol in a schematic or a Pin/Pad of a physical Component. Thus, at least two types of Netlist exist for each LEP. IGES permits both logical and physical Flow Associativities to exist in the same file. The presence of both a logical and a physical netlist in the same file shall not be conformant to this AP. This discussion is limited to the physical realm.

Entities required:

Flow Associativity Entity (Type 402, Form 18), Connect Point Entity (Type 132), and LEP Join Geometry (below).

Geometry -

In the Physical realm, each Connect Point Entity (Type 132) represents a Component Pin or a Component Pad or a stand-alone Via. The Join geometry, representing the electrical connections between and among Connect Points, may be selected from the following LEP Geometry Objects: Line Entity (Type 110), Circular Arc Entity (Type 100), Composite Curve Entity (Type 102), and Simple Closed Planar Curve Entity (Type 106, Form 63). Each Closed Curve shall reference one *LEP SEMANTIC PROPERTY* which defines its function. Each Line, Arc, or open Composite Curve shall reference two LEP properties: Line Widening Property Entity (Type 406, Form 5) and one *LEP SEMANTIC PROPERTY* named conductive_trace_function. All geometry shall lie entirely in the Z = 0.0 plane.

Annotation -

Conformance to the LEP AP does not require any Annotation Entities in the representation of the Physical Signal.

The Flow Associativity Entity (Type 402, Form 18) collects the Connect Point Entities (Type 132) for each Signal. Network Subfigure Instance Entities (Type 420) shall reference each Connect Point Entity (Type 132) corresponding to a Pin or Pad of the Component represented in the Physical Netlist. LEP Geometry Entities or Flash Entities may depict the location of each Connect Point in the Netlist.

Associativities -

The Flow Associativity Entity (Type 402, Form 18) models each Signal in the Netlist.

Properties -

Mandatory Properties:

Each Closed Curve Join shall reference one *LEP SEMANTIC PROPERTY* which defines its function. The applicable LEP functions for Join geometry include: planar_shape_function {signal, power, ground, ..}. Each Line, Arc, or open Composite Curve Join shall reference two LEP properties: Line Widening Property Entity (Type 406, Form 5) and one *LEP SEMANTIC PROPERTY* named conductive_trace_function {signal, power, ground, ..}.

Optional Properties:

The *LEP SEMANTIC PROPERTY* named pad_tech_type attached to the Connect Point may state whether the Pin is of SMT or through-hole design. The "pin1" *LEP SEMANTIC PROPERTY* shall appear on those Connect Points bearing that designation.

6.5.4.8. LEP Physical LEP Signal Carriers (Traces, Shapes) (1xx, 230)

The Physical LEP Signal Carriers constitute the Join defined in IGES 3.6.3.2. Taken together, the Joins provide the connectivity that the Logical Netlist requires. Lines having specified width depict conductive Traces of similar width and a specified nominal thickness. Some Joins are irregularly shaped areas of conductive material of a specified nominal thickness.

Entities required:

The Join Geometry, depicting the electrical connections between and among Connect Points, may be selected from: Line Entity (Type 110), Arc Entity (Type 100), full Circular Arc Entity (Type 100), Composite Curve Entity (Type 102), Flash Entity (Type 125), and Simple Closed Planar Curve Entity (Type 106, Form 63).

Geometry -

LEP Join Geometry is defined above under Entities required. Any LEP Join Geometry Entities may appear in conformant files. All geometry shall lie entirely in the Z = 0.0 plane. Flash Entity (Type 125) shall not reference a Transformation Matrix.

Annotation -

The Closed Curve(s) enclosing the conductive areas of LEP Joins shall (each) reference a Sectioned Area Entity (Type 230). That Sectioned Area shall reference a *LEP SEMANTIC PROPERTY* named planar_shape_function {signal, power, ground, ...}. No other Annotation Entities shall appear as part of a Physical Signal/netlist representation.

Structure -

LEP Physical PWB Signal Carriers shall reference no structure entities other than the Flow Associativity.

Associativities -

LEP Physical PWB Signal Carriers shall reference one Flow Associativity Entity (Type 402, Form 18) for each Join.

Properties -

Mandatory Properties:

Each Closed Curve Join shall reference one*LEP SEMANTIC PROPERTY*) named planar_shape_function which defines its function. Applicable LEP functions for Join geometry include: signal, power, and ground.

Each Line, Arc, or open Composite Curve Join shall reference two LEP properties:

Line Widening (type 406 form 5) and

LEP *LEP SEMANTIC PROPERTY* named conductive_trace_function.

The *LEP SEMANTIC PROPERTY* named pad_tech_type attached to the Connect Point shall state whether the Pin is of SMT or thru-hole design. If it is a thru_hole design, a further choice among three design intentions shall be made. The three valid alternatives are: thru_hole_anti, thru_hole_regular, and thru_hole_thermal.

The "pin1" *LEP SEMANTIC PROPERTY* shall appear on those Connect Points bearing that designation.

6.5.4.9. LEP Physical PWB Features: Fiducials

A Fiducial provides a reference location for accurate positioning of other LEP features.

Entities required:

Any LEP Geometry *CLOSED CURVE* may depict the *FIDUCIAL*.

Geometry -

Any LEP Geometry Entity may appear in conformant files. All geometry shall lie entirely in the Z = 0.0 plane. The Flash Entity (Type 125) shall not reference a Transformation Matrix.

Annotation -

Conformance to the LEP AP does not require any Annotation Entities in the representation of the *FIDUCIAL*.

Structure -

Conformance to the LEP AP does not require any Structure Entities in the representation of the *FIDUCIAL*.

Associativities -

Conformance to the LEP AP does not require any Associativity Entities in the representation of the *FIDUCIAL*.

Properties -

Mandatory Properties:

The following properties shall be referenced by each Closed Curve representing a *FIDUCIAL*:

The *LEP SEMANTIC PROPERTY* named fiducial having a first value from among (component/cluster/board/panel) and a second value from among (top/bottom)

6.6. LEP Entity Usage for Specific Applications

This section discusses tests of conformant files and their construction. For example, Netlist reports require Flow Associativities. This Section provides the additional detail about IGES entities and structures for given use applications. This information supplements and enhances the object descriptions in Section 5.3.

6.6.1. LEP Entity Usage for Design/Engineering Applications

6.6.1.1. LEP Entity Usage for Netlist Reports

The netlist is a report of the interconnection of all Component terminals (Pins or Pads) sharing the same signal. Notice this includes both the logical and physical forms. The logical form relates each signal in the database with all of the unique part_pins which are part of that signal. The physical form adds the X, Y location of each part_pin, and the connection geometry (Traces). This is the only instance in the LEP AP where the presence of both the logical flow and the physical flow Type flags of the Flow Associativity (Type 402, Form 18) in the same IGES file are appropriate and conformant. Such files also require use of counts of associated Flow Associativities and NF SPTR pointers.

This application does require a PDQ representation.

Entities required:

The Flow Associativity models each Signal. The LEP Geometry Entities depict the shape of the Join entities in the Physical Netlist. The LEP Geometry Entities may each reference one Line Widening Property Entity (Type 406, Form 5) to convey Trace width.

<u>Geometry</u> -

Logical:

> The logical form of the netlist shall use no geometry. Any geometry present shall be ignored.

Physical:

> Any LEP Geometry Entity may appear in conformant files. All geometry shall lie entirely in the Z = 0.0 plane. The Flash Entity (Type 125) shall not reference a Transformation Matrix. The geometry entities usually depict the Traces. Via depictions may employ geometry. A Connect Point Entity (Type 132) shall model each Via.

Both:

> Component through connections shall appear as Connect Points in Physical netlists. However, the physical characteristics of the Connect Point shall not appear in the logical form of the netlist.

<u>Annotation</u> -

The Flow Associativity Entity (Type 402, Form 18) parameter NAME1 shall provide the Signal name. The Connect Point Entity (Type 132) parameter CID shall provide the Pin name. The Network Subfigure Instance Entity (Type 420) parameter PRD shall provide the Reference Designator of the Component having the participating *PIN*.

Structure -

The Flow Associativity Entity (Type 402, Form 18) groups the Connect Point Entities (Type 132) for each Signal. Network Subfigure Instance Entities (Type 420) shall reference each Connect Point Entity (Type 132) corresponding to a *PIN* or Pad of the Component represented in the Physical Netlist and may do the same in the Logical Netlist. LEP Geometry Entities or Flash Entities may depict the location of each Connect Point in either Netlist.

Associativities -

The Flow Associativity Entity (Type 402, Form 18) shall model each Signal in either form of Netlist. IGES requires the TF (type of flow flag) to be set to 1 or 2 for each occurrence of the Flow Associativity when both logical (1) and physical (2) flows are present in one file. Corresponding signals reference each other using the Flow Associativity parameters NF and SPTR1 through SPTRNF in such cases.

Properties -

The *LEP SEMANTIC PROPERTY* named pad_tech_type attached to the Connect Point may state whether the Pin is of SMT or through-hole design. The "pin1" *LEP SEMANTIC PROPERTY* may appear on some of the appropriate Connect Points.

6.6.1.2. LEP Entity Usage for Component Placement Reports

Component placement refers to the assignment of Components to specific X, Y locations on the LEP. At times, particular Components must occupy well cooled spots or must adjoin other Components. The LEP component_preplace Property may identify such concerns. Component placement is also known as Component layout or LEP layout.

This Component placement application does require a PDQ representation.

Entities required:

Network Subfigure Definition Entity (Type 320) and Instance Entity (Type 420) represent the electrical parts. Subfigure Definition Entity (Type 308) and Singular Subfigure Instance Entity (Type 408) represent the mechanical parts.

The Network (or Singular) Subfigure Instance Entity (Type 420 or Type 408) locates the LEP Component. The LEP *CLOSED CURVE* depicts the shape of the LEP Component Outline. The LEP Closed Curve shall reference one *LEP SEMANTIC PROPERTY* named physical_outline.

Geometry -

A LEP Closed Curve having a physical_outline Property shall model each LEP Component Outline. Any LEP Geometry may be part of the LEP Closed Curve subject to the usual IGES constraints.

Associativities -

Conformance to the LEP AP does not require any associativities in the representation of the LEP Component Outline.

Annotation -

Conformance to the LEP AP does not require any Annotation Entities in the representation of the LEP Component.

Structure -

The LEP Closed Curve may reference one or more *LEP SEMANTIC PROPERTIES* named component_preplace, or component_instance_side in LEP AP conformant files.

Properties -

Mandatory Properties:

The Network (or Singular) Subfigure Instance Entity (Types 420 or 408) shall reference the following LEP properties:

The *LEP SEMANTIC PROPERTY* named component_std_name.

The *LEP SEMANTIC PROPERTY* named component_version.

The *LEP SEMANTIC PROPERTY* named physical_outline.

The *LEP SEMANTIC PROPERTY* named component_tech_type.

Optional Properties:

The Network (or Singular) Subfigure Instance Entities (Types 420 or 408) may reference the following LEP property:

The *LEP SEMANTIC PROPERTY* named component_preplace, and component_instance_side

6.6.1.3. LEP Entity Usage for Bill of Materials Reports

The Bill of Materials (BOM) report lists the Components which make up the LEP. The BOM lists part numbers, quantity used, and a brief description of the parts. Some BOMs also list reference designators. The report may include other semantic information as well. This BOM application does require a PDQ representation.

Entities required:

Network Subfigure Definition Entity (Type 320) and Instance Entity (Type 420) represent the electrical Components. Subfigure Definition Entity (Type 308) and Singular Subfigure Instance Entity (Type 408) represent the mechanical Components.

The Part Number Property Entity (Type 406, Form 9) attached to the Singular (or the Network) Subfigure Instance Entities (Type 408 or 420) provides the part number to the BOM report. The quantity used is calculated by the software. If needed, the PRD parameter of the Network Subfigure Instance Entity (Type 420) shall provide the Reference Designator data.

Geometry -

Any LEP Geometry Entities may be referenced by the Subfigure Definitions in the depiction of LEP Components.

Annotation -

Conformance to the LEP AP does not require any Annotation Entities in the representation of the LEP Component.

Structure -

The Network Subfigure Instance Entity (Type 420) shall reference one Network Subfigure Defini-

tion Entity (Type 320) and one or more BOM Properties for each electrical Component. The Singular Subfigure Instance Entity (Type 408) shall reference one Subfigure Definition Entity (Type 308) and one or more BOM Properties for each mechanical Component.

Associativities -

Conformance to the LEP AP does not require any associativities in the representation of the BOM.

Properties -

Mandatory Properties:

The Part Number Property Entity (Type 406, Form 9) shall provide part numbers for each Component. This property identifies 4 different part number families; the generic, military, vendor, and internal part numbers. The property shall provide at least one of these four part numbers for each Component.

Each Component instance represented by a Network Subfigure Instance Entity (Type 420) or a Singular Subfigure Instance Entity (Type 408) shall reference the following LEP properties:

The Part Number Property Entity (Type 406, Form 9).

Optional Properties:

Each Component instance represented by a Network Subfigure Instance Entity (Type 420) or a Singular Subfigure Instance Entity (Type 408) may reference the following LEP properties:

The *LEP SEMANTIC PROPERTY* named:

component_std_name
component_version
component_pkg_type
component_tech_type
component_layout_surface
component_insertion_origin
component_height
component_max_size
component_outline_overhang

6.6.2. LEP Entity Usage for Manufacturing Applications

Usage rules are laid down for providing usable information to support Photo-lithography tools, LEP Testers, Numerically Controlled Component processing and Panelization of PWB details. Concepts for support of Wire Wrap is included and Labelmaker application may be added later.

6.6.2.1. LEP Entity Usage for Photo-lithography Tools

Photo-lithography in this application refers to the creation of the master artwork used to manufacture the physical bare PWB or other LEP. This includes the Traces, planes, and Vias. It also includes annotation such as reference designators, Pin numbers, Component outlines, part numbers, tooling holes, Fiducials, and labels.

This photolithography application does NOT require a PDQ representation.

Entities required:

Any LEP Geometry Entity may depict part of the Traces, Vias, and annotation graphics needed for this application. The General Note Entity (Type 212, Form 0) may provide the text strings needed for other annotation.

Geometry -

Any LEP Geometry Entity may appear in conformant files. All geometry shall lie entirely in the Z = 0.0 plane. The Flash Entity (Type 125) shall not reference a Transformation Matrix. Receiving systems may balk at geometry more complex than a Line Entity (Type 110), Circular Arc Entity (Type 100), Copious Data Entity (Type 106), Composite Curve Entity (Type 102), or Flash Entity (Type 125).

Single entities shall depict each Fiducial and each tooling mark. The (full) Circular Arc Entity (Type 100) or the Flash Entity (Type 125) may serve some of these needs. The others may require a Composite Curve Entity (Type 102) or the Copious Data Entity (Type 106, Form 63). The Composite Curve shall gather the constituent entities when multi-entity representations are needed. That Composite Curve shall reference any properties providing semantics such as the *LEP SEMANTIC PROPERTY* named "fiducial".

Annotation -

Closed Curves may annotate the LEP by stylized depictions of Component outlines. The General Note Entity (Type 212, Form 0) may present Reference Designator and Pin text plus LEP identifiers.

Structure -

The simpler the structure, the better. Copious Data Entity (Type 106) may depict objects representable by polygons and Composite Curve Entity (Type 102) Entities may group Lines and Circular Arcs.

Associativities -

Conformance to the LEP AP does not require any associativities in the representation of the photo-tools.

Properties -

The Composite Curve Entity (Type 102), Circular Arc Entity (Type 100), Copious Data Entity (Type 106), Flash Entity (Type 125) or the Line Entity (Type 110) shall reference the following LEP properties:

> The Line Widening (Type 406, From 5) Property shall specify nominal Trace width. Thus, it shall not widen or narrow as the underlying geometry gets scaled.
>
> The *LEP SEMANTIC PROPERTY* named "fiducial" shall tag referenced objects.

6.6.2.2. LEP Entity Usage for LEP Test Aids

This application involves provision of information to support automated testing of an LEP. The tester needs probe locations tied to signal names. The tester also needs locations of Components which might interfere with probe insertion or removal. The object descriptions needed include those of the Traces, planes, padstacks and Vias. It also includes annotation such as Reference Des-

ignators, Pin numbers, Component outlines, Part Numbers, tooling holes, Fiducials, and labels.

This application does require a PDQ representation.

Entities required:

The Flow Associativity Entity (Type 402, Form 18), the Connect Point Entity (Type 132), and the Network Subfigure Instance Entity (Type 420) provides the "PDQ intelligence" (connectivity information) needed for this application. Network Subfigure Definition Entity (Type 320) and Instance Entities (Type 420) represent the electrical Components. If needed, the PRD parameter of the Network Subfigure InstanceEntity (Type 420) shall provide the Reference Designator data.

Any LEP Geometry Entity may depict part of the Traces, Vias, and annotation graphics needed for this application. The General Note Entity (Type 212, Form 0) may provide the text strings needed for other annotation.

Geometry -

A LEP Closed Curve having a physical_outline Property shall model each LEP Component Outline. Any LEP Geometry may be part of the LEP Closed Curve. All geometry shall lie entirely in the Z = 0.0 plane.

Single entities shall depict each Via, each connect point, each Fiducial and each tooling mark. The (full) Circular Arc Entity (Type 100) or the Flash Entity (Type 125) may serve some of these needs. The others may require a Composite Curve Entity (Type 102) or the Copious Data Entity (Type 106, Form 63). The Composite Curve shall group the constituent entities when multi-entity icons are needed. That Composite Curve shall reference any properties providing semantics such as the *LEP SEMANTIC PROPERTY* named "fiducial".

The Sectioned Area Entity (Type 230) shall depict filled areas such as irregularly shaped signal planes, ground planes or power planes. Only the (full) Circular Arc Entity (Type 100), Composite Curve Entity (Type 102), and Simple Closed Planar Curve Entity (Type 106, Form 63) shall be filled by the Sectioned Area Entity. Only the unfilled Fill Pattern (0) and the solid filled Fill Pattern (19) of the Sectioned Area Entity (Type 230) shall appear in conformant files.

Annotation -

Network Subfigure Instance Entities (Type 420) will provide information on Component location, Pin location, Signal Name, Reference Designator, Part number, and Pin text.

The Flow Associativity Entity (Type 402, Form 18) parameter NAME1 shall provide the Signal name. The Connect Point Entity (Type 132) parameter CID shall provide the Pin Name. The Network Subfigure Instance Entity (Type 420) parameter PRD shall provide the Reference Designator of the Component having the participating Pin.

Structure -

The Network Subfigure Instance Entity (Type 420) shall reference one Network Subfigure Definition Entity (Type 320) and one or more Test Properties for each electrical Component. The Network Subfigure Instance shall also reference one Connect Point Entity (Type 132) for each corresponding Connect Point in the referenced Network Subfigure Definition Entity (Type 320). Each Connect Point Entity (Type 132) shall provide its Pin Name in parameter PRD. Some instanced Connect Points Entities (Type 132) may model Vias rather than Component Pins. (Vias have no owner. In other words, the PSFI parameter of the Tyoe 132 has no valid pointer.) Each

owned and instanced Connect Point Entity (Type 132) shall provide a Pin name. The combination of the Reference Designator and the Pin name shall be unique within the IGES file in which it appears. Each instanced Connect Point Entity (Type 132) may reference a *LEP SEMANTIC PROPERTY* stating whether test probing is allowable or not. Each instanced Connect Point which does not reference either of those properties should not be probed. Each Flow Associativity Entity (Type 402, Form 18) shall provide a Signal Name (in NAME1). Each Flow Associativity shall reference all the Connect Points carrying the Signal Name found in NAME1.

Associativities -

The Flow Associativity Entity (Type 402, Form 18) models each Signal.

Properties -

Mandatory Properties:

Each Connect Point Entity (Type 132) shall reference one of the following two LEP properties:

The *LEP SEMANTIC PROPERTY* name:

allowable_test_point

or (the default) not_allowable_test_point

The Network Subfigure Instance shall reference the following LEP properties:

The *LEP SEMANTIC PROPERTY* named: component_instance_side.

Optional Properties:

The Connect Point Entity (Type 132) may reference the following LEP properties:

The *LEP SEMANTIC PROPERTY* name: pin1.

The Closed Curve may reference the following LEP properties:

The *LEP SEMANTIC PROPERTY* name: fiducial.

6.6.2.3. LEP Entity Usage for Numerically Controlled (NC) Production Files

Component Insertion, Component Pick and Place, and Component Verification and a concept for Wire Wrap are discussed below. The LEP AP files described below may be converted to MIL-D-28000 Class IV files as needed.

6.6.2.3.1. LEP Entity Usage for Component Insertion

This Component Insertion application does require a PDQ representation.

Component Insertion refers to the process of inserting through-hole Components of one or more distinct types into the PWB. All Components within each type are the same as far as the inserter is concerned. For example, one type may be a 10K 1/4w 5% resistor and another type may be a 22K 1/4w 10% resistor. Each type must be distinctly identified. The most common method to do this is with a Part Number Property Entity (Type 406, Form 9), using the internal Part Number Property Entity (Type 406, Form 9) value.

Automated insertion is concerned with two distinct rotations. The first rotation occurs during physical layout when placing the part from the LEP design library. For example, an axial Component is usually mounted with its axis parallel to the plane of the LEP. With the axis perpendicular

to the plane of the LEP, the Component has been rotated 90 degrees (or 270 degrees.) (The amount of rotation (90 or 270) matters little if the Component is non-polarized. For a diode or a polarized capacitor, though, it is important.) These rotations are communicated to the insertion machinery by means of the *LEP SEMANTIC PROPERTY* named component_placement_angle. The Transformation Matrix Entity (Type 124) is not applicable in such cases because there is no fixed frame of reference.

The other kind of rotation is beyond the scope of this AP. It tracks the motions of the head of the inserter and of the Component in the feeder.

Entities required:

The Network Subfigure Instance Entity (Type 420) models the Component inserted. It shall reference a Network Subfigure Definition containing a *CLOSED CURVE* which depicts the shape of the Component outline after insertion. It may also reference a Transformation Matrix Entity (Type 124) to model the rotation of the Component into position in model space. The Transformation Matrix shall not apply translations of the Component in the plane of the LEP. The X and Y coordinates within the Network Subfigure Instance Entity (Type 408) shall determine the lateral position of the Component on the LEP. The Network Subfigure Instance Entity (Type 420) shall reference a*LEP SEMANTIC PROPERTY*) named component_instance_side to state whether the Component is mounted on the top or the bottom of the LEP. Else, the default position is on top of the PWB. The Subfigure Instance may also reference a *LEP SEMANTIC PROPERTY* named component_instance_rotation if needed. The Network Subfigure Instance Entity (Type 420) shall reference other properties to provide semantic information to the inserter.

Geometry -

One *CLOSED CURVE* shall define the outline of the Physical Component in the X, Y plane of the LEP. Only the (full) Circular Arc Entity (Type 100), the Composite Curve Entity (Type 102), and the Simple Closed Planar Curve Entity (Type 106, Form 63) may appear as entities which define Component outlines. Each such *CLOSED CURVE* shall reference one LEP Property which defines its function. For this application, the LEP function needed is the physical_outline.

Any LEP Geometry Entity may appear as part of the Component Geometry. All LEP Geometry shall lie entirely in the Z = 0.0 plane.

Annotation -

Conformance to the LEP AP does not require any Annotation Entities in the representation of the Physical Component.

Structure -

The Network Subfigure Instance Type (Type 420) shall reference one Network Subfigure Definition Entity (Type 320) and one or more Component Insertion Properties for each electrical Component to be inserted. The Network Subfigure Definition (together with its Transformation Matrix) control the orientation and position of insertion.

Associativities -

Conformance to the LEP AP does not require any associativities in the representation of the Physical Component to be inserted.

Properties -

Mandatory Properties:

The Network Subfigure Definition Entity (Type 320) shall reference the following LEP properties:

The *LEP SEMANTIC PROPERTY* named:

component_insertable
component_insertion_origin

The Network Subfigure Instance Entity (Type 420) shall reference the following LEP properties:

The *LEP SEMANTIC PROPERTY* named:

component_insertion_force
component_insertion_height

The Connect Point Entity (Type 132) shall reference one of the following two LEP properties:

The *LEP SEMANTIC PROPERTY* name:

allowable_test_point
or (the default) not_allowable_test_point

Optional Properties:

The Network Subfigure Definition Entity (type 320) may reference the following LEP properties:

The *LEP SEMANTIC PROPERTY* named:

component_max_size

The Network Subfigure Instance Entity (Type 420) may reference the following LEP properties:

The *LEP SEMANTIC PROPERTY* named:

component_instance_side
component_instance_rotation
component_outline_overhang
component_placement_lead_diameter
component_placement_yn
component_placement_zspan
component_placement_body
component_placement_center
component_placement_angle
component_placement_form_code
component_placement_form_code_description
component_insertion_force
component_insertion_height

6.6.2.3.2. LEP Entity Usage for Component Pick and Place

This application does require a PDQ representation.

Component Pick and Place refers to the process of pasting SMT Components of one or more distinct types onto the LEP. All Components within each type are the same as far as the Pick and Place system is concerned. For example, one Component type may be a memory chip and another

Component type may be a multiplexer chip. Each Component type must be distinctly identified. The most common method to do this is with a Part Number Property Entity (Type 406, Form 9), using the internal Part Number Property value.

Automated Pick and Place is concerned with two distinct rotations. The first rotation occurs during physical layout when placing the part from the LEP design library. For example, an MCM chip Component is usually mounted with Pin 1 in the upper left hand corner. With Pin 1 in the upper right hand corner, the Component has been rotated 270 degrees. These rotations are communicated to the placement machinery by means of the *LEP SEMANTIC PROPERTY* named component_placement_angle. The Transformation Matrix Entity (Type 124) is not applicable in such cases because there is no fixed frame of reference.

The other kind of rotation is beyond the scope of this AP. It tracks the motions of the head of the Pick and Place system and of the Component in the Pick mechanism.

Entities required:

The Network Subfigure Instance Entity (Type 420) models the Component placed. It shall reference a Network Subfigure Definition containing a *CLOSED CURVE* which depicts the shape of the Component outline after placement. It may also reference a Transformation Matrix Entity (Type 124) to model the rotation of the Component into position in model space. The Transformation Matrix shall not apply translations of the Component in the plane of the LEP. The X and Y coordinates within the Network Subfigure Instance Entity (Type 408) shall determine the lateral position of the Component on the LEP. The Subfigure Instance shall reference a *LEP SEMANTIC PROPERTY* named component_instance_side to state whether the Component is mounted on the top or the bottom of the LEP. If it does not, the default position is on top of the LEP. The Subfigure Instance may also reference a *LEP SEMANTIC PROPERTY* named component_instance_rotation if needed. The Subfigure Instance shall reference other properties to provide semantic information to the Pick and Place system.

Geometry -

One Closed Curve shall depict the outline of the Physical Component in the X, Y plane of the LEP. Only the (full) Circular Arc Entity (Type 100), the Composite Curve Entity (Type 102), and the Simple Closed Planar Curve Entity (Type 106, Form 63) may appear as entities which define Component outlines. Each Closed Curve shall have one LEP Property attached to it which defines its function. For this application, the LEP function needed is the physical_outline.

Any LEP Geometry Entity may appear as part of the Component geometry. All LEP Geometry shall lie entirely in the Z = 0.0 plane.

Annotation -

Conformance to the LEP AP does not require any Annotation Entities in the representation of the Physical Component.

Structure -

The Network Subfigure Instance Entity (Type 420) shall reference one Network Subfigure Definition Entity (Type 320) and one or more Component Insertion Properties for each electrical Component to be placed. The Network Subfigure Definition (together with its Transformation Matrix) control the orientation and position of adhesion.

Associativities -

Conformance to the LEP AP does not require any associativities in the representation of the Physical Component to be placed.

Properties -

Mandatory Properties:

The Network Subfigure Definition Entity (Type 320) shall reference the following LEP properties:

The *LEP SEMANTIC PROPERTY* named:

component_layout_surface (top/bottom)
component_insertable (yes/no)

The Network Subfigure Instance Entity (Type 420) shall reference the following LEP properties:

Part Number Property Entity (Type 406, Form 9).

The *LEP SEMANTIC PROPERTY* named:

component_placement_force
component_placement_height
component_placement_angle
component_placement_depth_stop
component_placement_yn

Optional Properties:

The Network Subfigure Definition (type 320) may reference the following LEP properties:

The*LEP SEMANTIC PROPERTY*) named:

component_std_name
component_version
component_pkg_type
component_tech_type
component_layout_surface
component_insertion_origin
component_insertable
component_height
component_max_size
component_outline_overhang

The Network Subfigure Instance Entity (Type 420) may reference the following LEP properties:

LEP SEMANTIC PROPERTY named:

component_placement_body
component_placement_pickpoint_id
component_placement_preplace
component_placement_bonding
component_placement_bonding_material_specifications
component_placement_bonding_process_specifications
component_placement_center
component_placement_feeder

component_placement_force
component_placement_form
component_placement_form code
component_placement_form code_description

6.6.2.3.3. LEP Entity Usage for Component Verification

This application verifies the correct Components, properly oriented, are correctly positioned on the LEP. It needs Component outline (a Closed Curve), Part number, X, Y position, and "pin1" *LEP SEMANTIC PROPERTY* for each Component occurrence. This application does NOT require a PDQ representation.

Entities required:

The Network Subfigure Instance Entity (Type 420) models the LEP Component. The *CLOSED CURVE* depicts the shape of the LEP Component. The Network Subfigure Instance Entity (Type 420) shall reference one or more LEP Component Properties in LEP AP conformant files.

Geometry -

A Closed Curve having a placement_outline Property shall model each LEP Component in LEP AP conformant files.

Annotation -

Conformance to the LEP AP does not require any Annotation Entities in the representation of the LEP Component representation.

Structure -

The Network Subfigure Instance Entity (Type 420) shall reference one Network Subfigure Definition Entity (Type 320). The Network Subfigure Definition Entity (Type 320) shall reference a *CLOSED CURVE*. The Definition shall reference one or more LEP Component Properties in LEP AP conformant files.

Associativities -

Conformance to the LEP AP does not require any associativities in the representation of the LEP Component.

Properties -

Mandatory Properties:

The Network Subfigure Instance Entity (Type 420) shall reference the following LEP properties:

Part Number Property Entity (Type 406, Form 9).

The *LEP SEMANTIC PROPERTY* named placement_outline.

The proper Connect Point Entity (Type 132) of each Component shall reference the *LEP SEMANTIC PROPERTY* named pin1.

6.6.2.3.4. LEP Entity Usage for Wire Wrap

A simplified model of a 2 layer board will suffice for most wire wrap situations; for more complex designs you will see a multilayer PWB (with or without some mounted components) with

compliant pins friction-fit into the through hole barrels of the board. Wire wrap will usually extend off the board typically to connectors with wire wrap pins (in most cases, it is illegal or inadvisable to solder the solid wire used for wire wrap). Occasionaly, the motherboard will also include some flex-tape arrangement also (this is infrequent, but a possibility).

For the wire wrap board, the PWB, Section 6.5.4. is applicable; however the traces and associated Line Widening Property Entity (Type 406, Form 5) are omitted. The completed wire wrap representation will contain components whose pins are located only where a wire wrap pin exists. There are various CAE techniques employed to guide the layout of components on the PWB such as grids and wire wrap board graphics.

For the components, Section 6.5.2. is applicable; a padstack may be used, however the usual PWB lands may be omitted.

The netlist is merged such that the component pins are associated with a signal, thus Section 6.6.1.1. is applicable. Routing is routinely done for many wire wrap situations (especially for 3-D harnes situations, where the final assembly has connections in multiple planes and a "twist" has to be thrown in), but seldom is any of the routing described in any CAD data. The wire wrap product definition may be released or archived at this point.

The wire wrap product is completed by extraction (from the CAE system or from an IGES LEP file) of the netlist with the X, Y locations recorded for each "terminal" in the netlist. The extracted list is "post-processed"; sorted into a wire wrap-sequence order as determined to be optimum for the wire wrap machine used, and the wire lengths (usually precut to the machine requirements) selected. The completed wire wrapped board is continuity-tested (see Section 6.6.2.2.), and the components are inserted (for automated insertion Section 6.6.2.3.1. applies) in the identified locations.

Note that there is little control on the "path" taken by the wires; thus the wire wrap method of assembly is usually limited to low production runs, such as prototypes and special purpose test equipment. Some computer backplanes have utilized wire wrap construction because of the relatively low distributed capacitence between wires and the possibility to make field modifications to the connections.

6.6.2.3.5. LEP Entity Usage for Labelmaker

UNDER CONSTRUCTION.

6.6.2.4. LEP Entity Usage for Panelization

Panelization involves laying out two or more LEP assembly outlines on the surface of a substrate panel for processing efficiency considerations. Fiducials and test coupons are usually added to the layout design to enable better Quality Control. The information needed is the LEP outline (Singular Subfigure Definition Entity (Type 308) referencing a *CLOSED CURVE* having a *LEP SEMANTIC PROPERTY* named pwb_physical_outline Property, and those *CLOSED CURVES* which reference fiducial/board and fiducial/panel Properties.

This application does NOT require a PDQ representation.

Entities required:

One Closed Curve shall define the outline of the PWB. Two or three other *CLOSED CURVES*

depicting Fiducials are also needed. A Group Associativity Entity (Type 402, Form 7) without backpointers shall reference each entity that is to appear in a given panel.

Geometry -

Fiducials may be represented by single entities, such as the Flash Entity (Type 125), or multiple entities such as circles depicted by many short lines. Multi-entity Fiducials shall be subordinate to a Composite Curve Entity (Type 102). The single entity or Composite Curve Entity (Type 102) shall reference a Fiducial Property.

Annotation -

Conformance to the LEP AP does not require any Annotation Entities in the representation of the *CLOSED CURVE* depicting the PWB.

Structure -

One *CLOSED CURVE* shall define the outline of the PWB. Only the (full) Circular Arc Entity (Type 100), the Composite Curve Entity (Type 102), and the Simple Closed Planar Curve Entity (Type 106, Form 63) may appear as entities which define the PWB outline. That *CLOSED CURVE* shall reference the *LEP SEMANTIC PROPERTY* named top_level_assembly_instance. The other *CLOSED CURVES* needed shall reference the *LEP SEMANTIC PROPERTY* named "fiducial" and having the value "board" or the one having the value "panel" as described below.

Associativities -

A Group Associativity, without backpointers (Type 402, Form 7) shall reference each entity that is to appear in a given panel.

Notice that this associativity does not require backpointers from each entity back to the Group Associativity.

Properties -

Mandatory Properties:

The *CLOSED CURVE*(s) depicting fiducials outside the PWB outlines shall reference

The *LEP SEMANTIC PROPERTY* named fiducial/panel.

The other *CLOSED CURVES* depicting fiducials shall reference

The*LEP SEMANTIC PROPERTY*) named fiducial/board.

A *LEP SEMANTIC PROPERTY* named "panel_offset" shall be attached to each associativity to mark it as a panel associativity. In addition, the panel_offset property has two real values, X and Y, that specify the offset for this pattern in relation to the panel.

6.6.3. LEP Entity Usage for Drafting/Documentation Applications

Documentation of the LEP is primarily a manual operation. However, there are ways to automate parts of the documentation preparation processes. The differences between the information needed for documentation versus that needed for automated and semi-automated processes are discussed below. Following that is usage information about Assembly Documentation and a reference to an AP for Technical Illustrations. For this version, the discussion on entities to support Fabrication documentation is lamentably sketchy.

6.6.3.1. Major differences between the documentation and the PDQ model of a conformant LEP AP -

The PDQ model of a conformant LEP AP is at a specific orientation with a scale of 1.0. The documentation may be at any rotation(s) with any scale (e.g., to show fine detail). The picture may be cut into two or more sheets. One half of a scaled drawing then appears on one sheet, and the other half on the second sheet.

The PDQ model of a conformant LEP AP usually contains only a few annotation entities. The documentation contains many annotation entities, such as dimensioning and labeling.

The PDQ model of a conformant LEP AP exists in model space. The documentation often has many distinct views and may have a drawing space for each drawing entity.

6.6.3.2. Changes and additions to PDQ LEP AP

Changes to the DE for Documentation files:

> No hierarchy is needed,
> Line font may have more variability than eight colors,
> Line thickness may have more variability, and
> Components need not be subordinate.

Entities (Type 1xx) or entity forms beyond the LEP AP may be needed.

Geometry entities have been added:

> Unbounded Plane Entity (Type 108) (only as a clipping plane to a View Entity (Type 410) has been added for documentation). Additional forms of the Copious Data Entity (Type 106) include Forms 20 and 21, 31-38, and 40.

Dimensioning entities have been added:

> All 200 series entities are permitted in documentation files.

Other entities have been added for documentation files:

> Full color (including the Color Definition Entity (Type 314)); dimensions of various types; the View Entity (Type 410) and Drawing Entity (Type 404); plus various Properties- (e.g., for Units and Size).

The Flow Associativity Entity (Type 402, Form 18) is not needed for documentation files.

Type 420/320/132 entities:

> Strictly speaking, these connectivity entities are not needed for documentation files. For some users though, the loss of PDQ information may outweigh the ease of implementation.

Documentation of the LEP includes:

> Drafting/Dimensioning usually provides overall LEP dimensions, plus dimensions relating various LEP features such as tooling holes and mounting hardware.

6.6.3.3. LEP Entity Usage for Fabrication Documentation

This documentation includes: Fabrication instructions including layer order, substrate material

and thickness, special processing notes, and registration Fiducials.

Production Assembly information tells how to build a specific LEP product using detailed step by step instructions for a given manufacturing process.

6.6.3.4. LEP Entity Usage for Assembly Documentation

Production Assembly information tells how to build a specific LEP product using detailed step by step instructions for a given manufacturing process.

One common method used in manufacturing is the "station/operation" approach. This method divides the task of populating a LEP into several discrete steps. Each step begins with a check of the work done in the previous step. Then the task(s) of the current step are performed and the partial assembly is passed onto the correct location for the next step. For example, step one might inspect the unpopulated LEP for damage or irregularities. If the LEP seems good in that regard, it is placed in a holder, and then five resistors having identical part numbers are inserted into the PWB. Step two begins with an inspection to ensure the five parts supposedly added in step 1 are in position, and that each has (or could have) the correct part number. The operator performing step 2 then inserts six capacitors and four more resistors. Step 3 begins by inspecting the four resistors. Then the six capacitors are checked for polarity as well as correct part number and location.

The elapsed time to perform the tasks at each station may be adjusted to minimize bottlenecks. Each station is furnished with a color coded picture, indicating which parts added at the previous station are to be inspected, and which parts are to be added at this station. Manual creation of these process pictures may take quite a bit of resources and time, or, it may be automated using a PDQ LEP database.

If this process sheet creation application will not be automated, then MIL-D-28000 Class II, the Engineering Drawing subset may be invoked to assure conformance of any IGES drawing files created to provide color coded pictures at each station.

If this process sheet creation application will be automated, a PDQ LEP AP or data base may well be employed. Notice that the final IGES files that are generated by the automated process may need to be converted to MIL-D-28000 Class II IGES files. Colors and fill patterns may be assigned during that conversion process.

One additional file is required by the automated process. It is a file that tells which parts are placed at which station.

6.6.3.5. LEP Entity Usage for LEP Technical Illustrations

The CALS MIL-D-28000 Class I Technical Illustrations conformance requirements for Technical Publications shall apply for LEP Technical Publications. The LEP Technical Illustrations application shall add no entities, nor impose changes to existing Class I entities.

6.6.4. LEP Entity Usage for CAD to CAD exchange

Modifying the LEP AP IGES file to make it more legible to a particular brand or version of receiving CAD system is outside the scope of this AP. Vendors products may exist in this area.

6.6.4.1. LEP Entity Usage for Layer assignment

The assignment of particular LEP features, such as LEP Layers, to specific IGES Levels is outside the scope of this AP. In this AP, most semantic information is conveyed through the use of IGES Properties (Type 406).

6.6.4.2. LEP Entity Usage for Color

The custom of using colors to convey semantic information is outside the scope of this AP. A palette of 8 colors has been adopted to provide minimal visual variety.

6.6.4.3. Semantic Properties for LEP Applications

This AP requires the use of specific properties to communicate the intended use of some of the data encoded by means of the AIM. These properties are gathered into sections for particular Application Information Requirement areas and listed below. Each entry in the list is in the form of a string as it might appear in the Parameter Data of the Property Entity (Type 406). This serves to illustrate how name strings, (e.g., "3, 3Htop" for top) integers, (e.g., "1, 420" for 420) and real numbers (e.g., "2, 0.250" for 0.25) may appear in the IGES file sent to the processor(s). The *LEP SEMANTIC PROPERTY* is especially versatile in this regard.

Forms 1, 2, 3, 5, 6, 9, 10, 15, 24 and 27 of the IGES Property Entity (Type 406) may be found in conformant files. This AP specifies which entities are to reference particular Properties whenever that seems feasible. Some Properties may stand alone in the LEP AP conformant files. Many *LEP SEMANTIC PROPERTIES* are introduced in this AP to convey important semantic information and data for LEP applications.

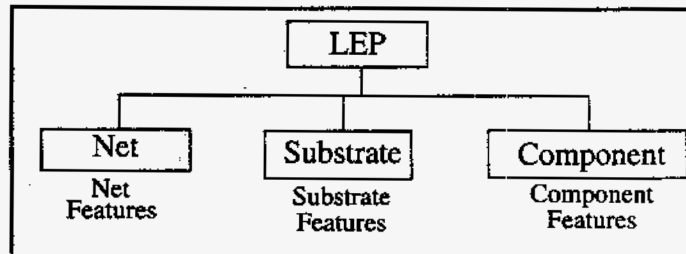


Figure 6-3. LEP Semantic Features

List of New Level Function Properties for LEP Applications

Processors should treat property text strings, both descriptions and values, as not case sensitive.

0=No value; 1=Integer; 2=Real; 3=Character string; 4=Pointer; 5=Not used; 6=Logical

Table 6-6. Level Function Properties (Type 406, Form 3)

Parameter Data (fragment)
0,9Htrace_top 0,9Htrace_bot 0,7Htrace_2 0,7Htrace_n 0,14Hsilkscreen_top 0,14Hsilkscreen_bot

List of New Generic Properties for LEP Applications

Table 6-7. Semantic Properties of Top Level LEP

Parameter Data (fragment)				
NP	Name	NV	TYP*	Value*
1,	27Htop_level_assembly_instance;			
4,	16Hlep_assembly_tag,	1,	3,	10Hmicro_proc ;
4,	25Hprinted_wire_assembly_tag,	1,	3,	9Hmux_demux ;
4,	32Hhybrid_microcircuit_assembly_tag,	1,	3,	9Hvideo_amp ;
6,	4Hdate,		3,	8Hmm/dd/yy,
			3,	13Hfile_created;
4,	26Hcomponent_default_padstack,	1,	3,	11HPAD_062_038;
4,	16Hschematic_number,	1,	3,	10HCK87654321 ;
4,	15Hpwb_part_number,	1,	3,	8H87654321 ; *
4,	15Hhmc_part_number,	1,	3,	8H87654321 ; *
4,	15Hmcm_part_number,	1,	3,	8H87654321 ; *
4,	15Hpwa_part_number,	1,	3,	8H87654321 ; *
4,	15Hpcb_part_number,	1,	3,	8H87654321 ; *
4,	15Hpca_part_number,	1,	3,	8H87654321 ; *
4,	17Hflexb_part_number,	1,	3,	8H87654321 ; *
4,	17Hflexc_part_number,	1,	3,	8H87654321 ; *
4,	24Hmicrostrip_b_part_number,	1,	3,	8H87654321 ; *
4,	23Hassembly_drawing_number,	1,	3,	10HAY87654321 ;
16,	24Hlep_physical_orientation,	7,	3,	9Hx_rot_nom,
			1,	45,
			3,	9Hy_rot_nom,
			2,	22.5,
			3,	9Hz_rot_nom,
			1,	0,
			3,	3Hdeg ;
16,	20Hlep_design_thickness,	7,	3,	3Htop,
			2,	0.27,
			3,	3Hbot,
			2,	0.125,
			3,	3Htot,
			2,	0.500,
			3,	2Hin ;

* These fields are repeated as required.

Table 6-8. LEP Semanic Properties of Network

Parameter Data (fragment)				
NP	Name	NV	Type	Value
4, 1,	16Hschematic_number, 14Hstandalone_via;	1,	3,	10HCK87654321;

Table 6-9. LEP Semanic Properties of Substrates

Parameter Data (fragment)				
NP	Name	NV	TYP*	Value*
6,	4Hdate,	2,	3, 3,	8Hmm/dd/yy, 13Hfile_creation
4,	16Hschematic_number,	1,	3,	10HCK87654321;
4,	10Hpwb_number,	1,	3,	8H87654321;
4,	15Hassembly_number,	1,		10HCK87654321;

* These fields are repeated as required.

NOTE: Also see Generic Properties of Closed Curve.

Table 6-10. Semantic Properties of Component and/or LEP

Parameter Data (fragment)				
NP	Name	NV	Type	Value
4,	15Habs_voltage_max,	1,	2,	120.0 ;
4,	9Htolerance,	1,	2,	0.01 ;
4,	10Htherm_cond,	1,	2,	18.0 ;
4,	8Htherm_jc,	1,	2,	ww.www ;
4,	7Htherm_r,	1,	2,	www.www ;
4,	14Hjunction_max_t,	1,	2,	125.0 ;
4,	29Hphysical_component_device_tag,	1,	3,	10Hquad_nand2 ;
4,	30Helectrostatic_discharge_rating,	1,	3,	3H5KV ;
4,	16Hcomponent_height,	1,	2,	0.250 ;
4,	26Hcomponent_outline_overhang,	1,	2,	0.038 ;

Table 6-11. LEP Semantic Properties of Component Definitions (Type 308 or 320)

Parameter Data (fragment)				
NP	Name	NV	TYP*	Value*
4,	18Hcomponent_std_name,	1,	3,	11HSIC.0003.20;
4,	17Hcomponent_version,	1,	3,	7HVer1.03;
4,	18Hcomponent_pkg_type,	1,	3,	3Haxi;
4,	19Hcomponent_tech_type,	1,	3,	3Hsmt;
4,	24Hcomponent_layout_surface,	1,	3,	5Hfront;
6,	26Hcomponent_insertion_origin,	2,	2,	x.xxx,
			2,	y.yyy;
16,	35Hcomponent_placement_feeder_location,	7,	3,	1Hx,
			2,	3.75,
			3,	1Hy,
			2,	5.25,
			3,	1Hz,
			2,	2.5,
			3,	2Hft;
4,	20Hcomponent_insertable,	1,	3,	3Hyes;
4,	16Hcomponent_height,	1,	2,	0.25;
8,	18Hcomponent_max_size,	3,	2,	xxx.xxx,
			2,	yyy.yyy,
				zzz.zzz;
4,	26Hcomponent_outline_overhang,	1,	2,	0.038;

* These fields are repeated as required.

Table 6-12. LEP Semantic Properties of Component Instances (Type 408 or 420)

Parameter Data (fragment)				
NP	Name	NV	Type	Value
4,	23Hcomponent_instance_side,	1,	3,	3Htop;
4,	18Hcomponent_preplace,	3,	2,	xx.xxx;
			2,	yy.yyy
			3,	3Htop;
1,	13Hcomponent_via;			
4,	14Hstandalone_via;			

Table 6-13. LEP Semantic Properties of Connect Point (Type 132)

Parameter Data (fragment)				
NP	Name	NV	Type	Value
1,	20Hallowable_test_point;			
1,	24Hnot_allowable_test_point;			
4,	13Hpad_tech_type,	1,	3,	3Hsmt;
4,	26Hcomponent_default_padstack,	1,	3,	11Hpad_o62_038;
1,	4Hpin1;			

Table 6-14. LEP Semantic Properties of Closed Curves

Parameter Data (fragment)				
NP	Name	NV	Type	Value
1,	17Hplacement_outline;			
1,	15Hthermal_outline;			
4,	22Hcomponent_keep_outline,	1,	3,	7Hkeepout;
4,	18Htrace_keep_outline,	1,	3,	7Hkeepout;
4,	16Hvia_keep_outline,	1,	3,	7Hkeepout;
1,	21Hplanar_shape_function,	1,	3,	3Hsignal;
4,	25Hconductive_trace_function,	1,	3,	3Hsignal;*
4,	28Hphysical_padstack_definition;*			
1,	26Hsubstrate_physical_outline ;			
1,	23Hcomponent_instance_side,	1,	3,	3Htop ;
4,	18Hcomponent_preplace,	2,	2,	xx.xxx,
8,			2,	yy.yyy;
1,	20Hlep_physical_outline;			
6,	26Hcomponent_default_padstack,	3,	3,	3Htop ;
			3,	11HPAD_062_038,
			3,	9Hcomponent,
			3,	3Htop;
4,	7Hfiducial,	1,	3,	6Htarget;

Table 6-15. LEP Semantic Properties of Non-Closed Curves

Parameter Data (fragment)				
NP	Name	NV	Type	Value
4,	25Hconductive_trace_function,	1,	3,	8HFlowName;
4,	21Hplanar_shape_function,	1,	3,	8HFlowName;
1,	9Htrace_bot;			
4,	7Htrace_n,	1,	1,	4;
1,	9Htrace_top;			

Table 6-16. LEP Semantic Properties of Feature Definitions

Parameter Data (fragment)				
NP	Name	NV	Type	Value
6,	8Hfiducial,	2,	3,	(see**list), (see*** list);
1,	28Hphysical_padstack_definition;			

** Values: 9Hcomponent, 7Hcluster, 5Hboard, 5Hpanel.

*** Values: 3Htop, 6Hbottom.

Table 6-17. LEP Semantic Properties of Vias

Parameter Data (fragment)				
NP	Name	NV	Type	Value
1,	13Hcomponent_via;			
1,	15Hstand-alone_via;			
1,	20Hallowable_test_point;			
1,	24Hnot_allowable_test_point;			
4,	17Htest_point_pwb_id,	1,	3,	5HTP321;
1,	16Hselected_test_point;			

Table 6-18. Production Engineering LEP Semantic Properties

Parameter Data (fragment)				
NP	Name	NV	Type	Value
0,	20Hpwb_physical_outline;			
2,	4Hdate,	2,	3,	8Hmm/dd/yy,
			3,	13Hfile_creation;
4,	26Hcomponent_default_padstack,	1,	3,	11Hpad_062_038;
6,	5Hpanel_offset,	2,	2,	xx.xx,
				yy.yy;
4,	14Hstation_number,	1,	1,	420;
4,	14Hstation_action,	1,	3,	6Hinsert;

Table 6-19. Insertion Machinery LEP Semantic Properties

Parameter Data (fragment)				
NP	Name	NV	Type	Value
4,	26Hcomponent_insertion_height,	1,	2,	0.025;

Table 6-20. Placement Machinery LEP Semantic Properties

Parameter Data (fragment)				
NP	Name	NV	Type	Value
4,	33Hcomponent_placement_lead_diameter,	1,	2,	0.025;
4,	22Hcomponent_placement_yn,	1,	3,	1Hn;
4,	25Hcomponent_placement_body,	1,	3,	4HTO22
6,	26Hcomponent_placement_center,	2,	2,	0.000,
			2,	0.000;
4,	29Hcomponent_placement_form_code,	1,	1,	7;
4,	41Hcomponent_placement_form_code_descri	1,	3,	9Htombstone;
8,	ption,	3,	2,	37.54,
	25Hcomponent_insertion_force,		2,	40.0,
			2,	47.5;
8,		3,	3,	3Hmax,
	28Hcomponent_physical_thickness,		2,	0.040,
			3,	2hin;

Table 6-21. Wire Wrap Application LEP Semantic Properties

Parameter Data (fragment)				
NP	Name	NV	Type	Value
1,	27Htop_level_assembly_instance ;			
1,	16Hlep_assembly_tag,	1,	3,	12Hproto_ww_134
1,	9wire_wrap;			
4,	5Hgauge,	1,	1,	26;
4,	10Hwire_color,	1,	3,	4Hblue;
1,	12Htwisted_pair;			
4,	10Hwrap_class,	1,	3,	1Hb;

Table 6-22. Production Test LEP Semantic Properties

Parameter Data (fragment)				
NP	Name	NV	Type	Value
1,	20Hallowable_test_point;			
1,	24Hnot_allowable_test_point;			
8,	28Hcomponent_physical_thickness,	3,	3,	2Hmax
			2,	0.040
			3,	2Hin;
6,	23Hselected_test_point_tag,	2,	3,	11HDITMCO_9100,
			3,	8Hx312y207;
4,	18Htest_point_pwb_tag,	1,	3,	11Hframe_reset;
4,	18Htest_point_nail_id,	1,	3,	5Hpt329;

6.6.4.4 LEP Entity Usage for Presentation (appearance)

Presentation aspects of the graphics produced by IGES files are especially important for Engineering Drawing and Technical Illustration applications. Other documents (called out in CALS MIL-D-28000 Classes II and I) provide more detailed guidance for those applications. The principal focus of this AP has been to enable mechanical/electronic exchange of Layered Electrical Product data as simply and unambiguously as possible.

6.6.5 LEP Entity Usage for CAD Simulation/Analysis

Simulation applications and Analysis need a broad range of product information appropriate to their purposes. Some of the information needed by such applications may be provided as prescribed in the LEP AP. Such information is primarily in the realm of physical description of the

product. Some of the semantic descriptions (particularly those of the connectivity among product components) may also prove useful in these applications.

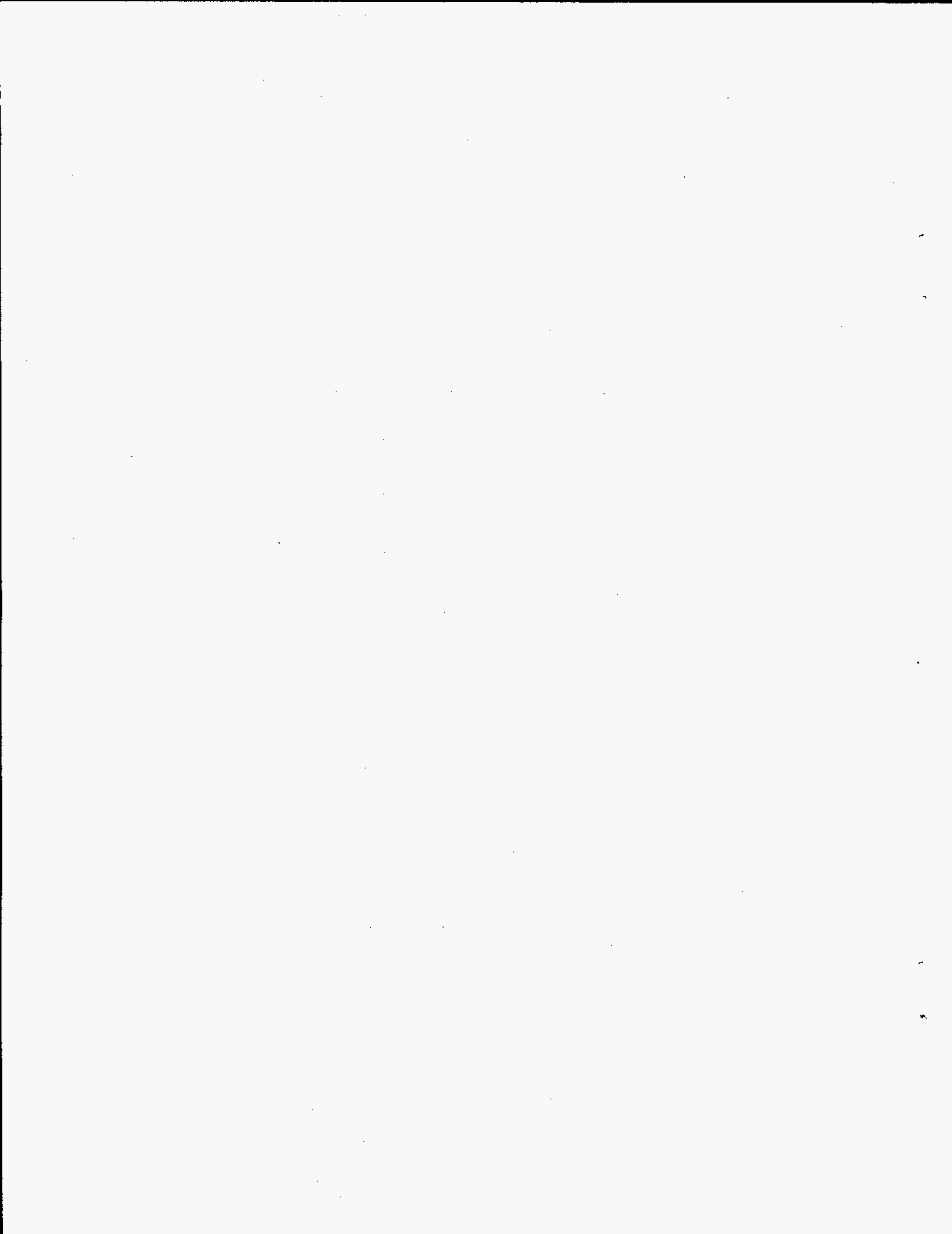
Other information needed by simulation and analysis applications may be captured using the IGES Tabular Data Entity Type (Type 406, Form 11). See the sections headed Electromagnetic Radiation Parameters (PTYPE = 22), Thermal Conductivity (PTYPE = 19), and Heat Capacity (PTYPE = 20) plus the other PTYPES for Finite Element data. More help is provided by the use of the Attribute Table Definition Entity Type (Type 322) and its companion Attribute Table Instance (Type 422). Tables 6 and 9 in the discussion of the Attribute Table Definition list such attributes as Capacitance, Delay Time, Junction Temperature, Tolerance and Units. Values for those and other listed attributes may be associated with appropriate LEP Components as described in IGES by referencing the applicable Attribute Table Instance in the second group of additional pointers discussed in IGES 2.2.4.4.2.

BLANK PAGE

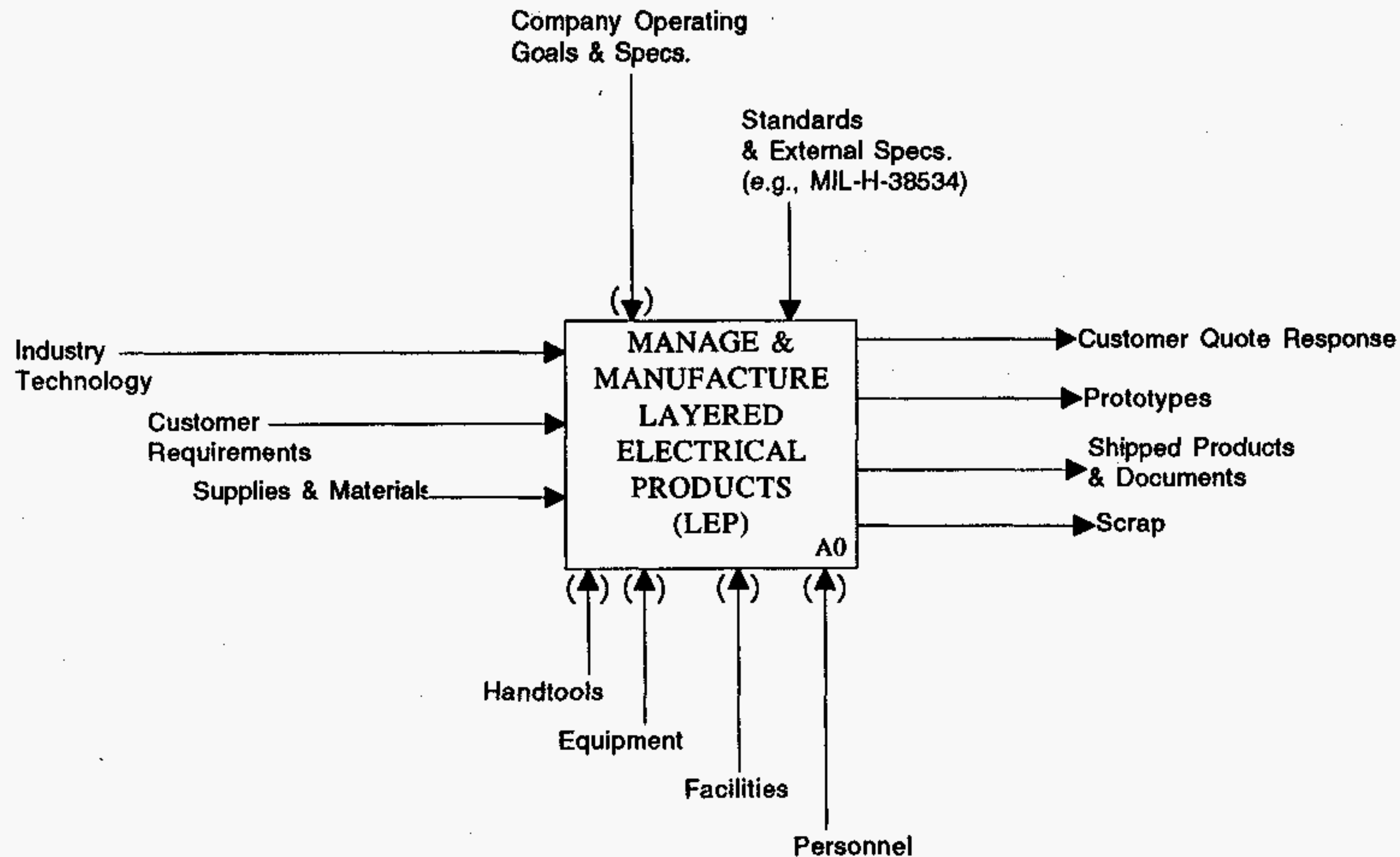
APPENDIX A. APPLICATION ACTIVITY MODEL

A. ACTIVITY MODEL CONTENTS

A-0	Model Context
A0	Manage and Manufacture Products
A0 Text	Text: Manage and Manufacture Products
A2	Perform Engineering
A2 Text	Text: Perform Engineering
A3	Assure Product Quality
A3 Text	Text: Assure Product Quality
A32	Perform Quality Appraisal
A32 Text	Text: Preform Quality Appraisal
A4	Produce Products
A4 Text	Text: Produce Productds
A41	Build Networks
A41 Text	Text: Build Networks
A42	Assemble Devices
A42 Text	Text: Assemble Devices
A43	Test Product
A43 Text	Text: Test Product

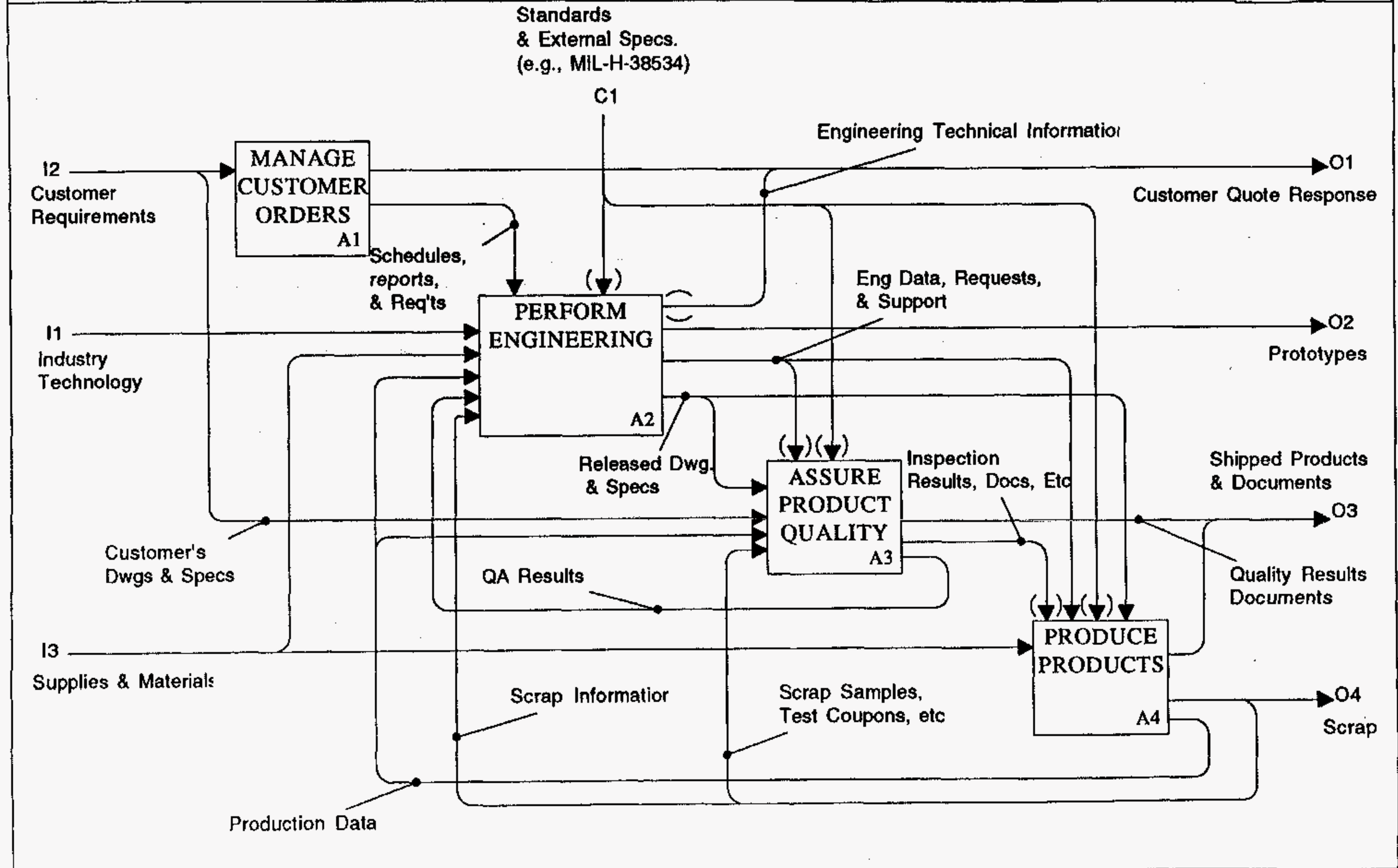


USED AT:	AUTHOR: Charles Azu PROJECT: NOSC Hybrid MicroCIN NOTES: 1 2 3 4 5 6 7 8 9 10	DATE: 10/31/91 REV: 3/17/94	WORKING DRAFT RECOMMENDED PUBLICATION	READER DATE	CONTEXT: Top



PURPOSE: To provide the manager with the information context needed for planning automation changes for his LEP design and fabricatic
The text describing each activity is intended to be generic; it is not meant to be proscriptive or to constrain processes bei
VIEWPOINT: The data exchange problem as seen by designers and manufacturers of LEP

NODE: A-0	TITLE: CONTEXT: APPLICATION PROTOCOL ACTIVITY MODI	NUMBER:


USED AT:
AUTHOR: Charles Azu
PROJECT: NOSC Hybrid MicroCIM
NOTES: 1 2 3 4 5 6 7 8 9 10
DATE: 10/31/91
REV: 11/26/91
WORKING
DRAFT
RECOMMENDED
PUBLICATION
READER
DATE
CONTEXT:
Standards
& External Specs.
(e.g., MIL-H-38534)
C1
Engineering Technical Informatioı
I2
Customer
Requirements
MANAGE
CUSTOMER
ORDERS
A1
O1
Customer Quote Response
Schedules,
reports,
& Req'ts
PERFORM
ENGINEERING
A2
Eng Data, Requests,
& Support
I1
Industry
Technology
O2
Prototypes
Released Dwg.
& Specs
ASSURE
PRODUCT
QUALITY
A3
Inspection
Results, Docs, Etc
Shipped Products
& Documents
O3
Customer's
Dwgs & Specs
QA Results
Quality Results
Documents
I3
Supplies & Materials
PRODUCE
PRODUCTS
A4
Scrap Informatioı
Scrap Samples,
Test Coupons, etc
O4
Scrap
Production Data
NODE: A 0
TITLE:
MANAGE AND MANUFACTURE PRODUCT
NUMBER:

USED AT:	AUTHOR: Charles Azu PROJECT: NOSC Hybrid MicroCIM NOTES: 1 2 3 4 5 6 7 8 9 10	DATE: 10/31/91 REV: 4/8/92 3/17/94	WORKING DRAFT RECOMMENDED PUBLICATION	READER DATE	CONTEXT: Top

MANAGE & MANUFACTURE HYBRID DEVICES (A 0)

1. Manage Customer Orders: Orders for products are received in the form of a requirements package. Some manufacturers require on the system definition of the "black box" that they are to design and build (SCD or data equivalent). The orders are analyzed to see if the manufacturing organization has the capability to produce the product. If the organization can produce the product, the customer is given a quote for work. A preliminary schedule is made based on the customer requirements. The schedule along with custor requirements are used as a control and input to the perform engineeri activity. Based on the Customer Requirements the applicable Military and Industry Standards and Specifications are chosen for the entire design, engineering, and production process. A Quality Function Deployment (QFD) phase may be entered during this activity. The QFD results can be translated into a type of product at the end of the phase.

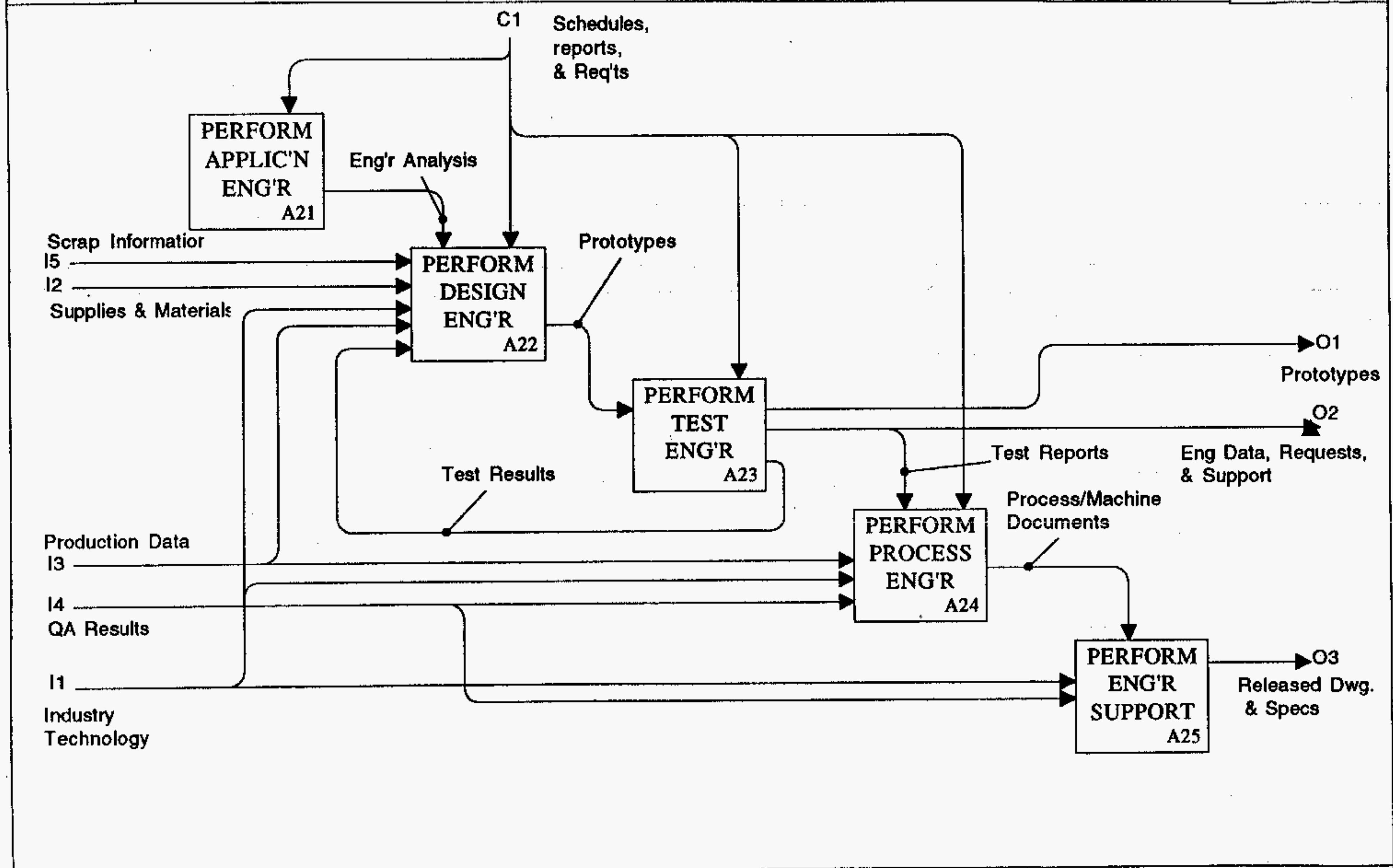
2. Perform Engineerng: The engineering of the product takes place using the customer requirements, government & industry standards a: guideline for the work done. Availability of supplies, materials, and new technology are taken into consideration during the design of a particular product. During this activity, all the operations take place that transform the customer's requirements into a set of documentation (schematics, netlists, test requirements, layout, etc.) and a released design that will be used to build prototype and production devices.

3. Assure Product Quality: The goal of this activity is to assure that the manufacturer's products meet both customer and government quali requirements. Production data is analyzed in this acitvity using SPC and design of experiments methodologies. Scrapped products are inspected to find their defects. Results of this activity are fed back to Perform Engineering. A quality plan is formed using customer requirements and it is updated using feedback from production. The quality plan along with inspection results are part of the documentation delivered to the customer.

4. Produce Product: In accordance with released design and specificat from engineering, the products are produced using the supplies and materials. The activity achieves this by planning, directing, and performing production activities that integrate materials, equipment, and personnel, to produce customer shipments. During the production process, some product that doesn't meet quality specifications is scrapped. Production data is taken and kept on products and their processes during production. This data is fedback to engineering and quality assurance. Approved products are packaged and shipped to th customer alona with their associated documentation.

NODE:	TITLE: A 0 T, MANAGE AND MANUFACTURE PRODUCT	NUMBER:

USED AT:	AUTHOR: Charles Azu PROJECT: NOSC Hybrid MicroCIM NOTES: 1 2 3 4 5 6 7 8 9 10	DATE: 10/31/91 REV: 1126/91	WORKING DRAFT RECOMMENDED PUBLICATION	READER	DATE	CONTEXT:



NODE: A 2	TITLE: PERFORM ENGINEERING	NUMBER:

USED AT:	AUTHOR: Charles Azu PROJECT: NOSC Hybrid MicroCIN NOTES: 1 2 3 4 5 6 7 8 9 10	DATE: 10/31/91 REV: 4/8/92 6/9/92 3/17/94	WORKING DRAFT RECOMMENDED PUBLICATION	READER DATE	CONTEXT: Top

PERFORM ENGINEERING, A2

1. Perform Application Engr.: Part of this acitvity consists of forming a design team to review the customer requirements to facilitate the product design. A few of the requirements to be reviewed consist of product package size, number of pins, current/voltage limitations, environmental considerations, and testing requirements.

2. Perform Design Engineering: In this activity the design of the produ substrate and the fully built up hybrid is accomplished. The substrate design is accomplished with the development of its layout and then its artwork. For the built up LEP, layouts are developed which are then used to complete the assembly details and generate th masks required for network fabrication. Material requirements for the network and assembly are developed and documented in drawings ca bill of materials. The design team reviews the complete product drawi and documentation. The team may run simulations of the circuit on a CAD machine as part of the review effort. The task in this activity is the building of prototype product to prove out the design concepts use in the product design. There is a tight coupling between this activ and Perform Process Engineering.

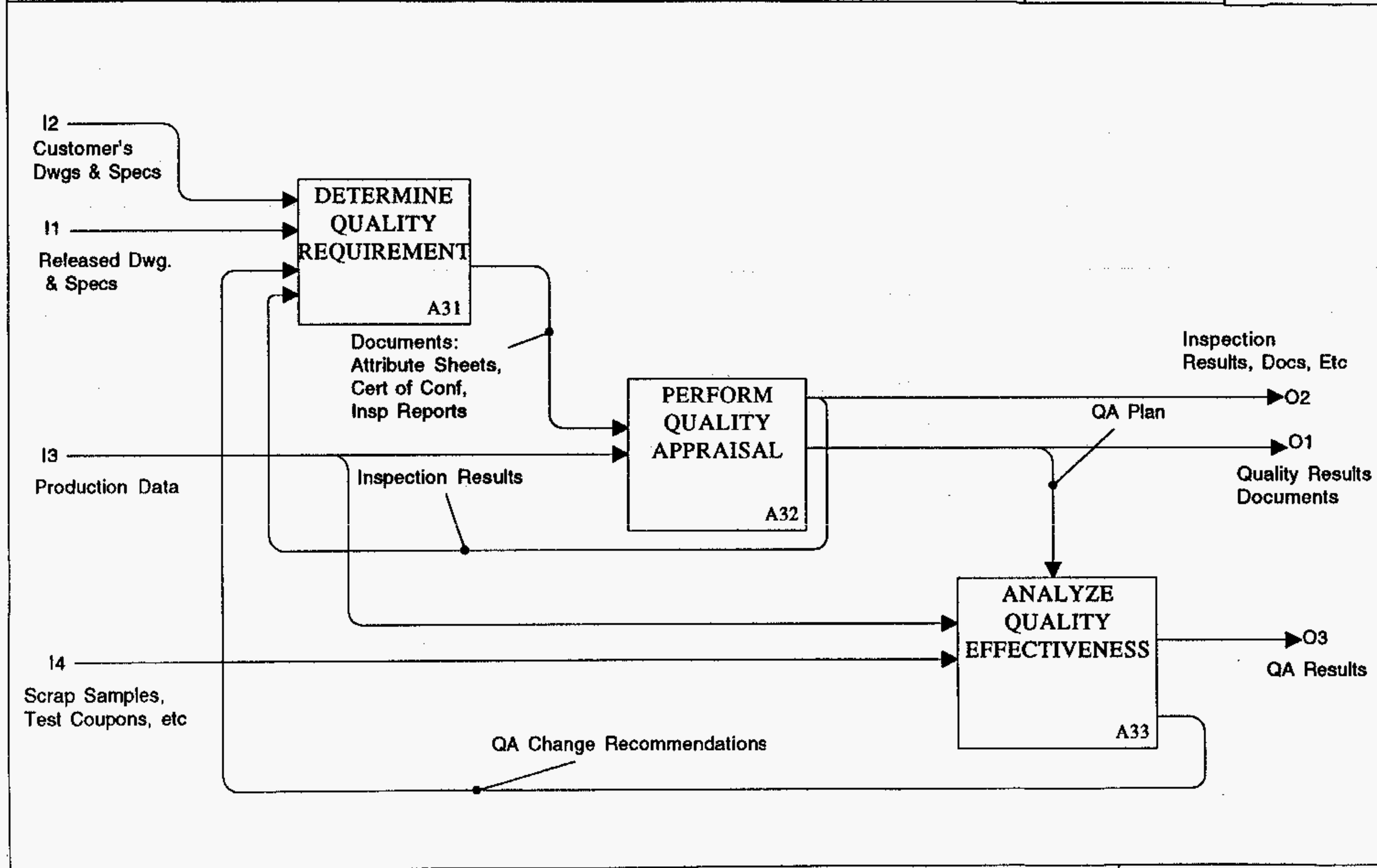
3. Perform Test Engineering: The purpose of this activity is to develop test systems. The test systems include those associated with electrical testing, active laser trimming, and burning-in-devices. Test systems are built for testing the product as well as the individual components on the product. In conjunction with the development of the test systems, test plans are designed. The prototypes built in the previous activity are tested according to the test plan using the testing systems. Problems encountered in the fabrication or testing of the prototypes lead to modification in the drawings and procedures used to build/test the product. When the prototypes have successfully passed all requirements, they are ready for production.

4. Perform Process Engineering: This activity consists of developing the procedures to produce the number of products required. Results from t prototype product build are used to develop the production processes and set production parameters (ie thru put rate thru a pick & place machine). Current technology and existing equipment are considered ir the development of the process. The usual governing factor in developir the process is the number of products of a certain type which will be built. Also taken into consideration are new technological developments. Process documentation is a result of this activity. Feedback from production is also taken in to update and refine the process parameters and documentation. There is a tight coupling betwe this activity and Perform Design Engineering.

5. Perform Engineering Support: This activity maintains configuration control through a variety of procedures including document control, software control, and change approval control. A vital part of this activity is the continuing support of the product throughout its life cycle. These support activities include: resolving production yield issues, failure analysis, providing the technical interface with customers, serving on the material review board, and developing ongoing product improvements.

NOTE- The Perform Design Engineering and Perform Process Engineerii are co-mingled in many organizations as part of a Concurrent Engineerir program. The two processes can and often do run in parallel.

NODE:	TITLE: A 2 T, PERFORM ENGINEERIN	NUMBER:



USED AT:
AUTHOR: Charles Azu
PROJECT: NOSC Hybrid MicroCIM
DATE: 10/31/91
REV:
NOTES: 1 2 3 4 5 6 7 8 9 10
WORKING
DRAFT
RECOMMENDED
PUBLICATION
READER
DATE
CONTEXT:
I2
Customer's Dwgs & Specs
I1
Released Dwg. & Specs
DETERMINE QUALITY REQUIREMENT
A31
Documents: Attribute Sheets, Cert of Conf, Insp Reports
PERFORM QUALITY APPRAISAL
A32
Inspection Results, Docs, Etc
O2
QA Plan
O1
Quality Results Documents
I3
Production Data
Inspection Results
ANALYZE QUALITY EFFECTIVENESS
A33
O3
QA Results
I4
Scrap Samples, Test Coupons, etc
QA Change Recommendations
NODE: A 3
TITLE: ASSURE PRODUCT QUALITY
NUMBER:

USED AT:	AUTHOR: Charles Azu PROJECT: NOSC Hybrid MicroCIN NOTES: 1 2 3 4 5 6 7 8 9 10	DATE: 10/31/91 REV: 4/8/92 3/17/94	WORKING DRAFT RECOMMENDED PUBLICATION	READER DATE	CONTEXT: Top

ASSURE PRODUCT QUALITY (A3)

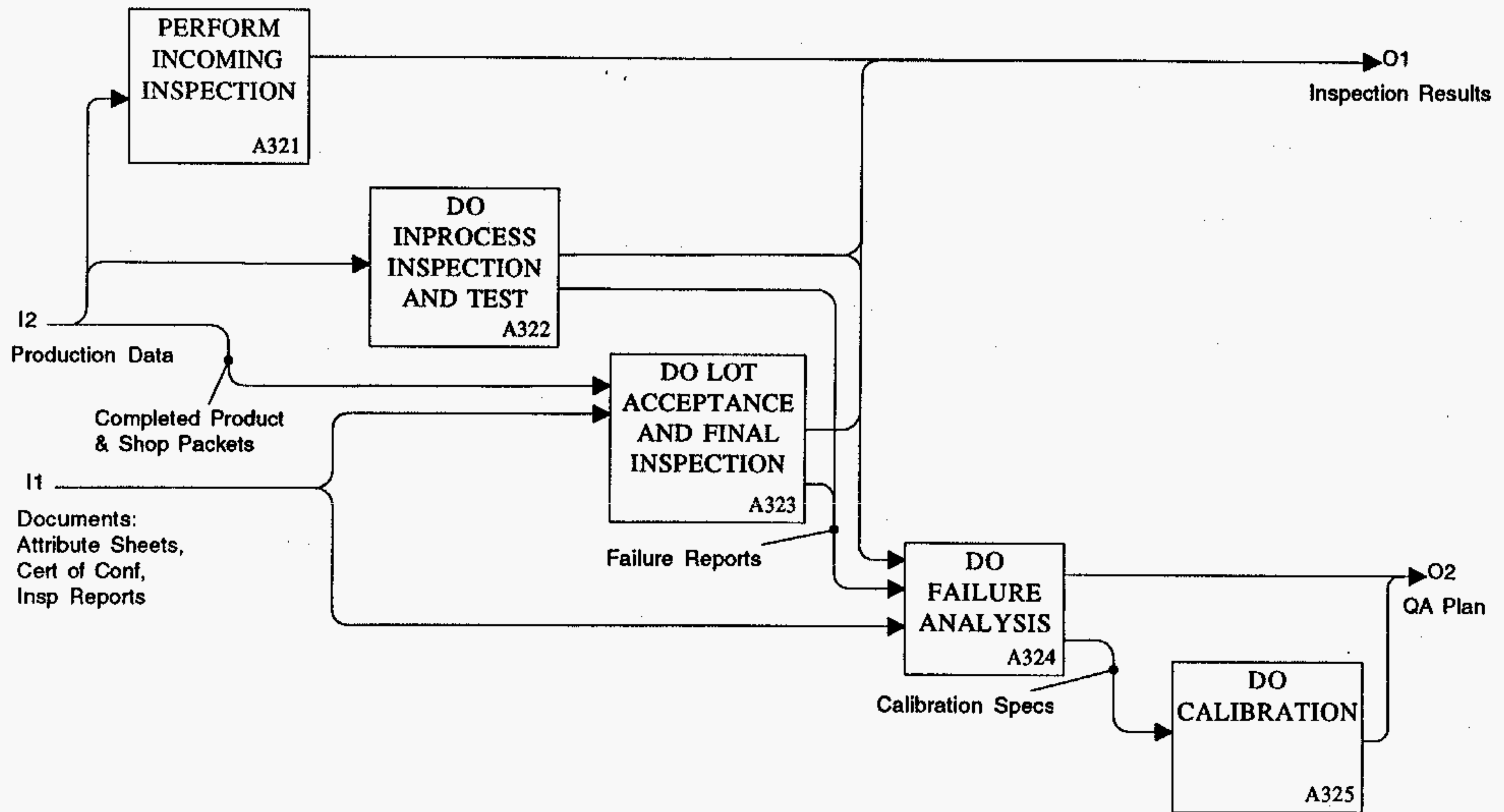
1. Determine Quality Requirements: Based on the drawings and engineer data on the product, and the customer requirements for quality and reliability, the quality assurance plan for the product production are established. The quality plans are put in place for implementation on a factory wide basis as the product is moved into production. The company's internal quality plans form part of the overall plan.

2. Perform Quality Appraisals: Inspect/Audit manufacturing and quality (area and process) periodically to assure continued conformance to internal and customer specification requirements. Results from this activity may be used in a continuous process of refining quality plans.

3. Analyze Quality Effectivness: The quality program is analyzed for its effectiveness. Part of this analysis consists of verifying that the process and testing equipment used in the production of LEPs are properly calibrated and are functioning correctly. Records are kept on calibrations and other adjustments to the equipment.

NODE:	TITLE: A 3 T, ASSURE PRODUCT QUALIT'	NUMBER:

USED AT:	AUTHOR: Charles Azu PROJECT: NOSC Hybrid MicroCIM NOTES: 1 2 3 4 5 6 7 8 9 10	DATE: 10/31/91 REV:	WORKING DRAFT RECOMMENDED PUBLICATION	READER	DATE	CONTEXT:



NODE: A 32	TITLE: PERFORM QUALITY APPRAISAL	NUMBER:

USED AT:	AUTHOR: Charles Azu PROJECT: NOSC Hybrid MicroCIN NOTES: 1 2 3 4 5 6 7 8 9 10	DATE: 10/31/91 REV: 4/8/92 3/17/94		WORKING DRAFT RECOMMENDED PUBLICATION	READER DATE	CONTEXT: Top

PERFORM QUALITY APPRAISAL (A 32)

1. Perform Incoming Inspection: Parts and materials needed to build a product are checked for compliance to standards specified by Engineeri An implicit part of this activity is supplier qualification, where the suppli qualified based upon the results of incoming inspection of the supplier's product. Results of this process are passed supplies, failed supplies, and a supplier qualification report.

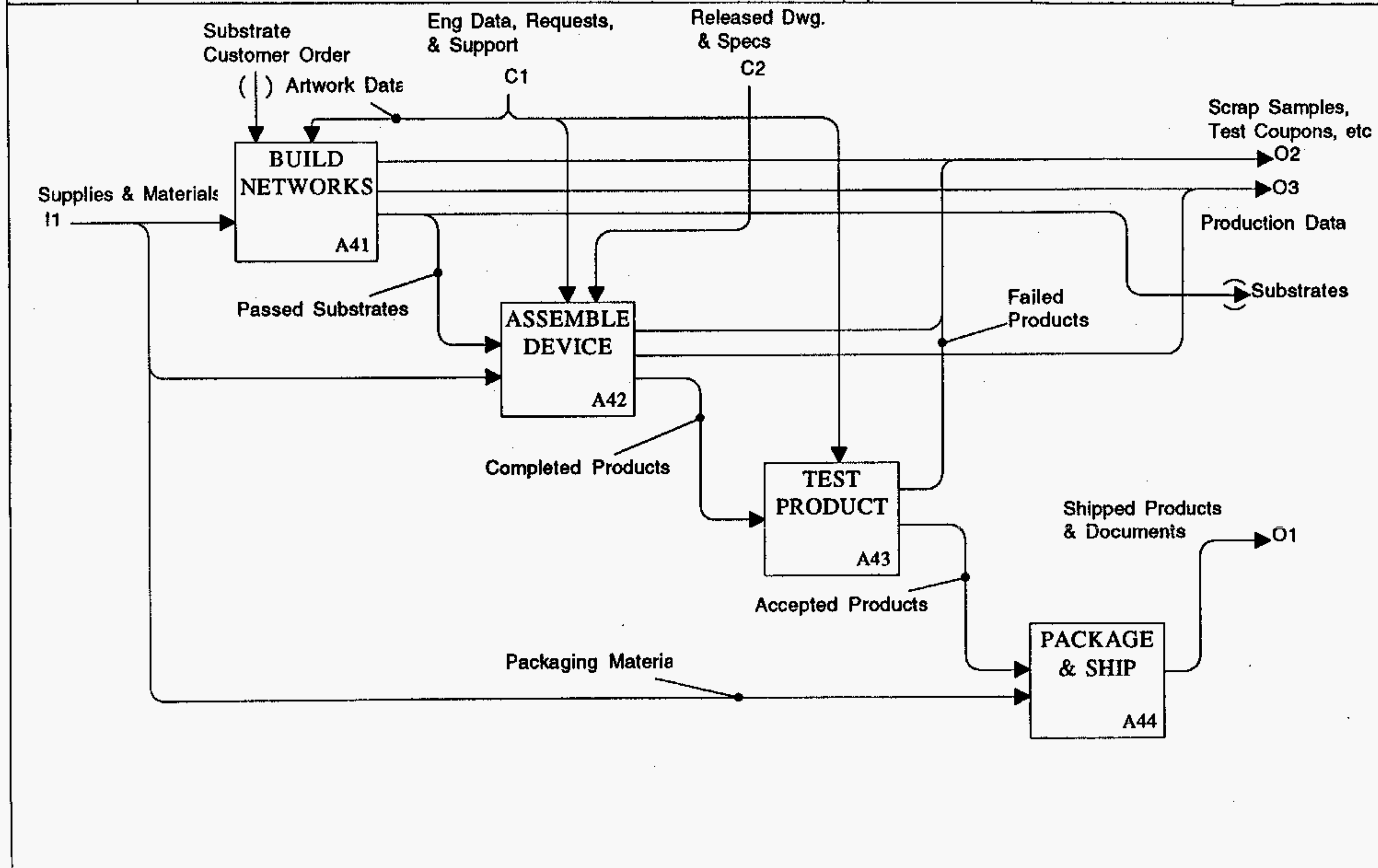
2. Inprocess Inspection and Test: Substrates and plated hybrids may be periodically checked for compliance to standards during production. The method and amount of inspection is established in the quality plan. Inspection mostly consist of visual and electronic. Product that fails this activity may be either reworked or scrapped. Some supplier qualification is done in this activity and this is fedback to the Perform Incoming Inspection process.

3. Final Inspection & Lot Acceptance: This activity consists of a final visual inspection of the built up hybrid before capsulization. Successfully passed products become part of a lot which is accepted fo packaging and shipment.

4. Failure Analysis: Rework of failed products and substrates is done in this activity. Those products which can not be diagnosed on the shop floor repair area may be taken to engineering for analysis.

5. Calibration: Calibration of equipement may occur based on the result of the failure analysis, otherwise calibration is done as part of Analyze Quality Effectiveness.

NODE:	TITLE: A 32 T, PERFORM QUALITY APPRAISA	NUMBER:


USED AT:
AUTHOR: Charles Azu
PROJECT: NOSC Hybrid MicroCIM
NOTES: 1 2 3 4 5 6 7 8 9 10
DATE: 10/31/91
REV: 2/13/92
3/17/94
WORKING
DRAFT
RECOMMENDED
PUBLICATION
READER
DATE
CONTEXT:
Substrate
Customer Order
Artwork Data
Eng Data, Requests,
& Support
C1
Released Dwg.
& Specs
C2
Scrap Samples,
Test Coupons, etc
O2
O3
Production Data
Substrates
Supplies & Materials
I1
BUILD NETWORKS
A41
Passed Substrates
ASSEMBLE DEVICE
A42
Failed Products
Completed Products
TEST PRODUCT
A43
Shipped Products
& Documents
O1
Accepted Products
PACKAGE & SHIP
A44
Packaging Materia
NODE: A 4
TITLE: PRODUCE PRODUCTS
NUMBER:

USED AT:	AUTHOR: Charles Azu PROJECT: NOSC Hybrid MicroCIN NOTES: 1 2 3 4 5 6 7 8 9 10	DATE: 10/31/91 REV: 4/8/92 6/9/92 3/17/94		WORKING	READER	DATE	CONTEXT:
				DRAFT			Top
				RECOMMENDED			
				PUBLICATION			

PRODUCE AND TEST PRODUCT (A4)

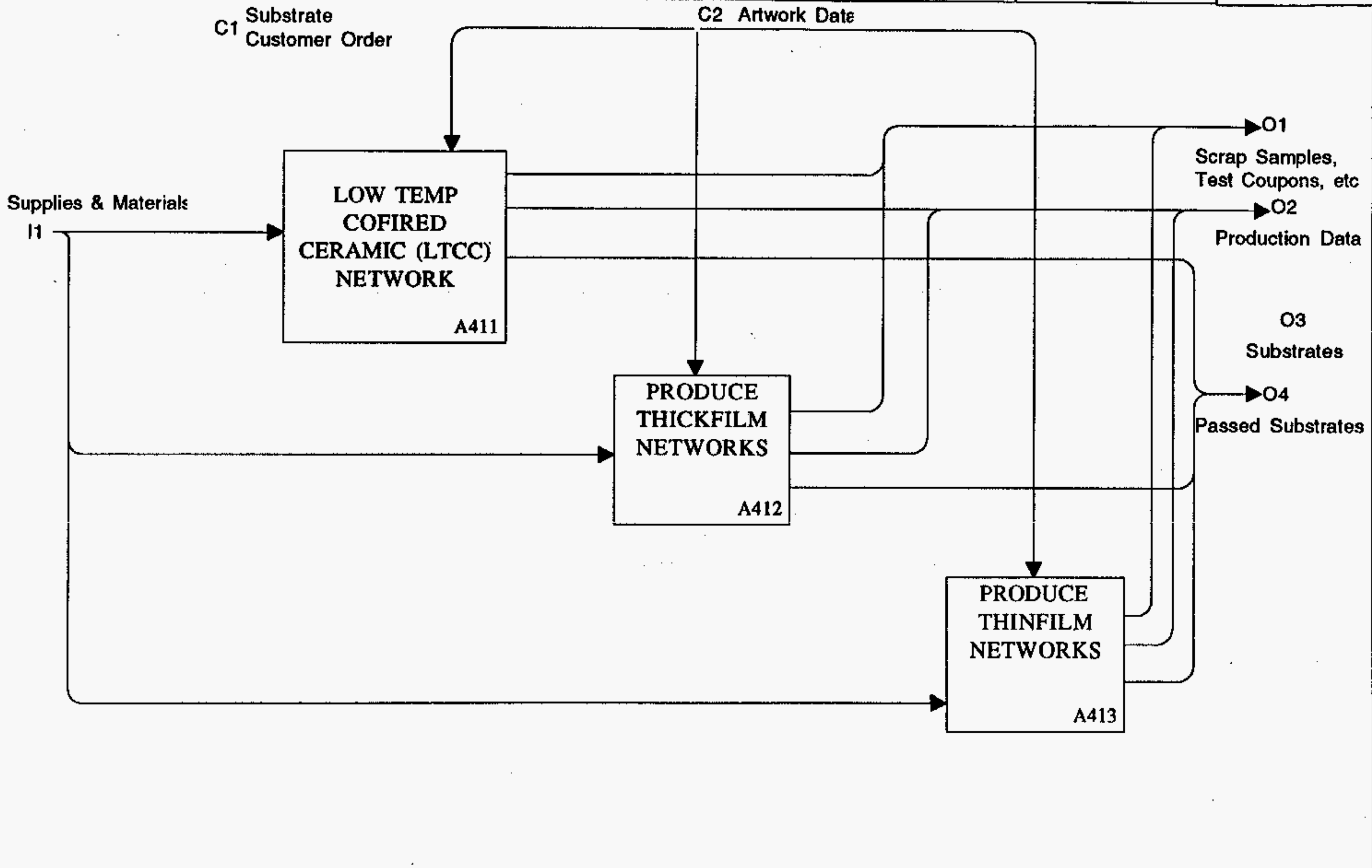
1. Build Substrate: Product substrates can be built using one of several methods including thick film, thin film, and green tape. Each method requires different technology and processing.

2. Build Product: In this step, the raw materials are transformed, in accordance with the design, into an LEP. The fabrication of an LEP consists of a sequence of fabrication steps such as screenings and firing of conductors an resistor geometries, application of appliqued devices and components, die a wire bonding, packaging, and various inprocess inspection steps. During th fabrication of the LEP, there may be tests performed (both physical and electrical) to ensure conformance to the tolerance specified in the design. Test results may be used to indicate inprocess corrections for successive devices to be fabricated.

3. Test Product: Testing of fabricated components and assemblies consists two types; testing electrical reliabilty and screening tests for thermal and mechanical failures. The test for electrical reliability include a preseal, pre burn, post burn-in, and a final acceptance test. Electrical tests confirm that all connections have been successfully made and that the circuit meets functional performance requirements. The screens for thermal/mecha failures include a stabilization bake, temperature cycling, constant acceleration, and leak tests. Products which are rejected from this activity are sent to be reworked. Any need for production improvement as a result of process is feedback to the Build Product (A42).

5. Package and Ship LEP: This section is shown for completeness. The delivery process may involve packaging, ensuring that documentation complete, and shipping. This data exists to the extent that it can be obtained from the automation data.

NODE:	TITLE: A 4 T, PRODUCE PRODUCT	NUMBER:


USED AT:
AUTHOR: Charles Azu
PROJECT: NOSC Hybrid MicroCIN
NOTES: 1 2 3 4 5 6 7 8 9 10
DATE: 10/31/91
REV:
WORKING
DRAFT
RECOMMENDED
PUBLICATION
READER
DATE
CONTEXT:
C1 Substrate Customer Order
C2 Artwork Date
Supplies & Materials
I1
LOW TEMP COFIRED CERAMIC (LTCC) NETWORK
A411
PRODUCE THICKFILM NETWORKS
A412
PRODUCE THINFILM NETWORKS
A413
O1
Scrap Samples, Test Coupons, etc
O2
Production Data
O3
Substrates
O4
Passed Substrates
NODE: A 41
TITLE:
BUILD NETWORKS
NUMBER:

USED AT:	AUTHOR: Charles Azu PROJECT: NOSC Hybrid MicroCIN NOTES: 1 2 3 4 5 6 7 8 9 10	DATE: 2/13/92 REV: 4/8/92		WORKING	READER	DATE	CONTEXT: Top
				DRAFT			
				RECOMMENDED			
				PUBLICATION			

A41 BUILD NETWORKS

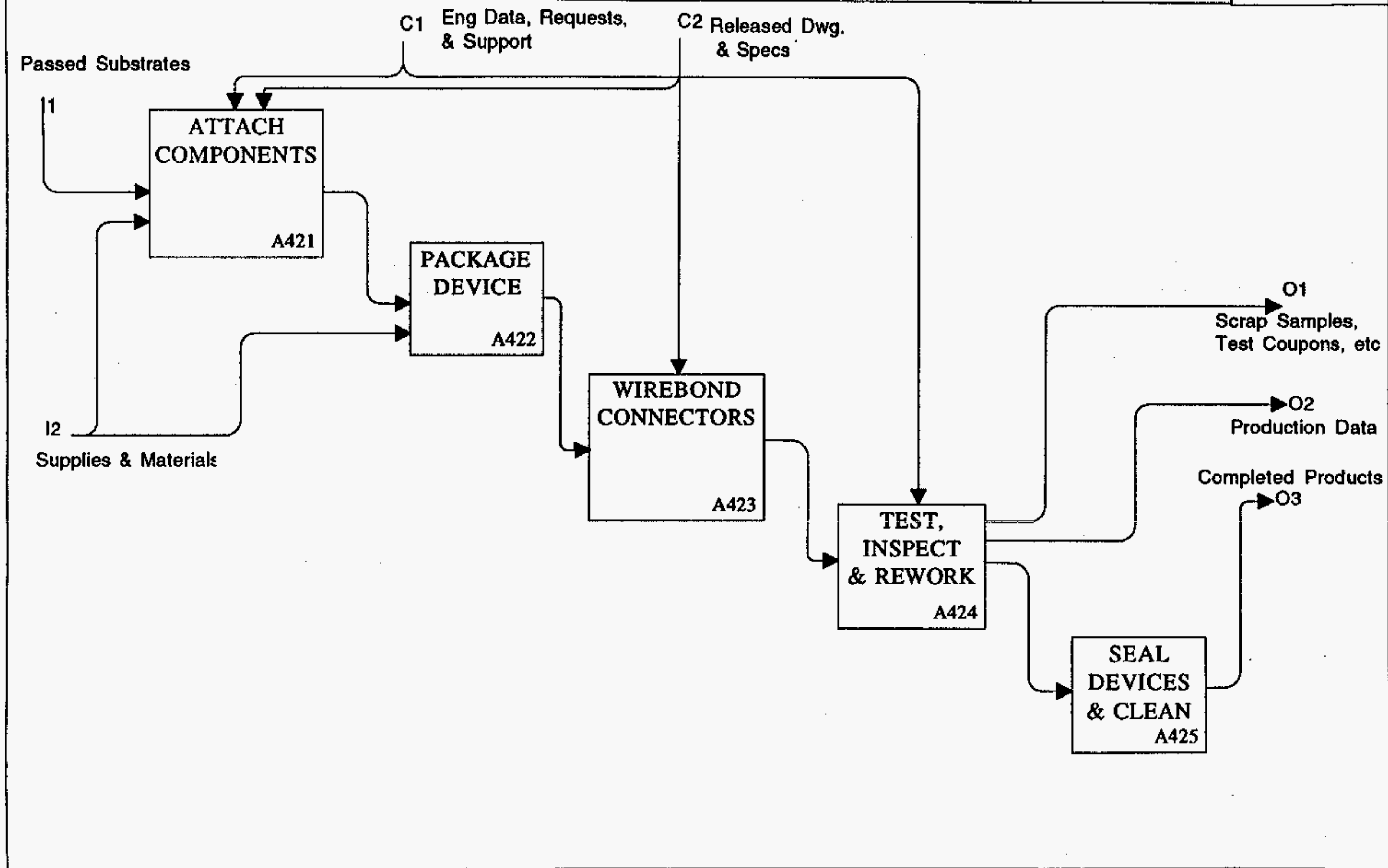
1. Hi/Low Temperature Co-fired Ceramics: Also called "green" tape, this substrate type is processed so that the individual layers of a multilayer substrate can be produced separately. Tl various layers are then laminated together in a stack and fired. Co-fired ceramics are used to fabricate high-density thick film multilayer substrates and packages.

2. Thick Film Networks: These networks are fabricated seque tially. The process involves using screen printing techniques to apply thick film conductors as well as resistive and dielectric pastes. After each screen printing pass (to apply a pattern of thick film material) the substrate is dried and fired at elevated temperatures to convert the film paste to a film with the required electrical and mechanical properties. A substrate usuall goes through several passes to deposit all layers.

3. Thin Film Networks: These networks are fabricated using vacuum vapor deposition, sputter deposition, and electroplatin to establish conductor and resistor metals. Photoplotting and chemical etching are used to define conductor, resistor, and polyamide geometries. Patterns are formed onto the substra using precision photolithography. This process involves applyir the photoresist, exposing the mask, developing the photoresist The circuits are baked at elevated temperature to stabilize resistor characteristics, relieve conductor stress, and polymeri the polyamide film. The annealed networks are then resistoı trimmed and broken into individual networks.

NOTE: For a particular LEP design, only one of these processes is invoked.

NODE: A 41 TEXT	TITLE: A 41 T, BUILD NETWORKS	NUMBER:


USED AT:
AUTHOR: Charles Azu
PROJECT: NOSC Hybrid MicroCIM
NOTES: 1 2 3 4 5 6 7 8 9 10
DATE: 10/31/91
REV:
WORKING
DRAFT
RECOMMENDED
PUBLICATION
READER
DATE
CONTEXT:
C1 Eng Data, Requests, & Support
C2 Released Dwg. & Specs
Passed Substrates
I1
ATTACH COMPONENTS
A421
PACKAGE DEVICE
A422
I2
Supplies & Materials
WIREBOND CONNECTORS
A423
TEST, INSPECT & REWORK
A424
SEAL DEVICES & CLEAN
A425
O1
Scrap Samples, Test Coupons, etc
O2
Production Data
Completed Products
O3
NODE: A 42
TITLE:
ASSEMBLE DEVICE
NUMBER:

USED AT:	AUTHOR: Charles Azu PROJECT: NOSC Hybrid MicroCIM NOTES: 1 2 3 4 5 6 7 8 9 10	DATE: 2/13/92 REV: 4/8/92 3/17/94	WORKING DRAFT RECOMMENDED PUBLICATION	READER DATE	CONTEXT: Top

A42 Assemble Devices

1. Attach Components: Elements such as silicon dies, capacitors, and resistors are attached to the network using appropriate epoxy, eutectic alloy, or solder. The epoxy is often screen printed onto the substrates. Elements are usually attached with pick and place robots. The elements are usually cleaned first. Once all elements are attached to the substrate, the substrate is baked at high temperature to cure the epoxy. This process may be done after the device is packaged depending upon a particular design and a company's methods.

2. Package Devices: The assembly networks are packaged into cases and attached with epoxy. The process contains several steps: cleaning the case, epoxying the network to the package, baking the packaged device to cure the epoxy, marking the package with the necessary information for identification, and baking the package and device once more to cure the markings. This process may be done before the components are attached depending upon a particular design and a company's methods.

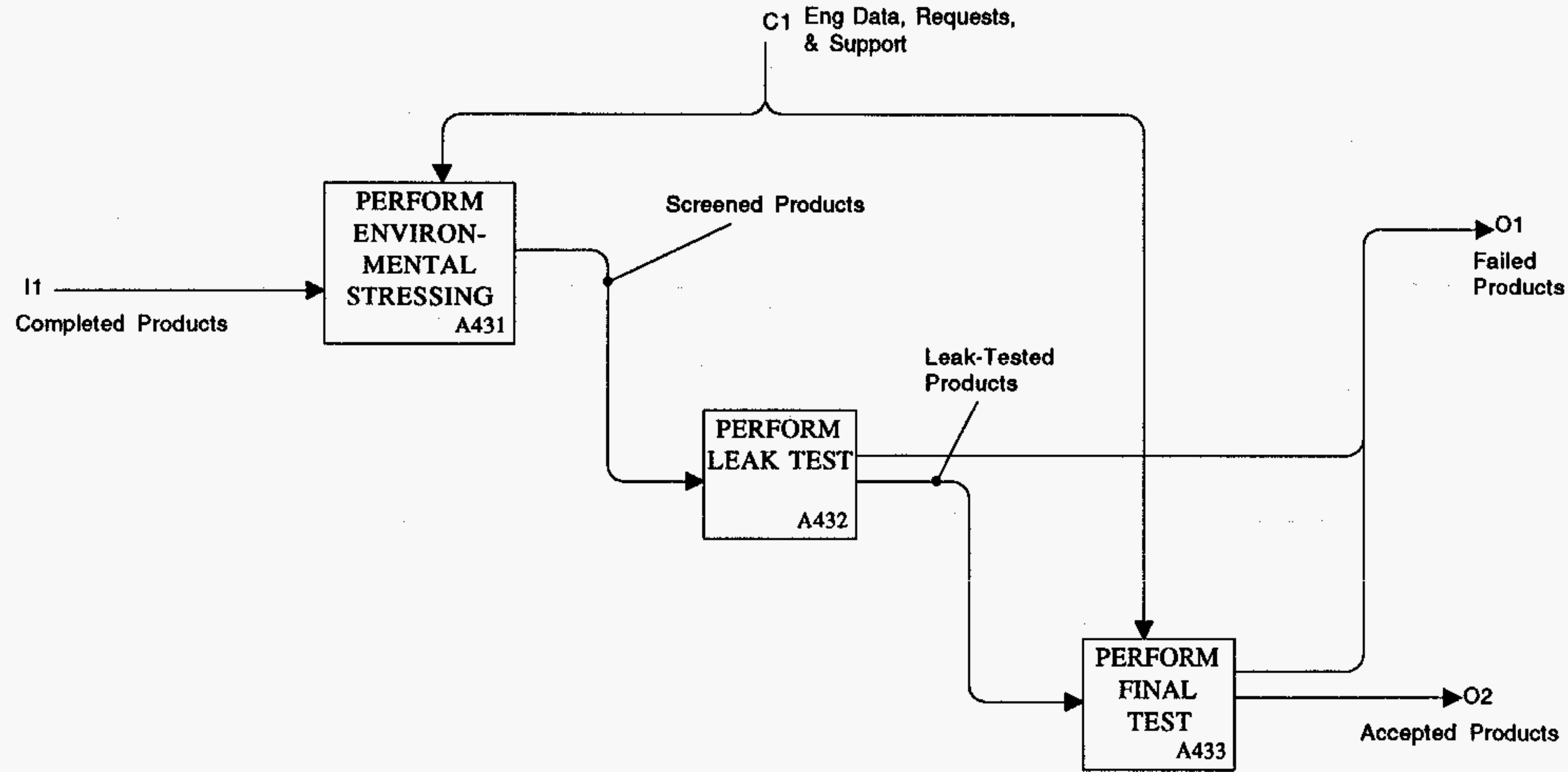
3. Wire Bond Connections: The packaged device is electrically connected to its elements and casing by wire bonds. The product is first plasma cleaned to remove any residue. Second the wire bond connections are made between substrate pads and components Third the wire bond connections are made between substrate and the leads of the case. These wire bond operations can be performed by an automatic wire bonder.

4. Test, Inspect, & Rework: After the hybrid has been wire bonded, the device must be bond pull tested usually with an automatic wire bond pull tester. The device is then electrically tested to insure that all connections have been made and that the device is functioning correctly. The product is visually inspected to insure quality. Products which are rejected during testing or inspection are reworked. Also, some active trimming and tuning of a device is done as a result of electrical test.

5. Seal Devices: Products are hermetically sealed by welding a cover onto the device. Once the device is sealed, leads preform test/inspection is performed.

NODE:	TITLE: A 42 T, ASSEMBLE DEVICES	NUMBER:

USED AT:	AUTHOR: Charles Azu PROJECT: NOSC Hybrid MicroCIM NOTES: 1 2 3 4 5 6 7 8 9 10	DATE: 10/31/91 REV: 3/17/94	WORKING DRAFT RECOMMENDED PUBLICATION	READER	DATE	CONTEXT:



NODE: A 43	TITLE: TEST PRODUCT	NUMBER:

USED AT:	AUTHOR: Charles Azu PROJECT: NOSC Hybrid MicroCIN NOTES: 1 2 3 4 5 6 7 8 9 10	DATE: 2/13/92 REV: 4/8/92 3/17/94		WORKING DRAFT RECOMMENDED PUBLICATION	READER DATE	CONTEXT: Top

A43 Test Product

During this activity the products are passed through environmental screening which include nitrogen bake, temperature cycling, and constant acceleration. The second part of the activity involves tes the product for leaks by pressurizing the device and monitoring for leaks. The final test is burn-in which removes infant mortality. Products which successfully pass these tests are moved to shippir Those that fail are sent to rework. Some manufacturers perform final electrical test then leak test. Manufacturers do this, despite the violation of Mil-Spec, because LEPs often fail more in electrical test than during leak test.

NODE:	TITLE: A 43 T, TEST PRODUCT	NUMBER:

Blank Page

APPENDIX B

APPLICATION PROTOCOL USAGE GUIDE

This section will include suggested algorithms for extracting various types of data from an IGES file which conforms to this AP. Eleven processes have been identified as candidates for algorithms of which six are included in this AP version:

1. Drill-Hole / Cutouts
2. Netlists
3. BOM / LOM
4. Schematic Graphics
5. Physical Artwork
6. Automatic Insertion
7. Simulation
8. ATE Data
9. Automatic Inspection Data
10. Process Extractor
11. Wire Wrap
12. Create/Update Die Component
13. Wirebond Extract

1. ALGORITHM GUIDELINES

To effectively represent such extractions algorithms a form of "pseudo-code" is employed; much like that used in many of the computer science journals (e.g., Communications of the ACM). The format has few rules, allowing the algorithm writer considerable flexibility. The algorithms are included as an assist to the programmer and are not intended to present requirements for code.

1.1. Algorithm Format

Each algorithm will be called a "process" and will have the following major structure:

```
PROCESS name;
   statement#1;
   statement#2;
   statement#3;
   statement#n;
ENDPROCESS;
```

Within each process different statements will be defined to perform operations of iteration, flow control, and decision-making. Each statement will now be described in detail.

1.1.1. STATEMENT GROUPING.

A statement is defined to be a single line operation or group of single lines enclosed in a BEGIN/ END group. So whenever we refer to "statement" we either mean a one-line statement or a group of many lines. All statements should be terminated with a semicolon (;). Each line of the statement can be any english-like or programming-like description of an operation. Some examples include:

a. Single Statement Example (four single statements):

```
get next record from IGES directory section;
i=i+1;
write out error message;
j=k/25.8;
```

b. Group Statement Example:

```
BEGIN
   get next record from IGES directory section;
   i=i+1;
   write out error message;
   j=k/25.8;
END;
```

1.1.2. COMMENTS.

An algorithm can contain comment lines. A comment line can appear anywhere in the algorithm and must be enclosed in a { } pair as follows:

```
j=k/25.8; {This is a comment}
{This is also a comment}

get next record;
write directory record;
```

1.1.3. IF/THEN/ELSE.

```
IF some_condition THEN
   statement#1;
   ELSE
      statement#2;
```

The following forms are also valid if each statement is in fact only one line:

```
IF some_condition THEN statement#1;
   ELSE statement#2;
IF some_condition THEN statement#1;
```

1.1.4. LABELS.

Any statement can have a symbolic label to be used as the address of a GOTO.

```
label:statement#1;
   statement#2;
   GOTO label;
```

1.1.5. GOTO.

The GOTO statement has the form:

```
GOTO label;
```

1.1.6. CASE.

A multi-way decision "CASE" statement has the following form:

```
CASE (decision variable or expression);
   (value#1):statement#1;
```

```
        (value#2):statement#2;
        DEFAULT :statement#3;
    ENDCASE;
```

If the decision has a matching "value" then ONLY the set of statements for that value are executed. The DEFAULT is not required, but if present the default statements are executed if no values match.

1.1.7. FOR.

A for-loop construct is provided and it has the form:

```
    FOR index_variable=start_value, end_condition
        statement#1;
```

-- or --

```
    FOR index_variable+start_variable, end_condition
        BEGIN
            statement#1;
            statement#2;
            statement#3;
        END;
```

1.1.8. WHILE.

A while-loop construct is provided and it has the form:

```
    WHILE (condition)
        DO statement#1;
```

1.1.9. EXIT.

The exit statement causes the algorithm to terminate immediately:

```
    EXIT;
```

1.2. Guidelines For Writing IGES Extract Algorithms

When writing an extraction algorithm we suggest one follow a few guidelines to make the algorithm both easy to read and understand.

• Always indent nested constructs

• Try to keep each process short; one to three pages at the most.

• Keep the algorithms at a very abstract level. We assume the reader is very familiar with the IGES format and has access to the specification for which the algorithm was written.

2. IGES EXTRACTION ALGORITHMS

2.1. ALGORITHM: Drill Hole

```
IGES VERSION : 3.0

{-----------------------------------------------------------}
{ Date: July, 1986 Author: Bob Corey (LNL)                   }
{ IGES Version : 3.0                                        }}
{ Description : This algorithm will extract the data         }
{ necessary to punch an NC drill tape. The output will       }
{ have the following format:                                 }
{  DRILL SIZE   X_LOCATION  Y-LOCATION                       }
{                                                            }
{ NOTE : This algorithm will require the addition of many }
{ detailed steps to parse entities to be fully functional.}
{-----------------------------------------------------------}
PROCESS Drill_Hole;

{ ***** TO BE COMPLETED ******* }

ENDPROCESS;  {Drill_Hole}
```

2.2. ALGORITHM: Netlist Extraction

IGES Version : 3.0 and above

Purpose: Extract a signal netlist from a LEP AP IGES file.

Description: A netlist lists every signal on the PCB, and lists every connection point on each signal. Connection points include: component through hole pins, vias, and test points.

Information: Signal names, component reference designators, component pins, component pin names, vias, and test points.

```
Reset      U1-4, U11-4, U11-22, J1-6
Xcvr_123   R1-1, U11-6, J1-7, TP123
```

These lines show two different signals, "Reset" and "Xcvr_123". Each signal has four connection points, one of which is a test point.

Note: input IGES file MUST be CALS Class III

Entities used

420 Network Subfigure Instance: the physical component on the PCB

132 Connect Point
402.18 Flow Associativity Property.

DE information: None of the DE information is used.

```
/* -------------------------------------------------------------- */
PROCESS Netlist_extract
 {
/* -- 1 --- Initialize ------------------------------------------- */

CHARACTER_STRING iges_file_name = "iges_file.igs";
INOUT_FILE_HANDLE  infile;

/* -- 2 --- Open and read the part ------------------------------ */

infile = OPEN_FILE (File_name = iges_file_name,  Access = "read");

       /* set aside some to store all of the entities in the file */
MAKE_LIST ( List_name = "all_entities");
       /* and read them from the input file */
all_entities = READ_FILE(infile);

/* -- 3 --- Select all Connect Points ------------------------- */

      /* Set aside some memory to store all of the connect points.
       *  IGES type 132 are Connect Points */
MAKE_LIST ( List_name = "connect_points");

      /*  Look at each entity in the list of all of the entities.
       *  For each entity, if the entity is a connect_point,
       * add it to the list of connect_points.
       */
FOR_EACH  entity  IN_LIST_OF   all_entities
  {
    IF_THE ( entity.type  IS_EQUAL_TO  132 )
     /* IGES 132: connect_point */
    {
      ADD_LIST( List_name = "connect_points", entity);
    }
  }

/* -- 4 --- Select all the Net Names ----------------------------*/
/*   The  Flow Associativity Entity (402 form 18) represents a
     signal network */
```

```
MAKE_LIST ( List_name = "nets");

                /* now, look at every entity in our list of
                 * all of the entities. If the entity we're
                 * looking at is a net_name, add it to the
                 * list of nets
                 */
FOR_EACH  entity  IN_LIST_OF   all_entities
  {
    IF_THE ( entity.type  IS_EQUAL_TO  402  AND  entity.form
             IS_EQUAL_TO  18)
    {
       ADD_LIST( List_name = "nets", entity);
    }
  }

/* -- 5 --- Process the nets ----------------------------------- */

FOR_EACH  signal  IN_LIST_OF   nets
  {
                /* print out the signal name */
     PRINT(signal.parm_data.name1);

                /* Look at every connect point being pointed
                 * to by this particular signal. Fields
                 * "cptrn" of the parameter data record contain
                 * this list.
                 */
     FOR_EACH  connect_point  IN_LIST_OF   signal.parm_data.cptrn
      {
                /* Field 7 "cid" of the Connect Points parameter
                 * data record contains the name of the connect
                 * point. Field 14 "psfi" of the Connect Points
                 * parameter data record contains a pointer to
                 * the component it belongs to. We must get the
                 * reference designator of the owning component.
                 * Note: if field 14 "psfi" is empty, then this
                 * connect point is most likely a via or a test
                 * point.
                 */
         owner_component = connect_point.parm_data.psfi;
         IF( owner_component IS  null)
          {
```

```
            IF( StrLen(connect_point.parm_data.cid) IS_LESS_THAN  1)
                CONTINUE;
              ELSE
                PRINT(",",connect_point.parm_data.cid," ");
          }
        ELSE
                    /* print the reference designator */
            PRINT(",",owner_component.parm_data.prd,"-
                  ",connect_point.parm_data.cid);
      }
  }
/* ------------------------------------------------------------ */
 }
```

2.3. ALGORITHM: Bill of Materials

IGES VERSION : 3.0 and above

/* --- *

Algorithm for: Bill of Materials

Purpose: Extract a Bill of Materials (BOM) from a LEP AP IGES file.

Description: The bill of materials lists each part_type that is used on the board. For example, the following lines ae taken from a standard bom.

Information: Internal part number, description derived: count of parts for each part number. optional: component reference designators, item_number

ITEM_NUMBER	COMPANY PART NO.	COUNT	DESCRIPTION	REFERENCE
13	2-60001-2360	1	Resistor, 36	R46
14	2-60002-3383	12	Resistor, 383	R1 R6 R8 R9 R21 R22 R31 R36 R44 R45 R47 R48
15	2-60002-3590	8	Resistor, 590	R2 R7 R10 R11 R20 R23 R32 R37

These lines show that there are 3 part_types, all are resistors.

There will only be one part on the board of the part_type "Resistor, 36," (part_number "2-60001-2360"), and that will be R46. There will be 8 parts on the board of part_type "Resistor, 590" (part_number "2-60001-2360")

part_type: (aka call out) this is the identifyer for each type of part.
This can be thought of as a part number.
(These are all of the Network Subfigure Defns 320)
component: This is the physical part stuck on the board.
(These are all of the Network Subfigure Instn 420)

Implementation notes.

Note: input IGES file MUST be CALS Class III

Entities used

320 Network Subfigure Definition: the "library" component
420 Network Subfigure Instance: the physical component on the PCB
406.9 Component part number

DE information: None of the DE information is used in the BOM.

Note: when storing the component definitions (320) and instances (420), any properties they reference must be stored with them. For example, each component instance must also store it's part number, it's description, and any other relevant information.

Geometric processing is not required.

Backpointer processing is not required.

Subfigure processing (reading, storing, and retreival) is not required.

```
/* ------------------------------------------------------------------------ */
/*         PROCESS bill_of_materials                                         */
/*                                                                          */
/*                                                                          */
/* -- 1 --- Initialize ---------------------------------------------------- */
       CHARACTER_STRING iges_file_name = "iges_file.igs";
       INOUT_FILE_HANDLE  infile;
       INTEGER part_count;
       INTEGER component_count;
       INTEGER item_number;

part_count = 0;
/* part_count will count the number of individual part_types that are read in*/
component_count = 0;
/* component_count counts the number of components placed on the board for
         each part_type */
```

```
item_number = 0;
/* ---------------------------------------------------------------------------- */

/* -- 2 --- Open and read the part ---------------------------------------------- */

  infile = OPEN_FILE (File_name = iges_file_name,  Access = "read");

/* set aside some to store all of the entities in the file                        */
  MAKE_LIST ( List_name = "all_entities");
/* and read them from the input file                                              */
all_entities = READ_FILE(infile);
/* ---------------------------------------------------------------------------- */



/* -- 3 --- Select placed components -------------------------------------------- */

/* Set aside some memory to store all of the actual components.                   */
/*     IGES type 420s network subfigure instance                                  */

MAKE_LIST ( List_name = "components");

/*  Look at each entity in the list of all of the entities.                       */
/*        For each entity, if the entity is a component, add it to the            */
/*        list of components                                                      */
FOR_EACH  entity  IN_LIST_OF   all_entities
{
      IF_THE ( entity.type  IS_EQUAL_TO  420 )  /* IGES 420: component   */
       {
          part_count = part_count + 1;
          ADD_LIST( List_name = "components", entity);
        }
      }
/* ---------------------------------------------------------------------------- */



/* -- 4 --- Select the part_tpyes ------------------------------------------------ */
/*                                                                                */
/*   Get a list of all the part_types.  These are component definitions.          */

            /* set aside some memory to store all of the part_types
             *  IGES type 320s  network subfigure definition */
  MAKE_LIST ( List_name = "part_type_entities");

            /* now, look at every entity in our list of all of the entities.
             *  If the entity we're looking at is a part_type, add it to the
             *  list of part_types */
  FOR_EACH  entity  IN_LIST_OF   all_entities
```

```
  {
    IF_THE ( entity.type  IS_EQUAL_TO  320 )
     {
         part_count = part_count + 1;
         ADD_LIST( List_name = "part_type_entities", entity);
     }
   }
/* ----------------------------------------------------------------------- */



/* -- 5 --- Process the part_types ------------- -------------------------- */
/*                                                                          */
/*  Process the information.  The goal is to relate each part_type to all of
 *   the actual components of that part_type.
 *
 *  First, start up a new list which will hold all of the actual components
 *   for each part_type.
 *  Go thru each part_type in the list of   part_type_entities, and
 *   and cycle through the list of components.  If a component is an
 *   instance of our current part_type, add it to our list of
 *   components of this part type.
 *   When we've come to the end of the list of components, we are done with
 *   the task of matching, and it is time to print out this list.         */


FOR_EACH  part_type_entity  IN_LIST_OF   part_type_entitie
  {
             /* start up a new list */
    MAKE_LIST ( List_name = "store_match");
    item_number = item_number + 1;


/* . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . */
/* look at every component on the board (in list "components")             */
FOR_EACH  component_entity  IN_LIST_OF   components
  {
   /* The component_entity is a 420.  The first field in the parameter
    *   data section points back to the Network Subfigure Defintion
    *   of this component.
    *
    *  If the DE pointer of this 420 points to the DE number of the
    *   current Network Subfigure Defintion, then store this           */
   IF( component_entity.Parameter_data.DE   IS_EQUAL_TO
                     part_type_entity.DeNumber )
   {
       component_count = component_count + 1;
        ADD_LIST( List_name = "store_match", component_entity);
    }
  }
```

```
/* . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . */

  /*   print out bill of materials
   *
   *  13   2-60001-2360   1    Resistor, 36        R46
   *  14   2-60002-3383  12    Resistor, 383       R1 R6 R8 R9 R21
   *                                               R22 R31 R36 R44
   */

  PRINT_OUT( item_number );

      /* entity  406.9  holds the part number */
  FOR_EACH  entity  IN_LIST_OF   part_type_entity.Parameter_data.Props
    IF entity.Form IS_EQUAL_TO  9) PRINT_OUT(entity.Parameter_data.IPN );

  PRINT_OUT( component_count );
  PRINT_OUT( part_type_entity.NAME);

   /* print out the ref des of each of the components in the
    *  store_match list, they are 420's.  paramater field 9,
    *  called PRD is the  Primay Reference Designator        */
  FOR_EACH entity   IN_LIST_OF   store_match
    PRINT_OUT( entity.Parameter_data.PRD );

  PRINT_OUT( 'cr/lf' );
/* . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . */

CLEAR_LIST (List_name = "store_match");
   component_count = 0
 }
/* ----- End of Process the part_types ------------------------------------ */
```

2.4 ALGORITHM: Schematic (Graphics)

```
IGES VERSION  : 3.0


{-------------------------------------------------------}
{ Date  : June 23, 1986          Author : Bob Corey (LLNL)}
{ IGES Version : 3.0                                     }
{Description : This algorithm will extract all the data   }
{ necessary to represent the graphical part of an        }
{ electronics schematic.  The algorithm will extract the  }
{ graphic entities from the IGES file and "draw" them to  }
{ some unspecified output device (i.e., pen plotter,      }
{ raster display, etc.).  Each entity as well as the      }
{ attributes of COLOR and LINE WIDTH will be output.      }
{                                                         }
```

```
{ NOTE : This algorithm will require the addition of many }
{        detailed steps to parse entities to be fully     }
{        functional.                                       }
{---------------------------------------------------------}
PROCESS schematic_graphics;
   { start by looking for subfigure instances (420s) that }
   { have a type flag of LOGICAL.                         }
  WHILE (we still have unprocessed subfigure instances, 420s)
   DO BEGIN
      IF (parameter section entity type = 420 AND
         type flag = LOGICAL) THEN
         BEGIN

         { ****TO BE COMPLETED **** }

         END:
   END:
ENDPROCESS;  {schematic_graphics }

/* ------------------------------------------------------------ */
```

2.5. ALGORITHM: Physical Artwork

(***** UNDER CONSTRUCTION *****)

2.6. ALGORITHM: Automatic Insertion;

an Algorithm for: Pick/Place, and adhesive dispensing.

Purpose: Extract data that can be downloaded directly to surface mount component handling manufacturing equipment.

Description: Automated component handling manufacturing equipment such as pick and place may be driven using electronic data generated by CAE/CAD/CAM machines. This eliminates user entry error and drastically cuts time to market.

Information:

Required: component technology type, component part number, component reference designator, component placement side, component placement origin.

Derived: Component rotation is derived from a transformation matrix, if the component has one. Most component rotations will be orthogonal.

```
/* -----------------------------------------------------------*
 REF_DES    X       Y  Rotation  Part_Number
   BAT1  1.600  9.400  000
```

```
U4    0.100  0.600  000

Note: input IGES file MUST be CALS Class III
   Entities used:                               (Selection criteria)
  320     Network Subfigure Definition                      *
  420     Network Subfigure Instance  (ref des and x,y coords)
  124     Transformation Matrix
  406.9   Component part number
  406.27  Component technology type   (smt/thru_hole)     *
  406.27  Component placement side    (top/bottom)        *
 *406.27  Component placement origin  (x,y coords)


DE information: Transformation Matrix on Network Subfigure
                Instance.

Note: when storing the component definitions (320) and instances
      (420), any properties they reference must be stored with
      them. For example, each component instance must also store
      it's part number, it's description, and any other relevant
      information.

Note: The use of 406 form 27 entities reflect a defined and
      reccommended method of representing information
      consistently.  This is a new method, as of August 1994, and
      will not be found in files before this date. The pick/place
      algorithm is still valid, however, it is left to the user to
      define the method for defining technology, placement side,
      and origin.

Note: Component placement origin is used only if the placement
      origin is different than the the instance origin. For
      example, a resistor that is .5 inch long may be defined with
      pin 1 as it's origin, thus pin 2 is .5 inch away, and the
      center of the component is .25 inches from the origin. When
      the component is taken from the library and put on the
      board, pin 1 will be put the specified x,y coordinate.
      However, the pick and place machine may define it's
      placement origin as being in the center of the part,thus the
      placement origin does not coincide with the instance
      origin.
* ------------------------------------------------------------- */

/* ------------------------------------------------------------- */
```

```
PROCESS pick_and_place
 {
  The pick and place algorithm breaks down into a few simple steps.
  First, make a list of all of the appropriate part_types.  They
  must be surface mount, and they must be on the side we are
  extracting data for. Second, make a list of all of the
  appropriate placed components. They too must be surface mount,
  and they must be on the side we are extracting data for.

  Note: for demonstration purpose, we will assume we are looking
        for all surface mount components on the bottom side.
/* ------------------------------------------------------------- */

/* -- 1 --- Initialize ----------------------------------------- */

CHARACTER_STRING iges_file_name = "iges_file.igs";
INOUT_FILE_HANDLE  infile;
INTEGER part_count;
INTEGER component_count;
INTEGER item_number;

part_count = 0;      /* part_count will count the number of
                      * individual part_types that are read in  */
component_count = 0;/* component_count counts the number
                      * of components placed on the board for
                      * each part_type.                        */
item_number = 0;
/* ----------------  ------------------------------------------- */


/* -- 2 --- Open and read the part ----------------------------- */

infile = OPEN_FILE (File_name = iges_file_name,  Access = "read");

                /* set aside some to store all of the entities
                 * in the file                              */
MAKE_LIST ( List_name = "all_ents");
                /* and read them from the input file */
all_ents = READ_FILE(infile);
 /* ------------------------------------------------------------- */


/* -- 3 --- Make a list of the appropriate part_types ----------- */
```

```
                /* set aside some memory to store all of the
                 * part_types IGES type 320s network subfigure
                 * definition
                 */
MAKE_LIST ( List_name = "part_type_entities");

                /* Now, look at every entity in the list of
                 * all entities. If the entity is not a
                 * part_type (320), then see if it satisfies the
                 * selection criteria.  If it does, then add it
                 * to the list of part_types.
                 */
FOR_EACH  entity  IN_LIST_OF   all_entities
 {
   IF_THE ( entity.type  IS_EQUAL_TO  320 )
   {
     IF_THE ( entity.property.technology_type  IS_EQUAL_TO  "smt")
     {
       part_count = part_count + 1;
       ADD_LIST( List_name = "part_type_entities", entity);
     }
   }
 }
/* ------------------------------------------------------------- */
/* -- 4 --- Make a list of the appropriate placed components --- */
                /* Set aside some memory to store all of the
                 * actual   components. (IGES type 420s network
                 * subfigure instance)
                 */
MAKE_LIST ( List_name = "components");
                /*   Look at each entity in the list of all of
                 * the entities. For each entity, if the entity
                 * is a component, add it to the list of
                 * components.
                 *   The  technology_type  property is required
                 * on the part_type, but not on the component. We
                 * get the pointer from the component to the
                 * part_type in the first index field of the
                 * component, labeled "DE".
                 */
FOR_EACH  entity  IN_LIST_OF   all_entities

 {
  IF_THE ( entity.type  IS_EQUAL_TO  420 )
```

```
                /* IGES 420: component */
  {
    IF_THE ( entity.property.placement_side  IS_EQUAL_TO  "bottom")
    {
       part_type = entity.parm_data.DE;
         IF_THE ( part_type.property.technology_type
                IS_EQUAL_TO  "smt")
         {
           part_count = part_count + 1;
           ADD_LIST( List_name = "components", entity);
         }
     }
   }
 }
/* --------------------------------------------------------------- */

/* -- 5 --- Generate  the output ------------------------------- */
                /* There is an optional step that may be
                 * performed at this point (if desired) if
                 * there is any type of sorting, such as by
                 * part_type, physical board location, height,
                 * etc. There will be no special ordering in
                 * this example.
                 */

PRINT( top_of_form );

FOR_EACH  entity  IN_LIST_OF   components
  {
                /* print out the reference designator */
    PRINT(entity.parm_data.PRD);

                /* print out the x,y coordinates */
    x = entity.parm_data.X;
    y = entity.parm_data.Y;
                /*  If there is a  component placement location
                 * property attached to the component, it is
                 * because the placement origin of the component
                 * is different than the physical origin of the
                 * component.
                 */
    IF_THE ( entity.property.component_placement_origin  EXISTS)
      {
         x = Modify__Instance_to_Placement__Origin(ent);
```

```
        y = Modify__Instance_to_Placement__Origin(ent);
     }
    PRINT(x,y);

         /* print out the rotation */
    rotation = 0.0;
              /*  If there is a transformation matrix
               * (Type 124), calculate the rotational
               * component, and convert it to degrees
               * between 1 and 359. Note that rotation
               * should be around the placement origin,
               * that's why it is done first.
               */
    IF_THE ( entity.transformation_matrix_pointer_field
              IS_GREATER_THAN  0)
       rotation = CONVERT__Transformation_Matrix__to_Degrees
              (entity);

    PRINT(rotation);

              /* print out the part number */
    PRINT(entity.property.part_number);

  }
/* ------------------------------------------------------------ */
```

2.7. ALGORITHM: Simulation

(***** UNDER CONSTRUCTION *****)

2.8. ALGORITHM: ATE Data

(***** UNDER CONSTRUCTION *****)

2.9. ALGORITHM: Automatic Inspection Data

(***** UNDER CONSTRUCTION *****)

2.10. ALGORITHM: Process Extractor

(***** UNDER CONSTRUCTION *****)

2.11 ALGORITHM: Wire Wrap

Significant error reduction has been demonstrated using CAE layout and automated or semi-automated wire wrapping[1] as compared to manual layout, as most of the errors occur in transcribing the pin locations and their signal names. The IGES file requirements for a wire wrap

design are described in Section 6.6.2.3.4. The wire wrap extraction process described below follows the CAE layout of a design. A general sequencing and sorting algorithm—usually part of a machine process file preparation—is then given.

EXTRACT: The Netlist is extracted from the physical LEP file as defined in Section 2.2., adding the X and Y location of each "connect_point IN_LIST_OF signal.parm_data.cptrn" to each "pin number" entry.

Sort the pin/coordinate/reference_designator information as follows:

/*

DATE : September 12, 1994 Author: Gary Panzer

Description: This algorithm will take X-Y location data from each point of a net (defined as an electrical connection between two or more points), determine the best point to point interconnect sequence to achieve a minimal distance for each net, determine the quickest wirewrap sequence, and provide wirewrap operator support services.

PARAMETERS:

Nets (A real array of X, Y, Connection ID locations, and wire lengths)

Board_Columns (An array of character that describe the physical rows of the wire wrap card. i.e., A, BB, X)

Board_Rows (An array of character that describe the physical rows of the wire wrap card. i.e., 1, 99, AA)

Count_of_Nodes (Total number of nodes to connect)

Column_Distance (A real array of the X distance for each column from location 0,0. Note: columns are not always on equal spacings)

Row_Distance (A real array of the Y distance for each row from location 0,0. Note: rows are not always on equal spacings)

*/

BEGIN SORT

/* First determine if there are multiple wires needed for each net. For multiple wires, each net is listed sequentially in the array along with its sequence number. For single wire nets, the next net will have the sequence number 1. */

1. The layout process and the CAE library parts descriptions, which preceeds the extract and sort algorithms defined in this Appendix, is found in; Parks, C., et al "Digital Circuit Analysis, Documentation, Wire Wrap, and Test Generation from a Single Data Extract on CADDS III," Fourth Annual International ComputerVision Users Conference Proceedings, 9/82, Prime-ComputerVision Corporation, Bedford, MA 01730.

```
BEGIN
     FOR i=1 to Count_of_Nodes
          Find the total number of nodes in each net (string);
          For each net, use every location as a starting point
            and find the distance to all other points;
          Find the best (minimal) total wire length using each
            node as a starting location;
          Store this information into an array which includes
            X & Y location plus wire length;
     ENDIF
```

/* Now sort for the best sequence of wirewrap operations, do layer 1 first and start with the longest wires first - the short wires hold down the long ones, and start work from location 0,0 and move diagonally out. */

```
     FOR i=1 to Count_of_Nodes
          SORT(Nets, Wire_Layer_Sequence.increasing,
            Wire_Length.decreasing,
            Distance_Away.increasing);
     END
```

/* Now connect human readable wirewrap info to machine operations. This includes ASCII X-Y locations and wire bin number. Print out trace information. Also count the number of wires needed for each wire length. Typically the smallest wire that can be bought is 3.5 inches and the largest is 20 inches, with steps of 0.5 inches) */

```
     FOR a= 3.5 to 20.0 step 0.5
          PRINT(Count(Nets.Wire,a));
     END

     FOR i=1 to Count_of_Nodes
          Human_Data.X(i) = Board_Columns(i);
          Human_Data.Y(i) = Board_Rows(i);
          Human_Data.Bin_#(i) = (Round(Wire_Length-3.5,
            0.5));
          PRINT(Human_Data(i),Nets.X,Nets.Y,Wire_Length,
            Layer, Sequence_Number)
     END
```

/* Now control wirewrap machine operations */

```
     For i=1 to Count_of_Nodes
          OUTPUT(Nets.X,Nets.Y,Human_Data.Wire,
            Sequence_Number);
```

```
        END
    END
```
----

12. Create/Update Die Component

(***** UNDER CONSTRUCTION *****)

13. Wirebond Extract

(***** UNDER CONSTRUCTION *****)

APPENDIX C

TECHNICAL DISCUSSIONS

C.1. A Primary Parent for the ARM

View 4-1 contains 7 independent entities, as opposed to most models which have a single independent entity. The multiple independent entity situation was recognized in the early STEP modeling, with one of these models proposing the Enterprise as the single independent parent. The MicroCIM team has discussed adding such an entity, however did not do so because the scope defined for the model does not require such a model refinement. It was recognized that each ARM independent entity employs "ID" in the key. Each (name)-ID is defined in the Section 3 Glossary to include data which uniquely identifies the source or owning enterprise. Thus an Enterprise parent is implied.

C.2. Netlist Information

The Net entity on ARM View 4-3 has been recognized as *derived* information. Typically the Net consists of a Signal Name with the Reference Designators and Pin Numbers to be associated (i.e., connected) with that signal. The source of Net information is the functional design of the circuitry as presented by a schematic. The Schematic in any particular product design may have been "back-annotated" with net list information from the package design. The Schematic entity itself has not been decomposed into its fundamental entities in this version of the ARM.

C.3. Multiple Planning for a Product

The association of Process information with the Product model presents a problem when "re-planning" takes place. This would occur if the product is not completely processed to a single process definition (e.g., second sourcing or reprocurement). The team view was that the model provides a place for a particular user to record his planning information. Should the Product data be relocated, the site-specific planning would not be populated, and may be re-defined at the next cite.

C.4. Producibility Feedback

The use of "producibility" data in a concurrent engineering requires much more attention. As yet the team has discovered no commonly agreed to model of how things like "cost penalties for non-standard designs" or "more easily manufactured trade-offs" are to be placed into a database which engineering could access during layout. These sorts of dispositions seem to take place during "producibility reviews" which requires the knowledge of experienced operations people. As a starting point for capturing such information, several kinds of Rules have been included in the ARM.

C.5. Quality

Much of the Application Activity Model (AAM) has been reworded and modified to reflect the use of Total Quality Management (TQM) and Quality Function Deployment (QFD) being practiced. The team expects new attributes will be required for some of the ARM entities.

DISTRIBUTION:

1 NOSC
Manufacturing & Computer Integrated
Engr. Technology Branch, Code 936
Attn: Charles C. Azu, Jr.
271 Catalina Blvd.
San Diego, CA 92152-5000

1 Naval Air Warfare Center
Aircraft Division, Indianapolis
Attn: Garland A. Borden II
6000 E. 21st. Street CODE 961
Indianapolis, IN 46219-2189

2 Carderock DIV, NSLOC
Attn: Jack Brainin, Code 1251
Lisa V. Deeds, Code 2031
Bethesda, MD 20084-5000

2 AUTODESK, INC.
Attn: Edward Clapp
Joel S. Peterson
2320 Marinship Way
Sausalito, CA 94965

1 General Dynamics
Electric Boat Division
Attn: Dr. Burton F. Gischner
Attn: Gregory F. Morea
75 Eastern Point Road
Groton, CT 06340-4989

1 Grumman Data Systems
Attn: William B. Gruttke
5300 International Boulevard
Suite 105
North Charleston, SC 29418

1 WizWorx
Software Imagineering
Attn: Dennette A. Harrod
60 Thoreau Street, Suite 320
Concord, MA 01742

1 Mentor Graphics Corp.
Attn: Dave Kehmeier
Marketing Development Specialist
1940 Zanker Road
San Jose, CA 95112-4216

1 Valid Logic Systems
Attn: Randy Lawson
Two Omni Way
Chelmsford, MA 01824

1 IGES Data Analysis, Inc.
Attn: Bill Loye
6933 Lamotte Drive
Centerville, MN 55038

1 Naval Air Warfare Center
Attn: J. Carter McCrary, C2955
Attn: Bernadette Namauleg, C25512
Naval Warfare Center
China Lake, CA 93555-6001

1 International TechneGroup Inc.
Attn: Jeff Mawhirter
5303 DuPont Circle
Milford, OH 45150

1 Hughes Aircraft Co.
Product Engineering Laboratory
Attn: Paul A. Nelson
Bldg. 600, MS C131
P. O. Box 3310
Fullerton, CA 92634-3310

1 NIST
Automated Electronics Mfg. Programs
Center for Electronics and E. E.
Attn: Curtis H. Parks
Bldg. 220 Rm. B-344
Gaithersburg, MD 20899

1 CAD Software Consultant
Specializing in NASCAD, IGES
Attn: Alan N. Peltzman
431 Blossom Tree Drive
Annapolis, MD 21401

1 Dimensions International
FAA Technical Center
Attn: Joseph L. Preston, Jr.
CALS Specialist
Atlantic City Intl. Airport
MS AOS-530
Atlantic City, NJ 08405

1 Dazix, an Intergraph Company
Attn: Anthony Prince
Electronics Integration
One Madison Industrial Park
Mail Stop LR23B4
Huntsville, AL 35894-0001

1 Caterpillar Inc.
Attn: Ed Reid
41 TCA 359
Box 1875
Peoria, IL 61656

1 Honeywell Inc.
Attn: John Russell
MN51-1320
Commercial Flight Systems
8840 Evergreen Blvd
Minneapolis, MN 55433-6040

1 Arthur D. Little, Inc.
Attn: James Sneed
5300 International Boulevard
North Charleston, SC 29418-6936

1 Raytheon
Attn: Patrick Toomey
Manager, CAE Microelectronics Dept.
Microwave and Power Tube Division
Industrial Components Operation
465 Center Street
Quincy, MA 02169

12 NIST
Attn: Ellen Trager
IPO Executive Assistant
B220 Rm A226
Gaithersburg, MD 20899

1 IGES Data Analysis, Inc.
Attn: William E. Turcotte II
2001 N. Janice Avenue
Melrose Park, IL 60160

106 DOE/OSTI for distribution on UC-706

1	MS	0953	William E. Alzheimer, 2900
1		0955	Tom J. Young, 2901
1		0803	John F. Jones, 13200
1		0661	Michael H. Pendley, 13212
1		0661	Philip R. Kennicott, 13212
1		0319	Robert F. Rieden, 2604
1		9018	Central technical Files, 8523-2
5		0899	Technical Library, 13414
1		0619	Technical Publications, 13416
2		0100	Document Processing, 7613-2 For DOE/OSTI