NISTIR 90-4234

A PLANNING MODEL FOR UNIFYING INFORMATION MODELING LANGUAGES FOR PRODUCT DATA EXCHANGE SPECIFICATION (PDES)

Joan Tyler

U.S. DEPARTMENT OF COMMERCE
National Institute of Standards and Technology
Center for Manufacturing Engineering
Gaithersburg, MD 20899

NATIONAL PDES TESTBED

U.S. DEPARTMENT OF COMMERCE
Robert A. Mosbacher, Secretary
Lee Mercer, Deputy Under Secretary for Technology
NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY
Raymond G. Kammer, Acting Director

NIST

NISTIR 90-4234

A PLANNING MODEL FOR UNIFYING INFORMATION MODELING LANGUAGES FOR PRODUCT DATA EXCHANGE SPECIFICATION (PDES)

Joan Tyler

U.S. DEPARTMENT OF COMMERCE
National Institute of Standards
and Technology
Center for Manufacturing Engineering
Gaithersburg, MD 20899

September 1989

Issued January 1990

U.S. DEPARTMENT OF COMMERCE
Robert A. Mosbacher, Secretary

Lee Mercer, Deputy Under Secretary
for Technology

NATIONAL INSTITUTE OF STANDARDS
AND TECHNOLOGY
Raymond G. Kammer, Acting Director

A PLANNING MODEL FOR UNIFYING INFORMATION MODELING LANGUAGES FOR PRODUCT DATA EXCHANGE SPECIFICATION (PDES)

Joan Tyler

NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY
CENTER FOR MANUFACTURING ENGINEERING
FACTORY AUTOMATION SYSTEMS DIVISION

ABSTRACT

A voluntary group called the IGES/PDES Organization is leading the Product Data Exchange Specification (PDES) project. The goal is to create a specification focused on exchanging product models with sufficient information content as to be interpretable directly by advanced computer aided design and computer aided manufacturing (CAD/CAM) application programs. There are many technical issues that must be resolved if PDES is to become a standard. This paper describes a mechanism for resolving issues surrounding the use of multiple information modeling paradigms. Several diverse modeling languages have been used to define and formalize the information needed to fully represent the multiple disciplines needed for a complete product model. This prompted a committee of the IGES/PDES Organization to undertake a research project to unify these divergent paradigms. Some intermediate results of this project are discussed including the Planning Model, which is intended to serve as a baseline for future development of a neutral repository for storing semantics.

1. INTRODUCTION

The need for standardizing the exchange of product data has been nationally [13] and internationally recognized. Subcommittee 4 of the International Standards Organization (ISO) Technical Committee 184 (TC184), passed a resolution describing the need for such a standard (Resolution 1, ISO TC184/SC4, July 1984). This need has been reaffirmed on a number of occasions.

The initial development of PDES has required an enormous amount of technical effort. There have been hundreds of contributors from a wide spectrum of industry, academia, and government. The national and international standards organizations share the common goal of having a single standard. The first working draft of the specification has been submitted to ISO TC184/SC4 [21] and has been registered as Draft Proposal 10303.

It is the intent of the PDES project to fully support the needs for a complete product model as required by sophisticated applications in many areas of use. For instance, in the area of manufacturing, these application would include generative process planning systems, CAD-directed inspection and automated NC data generation. In the area of maintenance planning the applications may enable analyses of assembly/disassembly. This type of information coupled with the geometric data will allow PDES to communicate a complete product model. Achieving a quality specification which is useful across industry boundaries is of the utmost concern to the developers.

Information and data modeling capabilities have been and continue to be the fundamental technology used in developing the requirements for the applications. Developers need the flexibility to chose a modeling language to represent their domain. The use of several divergent modeling languages prompted the PDES Dictionary/Methodology Committee of the IGES/PDES Organization to develop a strategy for unifying the constructs which support these modeling languages.

The Dictionary/Methodology Committee has constructed a planning model to serve as a guide for merging the modeling languages.

The planning model (Figure 1) provides the scope for three distinct models. These three models provide a foundation for solving the issues of multiple language usage. The more explicit detailed levels associated with the interaction, decomposition, and application of these three models will be the subject of later papers.

A planning model gives us the ability to traverse through different levels of detail to get to a particular view. A Planning Model is analogous to consulting a map before driving across the country. A traveler would consult a country map before going to the detail of state and local maps to plan his trip in increasing detail.

The committee's planning model will be referenced throughout this paper as the PDES Unification Meta Model or as the "PUMM."

2. PUMM DESIGN:

2.1 Overview and Approach

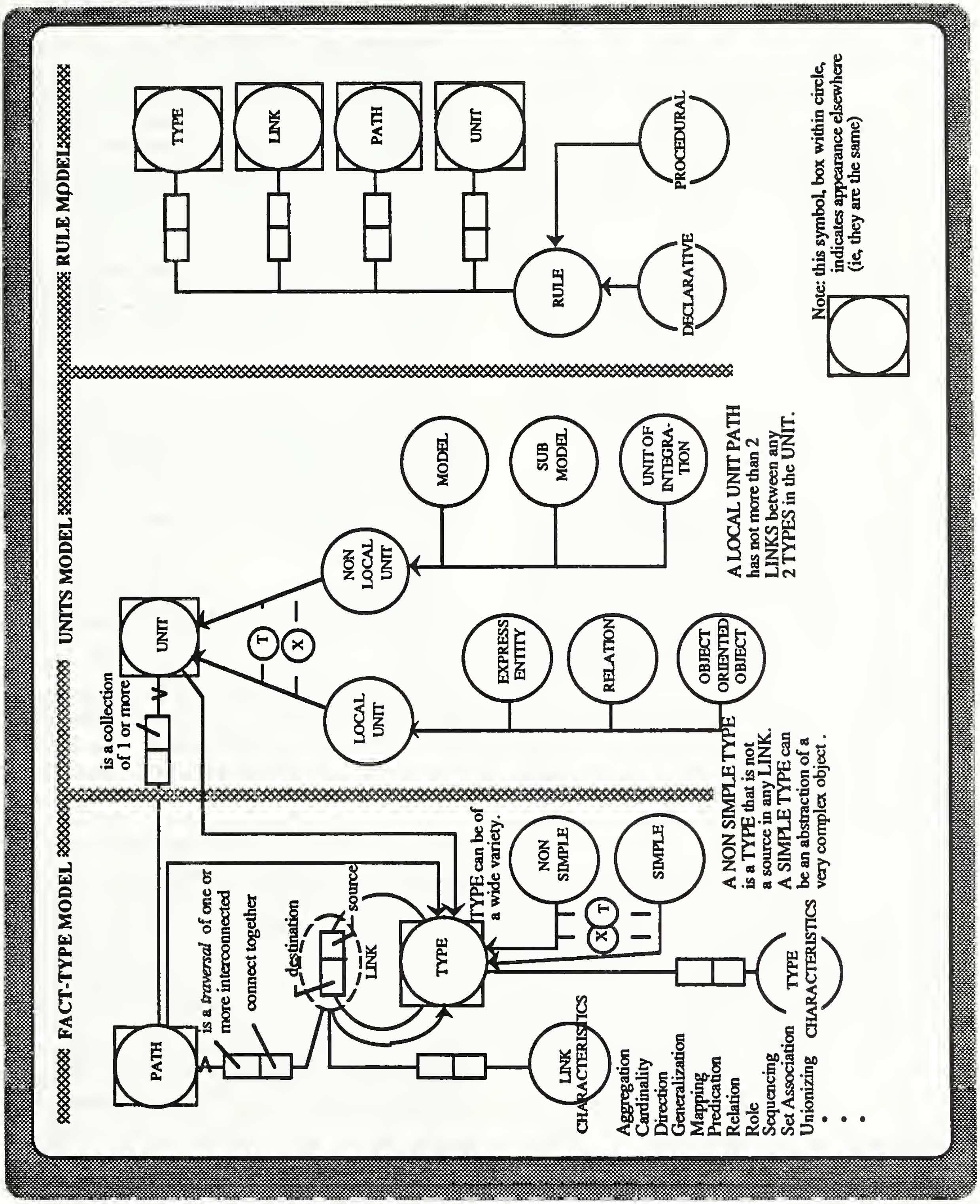
The Dictionary/Methodology Committee is building a core of primitives which can support many diverse languages. One of the key success factors is in the definition of the mappings or filters between these constructs and the modeling languages. The strategy for unification puts common modeling primitives in the PUMM and dissimilar constructs in the maps or filters. These constructs provide organization but are not carriers of meaning (semantics).

The PUMM will contain a core set of semantic primitives collected from a study of over twenty semantic modeling languages. The first three modeling languages targeted are EXPRESS, IDEF1X and NIAM. EXPRESS is a computer-sensible language that was conceived as a specification to define the relationships between data objects and the PDES physical file format. See *data abstraction* discussion in 3.1. IDEF1X was developed by the USAF ICAM project. Constructs consist of entity, attribute and relationship. Although IDEF1X supports modularity at the entity level (a form of data abstraction), it does not hide the important inter-entity linkages. See *semantic networks* discussion in 3.2. NIAM is used by PDES developers as a binary modeling language. Constructs consist of object and the fact or relationship with focus on the analysis of natural language sentences. See *semantic networks* discussion in 3.2. These are the most popular modeling languages currently in use by the PDES community. The Dictionary/Methodology Committee is designing a single abstract syntax for all three languages with the construction of the PUMM. It should be noted that the PUMM primitives will support a dictionary of "meaning," not of record or file structures.

The PUMM is divided into three major pieces: one defines the underlying *semantic network* (Fact Type Model), one defines the value-adding structural overlays (Units Model) for *data abstration*, one defines a procedural and declarative constraint language to constrain the semantic network and structural overlays (Rule Model).

The PUMM does not attempt to unify all known modeling approaches. It concentrates on unifying *semantic network* and *data abstraction* approaches because these seem to be the dominant approaches used within the PDES community. It allows both approaches and capitalizes on the commonalities of IDEF1X, EXPRESS and NIAM. It forces artifactual differences between the languages (sometimes only presentation differences) to be placed in presentation mappings.

The PUMM is a hybrid language which allows the building of semantic networks made from TYPEs and LINKs. The TYPEs represent a similar collection of objects. An object can be anything -- *abstract* or *concrete* -- *simple* or *complex*. A LINK can denote any kind of relationship. The PUMM will provide a diverse link annotation capability to allow a wide range of relationship expressions. Development of the PUMM will allow for precise modeling of successor semantics. A "little of everything" approach will be used initially to declare an unlimited number of distinguished relationships like "membership," "is-a," etc.



Planning Model

Figure 1

On top of the *semantic network* one may superimpose modular structure by defining objects called *UNITs*. This permits a modeling team to discover the underlying network complexity of the subject matter first and then add modularity. If the modeling team discovers the modular structure first, it can add detail later. In a sense this is what PDES is doing now with the EXPRESS entity building step (the modularization step).

The PUMM is quite flexible in that it scales up to model the relationships between large *UNITs* such as an entire PDES model. The PUMM does not prevent PUMM meta objects themselves from being instances of *TYPE*. The PUMM does not impose artificial boundaries between PUMM meta objects and universe of discourse[1] objects.

The heart of the PUMM is the *semantic network* complemented with a starter set of *LINK* types, providing users of the semantic repository an epistimological headstart. This idea was borrowed from the Brachman theory on epistimological nets [5]. The idea is to start off with a rich meta model instead of a lean one, to avoid having other PDES developers worry about inventing new connotations. Some relationships discovered in the PDES world appear to defy categorization. The possibility of using lambda [15] *LINKs* is well-founded. By categorizing *LINK*, we provide an *integration tool* as well as provide the ability to locate models that have similar *LINK* types. This could very well be the essence of the meaning to the term "*deep structure*." This is a term borrowed from the linquists but is used in a different sense. We use the term in the sense that given multiple languages each having its own superficial structure, there is another structure from which the surface structures of each of the languages can be reformulated. The ability to link fact types between different languages, perform *deep structure analysis* on these fact types will provide developers with a tool that can be used for various levels of integration.

[1]A formal description of a collection of abstract concepts. Usually captured in an information model.

Evolving clear and consensus definitions for the distinguished *LINK* types is critical to development of the PUMM *semantic network.* Some of the *LINK* types are classical and can be defined right out of a set theory or logic textbook; others are not classical but are important to the task. It is hoped that the list of distinguished *LINK* types will not suffer the enumeration problem of continually adding to the list. Some "rule" will have to be established as to what constitutes a distinguished *LINK* from a non-distinguished *LINK*. This is a dictionary meta-meta rule which can also be stored in the Meta Model.

3. PUMM Definition

The planning model for the PUMM (see Figure 1) is functionally decomposed into three models: *Fact Type Model, Units Model,* and *Rule Model.* Each of these models will be described along with its function.

MODEL	FUNCTION
Fact Type Model	Semantic Nets
Units Model	Data Abstraction
Rule Model	Constraints

The modular approach to development concentrates the design efforts of each model individually and then joins them to achieve full functionality. Additional models may also be required.

An attempt has been made to make the PUMM as general as possible in order to permit adapting to change. Thus, we are aiming at a programmable model. We believe that a high-productivity software development environment will permit such a programmable model. It is possible that additional models will be developed for interconnectivity. This implies a need for a Meta-Meta model (or a model's dictionary) in order to formally specify extensibility of the model. With this kind of programmable model environment, a family of dictionaries could be rapidly prototyped. Investigation is underway to use the Information Resource Dictionary System[3] (IRDS) [3] as a mechanism for this dictionary capability.

[3] ANSI Standard X3.138-1988.

3.1 Fact Type Model

This model defines the components of a *semantic network.* The Planning Model in Figure 1 provides the framework for the Fact Type Model. The underlying details for this model are not outlined in this paper. The major components are: *TYPE, LINK, LINK CHARACTERISTIC* and *TYPE CHARACTERISTIC.*

A *TYPE* is a collection of similar objects. An object can be concrete or abstract, simple or complex, lexical or non-lexical, etc. There is no limit on what a *TYPE* can be. A *TYPE* could even be a dictionary meta object. A *TYPE* could be an undefined package of bits, a binary image of a photograph, a paragraph of text, a single letter, an entire ship model, a vertex, a procedure, a rule, or a subnet. A *TYPE* can be a very artificial collection of objects based on some rule of similarity, i.e., "all pink tennis shoes with no shoe laces."

A *LINK* in the Fact Type Model represents a collection of similar relationships between objects contained within several *TYPEs.* There may be any number of *LINKs* between *TYPEs* in order to cover the many different perspectives of a relationship between objects. This *LINK* is also a binary relationship. Zimmerman represents the notion of PREDICATE as a *LINK* in his L+ language [29]. *LINK* also has the perspectives of *connotation, denotation, signification* and *reference.*[2] The *denotation* of a *LINK* is a set of binary relationships between objects. The *connotation* of *LINK* defines the membership rules that can be used to test the cartesian product of the members of the two *TYPEs* associated with the *LINK.*

The relation of *TYPEs* to each other is through directed *LINKs.* The PUMM allows for a diversity of *LINK* types. The important point that the PUMM is trying to communicate is that it must allow for a large number of meta *TYPEs* and for as many kinds of *LINK* types as possible. This a very different approach but it does serve to provide a basis for a diversity of modeling constructs.

After studying many modeling languages, a group of the same predicates begin to reappear. From these the following *LINK* connotations or "set of distinguished predicates" have been considered in designing the PUMM:

- Aggregation
- Cardinality
- Direction
- Generalization
- Mapping (function)
- Predication (description)
- Relation (interaction)
- Role
- Set Associations (inclusion,intersection....)
- Sequencing

Some *LINKs* will connect *TYPEs* that are concepts (intensional). These intensional links are not concerned with the population of the *TYPEs*; an example is a *subtype LINK.* Other *LINKs* connect together what the *TYPEs* denote; an example would be a *subset LINK.* Another kind of *LINK* could be the denoting *LINK.* This would link an intensional type (a concept) to an extensional type (a set) [21].

A *PATH* is a traversal through one or more connected links. The *LINK* is the smallest unit of organization in the PUMM. The *PATH* is the next highest level of organization (like an assembly level). *PATHs* represent products of binary relations, ie, *aR1b* connected to *bR2c* gives the path *aR1bR2c.* *RULEs* can also constrain *PATHs.*

3.2 Unit Model

A *UNIT* is a collection of *PATHs.* The *PATHs* could be disjoint or connected. Several *PATHs* could originate from the same *TYPE.* Branches of *PATHs* could be related to each other through *RULE.* A distinction is made between *local* and *non-local PATHs.* A *local PATH* never has more than two *LINKs* between any two types in the unit. A *non-local PATH* does not have this restriction.

---

[2] Signification - is used for naming concepts.
Denotation - is used for declaration of real world objects compatible with a concept.
Reference - is used for identification of particular real world objects.
Connotation - is used in the description of concepts.

Many kinds of record structures look like collections of local *PATHs*. Note that the *UNIT* does not have to be a conceptually relevant meta model object. Its main purpose is to facilitate the description of structures of information.

There is no limit put on the size of a *UNIT*. The entire PDES Integrated Product Information Model (IPIM)[1] could be considered as one *UNIT*. A single EXPRESS *entity* is a *UNIT*. An object-oriented "object" is a *UNIT*. The Electrical Functional Model[2] could be considered a *UNIT*. When a *UNIT* is used as a TYPE, all of its details are hidden, thus giving rise to the PUMM's own method for describing *data abstraction*. A *LINK* can be used to connect together two *UNITS*. It's interesting to note that just because two *UNITS* are connected together does not mean that there cannot be lower level connections between the two *TYPES*. The *LINK* could be used to indicate that one *UNIT* is a specialization on another *UNIT*.

3.3 Rule Model

The PUMM uses both declarative and procedural RULES to define constraints and rule-based knowledge that refers to TYPES and LINKS. RULES will be used to preserve the *intent* and *context* captured by the model in the original modeling language. Application of *RULES* is a subject of research and is still in the development stage. The application of *RULES* will be the topic of a seperate paper.

4. MODELING LANGUAGE RESEARCH

There are three modeling languages used extensively by developers within the PDES community. Our research has been directed at discovering the primitive modeling concepts that underlie these modeling languages.

Two approaches emerge from this research: the ***semantic network*** approach and the ***data abstraction*** approach. The IDEF1X and NIAM languages fall into the ***semantic network*** approach and the EXPRESS language falls into the ***data abstraction*** approach. .

PDES modelers have the freedom to work with languages which fall into either approach. Each approach has its strengths; therefore, a robust repository of semantic primitives should accomodate both approaches. The PUMM framework has been constructed with this in mind.

For additional information on semantic network models and related issues refer to descritions in [1], [2], [5], [6], [9], [12], [14], [18], [20], [22], and [25].

4.1 Data Abstraction

Data abstraction is the process of making abstract assemblies. By this we mean that a data object is constructed from other objects using various construction techniques such as aggregation, generalization and member association. Once the object has been constructed, the details can be abstracted and looked at only when needed. ***Data abstraction*** is identical to the process of modularizing a large system and is a natural system engineering approach. ***Data abstraction*** results in hierarchical assemblies of data objects just as real physical assemblies have assembly levels. ***Data abstraction*** leads to a modular solution and encourages top-down design. The modularization would be done almost on an arbitrary basis. Modularity is a tool which deals with how the structure of an object can make the attainment of some purpose easier.

The dominant relationship between data objects within ***data abstraction*** is "component of." Other relationships may be difficult to convey beyond "component of," without invoking a procedural language.

It is important to note that ***data abstraction*** contains nothing which inherently prevents it from being used to model a subject area that is highly interconnected. The typical approach is to model critical relationships as assemblies in their own right. This is known as *relationship objectification*. However, the ***data abstraction*** language must use a

---

[1] IPIM refers to Section 4 of ISO Draft Proposal 10303 [21].

[2] This is an Application Specific Model developed by members of the Electrical Committee of the IGES/PDES Organization.

modular and hierarchical approach to presenting the model, and often the meaning gets lost in the presentation.

4.2 Semantic Networks

Semantic networks provide graphical representations and the relationship types of **is-a, is-instance-of,** and **is-part-of**. The graphs are formed of data items connected by edges which are used for construction and for item placement within categories according to similar properties. One of the strengths of ***semantic networks*** is the ability to represent **is-a** (supertype/subtype) relationships with expanded usage to form unions of existing types. Another extension includes the idea of *subset* and *generalization* [26]. A *subset* **is-a** relationship arises when one type is contained in another. The *generalization* **is-a** relationship arises by partitioning its subtypes. Stated another way: subtypes can be disjoint but when put together they form the supertype.

The ***semantic network*** approach allows the designer to model subject matter as a collection of interconnected nodes and links. The links represent relationships between the nodes. Most ***semantic network*** languages allow extensive annotation of the link to develop the meaning of the relationship as much as possible.

Using ***semantic networks*** is quite different from using ***data abstraction.*** It allows the domain expert to defer modularization. ***Semantic networks*** usually allow for a more diverse relationship between objects. This eases the modeling process because the designer is not thinking about modularity while defining basic objects and relationships.

5. CONCLUSION

In summary, every attempt is being made to base the PDES Unification Meta Model on traditional linguistic, mathematical and ontological foundations. The PUMM is a representation mechanism for symbolizing these classical foundations in a very structured environment. It has not been necessary to invent anything new, our strategy is to apply sound theory to a new and innovative application. The ultimate goal of this work is to provide a neutral repository for storing semantics, not for the purpose of creating a new modeling language.

ACKNOWLEDGEMENTS

The author wishes to thank John Zimmerman of Allied Signal, Anne Williams of McDonnell Douglas, Bill Danner of NIST, Augusto Nieva of Digital Equipment Corporation and Roger Stumps of Boeing for their expert advice and valuable critique while developing this paper.

BIBLIOGRAPHY

[1] Bachman, C.; The Role Concept in Data Models. In Proceedings of the 3rd International Conference on Very Large Data Bases (Tokyo, Oct.6-8). IEEE, New York, 1977.

[2] Batini, C.,Lenzerini, M., and Navathe, S; A Comparative Analysis of Methodologies for Database Schema Integration. ACM Computing Surveys, 18, 4, 1986.

[3] Bloom, H., Furlani, C, Mitchell, M., Tyer, J. and Jefferson, D.; Information Resource Dictionary System: An Integration Mechanism for Product Data Exchange Specification, NISTIR 88-3862, 1988.

[4] Borgida, A.; Features of Languages for The Development of Information Systems at the Conceptual Level, IEEE Software 2, 1 Jan., 1985.

[5] Brachman, R.; On the Epistemological Status of semantic networks, Findler, 3-50, 1979.

[6] Brodie, M.L.; On the Development of Data models. In On Conceptual Modeling, M. L. Brodie, J. Mylopoulos and J. W. Schmidt, Eds. Springer-Verlag, New York, 1984.

[7] Bruce, Betram; Case systems for Natural Language, Artificial Intelligence 6, 327-360, 1975.

[8] Cardelli, L. and Wegner, P.; On Understanding Types, Data Abstraction, and Polymorphism. Computing Surveys, Dec., 1985.

[9] Carnap, R.; Meaning and Necessity-A Study in Semantics and Modal Logic,

University of Chicago Press, Chicago,1956.
[10] Chen, P.P.; The Entity-Relationship Model-Toward a Unified View of Data. ACM trans. Database Syst.1, Mar.,1976.
[11] Church, Alonzo; The Calculi of Lambda Conversion, Princeton University Press, Princeton, New Jersey, 1941.
[12] Guttag,J.; Abstract Data Types and the Development of Data Structures. Commun. ACM 20, 6 June, 1977.
[13] Henghold, W., Schumaker, J., and Baker, L.; Considerations for the Development and Implementation of PDES within a Government Environment, AFWAL-TR-89-8009,1989
[14] Hull, R. and King, R. ; Semantic Database Modeling: Survey, Applications, and Research Issues, ACM Computing Surveys, Vol. 19, 3.,1987.
[15] Hornstein, N.; Logic as Grammar, MIT Press, Cambridge, Mass., 1984.
[16] Mark, L., ;What is the Binary Relationship Approach? In Davis (Ed.) Entity-Relationship Approach to Software Engineering, North-Holland, Amsterdam, 1983.
[17] Meersman, R.; RIDL: A Query System as Support for Information Analysis, ECODO 32, 1981.
[18] Nijssen, G.M. and Halpin, T. A.; Conceptual Schema and Relational Database Design: A Fact Oriented Approach, Prentice Hall, Englewood Cliffs, N.J., 1989.
[19] Russell, B.; Introduction to Mathematical Philosophy, George Allen and Unwin LTD., London, 1919.
[20] Senko, M.E.; Conceptual Schemas, Abstract Data Structures, Enterprise Descriptions,Int. Comp. Symposium, ACM, Liege, Belgium 1977.
[21] Smith, B. and Rinadot, G.; Product Data Exchange Specification: First Working Draft, NISTIR 88-4004, National Technical Information Service (NTIS), Order PB 89-144794, 1988.
[22] Sowa,J.; Conceptual Structures: Information Processing in Mind and Machine, Addison-Wesley Publishing, Reading, Mass., 1984.
[23] Stachowitz, R.; A Formal Framework for Describing and Classifying Semantic Data Models. Information Systems, Vol 10, 1., 1985.
[24] Stoy, J.; Denotational Semantics: The Scott-Strachey Approach to Programming Language Theory, MIT Press, Cambridge, Mass., 1977.
[25] Su, S.Y.W., Krishnamurthy, v. and Lam, H.; An Object -oriented Semantic Association Model (OSAM*). In AI in Industrial Engineering and Manufacturing: Theoretical Issues and Applications, S. Kumara and R. L. Kashyap, Eds. American Institute of Industrial Engineers, Norcross, Ga., 1988.
[26] Tarski, A.; On the Concept of Logical Consequence:Logic, Semantics, Metamathematics, Clarendon, Oxford, 409-420, 1936.
[27] Teorey, T., Yang, D. and Fry, J.; A Logical Design Methodology for Relational Databases Using the Extended Entity-Relationship Model, Computing Surveys, Vol. 18, 2, 1986.
[28] Tsichritzis, D and Lochovsky, F; Data Models, Prentice-Hall, Englewood Cliffs, N.J., 1982.
[29] Zimmerman, J.; L+: An Abstract Modeling Language, unpublished manuscript, internal IGES/PDES Organization document, 1986.

NBS-114A (REV. 2-8C)

U.S. DEPT. OF COMM. BIBLIOGRAPHIC DATA SHEET (See instructions)	1. PUBLICATION OR REPORT NO. NISTIR 90-4234	2. Performing Organ. Report No.	3. Publication Date January 1990

4. TITLE AND SUBTITLE

A Planning Model for Unifying Information Modeling Languages for Product Data Exchange Specification (PDES)

5. AUTHOR(S)

Tyler, Joan E.

6. PERFORMING ORGANIZATION *(If joint or other than NBS, see instructions)*

NATIONAL BUREAU OF STANDARDS
DEPARTMENT OF COMMERCE
WASHINGTON, D.C. 20234

7. Contract/Grant No.

8. Type of Report & Period Covered

9. SPONSORING ORGANIZATION NAME AND COMPLETE ADDRESS *(Street, City, State, ZIP)*

10. SUPPLEMENTARY NOTES

☐ Document describes a computer program; SF-185, FIPS Software Summary, is attached.

11. ABSTRACT *(A 200-word or less factual summary of most significant information. If document includes a significant bibliography or literature survey, mention it here)*

A voluntary group called the IGES/PDES Organization is leading the Product Data Exchange Specification (PDES) project. The goal is to create a specification focused on exchanging product models with sufficient information content as to be interpretable directly by advanced computer aided design and computer aided manufacturing (CAD/CAM) application programs. There are many technical issues that must be resolved if PDES is to become a standard. This paper describes a mechanism for resolving issues surrounding the use of multiple information modeling paradigms. Several diverse modeling languages have been used to define and formalize the information needed to fully represent the multiple disciplines needed for a complete product model. This prompted a committee of the IGES/PDES Organization to undertake a research project to unify these divergent paradigms. Some intermediate results of this project are discussed including the Planning Model, which is intended to serve as a baseline for future development of a neutral repository for storing semantics.

12. KEY WORDS *(Six to twelve entries; alphabetical order; capitalize only proper names; and separate key words by semicolons)*

Data Abstraction, Data Exchange, Information Modeling, Integration, Mappings, Meta Models, Modeling, Modeling Languages, Product Data, Semantic Nets

13. AVAILABILITY

☒ Unlimited
☐ For Official Distribution. Do Not Release to NTIS
☐ Order From Superintendent of Documents, U.S. Government Printing Office, Washington, D.C. 20402.
☒ Order From National Technical Information Service (NTIS), Springfield, VA. 22161

14. NO. OF PRINTED PAGES

11

15. Price

A02