STEP PRODUCT INFORMATION MODELS IN AGILE MANUFACTURING

Mangesh P. Bhandarkar, Rakesh Nagi[1]

Department of Industrial Engineering, 342 Bell Hall

State University of New York at Buffalo, NY 14260

Telephone: 1-(716)-645-2357 x 2103

Fax: 1-(716)-645-3302

E-mail: nagi@eng.buffalo.edu

---

[1] To whom correspondence should be addressed

ABSTRACT

This paper focuses on the product information needed to support the emerging trend of agile manufacturing. We address the issue of consistent product information being available during the design and manufacturing stages of the products life cycle. We have described a means of converting product design data from Initial Graphic Exchange Specification (IGES) format into Standard for the Exchange of Product Model Data (STEP) format. We are developing translators which converts product design information in the form of an IGES drawing into STEP drafting specifics and another translator which extracts out form feature information for product manufacturing. We have also developed computer code to help us visualize the STEP drafting file. Important elements in this implementation include EXPRESS, C++ and ST-Developer.

Keywords: IGES, STEP, Application Protocol, Form Features, EXPRESS, ST-Developer.

1 INTRODUCTION

The past few years have seen a change in manufacturing strategies and technologies. The markets have become more and more fragmented and competitiveness has increased. For a manufacturing enterprise to be competitive and thrive in such a dynamic environment it must be capable of adjusting rapidly to the market. This has lead to the emergence of a new manufacturing paradigm called Agile Manufacturing, where multiple companies cooperate under a flexible virtual enterprise structure. The ideas is to make products of better quality in the least amount of time and at the best possible price, to result in the highest degree of customer service/satisfaction.

Agile Manufacturing makes use of modern information technology to form virtual enterprises. Partners must maintain a high amount of communication and data exchange between themselves for the success of the virtual enterprise. Data exchange does not mean just verbal communication, but the companies must exchange complete information about their products, all the way from design, manufacturing to inspection and shipping. This information should be available to each relevant partner over the entire life cycle of the product. This pressing need for information exchange has lead to the development of the STandard for the Exchange of Product model data (STEP) standard (ISO 10303). STEP aims at providing product information in a platform independent format, which renders it suitable for easy data exchange, over the various phases of the product life cycle.

Development of a new standard has introduced a need for translators which convert data files from the legacy format to the new format. This paper focuses on the development of a translator which converts IGES drawing information into STEP drafting information, for designing needs, and then extracting out form feature information from the IGES file into STEP form features, for manufacturing needs.

The paper is organized in the follows. We first present a brief overview of IGES and STEP followed by the literature review. We then describe the current problem and the work done in terms of coding and testing of the ‘design translator’. We then describe the work that is currently in progress for the ‘form feature’ translator. Finally we conclude with a discussion and proposed further work.

2 BACKGROUND

This section gives a brief overview of the IGES standard and how data is arranged in an IGES file. We have given a few reasons as to why IGES did not gain much popularity. We have also taken a look at the STEP standard and it benefits over IGES.

2.1 Initial Graphic Exchange Specification

IGES covers a wide range of application areas including electrical, plant design, as well as mechanical applications. IGES provides a standard format by which the user can transfer the data from one CAD system to another. A standard format like IGES requires two levels of processing:

1. A pre-processor takes the CAD data as described in the system specific format and converts it into the IGES format.
2. At the other end the post-processor converts data back from IGES to a format that can be understood by the CAD system.

IGES has information structures or entities to be used for digital representation and communication of product definition data. Data is represented in an application-independent format that enables exchange of product definition across the various CAD systems. The fundamental unit of data in an IGES file is the entity. The product is described in terms of geometry entities and non-geometry entities.

Geometry entities represent the definition of the physical shape and include points, curves, surfaces, solids and relations which are collections of similarly structured entities. Non-geometry entities provide a viewing perspective of which a planar drawing is composed, by providing annotation and dimensioning appropriate to the drawing. Non-geometry entities further serve to provide specific attributes or characteristics for individual entities. Non-geometry entities include view, drawing, dimensions, text, notation, witness lines and leaders.

An IGES file consists of five or six sections. Each section is identified in the IGES file by a code printed in the output file, as shown in the Table 1. IGES data can be represented in either ASCII or binary format. The code identifying the section is printed in column 73 of each line. Columns 74-80 show the sequence number of each line within the section. The sequence number is a seven digit string starting from 0000001, increasing sequentially according to the number of lines in the section.

Table 1 *IGES file structure and codes*

Section	**Code**
Flag Section (optional)	B/C
Start Section	S
Global Section	G
Directory Entry Section	D
Parameter Data Section	P
Terminate Section	T

Flag section The Flag Section of the file is used to indicate whether the file is in the Binary format or in the Compressed ASCII format. This section is optional.

Start Section The Start Section of the file is designed to provide a human-readable prologue to the file. There must be at least one start record. All records in this section have the letter S in column 73 and sequence number in column 74 through 80. Columns 1-73 need not be formatted in any special way except that the ASCII character set must be used.

Global Section The Global Section contains information describing the pre-processor and information needed by the post-processor to handle the file. All records in this section have a G in column 73. The parameters in the global section contain information that describes the parameter delimiter and record delimiter characters, file name, system ID, units, date and time of the file generation, file version number, etc.

Directory Entry Section The Directory Entry Section has one directory entry for each entry in the file. The directory entry for each entity is fixed in size and contains twenty fields of eight characters each, spread over two consecutive eighty character lines. The data in each section is right justified and consists of either pointer constants or integer constants. The purpose of this section is to provide an index for the file and to contain general attribute information for each entity. Each entity has an identification number which describes the type of the entity and hence how the corresponding data is represented in the parameter data section. The 1st and 11th fields contain the identifying number. The second field contains a pointer to the first line of the parameter data record for the entity. Other fields contains information line style, color, level of drawing on which the drawing resides, view data, entity label, etc. Column 73 contains the letter D for this section and columns 74-80 contain the sequence number within the section.

Parameter Data Section The Parameter Data Section of the file contains the data associated with each entity. The first field always contains the entity type number. The remainder of the record contains data to define the particular entity. Column 73 contains the code letter P for this section, and columns 74-80 contain the sequence number.

Terminate Section This section consists of only one line which terminates the file. The terminate section provides summary information about the number of lines used in each section of the whole file. Each field in the Terminate record contains a section identifier which is left-justified in the field, and the last sequence number used in that section.

IGES has been used in the past but the files are difficult to understand because of the complicated way in which the data is arranged. If there is an error in the IGES file, it is very difficult to find the mistake and correct it. Errors may arise due to changes made in the file. CAD vendors have developed their own flavors for the IGES files which may not be consistent between various vendors.

IGES does not provide data that may be relevant to applications other than drawing or 3D modeling. For example the designer may be interested only in the drawing or the geometry of the part to be manufactured but the process planner would be more interested in the form features and the manufacturability of the part and the inspection department may be interested in the tolerance information for the part. Each of the departments would share the same IGES file and it is left to the interpretation of the individual to derive the relevant information. All these reasons lead to the development of the STEP standard.

2.2 STandard for the Exchange of Product Model Data

STEP provides representation of product information along with the necessary mechanisms and definitions to enable product data to be exchanged. The exchange is among different computer systems and environments associated with the complete product life cycle including design, manufacturing, inspection, maintenance and disposal. The information generated about a product during these processes is used for many purposes.

STEP is organized as a series of parts, each published separately. These parts fall into one of the following series: description methods, integrated resources and application protocols. STEP uses a formal information modeling language, EXPRESS (Schenck 1994), to specify the product information to be represented. The use of a formal language enables precision and consistency of representation and facilitates development of implementations. STEP uses application protocols (AP's) to specify the representation of product information for one or more applications. For example the schema representing product shape and assembly information is AP203 also known as "Configuration Controlled Design," the schema used for representation drafting information is called AP202 "Associative Draughting" and the schema for process planning using form feature information is called AP224 "Mechanical product definition for process planning using form features."

STEP enables all people contributing to the design, manufacturing, marketing and supply of a product and its components to contribute to, to access, and to share information. STEP aims at eliminating the concept of "islands of automation." STEP also aims at uniting manufacturing efforts among corporate partners, distant subsidiaries and suppliers across diverse computer environments. It is intended to fit in completely with the new emerging paradigm of virtual enterprises and agile manufacturing.

STEP clearly emerges as a superior standard as compared to IGES. It takes care of issues that deal with the diversified engineering applications and covers security issues. IGES has been used in the past and now with the popularity of the STEP standard, there is a need for a translation service between the legacy format and the new standard.

Having now described the IGES and the STEP standard and the need for a translator between these two standards, the next section gives a literature review about existing work done in this area.

3 LITERATURE REVIEW

The authors are not aware of any large scale efforts to write a translator from IGES to STEP. There may however be certain small scale translators available which have been written for particular applications. Basu and Kumar (1995) have written a written a translator for finite element analysis. The translator extracts mesh entities (nodes and elements) from an IGES file and a simple FORTRAN 77 program converts these into relevant entities for Finite Element/Boundary element analysis.

Ssemakula and Satsangi (1990) have done some work with IGES and PDES (PDES was the precursor to STEP in the US) for computer integrated process planning. However they have not written any direct translation between IGES and PDES.

Feature extraction has been a widely researched field with a lot of work already been done in this area. We will present below some of the important research done in this area. A more elaborate discussion of the research in this field can found in Salomons *et al.* (1993).

Shah and Mathew (1991) have looked at STEP form-feature information. There have been some problems during the exchange of the feature data due to the lack of position/location information, multiple representation of a single feature, representation of standard popular profiles in tedious data structures, and non unique mapping of features between different systems.

Joshi and Chang (1988) have done pioneering work in this field. They use the concept of attributed adjacency graph (AAG) to recognize machining features from a boundary representation (B-Rep) of the solid. Su *et al.* (1995) have presented an Euler operator-based feature encoding module, for developing an integrated form-feature hybrid design and manufacturing system. Gu *et al.* (1995) have developed a form feature recognition and operation selection procedure using B-Rep solid model to construct an attributed adjacency matrix (AAM).

Qamhiyah *et al.* (1996) have presented a boundary-based procedure for the sequential extraction of form-features of objects with planar surfaces, from CAD model. The authors have developed form-feature classes into which the part is sequential classified until all the features are resolved. Gavankar and Henderson (1989) have developed a graph-based feature extraction technique to identify protrusions and depressions from boundary models. Meeran and Pratt (1993) have developed an experimental form feature recognizer which uses 2D drawings as inputs. The system is designed to meet the process-planning requirements of simple prismatic machined parts.

Liu *et al.* (1996) and Ling and Narayan (1996) have designed form-feature extraction schemes. Both the aforementioned works take the IGES format as input to the feature extractor, and classify all the features present on the workpiece whose information is in the file. The extracted form-features are stored in an object-oriented format.

All the above mentioned research have developed feature extraction schemes using various techniques and using several varied forms of input however the extracted features do not render themselves easily to data exchange as warranted by the needs of Agile Manufacturing. The features are also not stored in a standard format. The feature information would be lost and would have to be regenerated, this could lead to errors in the features extracted and other activities like process planning which make use of the feature information.

Due to these reasons, we have decided to extract out feature information and store them into a standard format, like STEP, which renders itself easily to data exchange.

STEP has gained considerable popularity in the recent past, mainly due to the active support for STEP from the automobile and the aerospace industries. There is therefore a need to have translators from legacy format to STEP. This is especially relevant considering that the aircraft and automobile companies have data in the IGES format which need to be converted to the new standard.

4 APPROACH

For our current project we have concentrated on the geometry, topology and drafting information for the 'design translator' and form feature information for the 'feature translator'. The STEP schema which is used for the design translator is AP202, Associative Draughting Schema, and the schema for the feature translator is AP224, Mechanical product definition for process planning using form features. AP202 has recently been released as International Standard (IS) whereas AP224 is a Draft International Standard (DIS).

4.1 STEP Application Protocols

4.1.1 Application Protocol 202 (Associative Draughting)

AP202 is the STEP schema which deals with drafting and 2D-drawing data. The geometry for the part is represented as 3D information and projected to a 2D plane to get the required drawing sheet. AP202 consists of:

- Geometry information: lines, circles, conics, surfaces.
- Topology information: faces, shells, edges, vertices, boundary representation solids(-Rep).

- Annotation information: dimensions and annotations.
- View and drawing information: views and view dependent drafting, annotations, drawing sheet size and layout, and layering information.
- Other information: product name, manufacturer's name, certain types of security issues.

Assemblies cannot be incorporated into AP202 because it only contains information about individual piece parts. Constructive Solid Geometry (CSG) solids cannot be converted into STEP (AP202 to be specific) directly, as AP202 uses B-Rep to represent solid models. CSG solids can be converted by using a solid modeler to convert CSG into the B-Rep format. For this reason we have eliminated CSG entities and concentrated our efforts on converting the rest of the IGES entities.

4.1.2 Application Protocol 224 (Mechanical product definition for process planning using form features)

AP224 is the STEP schema which is used to specify the requirements for the representation and exchange of information needed to define product data necessary for manufacturing single piece mechanical parts. AP224 covers the following:

- Product data that defines a single piece machined part to be manufactured.
- Product data that is necessary to track down the customer order.
- Product data necessary to identify the status of a part.
- Form features that are necessary for defining shapes necessary for manufacturing.
- Products that are manufactured by milling or turning.

Assemblies, representations of composite materials, sheet metal manufacturing data and data pertaining to the design phase of the product development cannot be represented in AP224. For the current work we are concentrating on milled prismatic parts. Identifying form features of turned parts consists of our ongoing effort.

4.2 Design Translator

The first step in developing the translator was deciding the mapping between IGES and STEP entities. The mapping is based on the perception and understanding of the IGES and AP202 definitions of entities. A few samples of the mapping are presented below and a more detailed mapping is included in Bhandarkar (1996).

1. IGES line entity is mapped into an AP202 trimmed curve. Line entities in IGES are used to represent a finite line bound between the end point. In STEP (AP202) a line is infinite and, defined by a vertex and a direction. Hence to convert an IGES line entity into AP202 we have to model it as a trimmed curve having its basis curve as a line. Complete circles and ellipses are mapped into their respective STEP counterparts. If on the other hand the conic is not complete then it is mapped as trimmed curves with the basis curve as the corresponding conic.

2. IGES linear dimension entity is mapped into an AP202 linear dimension. An IGES linear dimension entity is a composite entity consisting of one general note entity, two leader arrow entities and two witness line entities. The general note entity is mapped into an AP202 annotation text occurrence, each leader arrow entity is mapped into an AP202 leader terminators and each witness line entity is mapped into an AP202 annotation curve occurrence. All the AP202 entities are grouped as a *set of drafting callout elements* and then tied back to the linear dimension.

3. IGES view entity is mapped into an AP202 view volume. The view volume has the required information to convert from drawing space coordinates to view space coordinates. There is also information required to clip the geometry for partial views of the parts.

4. IGES manifold solid B-Rep entity, which is a solid model representation entity in the B-Rep format, is mapped into an AP202 B-Rep with voids. The B-Rep with voids contains a closed shell which forms the outer loop of the B-Rep entity and a set of oriented closed shells, which form the voids of the B-Rep.

Most of the IGES entities have been mapped into AP202 and some work is in progress to map the remaining entities. For example we are working on mapping the sectioned area entity and also dealing with the special types of fonts available in IGES.

4.3 Feature Translator

The feature translator is the second part of the project. Work is currently being performed on the development of the translation algorithm and coding it to extract out the form features from the

IGES file. The feature translator would function using the following steps

1. In the first step, the IGES file is read into memory. The file is parsed to find out if there are any objects of the type Manifold Solid B-Rep Object Entity (MSBO). The MSBO is the IGES entity used for representing topological information in the B-Rep format. MSBO consists of a number of connected components called shells. Each shell is composed of faces which have underlying surface geometry. The faces are bounded by loops of edges having curve geometry. The edges are bounded by vertices whose underlying geometry is a point.

2. If there is no MSBO then the IGES file contains only geometry information and the form-feature information cannot be extracted from the file. If there is an MSBO then the feature extractor proceeds further. The extractor then scans the solid for presence of any voids in any of the shells. Voids essentially represent material that has been removed from the surface of the solid, thus presence of a void definitely indicates a form-feature. The surface forming the voids is then analyzed to classify the void as a hole or a pocket or another appropriate form feature.

3. After the voids are identified and classified the extractor scans through each of the shells and tries to identify other form-features like protrusions, slots, steps, etc.

4. Ambiguities if any would be resolved by prompting the user for input regarding breaking ties among multiple possible features.

Thus we would have combination of automatic and user assisted form feature extraction mechanism.

5 IMPLEMENTATION

We have used C++ and object oriented B programming for the translators. The EXPRESS entities in AP202 and AP224 are compiled into C++ classes using the EXPRESS to C++ compiler developed by STEP Tools Inc. Class extension functions are then written to convert from IGES to AP202 and for extracting out the form features from the IGES data.

The design translation from IGES to AP202 is done in the following steps:

1. An IGES specification has been written in the EXPRESS language and so each IGES entity is available as a class in C++. The IGES reader goes through the file and reads in each entity in the file. First the global section which contains general information about the file like the name of the file authors name, company name, the global units for the file is read. Next the reader goes to the Directory Entry section and creates an instance of the object as defined by the entity number. Then it jumps to the corresponding Parameter data section of the IGES file and reads in the parametric data values. At this point the entries in the file are available as objects in the ROSE library. ROSE is a object oriented database (Hardwick 1991) developed at Rensselaer Polytechnic Institute. The ROSE database works off the compiled EXPRESS definition of the schema.

2. After the IGES file has been read into the ROSE library, the code goes through each of the entities and maps it into the corresponding AP202 entity.

3. At the end of stage 2 all the IGES entities have been converted into AP202. The mapped AP202 shape elements are then parsed and grouped into a *geometric_set* or *advanced_brep_shape_representation*,

depending on their type. The annotations are arranged into their respective views to correspond with the right view of the geometry. The drawing sheet layout is set to complete the STEP AP202 data.

4. The STEP data is then, saved in the ‘ROSE’ format and can be converted into a STEP Part 21 file format using a format utility provided with ST-Developer by STEP Tools Inc.

The translator has been tested and check with at least 10 IGES files and any bugs that have surfaced during these trials have been corrected. Files tested include those from the IGES manual and others obtained from various sources.

The feature translation is done is in the following steps:

1. The IGES reader parses through the IGES file and reads in all the entities into memory.

2. The code then looks for an MSBO entity. If there are no MSBO's then form-feature information cannot be found from the available IGES data.

3. If MSBO entity is found, the code parses through the MSBO and identifies any voids if present and creates corresponding AP224 object and populates them.

4. If no voids are found, the surface and edge data is examined to identify other form features like steps, protrusions, cutouts, etc.

5. After all the form features are identified, the STEP data is stored in the ‘ROSE’ format, which can be converted into a STEP Part 21 file format using the format utility.

The code for the feature translator is currently being developed and then the translator would be tested with sample test files to correct any bugs in the code or the feature recognition algorithm.

6 CONCLUSION AND FURTHER WORK

In the agile and virtual manufacturing environment, the STEP international standard for product data exchange is becoming increasingly popular over the legacy format such as IGES. IGES has been found to be restrictive because it does not capture data about the product through its life-cycle. To make the transition from IGES we are developing translators that would convert IGES drawing data into STEP drafting information and form-feature information. It is expected that this work will gain more importance once CAD vendors develop STEP support to their software products.

Work has also been performed to help visualize the AP202 files. We have used the Open Inventor Graphic modeling package (Wernecke, 1994) available for the Silicon Graphics platform for visualization. The visualizer helps to view the part being designed and with the capabilities provided by the software helps to view the part in totality. This enables the user to identify the form-features in the part and help resolve any ambiguities.

We are currently working on the form-feature translator. After the translator is sufficiently tested, the software can be extended in scope to take the form-feature information and develop a process plan for part manufacturing or even convert form-features into CNC code for part manufacture. We hope that through our efforts we have partially satisfied the information needs in the Agile Manufacturing environment. The work will gain greater importance as STEP is becoming more widely used all over the world.

ACKNOWLEDGMENTS

We would like to thank every one at STEP Tools Inc., Rensselaer Technology Park, Troy, NY for all their help to complete this project.

Rakesh Nagi acknowledges the support of the National Science Foundation career grant DMI-962409.

REFERENCES

1. Basu, D. and Kumar, S. (1995). “Importing MESH Entities through IGES/PDES,” *Advances in Engineering Software*, **23** (3), 151-161.

2. Bhandarkar, M. (1996). “STEP form feature modeling,” working thesis. Department of Industrial Engineering, State University of New York at Buffalo.

3. Gavankar, P. and Henderson, M. R. (1989). “Graph-based Extraction of protrusions and depressions from boundary representations,” *Computer Aided Design*, **22** (7), 442-450.

4. Gu, Z., Zhang, Y. F. and Nee, A. Y. C. (1995). “Generic form feature recognition and operation selection using connectionist modeling,” *Journal of Intelligent Manufacturing*, **6**, 263-273.

5. Hardwick, M. and the RPI DICE team (1991). “*ROSE 3.0 User Manual*,” Rensselaer Design Research Center, Rensselaer Polytechnic Institute, Troy, NY.

6. “*Initial Graphics Exchange Specification (IGES) Version 4.0*,” (1988) US Department of Commerce, National Institute of Standards and Technology, Gaithersburg, MD, USA.

7. Joshi, S. and Chang, T. C. (1988). “Graph-based heuristics for recognition of machined features,” *Computer Aided Design*, **20**, 58-66.

8. Ling, Z. K. and Narayan, S. (1996). “Heuristic Feature Recognition from a CAD Model,” *Integrated Computer-Aided Engineering*, **3** (3), 212-224.

9. Liu, S. C., Gonzalez, M. and Chen, J. G. (1996). “Development of an automatic part feature extraction and classification system taking CAD data as input,” *Computers in Industry*, **29**, 137-150.

10. Meeran, S. and Pratt, M. J. (1993). “Automated Feature recognition from 2D drawing,” *Computer Aided Design*, **25** (1), 7-17.

11. Owen, J and Woodall, A. (1990). "STEP application protocols: why and how," Document ISO TC184/SC4/WG1N465, National Institute of Standards and Technology, USA.

12. "*Part 202, Associative Draughting*," (1996). Document ISO TC184/WG3 N525(T6), National Institute of Standards and Technology, USA.

13. "*Part 224, Mechanical product definition for process planning using form features*," (1996). Document ISO TC184/WG3 N264(T7), National Institute of Standards and Technology, USA.

14. Qamhiyah, A. Z., Venter, R. D. and Benhabib, B. (1996). "Geometric Reasoning for the extraction of form features," *Computer Aided Design*, **28** (11), 887-903.

15. Salomons, O. W., Van Houten, F. and Kals. H. (1993). "Review of Research in Feature-Based Design," *Journal of Manufacturing Systems*, **12** (2), 113-132.

16. Schenck, D. A and Wilson, P. (1994). "*Information Modeling: The EXPRESS Way*," Oxford University Press, USA.

17. Shah, J. and A. Mathew, (1991). "Experimental investigation of STEP form feature information," *Computer Aided Design*, **23** (6), 282-296.

18. Shah, J., Mantyla, M. and Nau, D. (1994). Manufacturing Research and Technology; 20, "*Advances in feature based manufacturing*," Elsevier, The Netherlands.

19. Shu-Chu, L., Gonzales, M. and Chen J. G. (1996). "Development of automatic part feature extraction and classification system taking CAD data as input," *Computers in Industry*, **29**, 137-150.

20. Ssemakula, M. E. and Satsangi, A. (1990). "Application of PDES to CAD/CAPP integration," *Computers and Industrial Engineering*, **18** (4), 435-444.

21. Su, C. J., Sun, T. L., Wu, C. N. and Mayer, R. J. (1995). "An integrated form-feature-based design system for manufacturing," *Journal of Intelligent Manufacturing*, **6**, 277-290.

22. Wernecke, J. (1994). "*The Inventor Mentor: programming Object-oriented 3D graphics with Open Inventor, release 2*," Addison-Wesley Publication Company, USA.