Mechanical and Aerospace Engineering - Dissertations

College of Engineering and Computer Science

12-2012

Enriching Step-Based Product Information Models to Support Product Life-Cycle Activities

Mehmet Ilteris Sarigecili
Syracuse University

Recommended Citation

Sarigecili, Mehmet Ilteris, "Enriching Step-Based Product Information Models to Support Product Life-Cycle Activities" (2012). *Mechanical and Aerospace Engineering - Dissertations.* Paper 76.

Abstract

The representation and management of product information in its life-cycle requires standardized data exchange protocols. Standard for Exchange of Product Model Data (STEP) is such a standard that has been used widely by the industries. Even though STEP-based product models are well defined and syntactically correct, populating product data according to these models is not easy because they are too big and disorganized. Data exchange specifications (DEXs) and templates provide re-organized information models required in data exchange of specific activities for various businesses. DEXs show us it would be possible to organize STEP-based product models in order to support different engineering activities at various stages of product life-cycle. In this study, STEP-based models are enriched and organized to support two engineering activities: materials information declaration and tolerance analysis. Due to new environmental regulations, the substance and materials information in products have to be screened closely by manufacturing industries. This requires a fast, unambiguous and complete product information exchange between the members of a supply chain. Tolerance analysis activity, on the other hand, is used to verify the functional requirements of an assembly considering the worst case (i.e., maximum and minimum) conditions for the part/assembly dimensions.

Another issue with STEP-based product models is that the semantics of product data are represented implicitly. Hence, it is difficult to interpret the semantics of data for different product life-cycle phases for various application domains. OntoSTEP, developed at NIST, provides semantically enriched product models in OWL. In this thesis, we would like to present how to interpret the GD & T specifications in STEP for tolerance analysis by utilizing OntoSTEP.

ENRICHING STEP-BASED PRODUCT INFORMATION MODELS TO SUPPORT PRODUCT LIFE-CYCLE ACTIVITIES

By

Mehmet Ilteris Sarigecili

B.S. Cukurova University, Turkey
M.S. Cukurova University, Turkey

DISSERTATION

Submitted in partial fulfillment of the requirements for the degree
of Doctor of Philosophy in Mechanical and Aerospace Engineering
in the Graduate School of Syracuse University

December 2012

Acknowledgements

I would like to express my sincere gratitude to my academic advisor, Professor Utpal Roy, for his guidance, support and encouragement throughout my program of study and research work. It has been a pleasure to work with him.

I would take this opportunity to thank Dr. Shalabh C. Maroo, Dr. Young B. Moon, Dr. Jianshun Zhang and Dr. Sudarsan Rachuri and Dr. Riyad S. Aboutaha for consenting to serve as members of my dissertation committee. I also would like to thank Dr. Ram D. Sriram and Dr. Sudarsan Rachuri at the National Institute of Standards and Technology for providing funding to my research.

Further thanks go to Dr. Mehmet Murat Baysal for his constant support, for fundamental and theoretical discussions, and for his companionship and encouragement; to my office mates Dr. Lowell E. Lingo Jr., Bicheng Zhu and Shahed Shojaeipour; especially to Dr. Besir Sahin at Cukurova University for his faith in me and for supporting me in coming to Syracuse University.

Finally, and most important, I would like to express my sincere appreciation and thanks to my family, my dad Huseyin, my mom Fazilet, my sisters Neslihan and Aysegul, my brother Ali, my niece Derya and my brother-in-law Kemal for their infinite love and patience, without which it would have been impossible for me to complete my research work. I also would like to thank my all other relatives and friends for their constant supports and prayers which gave me ability to tackle challenges head on.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Introduction and Motivation

In today’s competitive business environment, companies need to collaborate more, which requires them to share and exchange product information among heterogeneous computer systems. The only feasible way to achieve this is to have a common information model that is capable of representing requirements in several domains. ISO 10303 (ISO 10303-1, 1994) provides such a standard information model; it was developed to enable the exchange of product data among the different computer systems used throughout a product’s life-cycle. It is called the *standard for exchange of product model data* (STEP), and it provides a neutral format for the representation of product data. In this thesis, enriching STEP-based product models to support two engineering activities in a product life-cycle is discussed: substance level materials declaration and tolerance analysis. Additionally, this research work is improved by (1) organizing the information requirements to support these activities, and (2) representing semantics of product data explicitly so that computers can understand these semantics.

The substance level information for any product that is supplied or manufactured is needed to be exchanged between members of a supply chain due to increasing environmental regulations. These regulations in certain industries, such as the automotive and electronic industries, require manufacturers and suppliers to develop programs to examine their products closely to improve product recyclability, eliminate the presence of certain hazardous substances, and implement take-back programs in a cost-effective manner. Since compliance to these regulations requires declaration of materials information, industry standards are developed to help exchange the required substance data. One of these standards is the IPC-175X series standards (i.e. 1751, 1752 or 1756), published for electrical and electronic products by "Association Connecting Electronics Industries" (IPC, 2012). This standard also covers JIG-101 (JIG-101, 2012), RoHS (RoHS, 2002), and REACH (REACH, 2006).

The implementation of IPC-175X standards helps create simple, standard forms to exchange substance level information in digital format, i.e. as an XML file. A Java-based implementation of IPC-175X standards is developed by the National Institute of Standards and Technology (NIST), which is called SCRIBA (SCRIBA, 2012). However, the information entered in the IPC-175X forms is not derived directly from any product life-cycle management (PLM) system. So, it requires a lot of manual efforts to identify and extract the relevant information from the IPC-175X data forms, and then to match it with information available in the PLM system to ensure the material compliance necessary for the regulation directives. It could have been better, if the information required for IPC-175X standards could be directly extracted from a PLM system. The information exchange would be much more efficient, consistent, and unambiguous. It necessitates a detailed investigation of all appropriate PLM standards, in order to find out whether the information required for IPC-175X standards are already available. Therefore, in this thesis, we studied the information content required for IPC-1751 and IPC-1752, and investigated whether the information could be extracted from STEP standards (AP 203 and AP 214) and its integrated resources (namely Part 41, Part 43, Part 44, and Part 45).

Another subject of this study is to organize STEP product models for data exchange in accomplishing engineering activities. STEP data models (or schemas) are represented in EXPRESS (ISO 10303-11, 1994) as a network of concepts involving entities and the properties of these entities. The representation of instance data according to EXPRESS models is defined in STEP Part 21 (ISO 10303-21, 2002). EXPRESS-based STEP information models are syntactically correct and well defined, and the schemas are well suited to representing the syntax of the product model; however, it is very difficult to populate these models with product data because they are too large and disorganized. For example, the implementable parts of STEP, application protocols (APs), represent product data models for a particular application domain, e.g., configuration-controlled product design (AP 203 (ISO 10303-203, 1994)), core automotive product data (AP 214

(ISO 10303-214, 2003)), product life-cycle support (AP 239 (ISO 10303-239, 2003)), etc. Support for, and compliance with, all of the information models in an AP are not practical for any single software application. Therefore, conformance classes (CCs) were created as subsets of APs to support file-based data exchange scenarios. The current CCs consist simply of a list of entities; these also are not very helpful for populating product data.

OASIS Product Life Cycle Support Technical Committee (OASIS, 2008) developed data exchange specifications (DEXs) (DEX, 2012) to serve as re-organized information models of AP 239 to support data exchange for activities of any business scope (Figure 1). A DEX is same as a CC in an AP; however, a DEX supports additional usage guidance, defined by *Capabilities*, and is semantically enriched by *Reference Data*. Each capability provides guidance and rules for a consistent representation and application of generic business concepts. *Templates* in each capability provide patterns for structuring and instantiating data; they unambiguously present required entities and the way these entities should be populated to represent patterns.

Similar to DEX/Template approach, we would like to develop functionality-based CCs (FCCs) to provide required information for successfully carrying out an engineering activity. This paper discusses the development of the FCC for 1-D tolerance analysis. This development process requires the enrichment of the available STEP information model with GD & T. In addition to that, the assembly constraint information models are also modified for the purpose of this study.

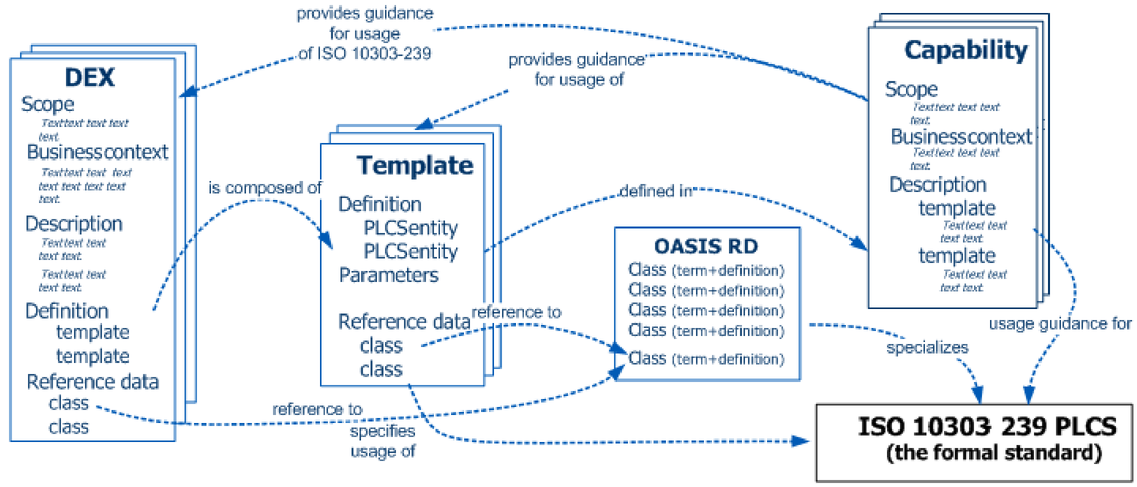


Figure 1. Structure of Data Exchange Specifications (DEXs) (DEX, 2012)

The information requirements in proposed FCCs are grouped into several information layers. They are arranged hierarchically, such that the higher-level information requirements collect necessary information from lower levels. It is because certain data instances cannot be created without instantiating some other data instances. Furthermore, the necessary information layers might be different for each engineering activity. The development of the FCCs is carried out in four stages:

1. identification of the required information for a particular functional activity,
2. grouping the information into hierarchical information layers,
3. mapping the information to the STEP entities for a standardized representation, and
4. developing templates as patterns for populating product data.

Another issue with EXPRESS-based STEP information models is that these models have a great difficulty in expressing explicit data semantics and make them available at different product life-cycle phases for different application domains. For example, though a limited tolerance specification of a product is possible in STEP (using AP 214 constructs), unless the semantic interpretations of those tolerance specifications are explicitly available, a user cannot apply those specification data in areas like tolerance analysis, product manufacturing, assembly or inspection. Recent work at NIST on developing semantically enriched STEP product models in OWL 2 (Web Ontology Language) (OWL, 2012) called OntoSTEP (Barbau et al., 2012) shows us it would be possible to develop a consistent formal model (including both syntactically and semantically correct information) for products that is useful in carrying out effective computational (both quantitative and qualitative) analyses in different domains of applications as they may be required in different product life-cycle phases.

In this thesis, we would like to develop the semantic interpretations of the GD & T (geometric dimension and tolerance) design specifications and use it for the linear, stack-up tolerance analysis. This requires first the development of a tolerance analysis oriented information model (based on the given GD & T specifications) in EXPRESS. This embellished GD & T model would be then merged with the GD & T model available in AP 214. In the next step, this combined STEP schema (model) is translated in OWL 2 using OntoSTEP plug-in so that the OWL model of the GD & T specifications is now available for further reasoning purposes. The SWRL (SWRL, 2012) rules inferred by the Pellet (Pellet, 2012) reasoning tool are used to map these GD & T specifications to the specifications needed for tolerance analysis.

1.2 Objectives of this Dissertation

To summarize, the main objectives of the thesis are:

- **Extraction of the materials declaration information requirements (i.e., IPC-175X) from an international standard that supports complete product life-cycle. (i.e., STEP):** All the necessary information required for creating the IPC-175X standard forms are extracted from the integrated resources of STEP. However, since the integrated resources are not directly implementable, and application protocols (APs) are only implementable parts of STEP, we further studied two (2) APs (AP203 and AP214) to see how much information from the IPC-175X standards could be extracted practically from STEP. A case study of a simple gearbox has been considered to illustrate the implementation of the concept.

- **Organizing STEP-based product models to support engineering activities and populate product data easily:** The available STEP models are rearranged to provide required information for successfully carrying out 1-D stack-up tolerance analysis. For tolerance analysis, the current STEP information models have to be enriched with

GD & T information. It is also needed to modify the available assembly constraint models in STEP.

- **Development of templates:** The patterns for structuring and populating data have been identified from available GD & T representation models (i.e., Part 47 of STEP).

- **Semantic interpretation of STEP data**: Semantic interpretations of GD & T specification have been developed to be able to use in tolerance analysis.

1.3 Organization of this Dissertation

Chapter 2 reviews the literature on (a) IPX-175X standards and studies on materials declaration, (b) structure of STEP, conformance classes in STEP and DEX/Template, and (d) studies on semantic product data representations.

In chapter 3, the extraction of IPC-175X information requirements from available STEP integrated resources are presented. In chapter 4, development of functionality-based CCs for 1-D stack-up tolerance analysis is discussed. This is accomplished by identifying the required information to carry out tolerance analysis, mapping the requirements to available STEP resources and finally creating EXPRESS-G definitions for the requirements that cannot be mapped directly to STEP resources. To improve the implementation of these models, templates for GD & T specifications have been developed as patterns for populating product data in chapter 5. Then, developed tolerance analysis information model is used for the interpretation of GD & T specifications to carry out tolerance analysis in chapter 6.

In the final chapter (chapter 7), contributions in the fields of STEP product modeling, semantic representations and substance level materials declaration are earmarked.

Chapter 2

Review of Related Research

In this chapter, the works in three subject areas are reviewed: (i) materials declaration standards and related studies, (ii) structure of ISO 10303 – STEP standard and (iii) representations of product data semantics.

2.1 The Material Declaration Standards and Related Studies

IPC-175X standards are developed in order to provide a standard data declaration mechanism between members of a supply chain. They are:

- IPC-1751 Generic Requirements for Declaration Process Management (IPC-1751A, 2012)
- IPC-1752 Materials Declaration Management (IPC-1752A, 2012)
- IPC-1756 Manufacturing Process Data Management (IPC-1756A, 2012)

The generic requirements for declaration process management are defined by IPC-1751, which is also mandatory for other IPC-175X standards. The requirements include: the names and the contact information for the requester and the supplier, and the product related information. IPC-1752 standard defines the data requirements for reporting material declaration for RoHS compliance, material group declaration for JIG-101 and

REACH standard, and also material composition declarations at homogenous material levels. IPC-1756 defines the requirements for exchanging manufacturing data for electrical and electronic products between members of a supply chain. The data requirements for the IPC-175X standards are represented by the XML schema, which is used to validate the XML data files.

JIG-101 standard (JIG-101, 2012), which is included in IPC-175X, is the Joint Industry Guide Material Composition Declaration for Electronic Products. It is published by the Electronic Industries Alliance, Japan Green Procurement Survey Standardization Initiative, and the Joint Electronic Device Engineering Council (JEDEC). JIG-101 is published for the electrical and electronic products, and it lists the materials and substances to be disclosed by suppliers when those materials and substances are present in products and sub-products.

The other regulatory directives that are covered in IPC-175X are RoHS (RoHS, 2002), and REACH (REACH, 2006). RoHS is the Restriction of the Use of Certain Hazardous Substances in Electrical and Electronic Equipment. It basically restricts six materials (i.e., lead, mercury, cadmium, hexavalent chromium, polybrominated biphenyls and polybrominated diphenyl ether) at the homogeneous material level. In RoHS, the homogeneous material means that the material which can be separated mechanically, like cutting, grinding, etc. REACH is regulation for the Registration, Evaluation, Authorisation and Restriction of Chemicals. It requires companies to declare the presence of certain “Substances of Very High Concern” (SVHC) in the products that are sold in Europe.

In the literature, there exist several studies for the integration of materials compliance management data with the product life-cycle data. Zhou et al. (2008) developed an integration framework to combine BOM data and material content information. In their proposed system, they create their own declaration form to exchange data. They also use XML user interface language (XUL) (XUL, 2012) and Simple Object Access Protocol (SOAP) (SOAP, 2012) technologies for the integration. In another study, Zhou et al. (2009) developed a framework to integrate the BOM XML files from heterogeneous PLM systems to the RoHS compliance management system. Gong et al. (2011) proposed a green design control system, which controls workflow and related functions of a product’s life-cycle. However, these studies do not work with international standards, and the extraction of the materials information from the PLM system cannot be done automatically.

There are also commercial products that implement IPC-175X standards in their PLM systems. The most popular ones are ENOVIA from Solidworks (ENOVIA, 2012), Windchill from PTC (Windchill, 2012) and Teamcenter from Siemens (Teamcenter, 2012). These products use their own proprietary in-house information extraction mechanisms. The need to represent the required IPC-175X information in an international standard arises in order to collaborate with different systems seamlessly.

2.2 Structure of ISO 10303 (STEP)

STEP (ISO 10303-1, 1994) provides a complete product information model and facilitates the product data exchange between different computer systems used throughout the product life-cycle. STEP is developed in series of parts (See Figure 2). They are description methods, integrated resources, application protocols, implementation methods, and conformance testing. The integrated resources of STEP are developed to represent abstract, conceptual models of product data. These models are then used in the development of application protocols. The integrated resources are

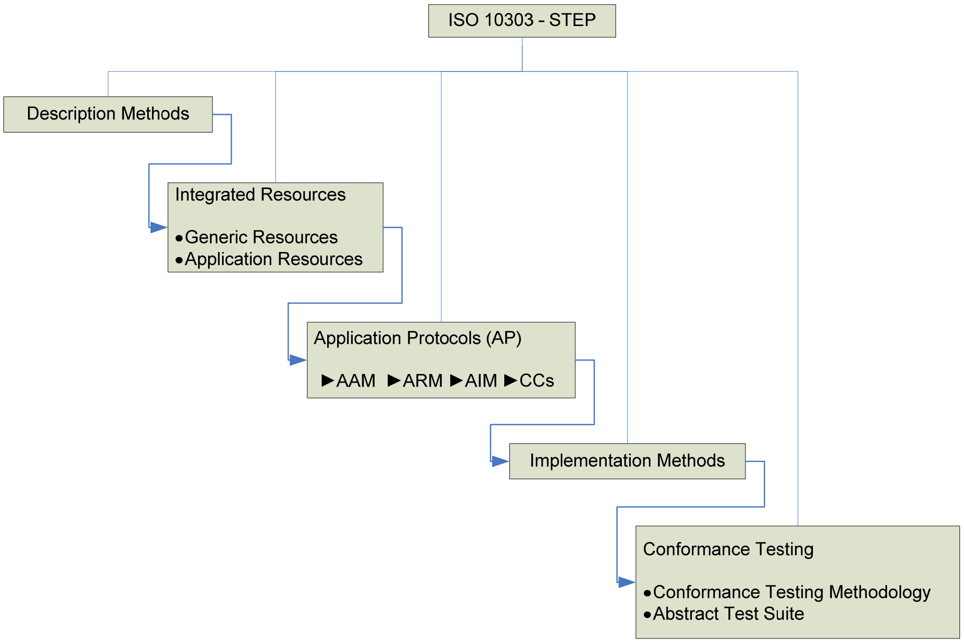


Figure 2: Structure of ISO 10303 – STEP (Sarigecili et al., 2009)

represented with 40 series parts of STEP. In this thesis, Part 41 (ISO 10303-41, 1994), Part 43 (ISO 10303-43, 2004), Part 44 (ISO 10303-44, 1994), Part 45 (ISO 10303-45, 1997) and Part 47 (ISO 10303-47, 1994) will be studied to represent information requirements for IPC-175X and tolerance analysis, as needed.

A brief explanation of these integrated resources would be as followings: Part 41 of STEP is known as the "Fundamentals of product description and support." It basically defines the conceptual models for the product definition, product property definition and representation, management resources, document representation, person and organization representation, date and time representation, external reference representation, support resource representation, the representation of the measurement with units and values, action representation, certification representation, approval representation, contract representation, security classification representation, group representation, and effectivity representation. Part 43 of STEP is called the "Representation structures." It basically defines the constructs for the representation of property definitions or shape definitions. Part 44 of STEP is called the "Product Structure Configuration." It is developed to represent the product structure, product concept, and configuration information. Part 45 of STEP is called "Materials." It is used to define materials information, material property information, and representation of material properties with uncertainty information. Part 47 of STEP is called "Shape Variation Tolerances." It is used to represent dimensions and tolerances of a product.

Application protocols (APs) of STEP are represented by 200 level series. The most commonly implemented ones are AP 203 and AP 214. AP 203 is the configuration controlled 3D design of the mechanical parts and assemblies. It is used to exchange product data for the mechanical parts and assemblies. For the materials declaration purposes, the product definition, configuration control, three-dimensional shape representation, materials, and their composition of chemical substance data can be represented by implementing AP 203 information models as needed. On the other hand, AP 214 is the core data for the automotive mechanical design processes. AP 214 covers most of the information that is covered by AP 203 but not all information. Some of the information (required by IPC-1752) that can be represented in AP 203 is not defined in AP 214. Therefore in order to extract the required information for IPC-1752 from STEP files we need to use AP 203 only. If a CAD vendor implements only AP 214 it will not be possible to extract all the information.

2.2.1 Conformance Classes in STEP

In the old structure of STEP, the APs consist of four parts (ISO 10303-11, 1994; Kemmerer, 1999): (1) an application activity model (AAM), (2) an application reference model (ARM), (3) an application interpreted model (AIM) and (4) conformance classes (CCs).

The implementable subsets of an AP are current CCs in the form of a list of entities. For example, AP 203 1st ed. (ISO 10303-203, 1994), which is the “Configuration

Controlled 3D Design of Mechanical Parts and Assemblies," has six CCs. The first is the mandatory CC, which groups the entities for the configuration control data without shape information. The other five CCs provide a list of entities for different shape representation models.

Any STEP-certified CAD system implements a particular set of CCs defined in an AP. As result of the implementation, a Part 21 STEP file is generated to represent product data. However, the information inside the STEP files is disorganized. A STEP file created by a translator includes a huge amount of data, and it is very difficult to query the data that represent a set of concepts. For example, how the dimensions and tolerances of a pattern of holes can be queried is not known. What should be specified as input? What should be retrieved as output?

In order to improve implementation of the CCs, we propose a layered approach for grouping the information requirements for several activities in a product life-cycle, as outlined in a previous study (Sarigecili et al., 2009). The layered approach proposes the establishment of a basic level of information requirements, in order to start defining the activity, and other levels of information requirements are added hierarchically. At each level, the information models defined by the integrated resources available in STEP are utilized as much as possible to represent the information requirements of an activity. In this way, we aim to make it easier to navigate through the information requirements of different levels and define what activities can be carried out by conforming to a specific CC.

It is important to note that there are some structural problems in implementing CCs from APs (ISO TC 184/SC4 N1161, 2001; Feeney, 2002):

1. It is not possible to implement a combination of subsets of multiple APs or extend existing APs to meet a business need.
2. It is not possible to reuse translators developed particularly to support one AP for developing and implementing another AP with the same, or similar, requirements.

The ISO developed modular application protocols (ISO TC 184/SC4 N1161, 2001) to make it possible to reuse developed information models for APs, thereby decreasing the time and effort needed to develop them. In the new modular structure, the information requirements are harmonized first for applications and then modeled as application reference models (ARMs). Finally, these ARMs are mapped to EXPRESS to produce module-interpreted models (MIMs) as standardized representations in application modules (AMs). Harmonizing the requirements before standardizing them prevents the creation of different dialects in the mapped standardized representations for the same application. It also facilitates the reusability of the AMs.

Application protocols that have modular structures also have CCs. For example, AP 203 2nd ed. (ISO 10303-203, 2005) has only one CC, i.e., CC7 — configuration-controlled 3D design of mechanical parts and assemblies, with additions. The additions consist of 77 different conformance options in AP 203 2nd ed. Conformance with this CC7 requires the support of all ARM and MIM elements defined in the application modules 10303-1022 (ISO 10303-1022, 2010) and 10303-1023 (ISO 10303-1023, 2004). 10303-1022 is the application module for part and version identification, whereas 10303-1023 is the part view definition. As an example of conformance options in AP 203 2nd ed., consider "co39," which is called the "conformance option GD & T representation." It offers the capability to represent geometric dimensioning and tolerancing information about the shape of a product. In "co39," any other conformance options and application modules that are required to be supported are also given.

As can be seen from the above explanations of CC7 and "co39," the requirements are implementable as small groups, and this makes it easier to implement them. However, the information requirements for a particular engineering activity (e.g., tolerance analysis) and the organization of the information requirements are not available. The user has to collect all the information as needed. At least there is some organization in the STEP information models, but they do not satisfy the exact needs of a user. Users do not know what AP, what CC or what application modules they require for a particular engineering activity.

In order to provide usage guidance and rules in data exchange for a particular business activity through the use of templates from AP 239 information models, OASIS PLCS TC developed data exchange specifications or DEXs (DEX, 2012). As an example, consider the template assigning_reference_data (PLCS template, 2012), which represents the pattern which is repeatedly used in AP 239 information models for the assignment of a class to something. The definition of the specified class is provided in an external reference data library. The information model of this template is shown in Figure 3.

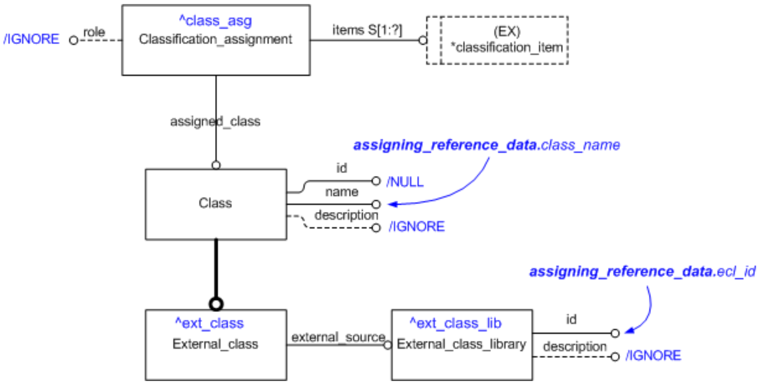


Figure 3: Assigning_reference_data template (PLCS template, 2012)

In a template representation, entities that need to be instantiated are presented as reference parameters. This is shown by an annotation character “^” preceding a short reference parameter name in blue for the entity. In Figure 3, classification_assignment, external_class and external_class_library entities have these reference parameters, indicating that these are entities instantiated in this template. In a template, whether attributes are optional or mandatory, and what the input parameters for attributes are, are defined as well. For example, assigning_reference_data.class_name and assigning_ reference_data.ecl_id, shown with blue arrows, are the input parameters that need to be provided by users. More details on DEXs/Templates can be found in (DEX, 2012).

In this study, we would like to develop templates based on repeatedly instantiated patterns in GD & T representation information models. These models are defined in Part 47 of STEP. The templates could then be used in functionality-based conformance classes for any engineering activity, such as tolerance analysis.

2.2.2 STEP-based Product Models in Literature

In the literature, there are several studies that use STEP-based product models to integrate information from different domains in a product life-cycle. These studies do not address either how to embellish current CCs or how to integrate the information needed for tolerance analysis in a collaborative environment.

For example, Xie et al. (2008) offered a generic product modeling framework that uses the STEP standard and integrates product design, manufacturing and assembly activities. Shaharoun et al. (1998) developed a product model for plastic products to define the geometric data representation of the injection molding features, employing the AP 203 framework. Sharma and Gao (2002) used AP 224 in developing the STEP-enabled Manufacturing Planning System (SMPS) to generate process plans automatically. Tang et al. (2001) developed a die and product integrated information model for stamped part and die development. Their model is also based on the integrated resources of STEP. In their model, the integrated resources are used to define information for shape representation, tolerance, materials, resources for manufacturing, actions to be taken for changes in design, product definition, and product structures for stamped parts and dies.

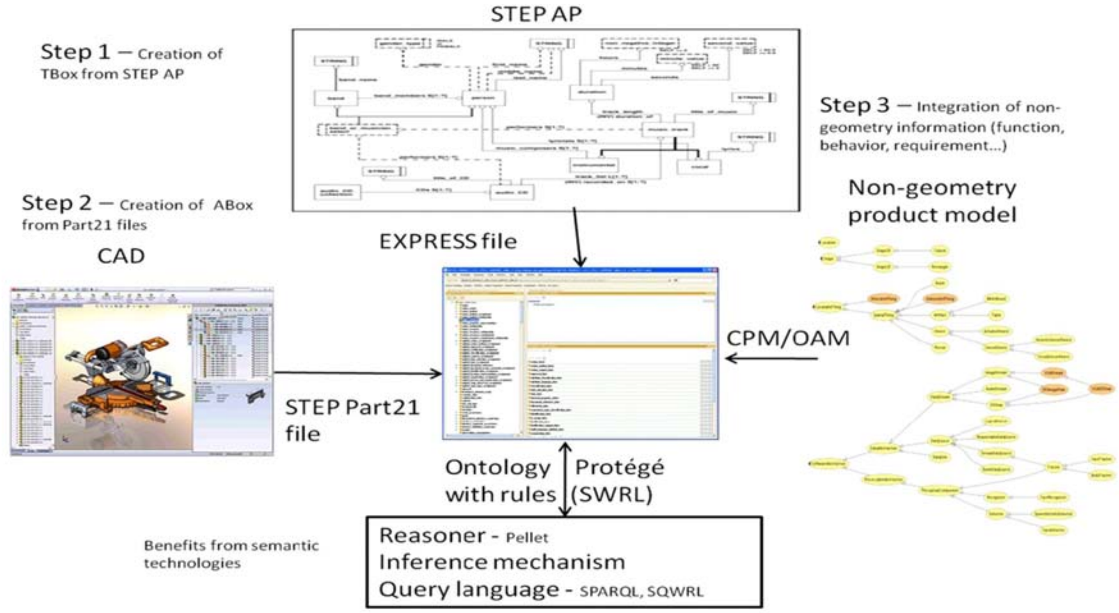
Zha and Du (2002) developed a system to integrate an assembly planning system with other CAD systems. In their system, the information for shape, feature, tolerance, product definition and assembly structure were defined based on STEP-integrated resources. Al-Ashaab et al. (2003) proposed the SPEED (Supporting Plastic enginEEring Development) system. This system helps in the sharing of injection mold information among design, analysis and manufacturing workers over the internet. A team at the University of Auckland, New Zealand, (Xie et al., 2008; Yang et al., 2008; Zhou, 2009-a) reviewed product-modeling techniques and proposed a generic product-modeling framework to integrate information on product design, manufacturing and assembly. They defined EXPRESS data models in addition to STEP-integrated resources, in order to exchange data. Barreiro et al. (2003) proposed an information model to integrate inspection processes, based on coordinate measuring machines with other product life-cycle activities. They used STEP resources and defined two models in EXPRESS representation: the product model and an inspection process model. Ming et al. (1998) proposed EXPRESS-based information models for Computer-Aided Process Planning to be used in a computer-integrated manufacturing environment. Their CAPP system brought together part, process plan and production resource information models.

Liang et al. (1999) developed an integrated product data sharing system for CAD, CAM and FEA. STEP-based information models from the integrated resources of AP 203, AP 209 (ISO 10303-209, 2001), AP 214 and AP 224 (ISO 10303-224, 2006) were used as the information-sharing environment. In order to optimize the data integration level and data processing efficiency, they defined a unit of functionality (UoF) as a group of data that is shared among different applications. Each UoF is a subset of an AP. The primary units of operation in their system are the UoFs. Because of this, they defined the relationships among different UoFs. However, because entities in different UoFs might be related to each other, the relationships between these entities also had to be defined. As a result, even though they tried to use information constructs from several application protocols, their information groups are bigger in context, e.g., representing FEA results, which is not sufficient for representing the capabilities of applications.

2.3 Representations of Product Data Semantics

Since semantics representation part of this study is based on the NIST's work on OntoSTEP (Barbau, 2012), it is necessary to review briefly the details of OntoSTEP to understand the proposed methodology.

OntoSTEP is developed to provide semantic product models which include geometry, function and behavior information. In OntoSTEP, the geometry information is defined by the STEP models whereas function and behavior information are defined by the Core Product Model (CPM) (Fenves, 2001) and the Open Assembly Model (OAM) (Baysal et al., 2004). The steps required to develop OntoSTEP based product model is given in Figure 4. At first, the EXPRESS model of a STEP application protocol is translated into OWL 2 via OntoSTEP plug-in (OntoSTEP, 2012), and the OWL schema of the STEP AP is created in Protégé. In the second step, the STEP file which includes the physical product data that is encoded with respect to the STEP AP is translated to the OWL individuals. In the third step, the ontology representation of the CPM/OAM model (Fiorentini et al., 2007) is merged into the OntoSTEP ontology to represent the function and behavior information for products.


STEP AP
Step 1 – Creation of TBox from STEP AP
Step 3 – Integration of non-geometry information (function, behavior, requirement...)
Non-geometry product model
Step 2 – Creation of ABox from Part21 files
CAD
EXPRESS file
CPM/OAM
STEP Part21 file
Ontology with rules
Protégé (SWRL)
Benefits from semantic technologies
Reasoner - Pellet
Inference mechanism
Query language - SPARQL, SQWRL

CAD - Computer Aided Design **CPM** - Core Product Model **OAM** - Open Assembly Model **SWRL** - Semantic Web Rule Language **SQWRL** - Semantic Query-Enhanced Web Rule Language **SPARQL** - SPARQL Protocol And RDF Query Language **OWL** - Web Ontology Language

Figure 4: Representation of OntoSTEP framework (Barbau et al., 2012)

In the literature, there are some other relevant studies that focus on the ontology-based product models to capture the semantics of the product data, as well. Kim et al. (2006) developed the assembly design ontology to represent explicitly and formally the assembly design information to map the assembly constraints to the design intent in a heterogeneous design environment. The OWL and SWRL are used in the development of their ontology. Even though it is possible to represent the assembly design information explicitly in OWL, their work is not integrated with STEP. For the interpretation of the GD & T information for tolerance analysis, all product structure, geometry and GD & T information is needed to be developed in their ontology. In our work, we use the STEP information models to have both standardized and explicit semantic information representation.

Panetto et al. (2012) developed ONTO-PDM to facilitate the interoperability among product data management (PDM) systems, enterprise resource planning (ERP) systems and manufacturing execution systems (MES). ONTO-PDM (framework shown in Figure 5) is the integration of ISO 10303 – STEP and IEC 62264 (2002) which is the standard for enterprise control system integration. In that study, the concepts are compared between the two standards first and then, then mapped to conceptualized UML models for representation of the information regarding the common concepts. They used the first order logic to represent the semantics of each construct of the standards conceptual models. First order logic axioms formalize the representation of mappings between the concepts of ISO 10303 and IEC 62264. The developed ONTO-PDM provides a product-centric interoperability. Its scope is the higher level data for the management of the information. However, in our study the fine level technical data has to be represented explicitly and interpreted for different application domains.

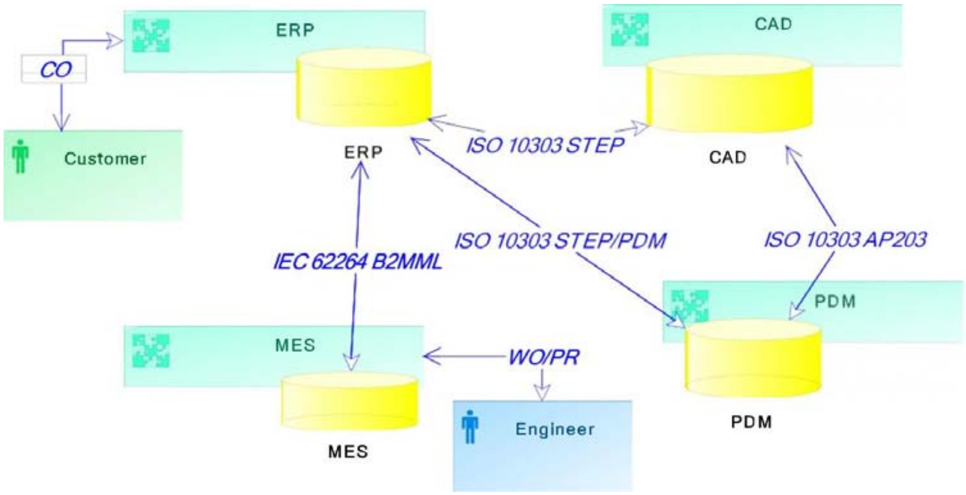


Figure 5: ONTO-PDM architecture (Panetto et al., 2012)

Matsokis and Kiritsis (2011) developed an ontology based semantic model for the representation and management of product life-cycle data/knowledge by mapping the UML model of Product Data and Knowledge Management Semantic Object Model (SOM) defined in Promise-PLM European project (PROMISE, 2012). They used description logic, SWRL and Pellet reasoning mechanisms to check the consistency of the ontology, to reclassify the classes to the correct position in the model.

The SOM is a product information model with capabilities of information tracking and flow management. The SOM is particularly applicable to the use, service and maintenance phases of the product life-cycle.

Chapter 3

The Harmonization of the Standards: ISO 10303 - STEP and IPC-1752 Materials Declaration

In this section, all the necessary information required for creating the IPC-175X standard forms are identified and checked whether the requirements are present in the integrated resources of STEP. However, since the integrated resources are not directly implementable, and application protocols (APs) are only implementable parts of STEP, two (2) APs (AP203 and AP214) are studied to see how much information from the IPC-175X standards could be extracted practically from STEP. A case study of a simple gearbox is also considered to illustrate the implementation of the concept.

3.1 Extraction of IPC-1751 Information Requirements

3.1.1 IPC-1751 Information Requirements

The overall scope of IPC-175X is to declare the substance level information for a product. IPC-1751 is mandatory for other IPC-175X standards. The UML information model for IPC-1751 is given in Figure 6. According to this model, a material information declaration can be initiated either by a request from a member of the supply chain or by a company that would like to declare it on their website or in their catalogs. A request/reply scenario requires defining a contact person both for making the request and for responding the request, as shown in Figure 6. In addition to the personal information, the company information, the contact information for the person like email address, physical address, and phone number have to be defined for both requesting/replying people. In IPC-175X, more than one physical address and phone number for a person may be defined. In the second scenario, where a company would like to declare the materials information by itself, the same information is required, except the requested portion because there will be no one requesting the information.

The product information for which the material information declaration is being made has to be defined as well. In IPC-1751, the product information consists of product identification (e.g. product number, product name, etc.) and the product structure information that is captured by the sub-product definitions (Figure 6). In IPC-1751 standard, the following information related to the product has to be defined in order to satisfy the substance level information exchange requirements:

- requester product number,
- requester product name,
- manufacturer product number,
- manufacturer product name,
- manufacturer product version,

- manufacturer site,
- effective date,
- product amount,
- unit of measure, and
- unit type.

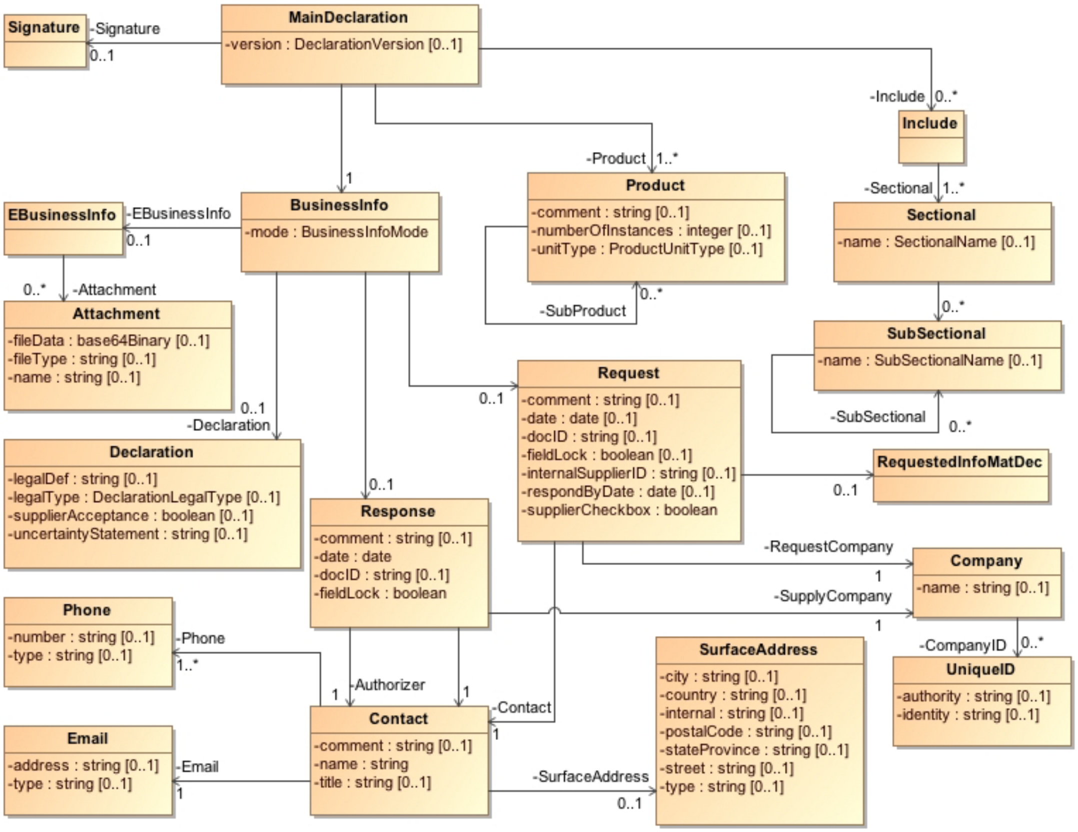


Figure 6: UML information model for IPC-1751 (IPC-1751A, 2012)

In the above list, the requester product number, requester product name, manufacturer product number, manufacturer product name, manufacturer product version, and manufacturer site are self-explanatory. The effective date represents the date for which the declared product information is effective. The product amount defines the value of the total amount of the product mass. This information should also specify the unit of measure, like milligrams, grams, kilograms, parts per million, or mass percent. The unit type identifies the basis of the quantification. For discrete products, the unit type would be “each”. For potentially boundless products such as a length of wire, a sheet of laminate, or a liquid, the unit type would be per meter, per square meter, or per liter, respectively.

The definition of the product structure in IPC-1751 is given in Figure 6. A product can have 0 or more sub-products. Each sub-product is also a product and should have the same information as defined previously for the products. In addition, each sub-product should have another attribute, which is representing the number of instances of the defined sub-product in the parent assembly. It is like a traditional bill of material (BOM) structure.

Other than the product related information, the date for the declarations, the liability of the declarations, i.e. legal definitions, and the document identification for the declarations have to be defined.

For the request/reply declaration, the date when the request is made and the expected response date are required in IPC-1751. Also, the date representation should comply with the XML schema date definition. In addition, the document ID for both a request and a reply has to be captured in order to keep consistency in between the declarations.

In Figure 6, the response to a reply should have a legal definition of the declaration. The legal definitions can be either a standard declaration, which is given in IPC-1751, or a custom one defined by the companies. The legal definitions as well as the uncertainty information of the declaration should be given under the declaration, as evident from Figure 6. The details of the sectional, sub-sectional, and material information will be defined under IPC-1752.

In short, these are the information requirements that need to be extracted from the STEP standard. In the next section, the extraction of these information groups will be discussed.

3.1.2 Extraction of IPC-1751 Information requirements from Integrated Resources of STEP

The information requirements of IPC-1751 cannot be mapped on a one-to-one basis into STEP standards. This is because the same information is conceptualized differently in IPC-175X and STEP standards. Also, the data structure and representation are completely dissimilar. In this section, the concepts of IPC-1751 will be compared to the concepts in the integrated resources of STEP, in order to identify the availability of the information first and then to find a way to extract the information consistently. The representation of the information requirements of IPC-1751 in STEP standards are described in five sub-sections: (1) the representation of contact information, (2) the declaration of the date, (3) the representation of a document and its properties, (4) the identification of a product and (5) the structure of a product.

3.1.2.1 The Representation of Contact Information

The contact, company, and the communication information of IPC-1751 can be extracted from Part 41 of STEP. The first difference is in naming the concepts. Part 41 calls person, organization, and address, instead of the terms contact, company, and surface address of IPC-1751, respectively. Another difference is that the address of the company is inferred from the contact address (Figure 6) in IPC-1751. In Part 41, person and organization information are defined separately, as shown in Figure 7. The entities* `organizational_address` and `personal_address` are subsets of the entity `address`. The address information is attached to both the person and the company information, separately. This means every person and company will have contact information. The association between person information and organization information is achieved by `person_and_organization`, as shown in Figure 7. In IPC-1751, every company should have a unique identifier and the organization that assigns the unique ID. In Part 41, `organization` information is defined by `id`, `name`, and `description` attributes. An additional attribute to define a “unique ID authority” will be incorporated as explained below.

* We use `Courier New font` to represent EXPRESS definitions.

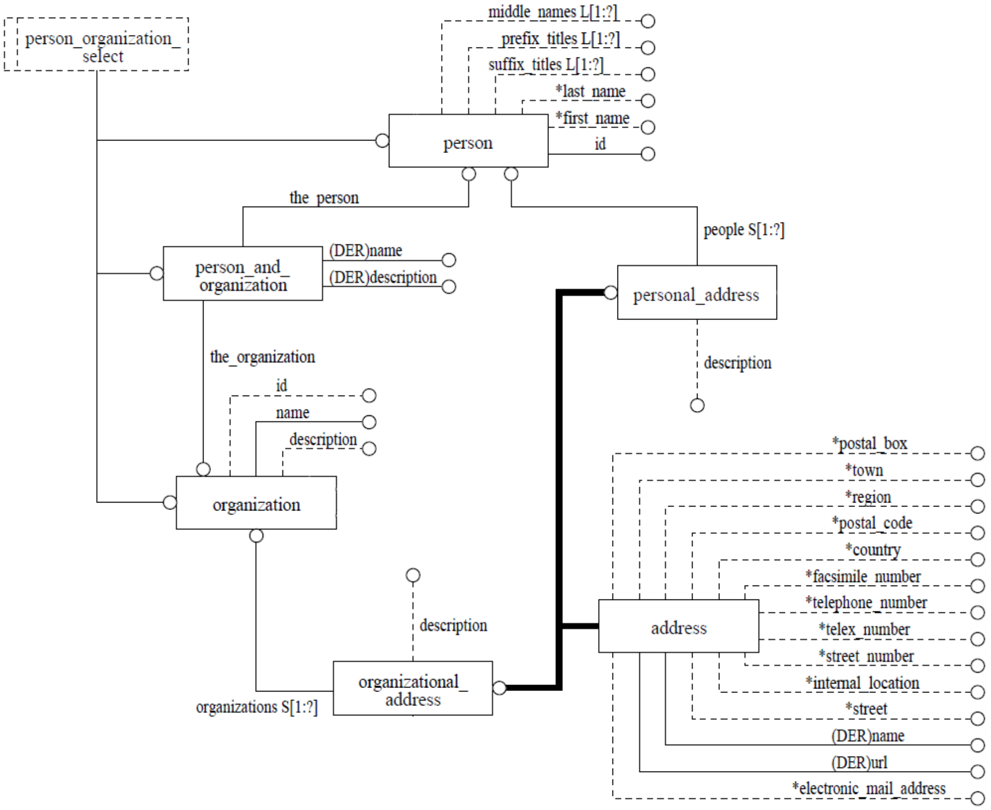


Figure 7: The EXPRESS-G representation of person, organization and address information in Part 41 (ISO 10303-41, 1994)

When the address, email, and phone number representations of IPC-1751 are compared to those of Part 41, it is observed that phone number and email address are defined as separate concepts from the address in IPC-1751. In Part 41, they are all defined under `address` entity. The `address` definition of Part 41 covers more details than IPC-1751. Some of the same concepts are represented by different terminologies in Part 41 and IPC-1751. For example, IPC-1751 has city and state province, whereas Part 41 calls them town and region, respectively. However, the required information of address, phone, and email definitions in IPC-1751 can be extracted completely from the Part 41 address definitions. In Part 41, the role of a person and role of a company can be assigned to related person and company, as shown in Figure 8. The `person_role` can be used to represent the title definition of a person in IPC-1751. The `organization_role` can be defined as “supplier” or “requester” in order to identify the company that is supplying or requesting the materials information as required in IPC-1751. In order to represent the organization that assigns the unique ID in STEP, the `organization_role` can be assigned as “unique ID definition” for the organization that assigns it and then use a relationship between this organization and any other company, as shown in Figure 9.

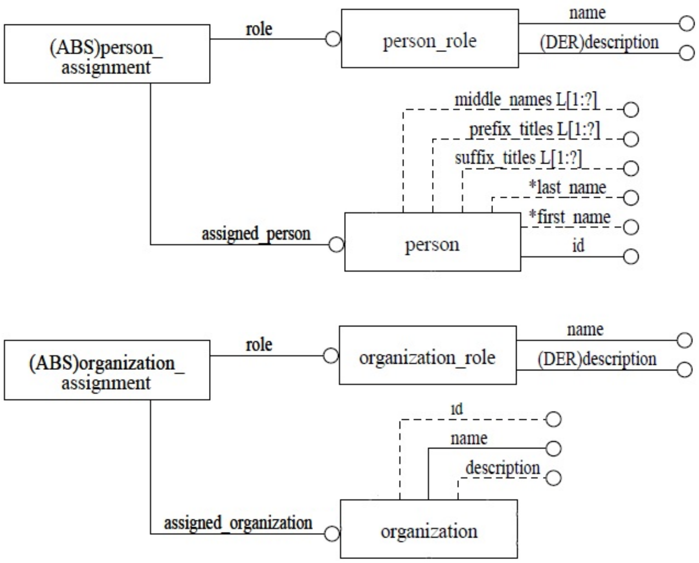


Figure 8: The EXPRESS-G representation of role assignment to person and organization in Part 41 (ISO 10303-41, 1994)

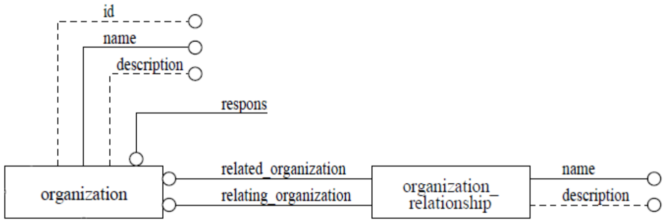


Figure 9: The EXPRESS-G representation of organization relationship definition in Part 41 (ISO 10303-41, 1994)

3.1.2.2 The Declaration of the Date

In IPC-1751, the date information has to be defined to identify the request date and the expected response date for the request/reply declaration. The XML schema date definition has been used for the date representation, i.e. year-month-day representation. In Part 41, the available data structure for date definition has more definition types than IPC-1751. In Figure 10, the EXPRESS-G representation for date definition, date role, and date assignment are given. The date definition has three different representation types: `calendar_date` represents year number, day number, and month number; `ordinal_date` only represents year number and the day number of the year; and `week_of_year_and_day_date` represents year number, the week number of the year, and optionally the day number of the week. Any of this representation can be easily converted to XML schema date type. The `date_role` entity will be used to identify the request date and respond-by date of IPC-1751. The association of the role to any date is achieved by the `date_assignment` entity, as shown in Figure 10.

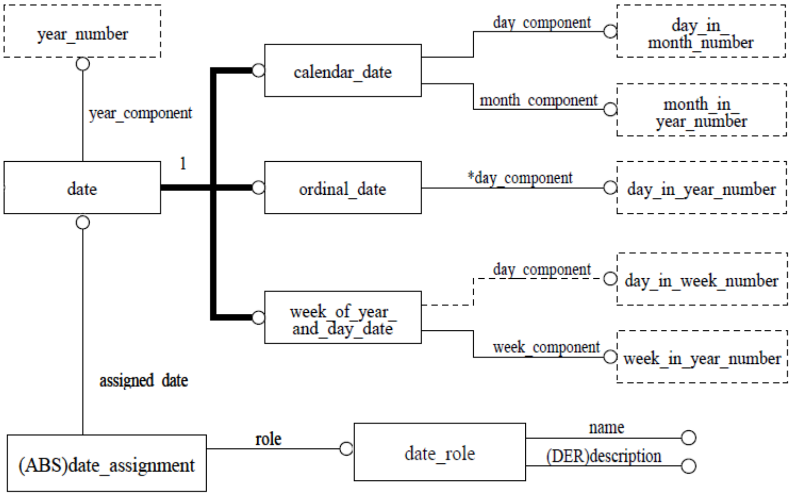


Figure 10: The EXPRESS-G representation of date and date role definition in Part 41 (ISO 10303-41, 1994)

3.1.2.3 The Representation of a Document and Its Properties

The document ID concept of IPC-1751 which is required for keeping consistency in between the declarations, can be represented by `document` definition in Part 41, as shown in Figure 11. The `document` entity has an *ID,* a `name`, and an optional `description` attributes. The `ID` attribute will be used to define the document ID. The `name` attribute will be defined as either "request," "reply," or "distribute." Each `document` has an association with `document_type` which can be used to define the type of document. For the purpose of this study, it is suggested that the document type is defined as "IPC-1752." The `document_representation_type` entity distinguishes the document format; e.g., a digital format, physical format, etc. The association between a document and a product is defined by the entity `document_product_association`. In this definition, related product can be any of the `product`, `product_definition`, or `product_definition_with_formation` entities of STEP. However, for the purpose of this study, the `product_definition_with_formation` will be allowed as the related product because only this entity shows the version information of the product definition. The relationship between a request document and the reply document can be captured by the entity `document_relationship`. In this definition, the request document should be specified by the relating document and the reply document should be specified by the related document. The four types of declaration that are defined as classes A, B, C, and D in IPC-1751 can be represented by the subclass of the `document` entity which is called `document_with_class`.

A standard legal definition or a custom defined legal definition is required for IPC-1751 forms to declare the liability of the company by providing the data in the IPC-175X forms. It is captured as a `document` of Part 41. The document type should be defined as “legal definition of declaration.” The `name` attribute of the `document` will be defined as either a standard legal definition or custom legal definition. In these documents, the legal definition, uncertainty information, and supplier acceptance have to be defined.

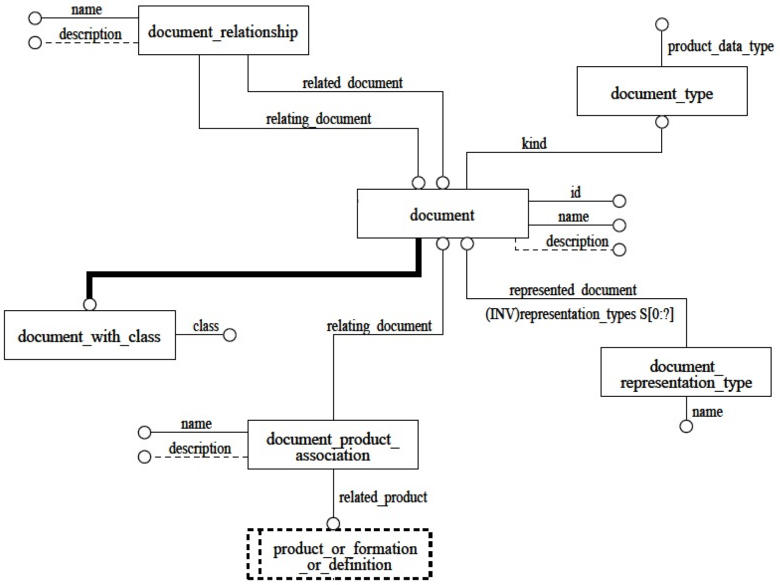


Figure 11: The EXPRESS-G representation of document definition in Part 41 (ISO 10303-41, 1994)

3.1.2.4 The Identification of a Product

The information required for the identification of the product will be captured from Part 41 of STEP. The related EXPRESS-G definition is given in Figure 12. `Name` and `id` attributes of the `product` entity correspond to the product name and the product number in IPC-1751 respectively. The product version information in Part 41 is captured by the entity `product_definition_formation`. The `id` attribute of this entity corresponds to the product version information. Each product version might have different product views like design, maintenance, etc., which is represented by the `product_definition` in Figure 12. Additionally, the entities `product_context` and product_definition_context have to be defined in Part 41. The engineering discipline's point of view from which data is being presented is captured by the product_context entity e.g., electrical, mechanical, software, etc. On the other hand, the product definition type and the life-cycle stage from which the data is viewed are captured by the entity product_definition_context e.g., design, manufacturing, etc. The entity product_definition_formation_with_specified_source which is the subset of product_definition_formation specifies the source of the product as made, bought, or unknown. This additional information explained above cannot be defined in IPC-1751.

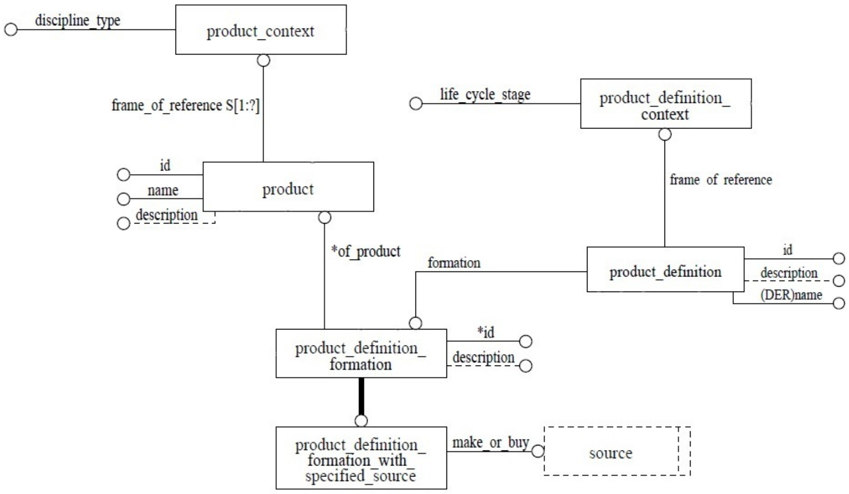


Figure 12: The EXPRESS-G representation of product definition in Part 41 (ISO 10303-41, 1994)

The effective date in IPC-1751 can be represented by the date entity as explained before. The mass of a product, unit of measure, and unit type information required for

IPC-1751 can be represented by the measure schema of Part 41. The measure schema has the required capability to define any value of measure with the unit, i.e. mass measure, length measure, etc. In this paper, only how to represent the product amount (i.e. mass of a product) and the unit of measure for the mass will be discussed (Figure 13) as an example. In STEP, any measure with a unit is represented by the entity `measure_with_unit`. It has several subsets, and each subset is different than the others (ONEOF property of STEP). The `measure_with_unit` entity has a `value_component` and a `unit_component` as evident from Figure 13. The `value_component` is a select type, namely `measure_value`, which has several data

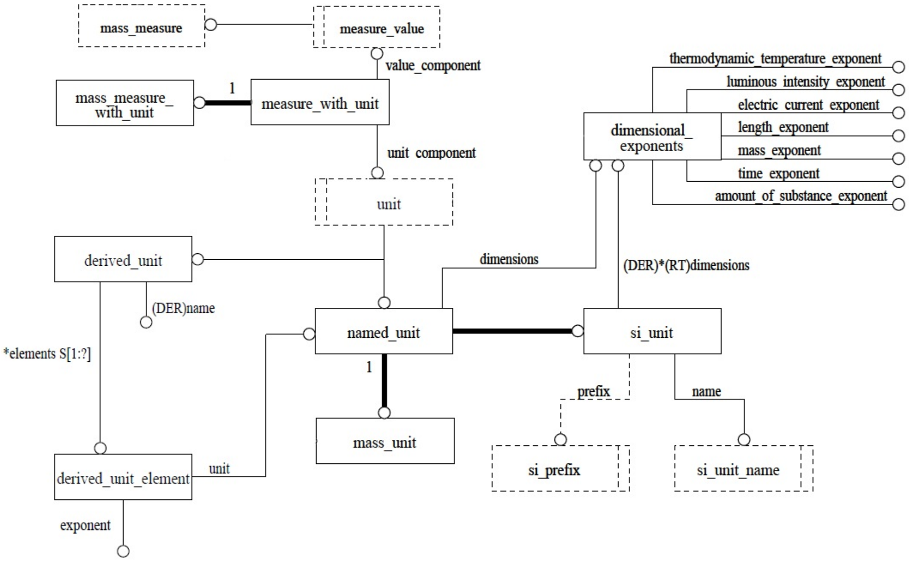


Figure 13: EXPRESS-G representation of mass measure with unit definition in Part 41 (ISO 10303-41, 1994)

types to be selected from. The `measure_value` select type has to be compliant with the `measure_with_unit` subset entity. For representing mass, the select type should be `mass_measure` which is compliant with `mass_measure_with_unit`. The `unit_component` of the `measure_with_unit` is also a select type, which is called `unit`. The select type `unit` can be either `derived_unit` or `named_unit`. The entity `named_unit` has also several subsets and each subset is different than the others. For representing mass, the subset of the `named_unit`, `mass_unit`, has to be instantiated. The `named_unit` is also superset of the `si_unit`. The `si_unit` has two attributes: a `prefix` and a `name`. The `prefix` attribute is a select type, which is called `si_prefix`, and it has the allowable values as “milli,” “centi,” “kilo,” etc. The `name` attribute of the `si_unit` is also a select type, which is called `si_unit_name`. For our case, the required `si_unit_name` is “gram.” Each `named_unit` and `si_unit` also has an attribute called `dimensions`. This attribute defines the `dimensional_exponents` for the unit. The `dimensional_exponents` capture the exponent values of the unit dimensions for the `named_unit` or the `si_unit`. To represent the “unit type” of IPC-1751, a `named_unit` with required `dimensional_exponents` can be instantiated.

3.1.2.5 The Structure of a Product

The product structure in STEP is defined in Part 44 (ISO 10303-44, 1994). Several types of product structure representation are possible in Part 44. In this paper, only the ones that can be used to represent the product structure of IPC-1751 will be discussed. The required entities are represented below in Figure 14.

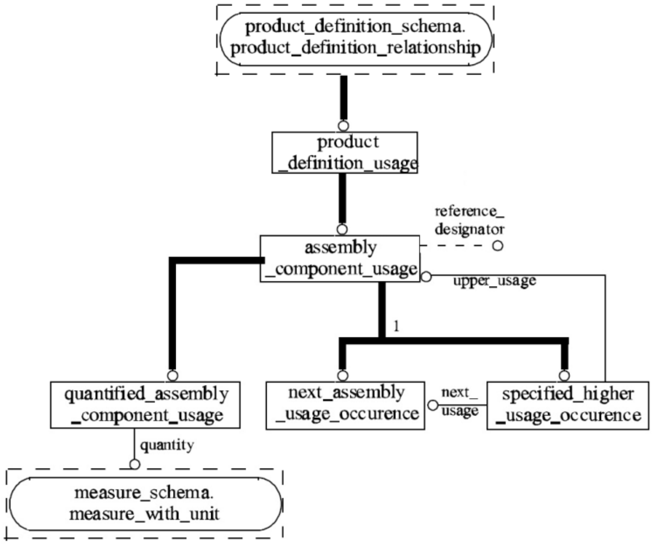


Figure 14: The EXPRESS-G representation of product structure in Part 44 (ISO 10303-44, 1994)

The traditional BOM in Part 44 can be represented by the entity `quantified_assembly_component_usage`. As evident from Figure 14, this entity is a subtype of the `entity product_definition_relationship`, and it has two attributes: `relating_product_definition` and the `related_product_definition`. The `relating_product_definition` is connected to the parent assembly, and the `related_product_definition` is connected to the part. The `quantified_assembly_component_usage` also states the number of components used in that particular assembly with the `quantity` attribute.

Another type of product structure definition, which is called the assembly tree structure, can be represented by connecting each component to its immediate parent assembly. This type of product structure can be represented by the entity next_assembly_usage_occurence of Part 44. If this type of representation is defined, then an algorithm is needed to convert this structure to the traditional BOM because each instance of the sub-product is represented by an instance of the next_assembly_usage_occurence entity. In IPC-1751, only traditional BOM is defined as the product structure definition.

In Part 44, a sub-assembly's components within an assembly can be identified by the entity specified_higher_usage_occurrence (e.g., the LCD screen of the lid sub-assembly of a laptop assembly). This entity is a subset of the product_definition_relationship (Figure 14). It has two attributes inherited from the parent: relating_product_definition and the related_product_definition. The related product definition is attached to the child product (e.g. LCD screen) that needs to be identified in higher parent assembly (e.g. laptop), which is not the immediate parent assembly (lid sub-assembly of laptop). The higher parent assembly, which is not the immediate parent assembly, is associated as the relating product definition. The intermediate parent assemblies are associated with next usage and upper usage attributes. This type of product structure also has to be converted to the traditional BOM for the IPC-1751 product structure definition.

3.2 Extraction of IPC-1752 Information Requirements from Integrated Resources of STEP

3.2.1 IPC-1752 Information Requirements

IPC-1752 (IPC-1752A, 2012) is the standard for the materials declaration management, which defines the information requirements of materials declaration for different regulations. With IPC-1752 data structure, it is possible to define and exchange the data of materials and substances that are present in products. In order to standardize the information requirements of materials declaration, there are four types of materials declaration in IPC-1752 that are called classes (Table 1). These classes can be combined in a single form for declaration as required. The UML information model for the declaration is also given in Figure 15.

In first type of declaration, which is class-A in Table 1, it collects simple true or false answers for a host of statements related to the product specifications. Each statement is called as a query in Figure 15. These statements can be standard ones that are given in IPC-1752, or it can be custom defined by the company. The second type of declaration, class-B in Table 1, requires the declaration of the materials information, which can be grouped under a specific category for the products. It is instantiated with a “material group list” in Figure 15. Each material group list has several “material groups.” The amount of each material group in the product has to be declared in terms of values and the unit of measure, as shown in Figure 15. The third type of declaration, class-C in Table

1, can be used to report JIG-101 substances at the product level. In order to declare this materials information, “Substance Category List,” in Figure 15, should be created. Each list has several “Substance Categories.” Each “Substance Category” has a “name,” “description of use,” “homogeneous material,” “threshold,” “concentration,” “substance,” and “amount” information. The homogeneous material information captures whether the substance defined is at the homogeneous level (True) or at the product level (False). The threshold information captures the level of threshold, whether the substance group is over threshold (true/false) and whether the substance is added intentionally (true/false). In Figure 15, the amount of each substance group in the product has to

Table 1: Material Declaration Classification in IPC-1752

Class Description	Declaration Type	Detailed	Requirements
A	− Reporting in Query/Reply format	Query/Reply	− Supplier provides responses to standard queries and/or optional custom queries.
B	− Material group reporting	Material Group	− Supplier states the amount of different groupings of materials within a product.
C	− JIG-101 substance category reporting for the product − Additional substance categories reporting at the product level	Substance summary groups	− Supplier provides mass and/or concentration of JIG-101 substance category at the product level if above thresholds. − Additional substance categories can be added and reported at the product level.
D	− Substances reporting at the homogeneous material level − JIG-101 substances and additional substances are accommodated	Full substances	− Supplier provides location, mass, substances at homogeneous material level.

be declared as a value and a unit of measure. Any substance in a substance group is defined by a “name,” “unique ID,” “amount,” and “exemption” information. The unique ID identifies each substance with the authority that assigns the ID to the substance. The exemption information for IPC-1752 is retrieved from the directives, such as RoHS and REACH. The last type of declaration, class D in Table 1, is used to declare the

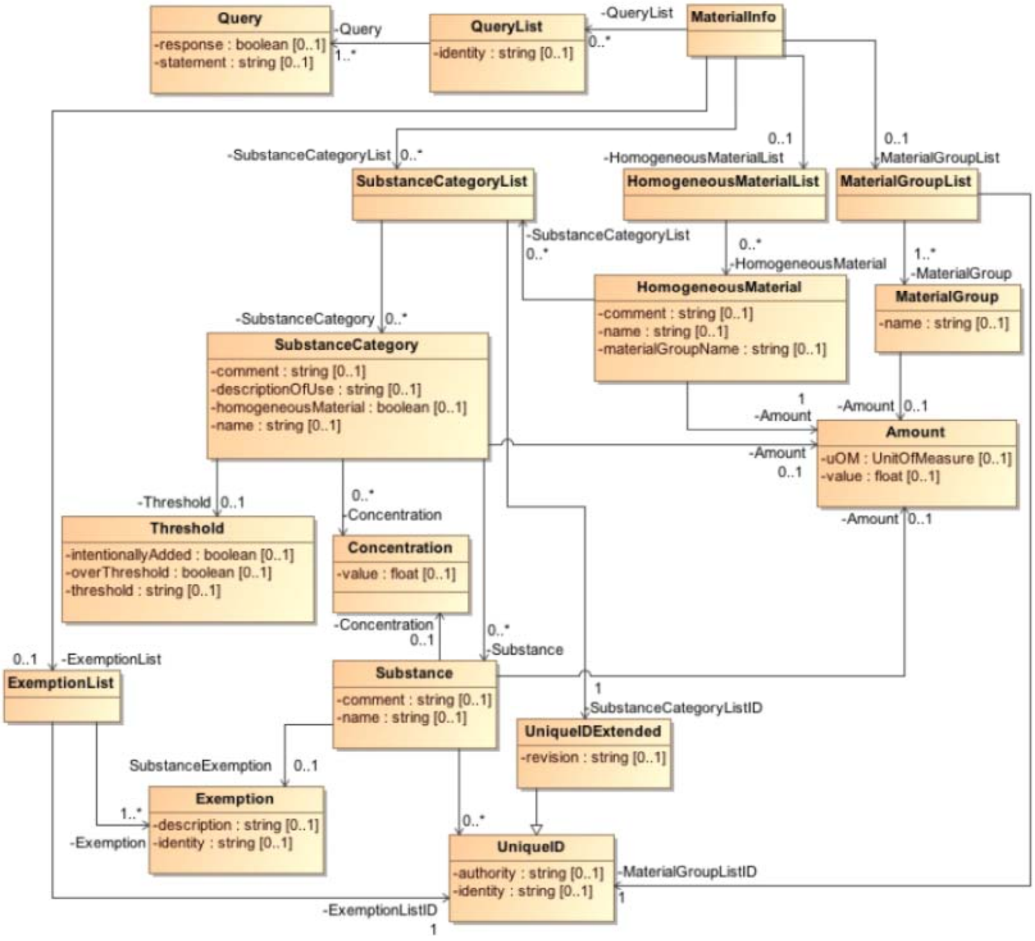


Figure 15: UML information model for IPC-1752 (IPC-1752A, 2012)

homogeneous substance level information in the product. For that purpose, a "Homogeneous Material List," in Figure 15, should be instantiated. Each homogeneous material might have a "name," a "Material Group Name," a "Substance Category List," or an "Amount" information.

3.2.2 Extraction of IPC-1752 Information requirements from Integrated Resources of STEP

For the purpose of this study, we would like to discuss: (1) material characterizations as needed for IPC-1752, (2) how to represent material and substance information of a product with STEP information model, and (3) how to create the required information for IPC-1752 declaration classes.

3.2.2.1 The Material Information needed for IPC-1752

The information requirements for IPC-1752 are discussed in detail in section 3.2.1. In this section, we would like to characterize the material information contents that need to be extracted from the STEP standards.

Since IPC-1752 is related to the materials declaration, the material names assigned to a product and the amounts of these materials have to be extracted. In addition, IPC-1752 also needs the declarations of the following information related to:

(1) materials that can be grouped under common categories and amounts of these material groups in a product;

(2) materials information for the whole product or, a small portion of that product to define the homogeneous material information (it is needed to describe any materials that can be separated from the base product). This is required for the regulatory directives like RoHS and REACH;

(3) exemptions defined in these directives that are available in the declaration form. For each exemption, whether it is true or not has to be specified in the IPC-1752 form for the product; and

(4) concentrations of the declared materials, whether the substance defined is added intentionally or not.

3.2.2.2 The Materials Representations Schema Available in STEP

The materials information is represented by Part 45 of STEP standards. The information model of Part 45 can be used to represent substance level information for compliance management. In Part 45, any material is defined as a product by instantiating the `product` entity, as given in Figure 16. An `id` and `name` can be defined for the material or substance. The `product_related_product_category` entity is used to differentiate between a raw material and a mechanical product. This entity is a subset of the product_category entity. The name attribute of the product_related_product_category should be raw material.

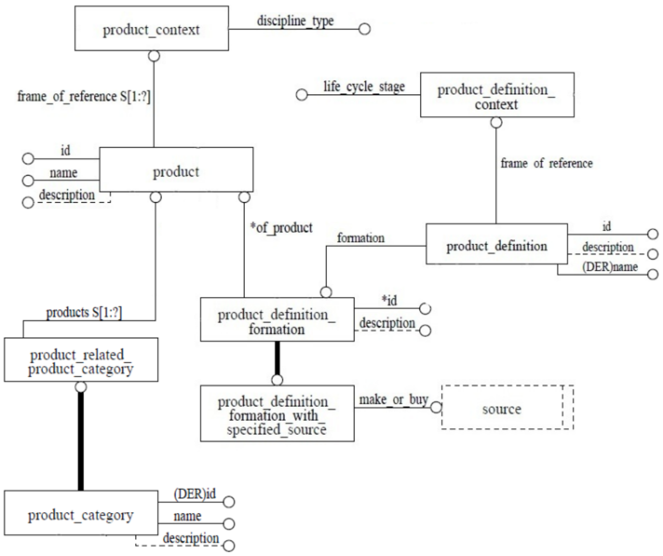


Figure 16: The EXPRESS-G representation of representing materials as product in Part 41 (ISO 10303-41, 1994)

In Part 45, any material designation is represented by the entity material_designation, as shown in Figure 17. This entity is associated to a select type, called characterized_definition, and it is used as a mechanism to associate the material designation to any product, product_definition, or only some part of the product shape definition. The characterized_definition is explained in detail later in this section (Figure 18). The material_designation entity will be used to represent the unique ID concept of materials and substances in IPC-1752.

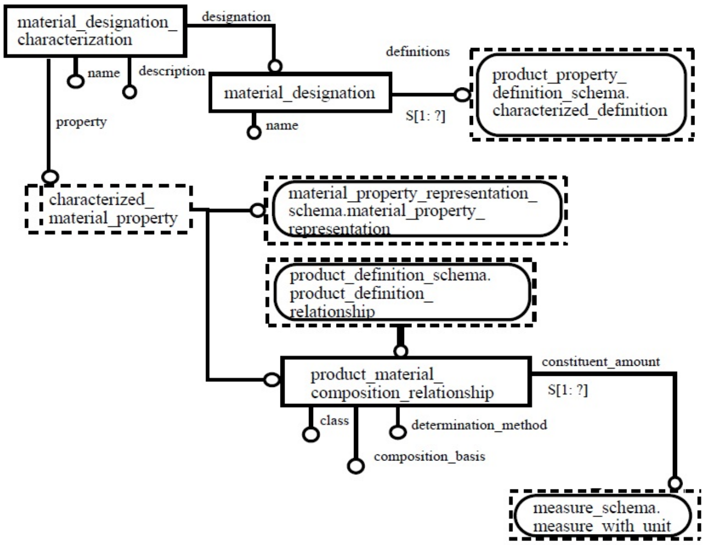

material_designation_ characterization
designation
definitions
product_property_ definition_schema. characterized_definition
name
description
material_designation
S[1: ?]
name
property
characterized_ material_property
material_property_representation_ schema.material_property_ representation
product_definition_schema. product_definition_ relationship
product_material_ composition_relationship
constituent_amount
S[1: ?]
class
determination_method
composition_basis
measure_schema. measure_with_unit

Figure 17: The EXPRESS-G representation of material designation and material composition in Part 45 (ISO 10303-45, 1997)

In Part 45, the mass of any material, such as the constituents of any material and the mass of any substance, have different ways of representation. In Part 45, either a material property definition or the product material composition relationship definition can be associated to a `material_designation` (Figure 17). This association is represented by the entity `material_designation_characterization`. In order to define the constituents of a material product, an instance of the `material_designation_characterization` has to be created for each constituent substance of a product. Each constituent is then identified by the entity `product_material_composition_relationship`. This entity aggregates the class information for the kind of relationship between a constituent and a homogeneous product (e.g. mixture, alloyed, chemically bonded, etc.), the amount of constituent, the

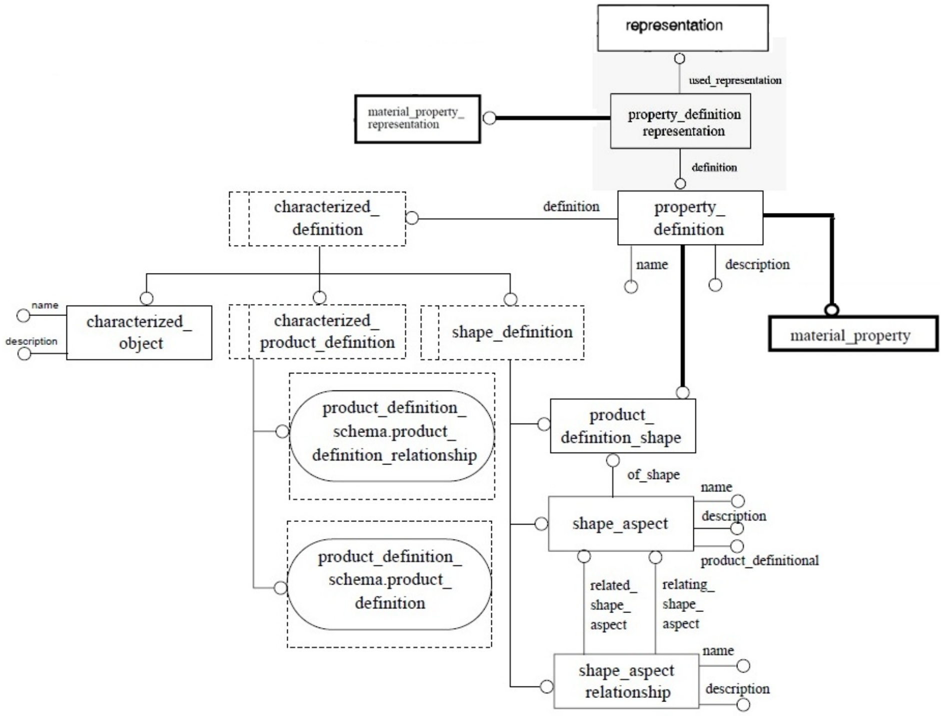


Figure 18: The EXPRESS-G representation of material property in Part 45 (ISO 10303-45, 1997)

basis of composition (e.g. volume, weight, mole, atoms, etc.), and the method (of determination) by which the amount of constituent is determined. This entity is a subset of the entity `product_definition_relationship`, and it inherits five attributes: `id`, `name`, `description` (optional), `relating_product_definition`, and `related_product_definition`. The `relating_product_definition` attribute should be connected to the `product_definition` of the base material. The `related_product_definition` attribute should be connected to the `product_definition` of the constituent material. The constituents of a product that are defined through the entity `product_material_composition_relationship` can be used as the substance representation at the product level for IPC-1752. The `constituent_amount` attribute represents the amount of the substance and the unit of measure. The representation of the amount of mass and the unit of mass measure are exactly the same as given previously. For each constituent material, a `material_designation` entity should be instantiated to define the ID of the constituent.

In order to represent the mass of a material or a product in Part 45, the entity `material_property` in Figure 18 will be used. It is a subset of `property_definition` which has `name`, `description`, and `definition` attributes. The `definition` attribute is connected to the `characterized_definition` select type. The `product_definition` entity (Figure 18) is selected for the `definition` attribute of the `material_property` through the `characterized_definition` select type for the product or material for which the mass has to be defined. Then, the `material_property_representation` entity (Figure 18) has to be instantiated, which is a subset of `property_definition_representation`. It inherits two attributes from the superset entity: `definition` and `used_representation`. The `definition` attribute will be connected to the instantiated `material_property`. The `used_representation` attribute will be connected to the `representation` entity. In Figure 19, how the `representation` entity is related to the measure_representation_item is shown. The measure_representation_item is a subset of both the representation_item and measure_with_unit. The value and unit of mass will be represented through measure_representation_item.

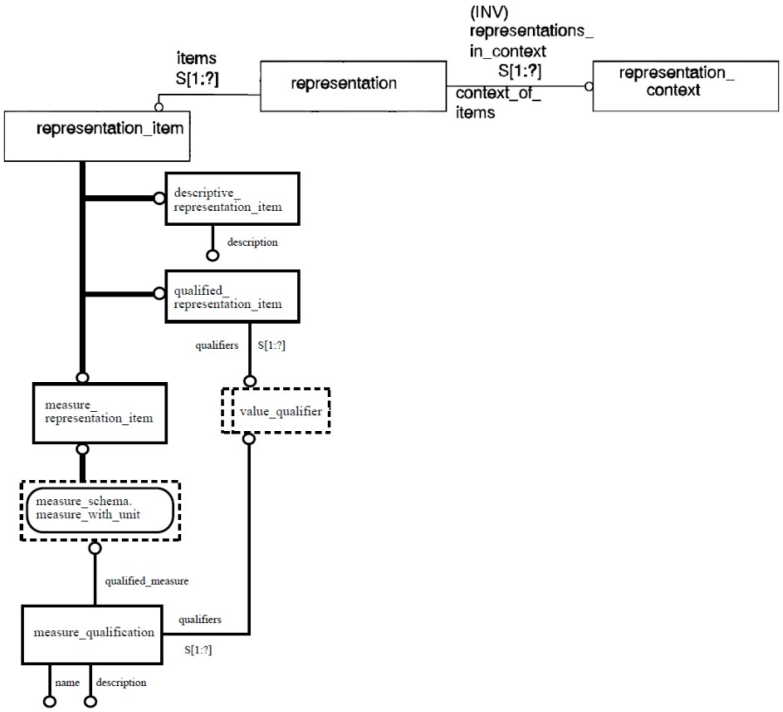


Figure 19: The EXPRESS-G representation of measure representation item in Part 45 (ISO 10303-45, 1997)

In order to represent the homogeneous level substance information of IPC-1752, the material_designation entity has to be instantiated for the given substance of interest (Figure 17). The definitions attribute of this entity should be connected to the characterized_definition select type. The allowable select types for the characterized_definition are shown in Figure 18. The shape_aspect entity

(Figure 18) is selected to define the homogenous level substance information. The `shape_aspect` defines an element of the shape, which is identifiable for a product. Hence, this entity is used to collect the information to identify the part of the product where homogeneous substance is defined. Then, substance information can be assigned through the `material_designation` entity, as explained above.

3.2.2.3 The Representation of IPC-1752 Material Requirements Using STEP Schema

All the information, which can be extracted from integrated resources of STEP, can be used to create each declaration class of IPC-1752 (Table 1). In this section, the representation of these declaration classes of IPC-1752 in STEP formats is discussed.

In order to represent queries defined in class-A declaration of IPC-1752, the `material_property` entity will be instantiated (Figure 18). The `name` attribute of the `material_property` will be defined as the exact query list name of the IPC-1752, e.g. "EURoHS-0508," The `definition` attribute will be connected to the `product_definition` entity for which the class-A declaration is being reported; the `used_representation` attribute of the `material_property_representation` will be connected to the `representation`; and it will be connected to a set of `descriptive_representation_items` (Figure 19). Each `descriptive_representation_item` will describe a query of the query list in IPC-

1752. Unfortunately, a “True/False” statement for any query that is required in IPC-1752 is not available for the `descriptive_representation_item`. Our solution to this problem is to define the “True/False” statement of any query inside the `descriptive_representation_item` after a column “:”. For example, “Product(s) meets EU RoHS requierement without any exemptions: True” will be instantiated as a `descriptive_representation_item`.

The class-B declaration of IPC-1752 requires the declaration of the material groups (e.g. non-ferrous metals) and amounts of these groups in the product. The material group information in IPC-1752 is actually derived information from the original material content information of a product. Because of that reason, the material group information should be defined as a material property definition. From Figure 18, the `name` attribute of the `material_property` will be defined as the material group name; the `definition` attribute will be connected to the `product_definition`. Then, the amount of mass for the material group can be represented through the `material_property_representation` as explained previously.

The substance information at the product level should be declared for class-C declaration of IPC-1752. This information can be extracted by the entity `product_material_composition_relationship`, as explained before.

The substance information at the homogeneous material level should be declared for class-D declaration of IPC-1752. This information required for the class-D forms of IPC-1752 can be extracted from the materials information that has been assigned directly to the shape_aspect which specifies the part of the product for the homogeneous material.

3.2.3 IPC-1752 Information Representation Capabilities of Application Protocols: AP203 and AP214

Both AP203 2nd Ed. (ISO 10303-203, 2011) and AP214 3rd Ed. (ISO 10303-214, 2010) can represent the product and the product structure information. The person, organization, personal_address, and organizational_address can be represented. However, the person_role is not defined in either AP203 or AP214. This is a problem for extracting the title information in IPC-1751 from AP203 or AP214. In AP214, the organization_role is represented. This entity is associated with the organization and with the organization_assignment. AP203 does not capture that information. Instead, AP203 has another entity, which assigns a role to both the person and organization. It is called person_and_organization_assignment. The role is defined by the entity person_and_organization_role. This causes a problem in identifying the separate title information for person and role information of a company for IPC-175X forms from AP 203 STEP files. Both AP 203 and AP 214 can capture the date information. Even though AP 214 has less representation capabilities for `date`, the XML date type can be easily extracted. The representation of the IPC-1752 declaration classes is mapped into the `document_with_class` entity. This entity is not available in AP203 and AP214. Other than this entity, the other representation capabilities explained in section 3.1.2.3 can be represented by both AP 203 and AP 214.

All Part 45 material property and material constituent representations with the uncertainty definitions can also be represented with AP 203. On the other hand, AP 214 can only capture the material property representation with limited uncertainty definitions. AP 214 cannot capture the coverage factor of uncertainty, which is a multiplier of the standard deviation for defining the uncertainty. Also, AP 214 cannot capture the material constituent information, as well as the relationships between several data environment definitions.

3.3 Case Study: A Gearbox

In this section, a case study of a simple gearbox assembly is implemented to show the extraction of STEP information. The parts of the gearbox and the assembly are created in a CAD system. The assembly is shown in Figure 20. A traditional BOM for the gearbox assembly is also given in Table 2.

The gearbox has one lower case (i.e. the lower half of the gear assembly) and one upper case (i.e. the upper half of the gear assembly) as a container. It has two shafts that hold two different spur gears. The spur gears have 20 mm and 40 mm pitch diameters.

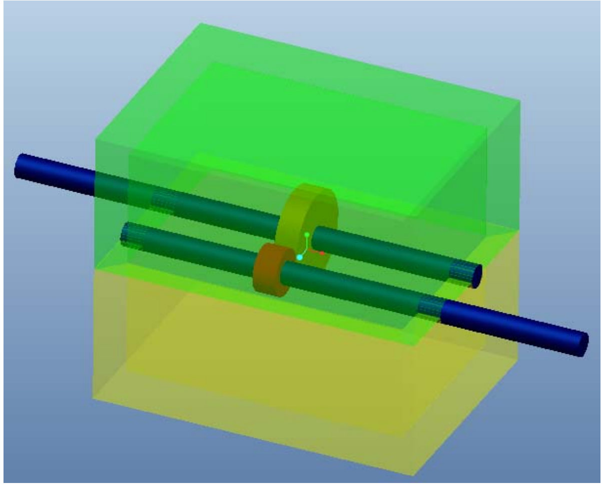
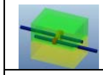
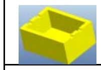
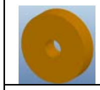
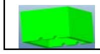
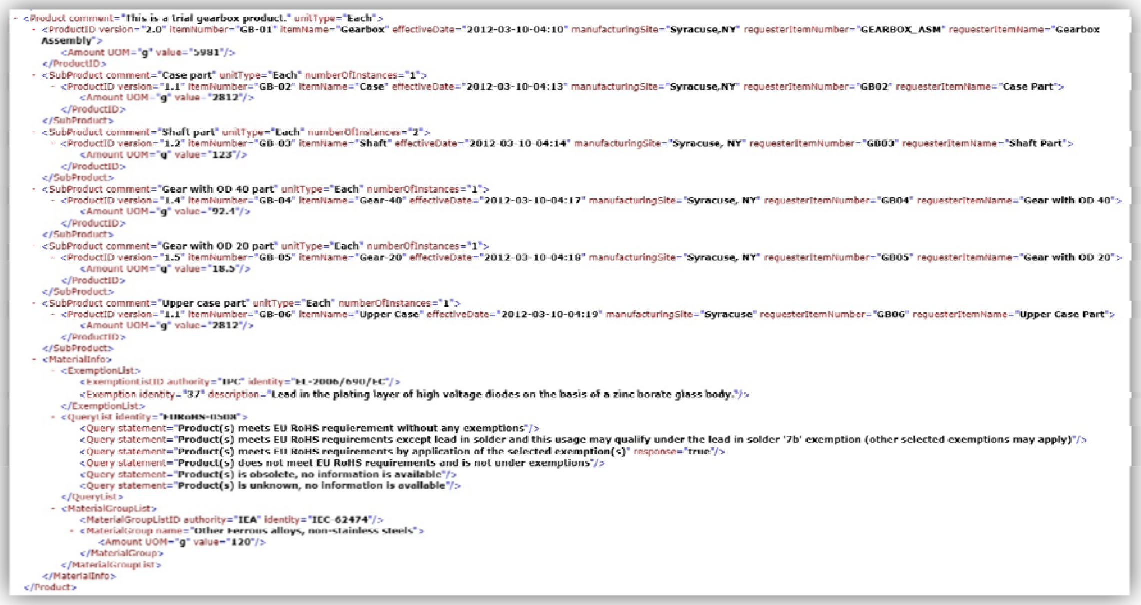


Figure 20: A typical gearbox assembly

Table 2: BOM for the Gearbox Assembly

Product Figure	Product Name	Quantity
	Gearbox assembly	1
	Case_lower	1
	Shaft	2
	Gear_40	1
	Gear_20	1
	Case_upper	1

A part of an IPC-1752 XML file that shows the product structure information is given in Figure 21. As it is evident from the XML representation, the product that is in context



```
- <Product comment="This is a trial gearbox product." unitType="Each">
  - <ProductID version="2.0" itemNumber="GB-01" itemName="Gearbox" effectiveDate="2012-03-10-04:10" manufacturingSite="Syracuse,NY" requesterItemNumber="GEARBOX_ASM" requesterItemName="Gearbox
    Assembly">
       <Amount UOM="g" value="5981"/>
    </ProductID>
  - <SubProduct comment="Case part" unitType="Each" numberOfInstances="1">
    - <ProductID version="1.1" itemNumber="GB-02" itemName="Case" effectiveDate="2012-03-10-04:13" manufacturingSite="Syracuse,NY" requesterItemNumber="GB02" requesterItemName="Case Part">
         <Amount UOM="g" value="2812"/>
      </ProductID>
    </SubProduct>
  - <SubProduct comment="Shaft part" unitType="Each" numberOfInstances="2">
    - <ProductID version="1.2" itemNumber="GB-03" itemName="Shaft" effectiveDate="2012-03-10-04:14" manufacturingSite="Syracuse, NY" requesterItemNumber="GB03" requesterItemName="Shaft Part">
         <Amount UOM="g" value="123"/>
      </ProductID>
    </SubProduct>
  - <SubProduct comment="Gear with OD 40 part" unitType="Each" numberOfInstances="1">
    - <ProductID version="1.4" itemNumber="GB-04" itemName="Gear-40" effectiveDate="2012-03-10-04:17" manufacturingSite="Syracuse, NY" requesterItemNumber="GB04" requesterItemName="Gear with OD 40">
         <Amount UOM="g" value="92.4"/>
      </ProductID>
    </SubProduct>
  - <SubProduct comment="Gear with OD 20 part" unitType="Each" numberOfInstances="1">
    - <ProductID version="1.5" itemNumber="GB-05" itemName="Gear-20" effectiveDate="2012-03-10-04:18" manufacturingSite="Syracuse, NY" requesterItemNumber="GB05" requesterItemName="Gear with OD 20">
         <Amount UOM="g" value="18.5"/>
      </ProductID>
    </SubProduct>
  - <SubProduct comment="Upper case part" unitType="Each" numberOfInstances="1">
    - <ProductID version="1.1" itemNumber="GB-06" itemName="Upper Case" effectiveDate="2012-03-10-04:19" manufacturingSite="Syracuse" requesterItemNumber="GB06" requesterItemName="Upper Case Part">
         <Amount UOM="g" value="2812"/>
      </ProductID>
    </SubProduct>
  - <MaterialInfo>
    - <ExemptionList>
         <ExemptionListID authority="IPC" identity="EL-2006/690/EC"/>
         <Exemption identity="37" description="Lead in the plating layer of high voltage diodes on the basis of a zinc borate glass body."/>
      </ExemptionList>
    - <QueryList identity="EURoHS-0508">
         <Query statement="Product(s) meets EU RoHS requierement without any exemptions"/>
         <Query statement="Product(s) meets EU RoHS requirements except lead in solder and this usage may qualify under the lead in solder '7b' exemption (other selected exemptions may apply)"/>
         <Query statement="Product(s) meets EU RoHS requirements by application of the selected exemption(s)" response="true"/>
         <Query statement="Product(s) does not meet EU RoHS requirements and is not under exemptions"/>
         <Query statement="Product(s) is obsolete, no information is available"/>
         <Query statement="Product(s) is unknown, no information is available"/>
      </QueryList>
    - <MaterialGroupList>
         <MaterialGroupListID authority="IEA" identity="IEC-62474"/>
       - <MaterialGroup name="Other Ferrous alloys, non-stainless steels">
            <Amount UOM="g" value="120"/>
         </MaterialGroup>
      </MaterialGroupList>
    </MaterialInfo>
</Product>
```

Figure 21: The XML representation of the gearbox product structure in IPC-1751 form

has to be identified, as explained before. In Figure 22, the EXPRESS-G instantiation of the `product` for the gearbox assembly has been shown. The product number and the product name are captured by `id` and `name` attributes of the `product` entity, respectively. The product version information is given by the entity `product_definition_formation_with_specified_source` which is a subset of `product_definiton_formation`. The `id` attribute defines the product version number. As explained in previous sections, a product version might have several product definitions for different application contexts, like design, maintenance, etc. In this case, the product_definition is represented for design view. The context of the product is given as the mechanical context in Figure 22.

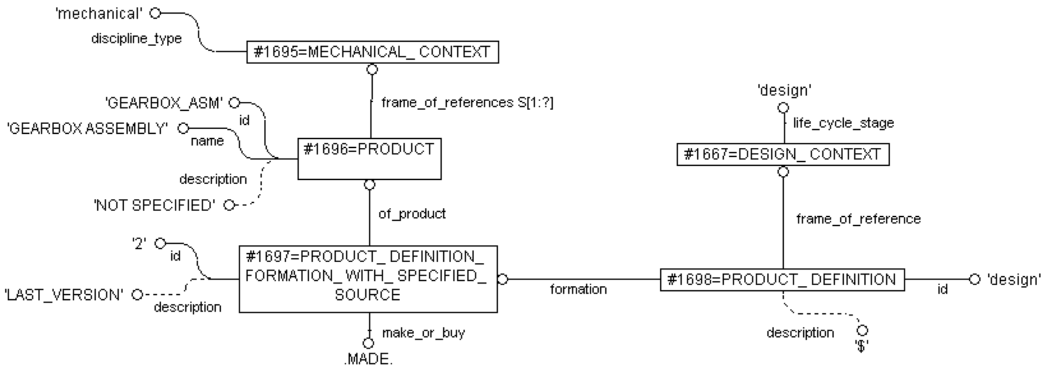


Figure 22: The EXPRESS-G instance representation of the product for the gearbox assembly

In IPC-175X forms, the product structure is represented as a simple traditional BOM (Figure 21). As explained in section 3.1.2.5, there are several ways to represent the product structure in STEP. It is possible to create the traditional BOM out of each product structure representation. In Figure 23, the traditional BOM is shown by instantiating the quantified_assembly_component_usage entity. For each component (i.e. product_definition), there is one quantified_assembly_component_usage entity. The related_product_definition attribute is connected to the component part, whereas the relating_product_definition attribute is connected to the parent assembly (i.e. gearbox product_definition). Then, the quantity attribute captures the number of component used in the parent assembly.

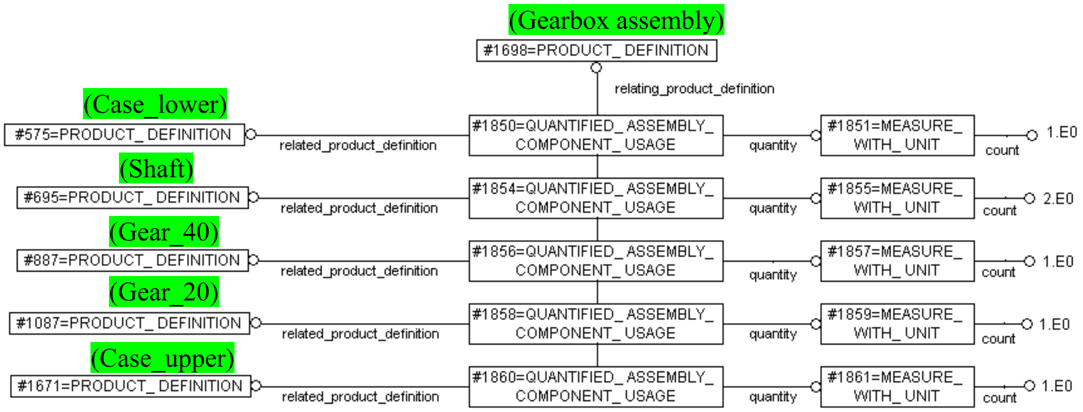

(Gearbox assembly)
#1698=PRODUCT_ DEFINITION
relating_product_definition
(Case_lower)
#575=PRODUCT_ DEFINITION
related_product_definition
#1850=QUANTIFIED_ ASSEMBLY_ COMPONENT_ USAGE
quantity
#1851=MEASURE_ WITH_ UNIT
count
1.E0
(Shaft)
#695=PRODUCT_ DEFINITION
related_product_definition
#1854=QUANTIFIED_ ASSEMBLY_ COMPONENT_ USAGE
quantity
#1855=MEASURE_ WITH_ UNIT
count
2.E0
(Gear_40)
#887=PRODUCT_ DEFINITION
related_product_definition
#1856=QUANTIFIED_ ASSEMBLY_ COMPONENT_ USAGE
quantity
#1857=MEASURE_ WITH_ UNIT
count
1.E0
(Gear_20)
#1087=PRODUCT_ DEFINITION
related_product_definition
#1858=QUANTIFIED_ ASSEMBLY_ COMPONENT_ USAGE
quantity
#1859=MEASURE_ WITH_ UNIT
count
1.E0
(Case_upper)
#1671=PRODUCT_ DEFINITION
related_product_definition
#1860=QUANTIFIED_ ASSEMBLY_ COMPONENT_ USAGE
quantity
#1861=MEASURE_ WITH_ UNIT
count
1.E0

Figure 23: The EXPRESS-G instance representation of BOM for the gearbox assembly

If the product structure is represented by the entity `next_assembly_usage_occurence`, it will be represented as shown in Figure 24. In this representation, the component parts are represented only once by the `product_definition` entity which collects all the information related to the component part. Then, there should be one `next_assembly_usage_occurence` entity instantiated for each component used in the parent assembly because the number of each component used in the parent assembly is represented by the instances of the `next_assembly_usage_occurence`. As shown in Figure 24, there are two instantiations of the `next_assembly_usage_occurence` for the shaft component which is compliant with our BOM (Table 2).

Another product structure representation in STEP is one that can be used to relate individual components within any sub-assembly or assembly in which they are included.

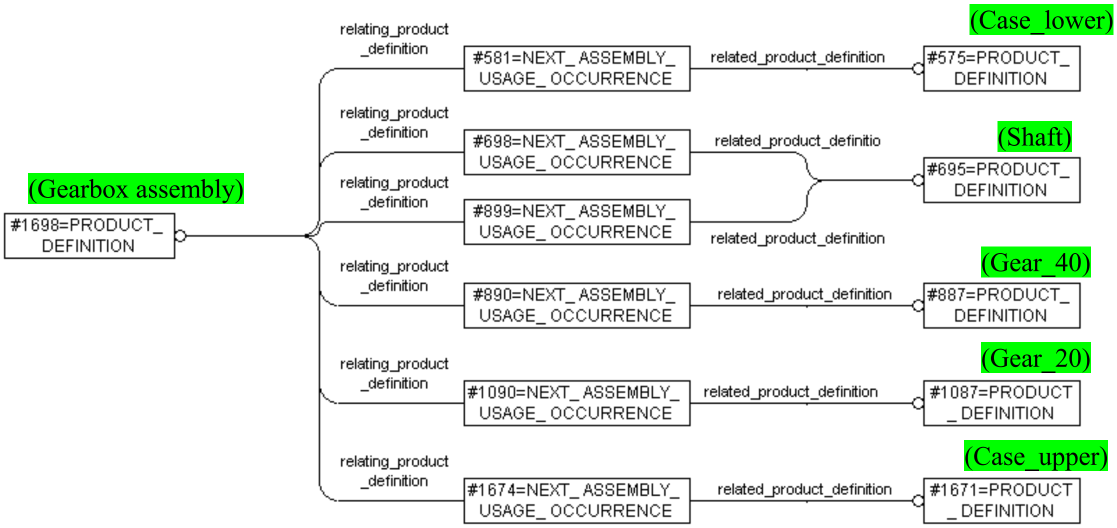

(Case_lower)
relating_product _definition
#581=NEXT_ASSEMBLY_ USAGE_OCCURRENCE
related_product_definition
#575=PRODUCT_ DEFINITION
relating_product _definition
#698=NEXT_ASSEMBLY_ USAGE_OCCURRENCE
related_product_definitio
(Shaft)
#695=PRODUCT_ DEFINITION
(Gearbox assembly)
#1698=PRODUCT_ DEFINITION
relating_product _definition
#899=NEXT_ASSEMBLY_ USAGE_OCCURRENCE
related_product_definition
(Gear_40)
relating_product _definition
#890=NEXT_ASSEMBLY_ USAGE_OCCURRENCE
related_product_definition
#887=PRODUCT_ DEFINITION
(Gear_20)
relating_product _definition
#1090=NEXT_ASSEMBLY_ USAGE_OCCURRENCE
related_product_definition
#1087=PRODUCT _DEFINITION
(Case_upper)
relating_product _definition
#1674=NEXT_ASSEMBLY_ USAGE_OCCURRENCE
related_product_definition
#1671=PRODUCT _DEFINITION

Figure 24: The simple assembly tree structure representation for the gearbox assembly in EXPRESS-G

This is achieved by the entity `specified_higher_usage_occurrence`. In Figure 25, an EXPRESS-G instantiation of this entity is represented for the shaft component in a gearbox. For the sake of this example, the shaft and the gear are combined to a subassembly. Then, this subassembly is defined as a component of the gearbox assembly. In our conceptual gearbox model for this case, there are two shaft and gear subassemblies. Two shafts are identical in the parent assembly. There is only one `product_definition` instance (i.e. instance #695 in Figure 25) which collects all the details of the shaft product. To identify one of the usages of the shaft component in two subassemblies of the shaft and gear, the `specified_higher_usage_occurence` entity is instantiated (i.e. instance #1866 in Figure 25). By this way, it is possible to identify the usage of the shaft in a particular subassembly in the gearbox assembly.

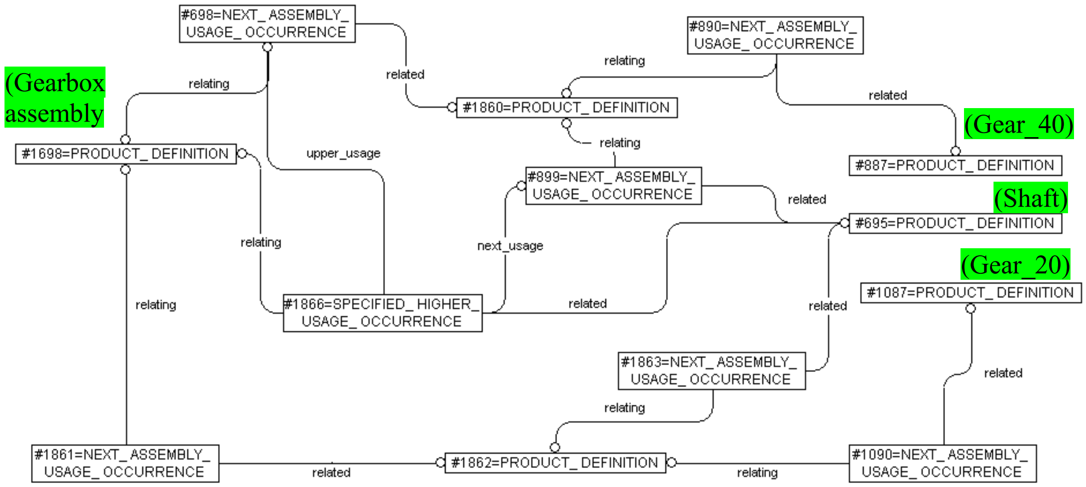

#698=NEXT_ ASSEMBLY_ USAGE_ OCCURRENCE
#890=NEXT_ ASSEMBLY_ USAGE_ OCCURRENCE
(Gearbox assembly)
relating
related
relating
related
#1860=PRODUCT_ DEFINITION
(Gear_40)
#1698=PRODUCT_ DEFINITION
upper_usage
relating
#887=PRODUCT_ DEFINITION
#899=NEXT_ ASSEMBLY_ USAGE_ OCCURRENCE
related
(Shaft)
#695=PRODUCT_ DEFINITION
relating
next_usage
(Gear_20)
#1087=PRODUCT_ DEFINITION
relating
#1866=SPECIFIED_ HIGHER_ USAGE_ OCCURRENCE
related
related
#1863=NEXT_ ASSEMBLY_ USAGE_ OCCURRENCE
related
relating
#1861=NEXT_ ASSEMBLY_ USAGE_ OCCURRENCE
related
#1862=PRODUCT_ DEFINITION
relating
#1090=NEXT_ ASSEMBLY_ USAGE_ OCCURRENCE

Figure 25: The complex assembly tree structure representation for the gearbox assembly in EXPRESS-G

In order to represent the material and substance information of any product in STEP, they need to be instantiated as a product because any substance or any material is represented as a product in STEP. The EXPRESS-G representation of iron as a raw material is shown in Figure 26.

In order to represent the material composition of any product, the `material_designation` should be assigned to the `product` in STEP. In Figure 27, plain carbon steel, AISI1040, will be assigned to a product which is called “block” for an EXPRESS-G representation. As shown in the figure, `material_designation` is associated to the `product_definition` entity for the “block” because a `product` might have several versions, where each version might have a different assigned material. The materials assignment to all other parts of the gearbox should be done the same way.

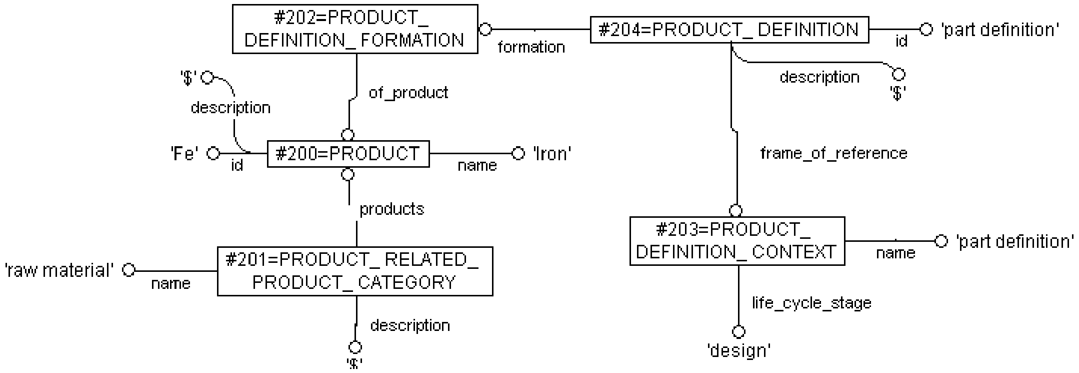


Figure 26: The EXPRESS-G representation of "Iron" as raw material

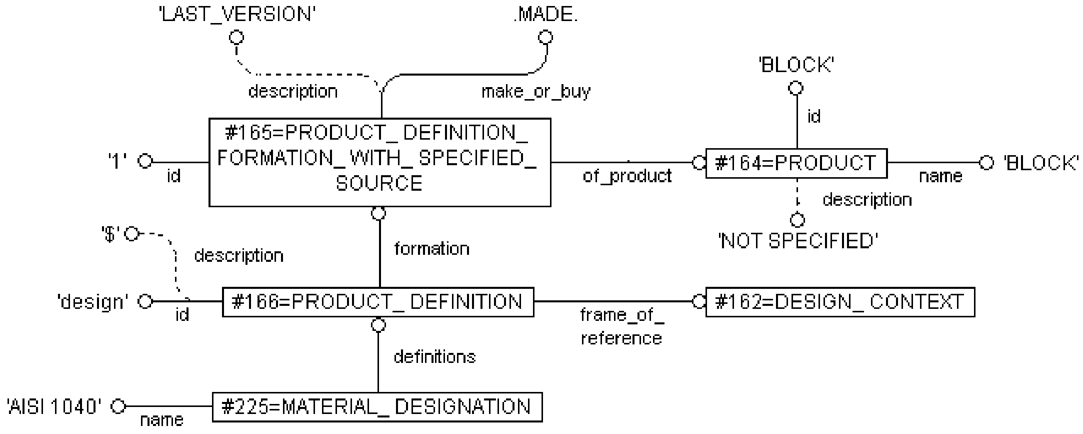


Figure 27: The EXPRESS-G representation of material designation for AISI 1040

In Figure 28, how to represent a constituent material of a homogeneous product (i.e. iron for AISI 1040 material), the amount of this constituent is represented. This is achieved by instantiating the `product_material_composition_relationship` entity. The relationship between `material_designation` and `product_definition` is achieved through `material_designation_characterization`. In Figure 28, an example "iron" constituent for AISI 1040 plain carbon steel is shown. For each constituent substance of product material, another `material_designation_characterization` has to be instantiated. In Part 45, the constructs for the uncertainty definition of the measured values are also given. The uncertainty definition is not required for IPC-1752, but is defined in STEP. In Figure 28, the uncertainty of the measure values for the 'iron' content has been associated as minimum and maximum values.

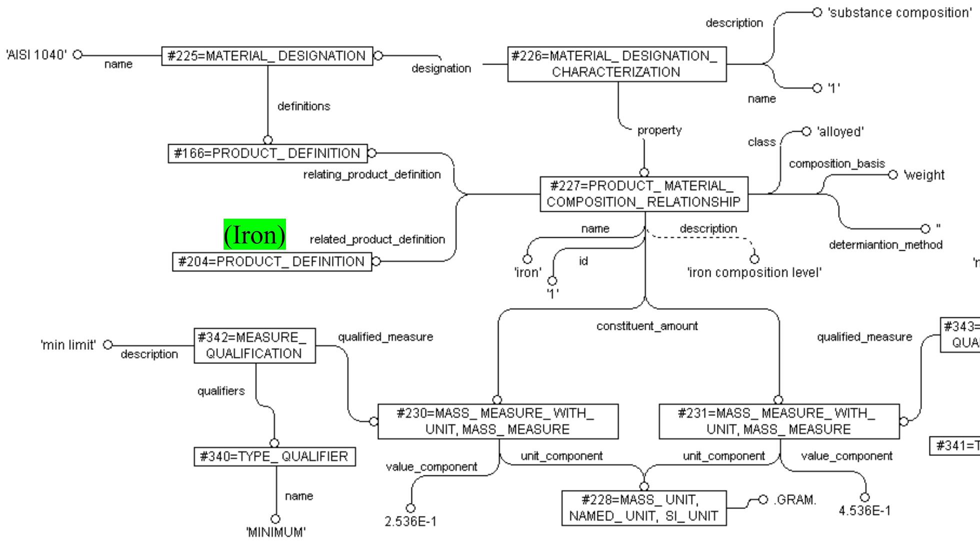


Figure 28: The EXPRESS-G representation of "Iron" as a constituent of a homogeneous product

A material property (e.g. density, mass, etc.) is assigned to a product through the `material_property` and `material_property_representation` entities. In Figure 29, the density of AISI 1040 material, which is assigned to the product "block," is defined. The value and the unit of the density are defined through the `material_property_representation` entity. The material property, density, has been selected on purpose to show how to instantiate a derived unit. Also, representing mass unit has been shown for constituents of a product in the previous example. In order to represent the mass of a product as a material property in Figure 29, `material_property` should be instantiated with a `name` attribute of “mass.” The `material_property_representation` should be connected to `representation`, and the `representation` should have `items` `mass_measure_with_unit` as instantiated in Figure 28.

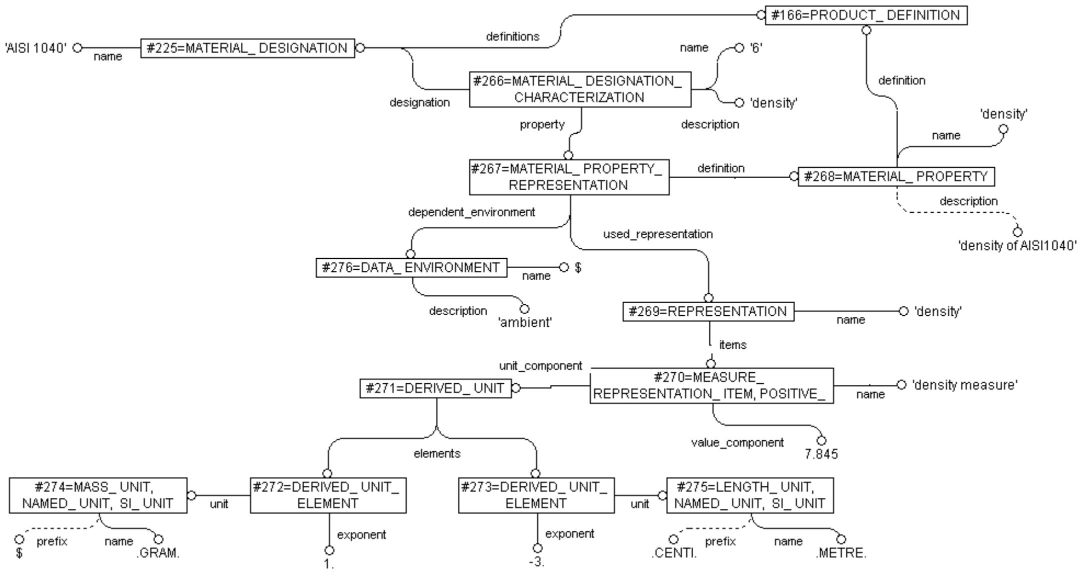
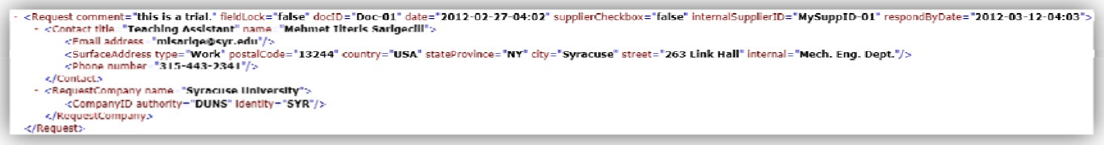


Figure 29: The EXPRESS-G representation of material property assignment

Part of an IPC-1752 XML file which shows the contact information and company information that have to be filled out in order to send the form is given in Figure 30. In Figure 31, example instantiation of `person` and `organization` has been shown. The `person` entity keeps `id`, `first`, `middle`, and `last names` for the person. The `organization` entity keeps the required `id` and `name` information for the organization. The relationship between the `person` and the `organization` is captured by the `person_and_organization` entity.



```
- <Request comment="this is a trial." fieldLock="false" docID="Doc-01" date="2012-02-27-04:02" supplierCheckbox="false" internalSupplierID="MySuppID-01" respondByDate="2012-03-12-04:03">
  - <Contact title="Teaching Assistant" name="Mehmet Ilteris Sarigecili">
      <Email address="msarige@syr.edu"/>
      <SurfaceAddress type="Work" postalCode="13244" country="USA" stateProvince="NY" city="Syracuse" street="263 Link Hall" internal="Mech. Eng. Dept."/>
      <Phone number="315-443-2341"/>
    </Contact>
  - <RequestCompany name="Syracuse University">
      <CompanyID authority="DUNS" identity="SYR"/>
    </RequestCompany>
  </Request>
```

Figure 30: The XML representation of request information in IPC-1751

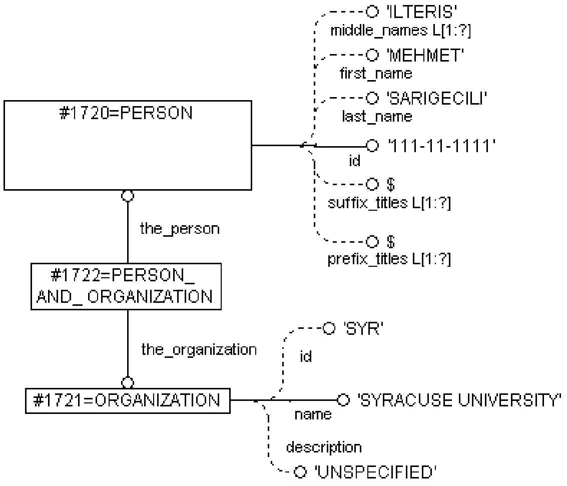


Figure 31: The representation of person and organization in EXPRESS-G

The contact information is represented for person and organization separately in STEP. In this section, only the `personal_address` will be shown as an example. In Figure 32, the `personal_address` keeps all contact information, like `address`,

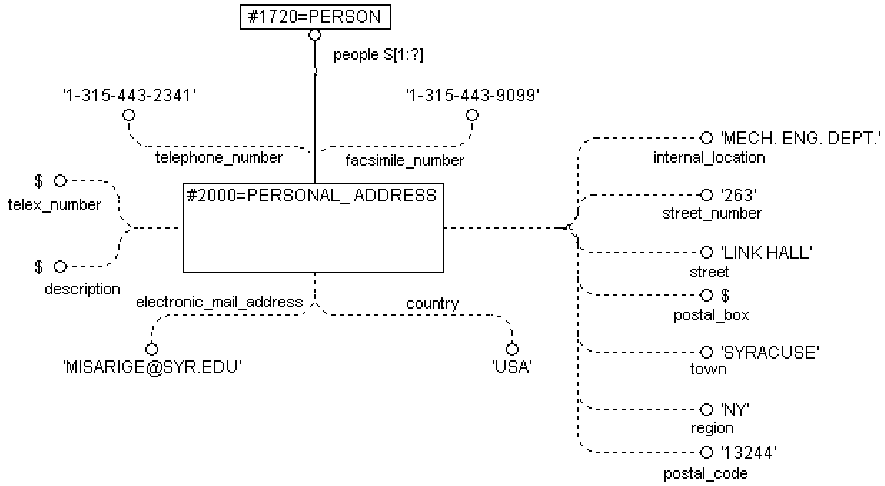


Figure 32: Example instantiation of personal address in EXPRESS-G

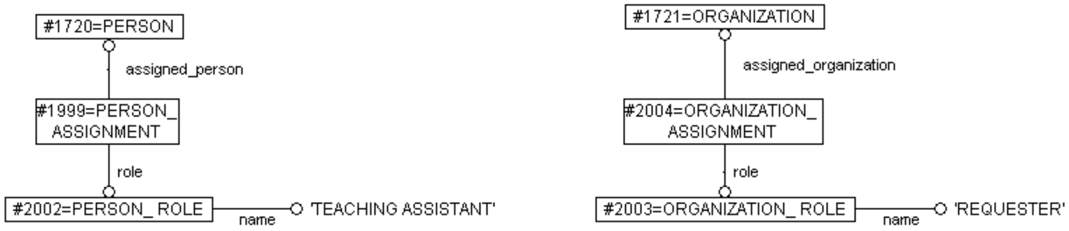


Figure 33: The representation of person and organization assignment in EXPRESS-G

phone, email, etc., and connects to the related person. The title of the contact person is captured by person_role, as shown in Figure 33. This entity is then connected to the person by the person_assignment. The organization_role will be used to define the requester and supplier companies. The organization_role is attached to the organization by the organization_assignment entity. The relationship between a company that requests information and a company that supplies information can be created by the organization_relationship, as shown in Figure 34. The organization_relationship is characteriezed as “materials declaration” by the name attribute.

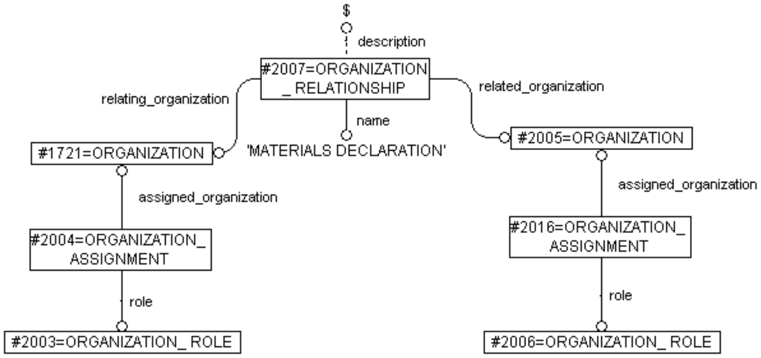


Figure 34: The representation of organization relationship in EXPRESS-G

The date information needs to be specified for the request date, response date, and the effective date. In STEP, as explained in previous sections, it has several ways to represent the date information. In Figure 35, for example, the calendar_date representation is shown. The year, day, and month information can be captured by the calendar_date. The type of the date (i.e., request date, response date or effective date) is captured by the date_role, and it is associated to the calendar_date by the date_assignment.

The document identification also has to be done for IPC-1752 (Figure 29). This is shown in Figure 36. The document_with_class entity, which is a subset of the document, keeps the id, name, and class information for the declaration document.

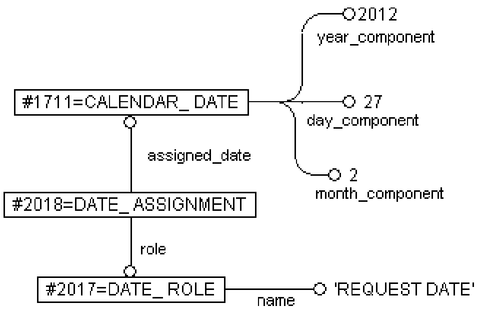


Figure 35: The representation of date in EXPRES-G

The document is attached to the product through the document_product_association. The document_representation_type represents that the document is in digital format. The document_type is defined as “IPC-1752” in Figure 36. The relationship between a request document and a response document is captured by the document_relationship.

The class-A declaration of IPC-1752 has queries to be answered as true or false (Figure 29). An EXPRESS-G representation of these queries is shown in Figure 37. The query list is represented as a material property, where each query statement is shown as a descriptive_representation_item. The material property is directly connected to the product (i.e. gear box) and the association between the query statements and the material property is define through material_property_representation and representation entities.

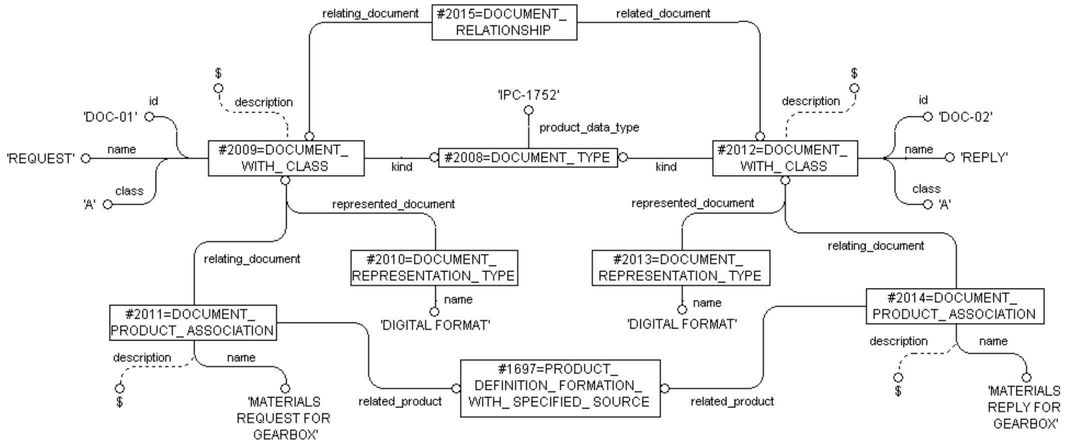


Figure 36: The representation of document and document relationship in EXPRESS-G

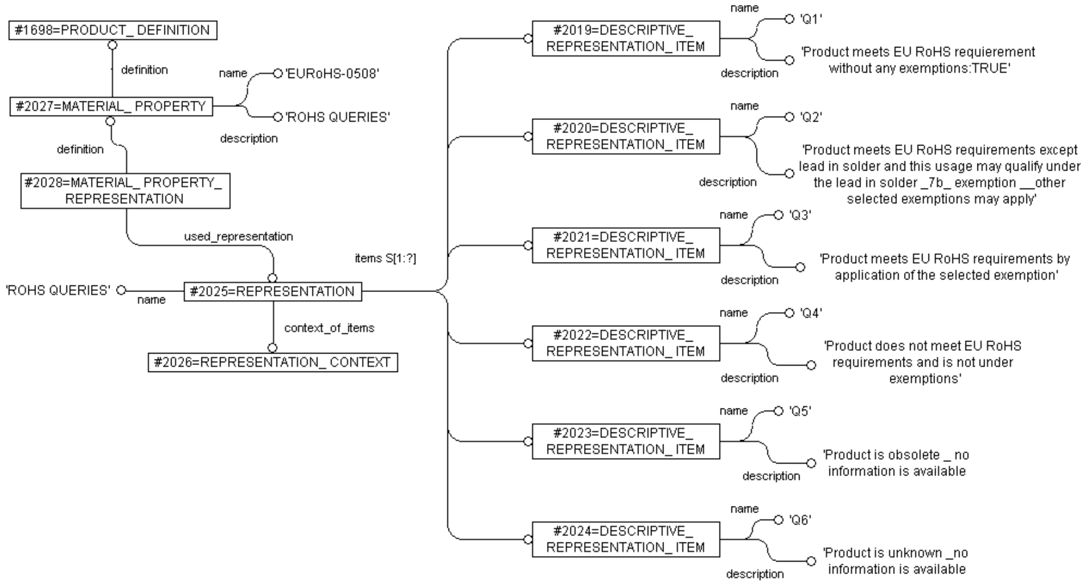

#1698=PRODUCT_ DEFINITION
definition
#2027=MATERIAL_ PROPERTY
name
'EURoHS-0508'
'ROHS QUERIES'
description
definition
#2028=MATERIAL_ PROPERTY_ REPRESENTATION
used_representation
items S[1:?]
'ROHS QUERIES'
name
#2025=REPRESENTATION
context_of_items
#2026=REPRESENTATION_ CONTEXT
#2019=DESCRIPTIVE_ REPRESENTATION_ ITEM
name
'Q1'
description
'Product meets EU RoHS requierement without any exemptions:TRUE'
#2020=DESCRIPTIVE_ REPRESENTATION_ ITEM
name
'Q2'
description
'Product meets EU RoHS requirements except lead in solder and this usage may qualify under the lead in solder _7b_ exemption __other selected exemptions may apply'
#2021=DESCRIPTIVE_ REPRESENTATION_ ITEM
name
'Q3'
description
'Product meets EU RoHS requirements by application of the selected exemption'
#2022=DESCRIPTIVE_ REPRESENTATION_ ITEM
name
'Q4'
description
'Product does not meet EU RoHS requirements and is not under exemptions'
#2023=DESCRIPTIVE_ REPRESENTATION_ ITEM
name
'Q5'
description
'Product is obsolete _ no information is available
#2024=DESCRIPTIVE_ REPRESENTATION_ ITEM
name
'Q6'
description
'Product is unknown _no information is available

Figure 37: An EXPRESS-G representation of queries for Class-A in IPC-1752

Chapter 4

Development of Functionality-Based Conformance Classes

In this section the development of the functionality-based CCs (FCCs) discussed. This necessitates (i) analyzing and identifying the required information for an engineering activity, (ii) grouping the information requirements in a hierarchical order, (iii) mapping the grouped information to available STEP resources to facilitate the standardized data exchange, (iv) defining any other pieces of information (that are not available in present STEP models) in EXPRESS, (v) breaking down the mapped information in groups (i.e., each information layer) into small units which are repeatedly instantiated as a unit. In the following sections, the development of FCCs for 1-D tolerance analysis is discussed in detail.

4.1 Identification of Information Requirements for 1-D Tolerance Analysis

In the development of FCCs for 1-D tolerance analysis (i.e., stack-up analysis), as a first step, the information requirements for tolerance analysis activity are identified from the industrial practices. A stack-up analysis is used for finding out the variation of an assembly (or a part) requirement by adding the dimensions and tolerances. Several sample problems related to stack-up analysis, collected from different sources (Fischer, 2004; Drake, 1999; Geng, 2004; Zhang, 1997; Neumann, 2003), have been analyzed in detail.

A complex bolted assembly is considered to present the required information for 1-D stack-up analysis. 3-D model of the assembly is shown in Figure 38. The sample tolerance analysis problem for this assembly is presented in Figure 39. The goal is to determine the minimum and maximum gap between parts 5 and 6 (i.e., between A and J). The details for the base plate (part 1) and the bracket (part 2) are given in Figure 40 and

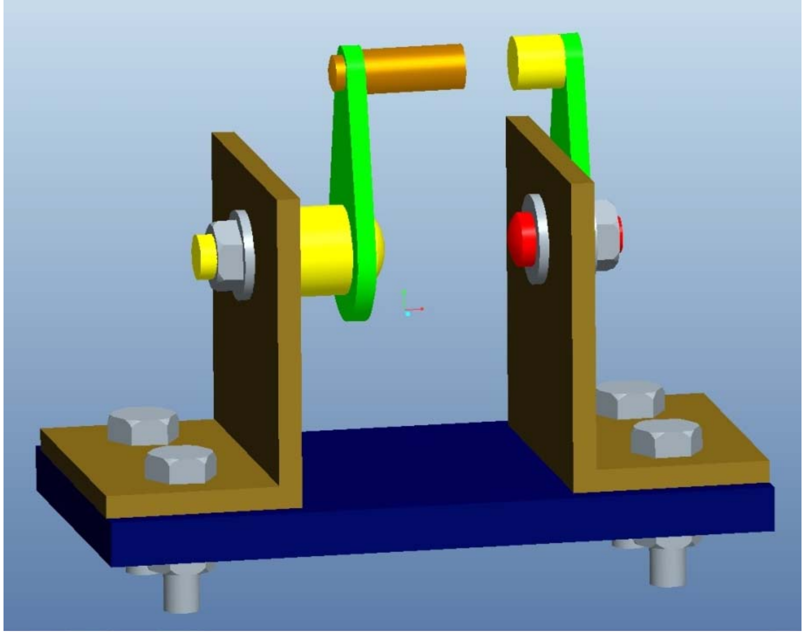


Figure 38: The assembly model

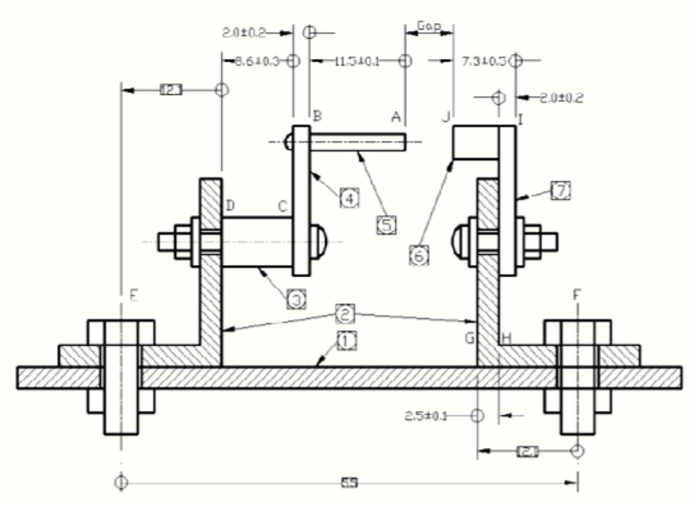


Figure 39: The complex bolted assembly with GD & T (Fischer, 2004)

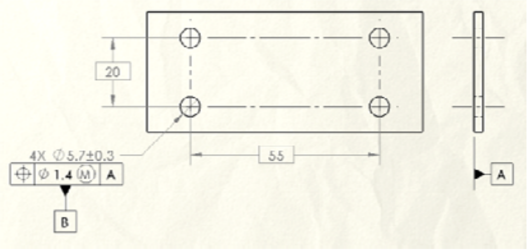


Figure 40: Details of the base plate (Fischer, 2004)

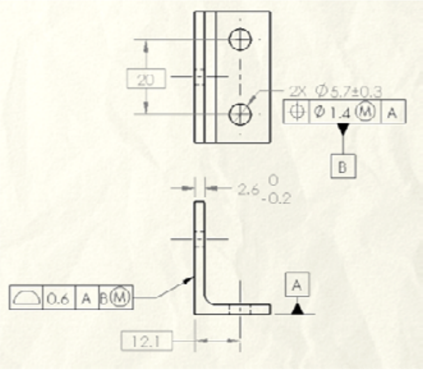


Figure 41: Details of the bracket (Fischer, 2004)

Figure 41, respectively. The GD & T specifications for these parts are shown in the figures. The specified geometric tolerances are as follows: (1) the patterns of holes in both the base and the bracket have positional tolerances, and (2) the bracket has a profile tolerance.

The manual tolerance analysis for this sample problem is shown in Figure 42. In the figure, every element required for the stack-up analysis is marked with a circled number. These elements and the explanations for them are tabulated in Table 3.

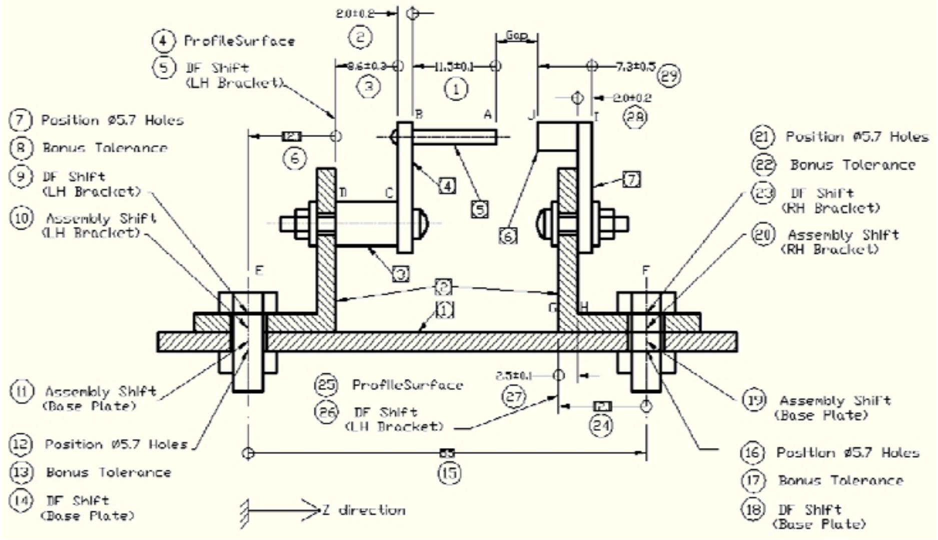


Figure 42: The stack-up analysis details for complex assembly (Fischer, 2004)

Table 3: The Stack-up Analysis for the Complex Assembly (Fischer, 2004)

Dim No	Part No	+	-	±	Description
1	5		11.5	±0.1	Dim: Pin Length
2	4		2	±0.2	Dim: LH Plate Thickness
3	3		8.6	±0.3	Standoff Thickness
4	2			±0.3	Profile of Flange Face on LH L-bracket
5	2			±1	Datum Feature Shift: ((5.7+0.3)-(5.7-0.3-1.4))/2=±1
6	2		12.1	±0	Dim: Flange Face – CL DF$_B$ Holes on LH L-bracket (Basic)
7	2			±0	Position of Dia 5.7 DF$_B$ Holes on LH L-bracket: N/A
8	2			±0	Bonus Tolerance: N/A
9	2			±0	Datum Feature Shift: N/A
10	2			±1	Assembly Shift: LH L-bracket Holes @ LMC: 6(H)-4(F)=2/2=±1
11	1			±1	Assembly Shift: Base Plate LH Holes @ LMC: 6(H)-4(F)=2/2=±1
12	1			±0.7	Position of LH Dia 5.7 DF$_B$ Holes on Base Plate
13	1			±0.3	Bonus Tolerance: (0.3+0.3)/2=±0.3
14	1			±0	Datum Feature Shift: N/A – DF$_A$ not a Feature of Size
15	1	55		±0	Dim: CL LH DF$_B$ Holes – CL RH DF$_B$ Holes on Base Plate (Basic)
16	1			±0.7	Position of RH Dia 5.7 DF$_B$ Holes on Base Plate
17	1			±0.3	Bonus Tolerance: (0.3+0.3)/2=±0.3
18	1			±0	Datum Feature Shift: N/A – DF$_A$ not a Feature of Size
19	1			±1	Assembly Shift: Base Plate RH Holes @ LMC: 6(H)-4(F)=2/2=±1
20	2			±1	Assembly Shift: RH L-bracket Holes @ LMC: 6(H)-4(F)=2/2=±1
21	2			±0	Position of Dia 5.7 DF$_B$ Holes on RH L-bracket: N/A
22	2			±0	Bonus Tolerance: N/A
23	2			±0	Datum Feature Shift: N/A
24	2		12.1	±0	Dim: CL DF$_B$ Holes – Flange Face on RH L-bracket (Basic)
25	2			±0.3	Profile of Flange Face on RH L-bracket
26	2			±1	Datum Feature Shift: ((5.7+0.3)-(5.7-0.3-1.4))/2=±1
27	2	2.5		±0.1	RH L-bracket Flange Thickness
28	7	2		±0.2	Thickness of RH Plate
29	6&7		7.3	±0.5	Thickness of RH Plate &Boss
Total		59.5	53.6	±10	Worst Case Tolerance – Sum of the plus/minus tolerance values

From Figure 42 and Table 3, the following information is identified as needed for stack-up analysis:

- Tolerances should be converted to symmetric bilateral format.

- Dimensions and tolerances are represented by a chain of dimensions (i.e., dimension loop): A→B→C→D→E→F→G→H→I→J→A.
- Nominal dimensions and symmetric dimensional tolerances from A through J in the loop have to be identified. For example, A→B is 11.5±0.1 mm. In this case, the nominal dimension is 11.5 mm and the symmetric tolerance is ±0.1 mm.
- Geometric tolerances should be converted to symmetric bilateral tolerances (for details (Fischer, 2004)). For example, the profile tolerance of the flange face in the L-bracket is converted to ± 0.3 mm.
- Bonus tolerances, datum feature shifts and assembly shifts should be defined in symmetrical bilateral format (details are explained in section 4.4.3 on page 84).

The nominal gap (i.e., 5.9 mm) is calculated by subtracting the negative sum (i.e., 53.6 mm) from the positive sum (59.5 mm) (Table 3). The worst-case variation is calculated as ±10 in Table 3. Hence, the stack-up analysis for the gap (Figure 42) yields 5.9±10 mm.

4.2 Grouping of Information Requirements into Hierarchical Information Layers

Grouping the required information into hierarchical layers necessitates analyzing the procedure to carry out tolerance analysis in computerized systems. A general procedure is given below:

- Geometric models of components are created.
- GD & T specifications are added to the geometric models of each component.
- Assembly model is created from component models. The assembly model contains assembly constraint (i.e., mating) and transformation definitions between components.

- The objective of the tolerance analysis (i.e, gap in between two faces) has to be defined.
- Any bonus tolerance, datum feature shift or assembly shift is created by CAD system.

In a data exchange / sharing scenario, the order of this procedure for tolerance analysis clearly represents the hierarchical layers of information that can be navigated. The information layers are shown in Figure 43. The information requirements are divided into two different domains: (i) product design and modeling and (ii) tolerance analysis. Since product design and modeling is a common domain for other activities as well, the information requirements specific to tolerance analysis is grouped separately.

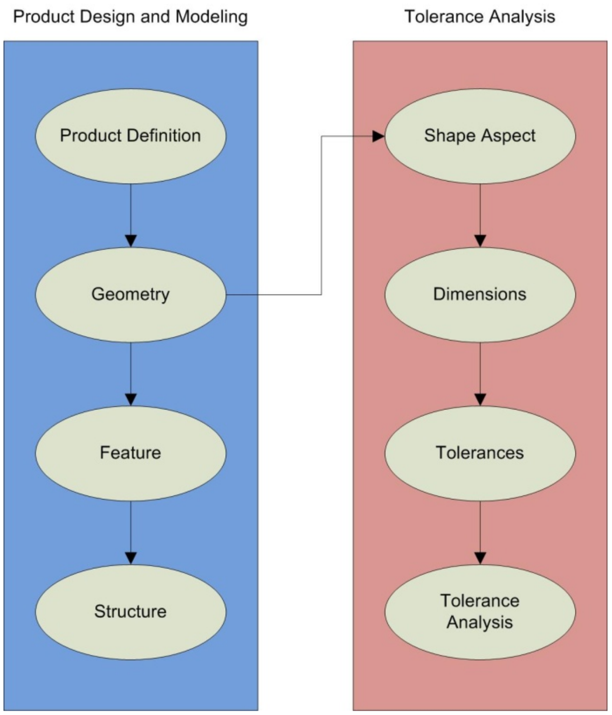


Figure 43: Information layers for tolerance analysis

4.3 Mapping of Information Requirements into Available STEP Resources

The information groups in Figure 43 have to be mapped to the available STEP resources. The information requirements for identifying any product by its name, version number and product context are defined in information layer 1 of the product design and modeling activity as shown in Figure 43. These information requirements are collected from the product definition schema of Part 41 of STEP.

The geometric representation of a product model is defined in information layer 2. The geometry information is available at Part 42 of STEP. In addition to Part 42, Part 43 of STEP has the necessary information requirements to associate the geometric definition of a product with its basic definition, i.e., its name, version number, etc.

Feature-level information is grouped in information layer 3. Feature definitions are taken from the form definitions of AP 214.

In the last layer, the structural information is defined. The assembly tree structure information is defined by Part 44 of STEP. The orientation information of components within the assembly is available in Part 43 of STEP. The information on assembly constraints between components is given in Part 109 (ISO 10303-109, 2004) of STEP. Besides the assembly constraint definitions relating to components, the information groups of product design and modeling are pretty well implemented in STEP by commercial CAD systems. For this reason, only the assembly constraint information model will be discussed in detail in section 4.4.1.

The information requirements for tolerance analysis, aside from product design and modeling, are given in Figure 43 as well. Once the product definition and geometry have been identified, the shape of a part has to be defined in terms of shape aspect, which includes dimensions and tolerances. There are several types of identifying shape aspects, e.g., a pattern of features conceived as a shape aspect, or a derived shape aspect, etc. In STEP, they are available in the shape aspect schema of Part 47. These information requirements are grouped in information layer 1. Size and location dimensions are defined by referring to the related shape aspects. Size and location information are grouped in information layer 2. These information requirements are available in the shape dimension schema of Part 47 of STEP. In layer 3, the dimensional tolerances and geometric tolerances are grouped. These tolerance information requirements are available in the shape tolerance schema of Part 47. This layer contains all the information needed to carry out tolerance analysis at the part level. The assembly level tolerance analysis requires assembly structure information as well. For this reason, the information requirements for assembly level tolerance analysis are grouped in information layer 4.

4.4 Information Requirements Unavailable in STEP Resources

The information requirements that are unavailable in STEP resources for stack-up tolerance analysis have to be represented in EXPRESS. In the previous section, the information requirements for the product design and modeling domain were mapped to

STEP except the representation of assembly constraints between components. In section 4.4.1, the modified information model for assembly constraint representation is discussed. Additionally, GD & T representation information were mapped to STEP in the last section, as well. However, the following information is not available in STEP: information for dimension loop that is created by chaining dimensions to each other, and the loop elements (section 4.4.2), dimensional variation items (section 4.4.3) and tolerance analysis process (section 4.4.4). In the following sections, the details of representing these information requirements in EXPRESS are discussed.

4.4.1 Modifying Assembly Constraint Information Model of STEP

Specification of the explicit geometric constraints among components of an assembly model is defined in Part 109. The assembly constraints defined there are descriptive, not mathematical. These constraints are subtypes of constraints defined in Part 108 (ISO 10303-108, 2005).

The assembly constraint representation available in STEP has to be modified for the following reason: In STEP, assembly constraint information is defined between two components of an assembly. The geometric entities that are constrained in these two components are identified by the constraint information. The constraint information also captures the transformation information (i.e., orientation and location) of each component with respect to each other. The transformation information is represented in STEP by `representation_relationship_with_transformation` entity. However, in a STEP file, product description of a component in an assembly is defined only once. Then, for each usage of the same component within the assembly (the same component might be used more than once in an assembly), the next_assembly_usage_occurence entity is instantiated to define a relationship between the component and the assembly. The transformation information for the usage of this component with respect to assembly is represented by the representation_relationship_with_transformation. Therefore, defining transformation information between components in assembly constraint representation excludes the usage information of each component within the same assembly. Because of that reason, we modified the assembly constraint representation in STEP by separately referring to the representation_relationship_with_transformation that captures the transformation information for each component usage within the assembly.

In Figure 44, the EXPRESS-G representation of the modified assembly constraint representation is shown. The assembly_geometric_constraint entity is used to define the relationships between rigid components of an assembly, which control the transformation matrices that position and orient the models of these components. The assembly_geometric_constraint has two subtype entities: binary_assembly_constraint and fixed_constituent_assembly_constraint. The first is used to precisely define binary constraints controlling the relationship between two components in an assembly; the second is used to define a fixed position and orientation of one component of an assembly. Two attributes added to represent the transformation information of each

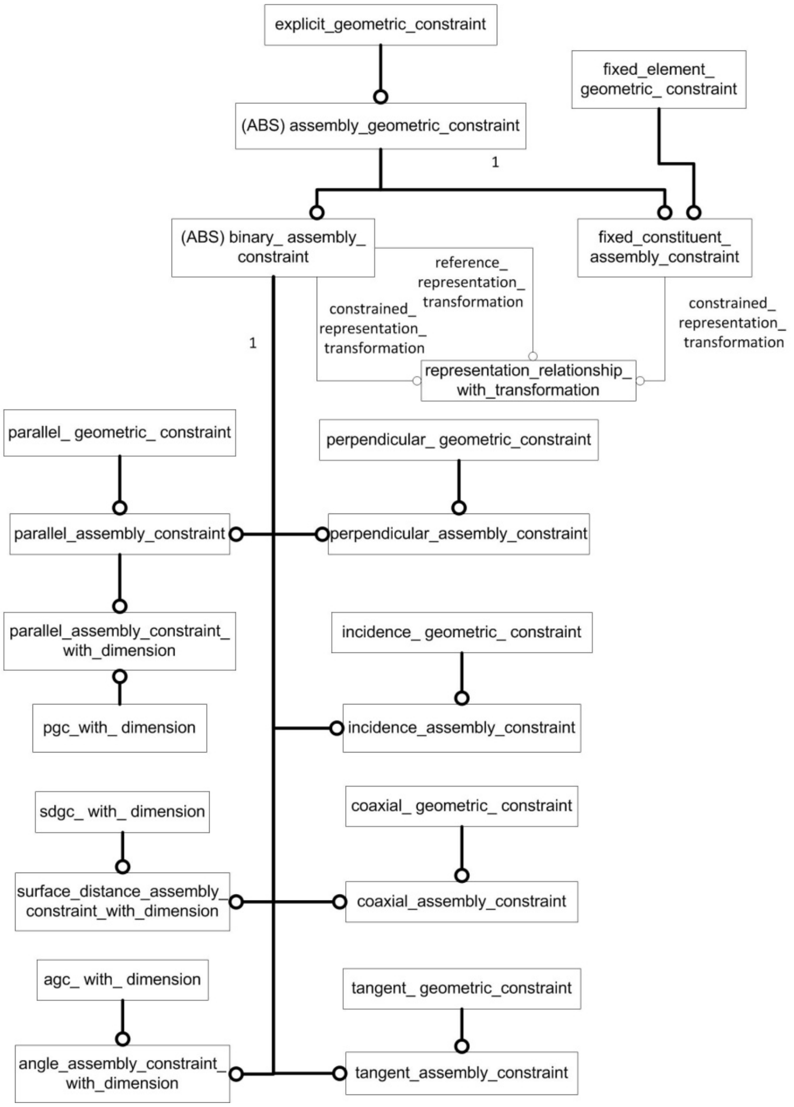

explicit_geometric_constraint
fixed_element_ geometric_ constraint
(ABS) assembly_geometric_constraint
1
(ABS) binary_ assembly_ constraint
fixed_constituent_ assembly_constraint
reference_ representation_ transformation
constrained_ representation_ transformation
constrained_ representation_ transformation
1
representation_relationship_ with_transformation
parallel_ geometric_ constraint
perpendicular_ geometric_constraint
parallel_assembly_constraint
perpendicular_assembly_constraint
parallel_assembly_constraint_ with_dimension
incidence_ geometric_ constraint
pgc_with_ dimension
incidence_assembly_constraint
sdgc_ with_ dimension
coaxial_ geometric_ constraint
surface_distance_assembly_ constraint_with_dimension
coaxial_assembly_constraint
agc_ with_ dimension
tangent_ geometric_constraint
angle_assembly_constraint_ with_dimension
tangent_assembly_constraint

Figure 44: Modified assembly constraint representation in EXPRESS-G

component with respect to the assembly are the `constrained_representation_transformation` and the `reference_representation_transformation` (Figure 44). In this way, it is guaranteed that the assembly constraints between components of an assembly are defined for the correct use of the component in the assembly.

Similarly, `constrained_representation_transformation`, attribute is defined for the `fixed_constituent_assembly_constraint` to represent the transformation information of the fixed component with respect to the assembly (Figure 44).

The entity definitions that are used as defined in Parts 108 and 109 are listed in Table 4, for ready reference. In the table, the entities in the right column are subtypes of the entities in the left column. The details can be found in (ISO 10303-109, 2004; ISO 10303-108, 2005) respectively.

Table 4: Entities Used from Part 108 and Part 109 for the Assembly Constraints

Entities used from Part 108	Entities used from Part 109
explicit_constraint	assembly_geometric_constraint
-	binary_assembly_constraint
defined_constraint	-
explicit_geometric_constraint	-
fixed_element_geometric_constraint	fixed_constituent_assembly_constraint
parallel_geometric_constraint	parallel_assembly_constraint
pgc_with_dimension	parallel_assembly_constraint_with_dimension
surface_distance_geometric_constraint	-
sdgc_with_dimension	surface_distance_assembly_constraint_with_dimension
angle_geometric_constraint	-
agc_with_dimension	angle_assembly_constraint_with_dimension
perpendicular_geometric_constraint	perpendicular_assembly_constraint
incidence_geometric_constraint	incidence_assembly_constraint
coaxial_geometric_constraint	coaxial_assembly_constraint
tangent_geometric_constraint	tangent_assembly_constraint
tangent_geometric_constraint	tangent_assembly_constraint

4.4.2 Dimension Loop and Loop Element

The analysis objective (i.e., the gap) for 1-D stack-up tolerance analysis is defined by the distance in between two particular faces in the assembly model. Then, starting from one face of the analysis objective each component's dimensions that affect the analysis objective are chained to each other to create a dimension loop, ending at the other face of the analysis. The dimension loop information is not available in STEP. Therefore, its EXPRESS definition is created in this study. The EXPRESS-G representation is shown in Figure 45. The dimension loop is defined as a list of loop elements. Loop elements are defined as a subset of `shape_aspect_relationship` which is available in Part 47. This entity defines a relationship between two `shape_aspect` definitions. Hence, the directionality of loop elements are defined by two attributes inherited from the `shape_aspect_relationship`: the directionality is from `relating_shape_aspect` to `related_shape_aspect`. Each loop element is used to identify the required dimension and tolerance value for the analysis. Additionally, the transformation information of each component that is used in the loop element definition has to be captured. This is defined by the following two attributes: `element1_transformation` and `element2_transformation`.

4.4.3 Dimensional Variation Items

For 1-D stack-up tolerance analysis, each variation item (i.e., each dimension and tolerance) that contributes to the variation has to be associated with a loop element in a dimension loop. The variations should be represented in symmetrical bilateral format.

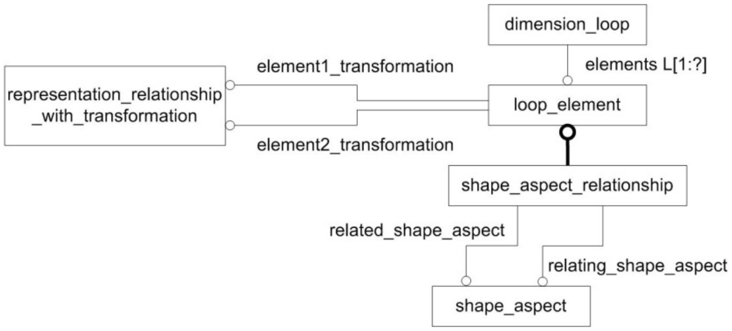


Figure 45: Detailed representation of the `dimension_loop` in EXPRESS-G

Hence, nominal dimensions and tolerances (section 4.4.3.1) and geometric tolerances (section 4.4.3.2) should be converted to symmetric bilateral tolerances. There are also variation items that are caused by assigned geometric tolerances. They are bonus tolerances (section 4.4.3.4), datum feature shifts (section 4.4.3.5) and assembly shifts (section 4.4.3.6). They have to be represented in symmetric bilateral format as well. All these variation items are collected under the select type `dimensional_variation` in Figure 46. To present definitions of these allowable entities neatly, each will be treated separately. Also, the explanation of these additional variation items caused by geometric tolerances necessitates the definition of virtual condition boundary and it is given in section 4.4.3.3.

4.4.3.1 Nominal Dimension

In a dimension loop, each loop element should have a nominal dimension and a dimensional tolerance, both of which affect the tolerance analysis. This is represented by the nominal dimension in Figure 47. To use dimension and tolerance representation available in STEP, the nominal dimension is defined as a subset of the

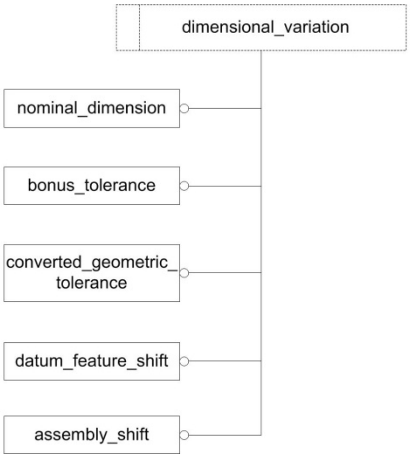


Figure 46: Detailed representation of the `dimensional_variation` in EXPRESS-G

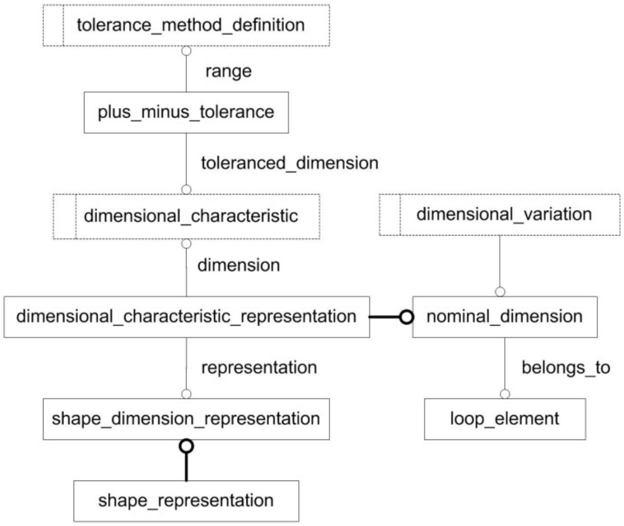


Figure 47: Detailed representation of the `nominal_dimension` in EXPRESS-G

`dimensional_characteristic_representation` of Part 47. The nominal dimension indirectly acquires a dimensional value and a plus/minus tolerance. The attribute `belongs_to` defines the association between the nominal dimension and the loop element.

4.4.3.2 Converted Geometric Tolerance

Any specification of geometric tolerance that affects the tolerance analysis has to be converted to plus/minus tolerances for linear tolerance analysis. In our EXPRESS-G representation (Figure 48), the values of the plus/minus tolerances for converted geometric tolerance are represented by available STEP representation, i.e. `tolerance_value`. Additionally, the geometric tolerance that is converted to symmetric bilateral tolerances is identified by the `g_tolerance` attribute. To associate the converted geometric tolerance with a loop element, the attribute `belongs_to` is defined. This attribute associates converted geometric tolerance with shape aspect. The constraint on this attribute is that the associated shape aspect has to be either the `relating_shape_aspect` or the `related_shape_aspect` of a loop element.

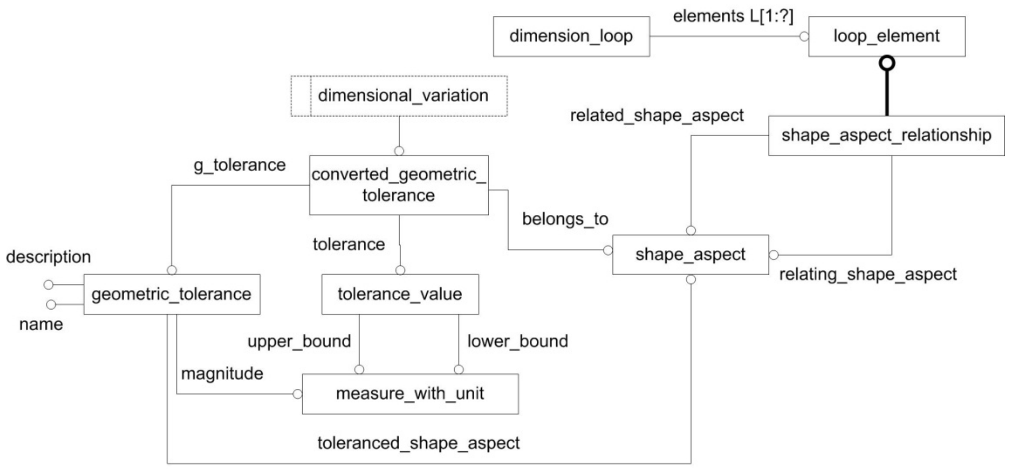


Figure 48: Detailed representation of the `converted_geometric_tolerance` in EXPRESS-G

4.4.3.3 Virtual Condition Boundary

A *virtual condition boundary* (Drake, 1999) is an imaginary boundary established for features of size with a geometric tolerance specification that is modified to MMC or LMC. Any geometric tolerance applied to a feature of size and modified to MMC creates a virtual condition boundary in the air adjacent to the feature surface(s). This boundary is the limit in air space where the feature shall not encroach. The MMC virtual condition boundary represents a restricted air space reserved for the mating part feature. In such a mating interface, the internal feature's MMC virtual condition boundary must be at least as large as that for the external feature. Similarly, any geometric tolerance applied to a feature of size and modified to LMC creates a virtual condition boundary which is embedded in part material, just beneath the feature surface(s). This boundary constitutes a restricted core or shell of part material into which the feature shall not encroach. The LMC virtual condition boundary assures a protected core of part material within a pin, boss, or tab, or a protected case of part material around a hole or slot.

The perfect geometric shape of any virtual condition boundary is a counterpart to the nominal shape of the controlled feature. *Virtual condition* (the boundary's fixed size) is determined by three factors: 1) the feature's type (internal or external); 2) the feature's MMC or LMC size limit; and 3) the specified geometric tolerance value. The details are given in Table 5.

In Figure 49, a pin fitting into a hole is represented to explain virtual condition boundary. Both pin and hole have perpendicularity tolerance with MMC modifier.

Perpendicularity tolerance applied to a feature of size, modified to MMC or LMC, establishes a virtual condition boundary beyond which the feature's surface(s) shall not encroach.

Table 5: The Virtual Condition Size for MMC and LMC

	Internal feature of size	External feature of size
MMC virtual condition	MMC size limit − geometric tolerance	MMC size limit + geometric tolerance
LMC virtual condition	LMC size limit + geometric tolerance	LMC size limit − geometric tolerance

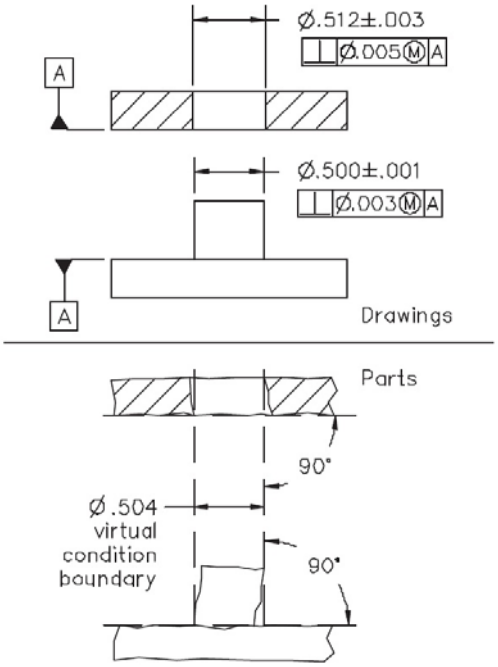


Figure 49: Using virtual condition boundaries for mating constraints (Drake, 1999)

For the pin in Figure 49, the diameter of the virtual condition boundary equals the pin's MMC size plus the perpendicularity tolerance value: Ø.501 + Ø.003 = Ø.504 in. Similarly, the diameter of the virtual condition boundary for the hole is calculated by subtracting the perpendicularity tolerance value from the hole's MMC size: ∅.509 - ∅.005 = ∅.504 in.

In this section, virtual condition boundary (VCB) representation using available STEP entities is explained. The VCB is defined as a property of the shape_aspect because it is a derived definition based on the feature of size and the geometric tolerance definition. In Figure 50, the name and description attributes of the property_definition should be set to "tolerance property" and "virtual condition boundary definition," respectively. This attribute is connected to the shape_aspect through the definition attribute. The value of the VCB is defined by the measure_representation_item. The name attribute of this entity is set as "VCB size." The value_component attribute shows the value of the VCB. This value representation is connected to the representation by the items attribute. The name

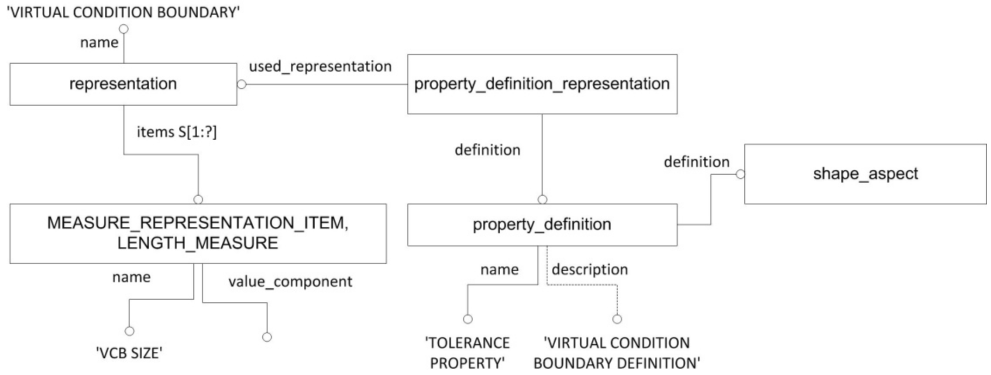


Figure 50: Detailed representation of the virtual condition boundary in EXPRESS-G

attribute of the `representation` is set to “virtual condition boundary.” Then, the association between the `representation` and the `property_definiton` is defined by the `property_definition_representation` in Figure 50.

4.4.3.4 Bonus Tolerance

Any geometric tolerance applied to a feature of size and modified to MMC or LMC establishes a bonus tolerance (Fischer, 2004) when the size of the as-produced feature deviates from the virtual condition boundary. In Figure 49, the diameter of virtual condition boundary for both pin and hole is set to Ø.504 in. The tolerance zone for the pin increases as its diameter deviates from MMC size limit (the largest pin) to LMC size limit (the smallest pin). Similarly, as diameter of the hole changes from MMC size limit (the smallest hole) to LMC size limit (the largest hole), its tolerance zone increases. This increase in tolerance zone is added like a bonus in tolerance analysis. Therefore, the bonus tolerance is the maximum amount of increase in tolerance zone. For the pin diameter it is Ø.001+Ø.001=Ø.002 in. For the hole diameter, it is Ø.003+Ø.003=Ø.006 in. After finding out any bonus tolerance, it has to be converted to symmetric bilateral tolerances.

The representation of bonus tolerance in EXPRESS-G is shown in Figure 51. In this figure, the plus/minus tolerance values for bonus tolerance are defined by `tolerance_value` entity available in STEP. Other than that, any information that contributes to the calculation of bonus tolerance has been captured as well: size dimension assigned to feature of size, specified geometric tolerance and material modifier. Bonus tolerance is also associated with shape aspect for relating this bonus tolerance to a loop element.

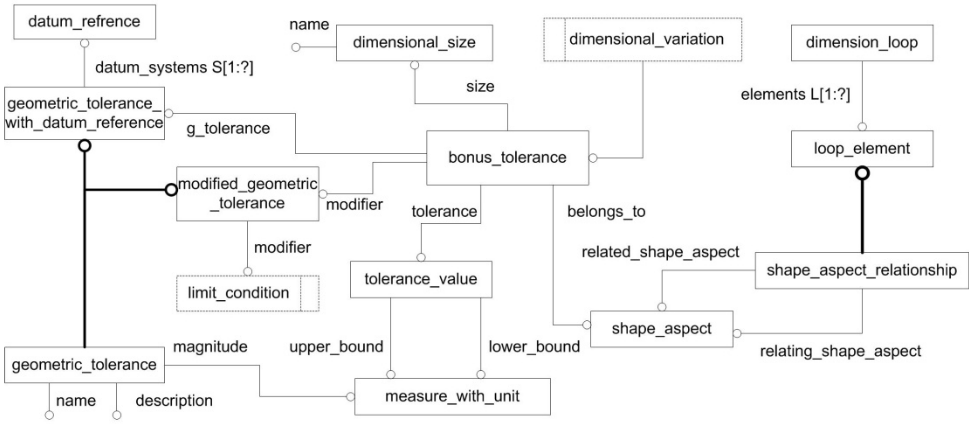


Figure 51: Detailed representation of the `bonus_tolerance` in EXPRESS-G

4.4.3.5 Datum Feature Shift

Datum feature shift (Fischer, 2004) is the maximum variation that may be caused when inspecting features related to datum features of size specified at MMC or LMC. For the inspection, the datum feature simulators have to be created. These simulators may be larger or smaller than the datum features. The maximum difference between the datum features and their simulators contributes to the maximum variation.

Any datum feature of size has to be simulated for its applicable virtual condition size, LMC size, or MMC size, whichever is applicable, according to the paragraph 2.11.3 in the ASME Y14.5M-1994 (ASME, 1994) standard. Two considerations must be made to determine which datum feature simulator size is appropriate:

1. Determine whether LMC or MMC is specified.
2. Determine if there is a geometric tolerance specified that controls the datum feature of size's center geometry as per the rules below:
    a. If a datum feature of size is not specified with a geometric tolerance that controls the datum feature's center geometry (such as its orientation or position), then the datum feature of size is simulated at its appropriate LMC or MMC size (Table 6).
    b. If a datum feature of size is specified with a geometric tolerance that controls the datum feature's center geometry (such as its orientation or position), then the datum feature of size is simulated at its appropriate virtual condition size (Table 6).

Table 6: Datum Feature Shift

Referred datum	Modifier	Internal Datum Feature	External Datum Feature
Datum referenced w/o modifier		0	0
Modified datum and datum feature w/o geometric tolerance		Upper limit – lower limit	Upper limit – lower limit
Modified datum and datum feature with geometric tolerance	RFS	N/A	N/A
	MMC	(Nominal size + upper limit) – (Nominal size – lower limit – geometric tolerance)	(Nominal size + upper limit + geometric tolerance) – (Nominal size – lower limit)
	LMC	(Nominal size + upper limit + geometric tolerance) – (Nominal size – lower limit)	(Nominal size + upper limit) – (Nominal size – lower limit – geometric tolerance)

As an example, a profile tolerance is specified with datum A as the primary datum and datum B at MMC as the secondary datum in Figure 52. The datum feature B is simulated at its MMC virtual condition size because the datum feature of size has a positional tolerance controlling its center geometry.

Datum feature shift calculation for the hole at its MMC virtual condition size is as follows:

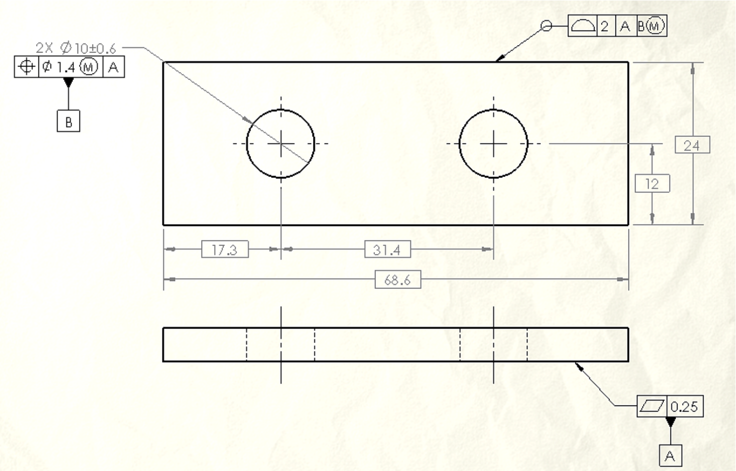


Figure 52: A datum feature simulated at its virtual condition size

1. MMC virtual condition size :

   Ø10.0 nominal size - 0.6 size tolerance - Ø1.4 positional tolerance = Ø8.0 mm.
2. This is the datum feature simulator size.
3. LMC (largest hole) = Ø10.0 nominal size + 0.6 size tolerance = Ø10.6 mm LMC size.
4. Datum feature shift = Ø10. LMC size - Ø8.0 MMC size = Ø2.6 mm datum feature shift.
5. Divide the datum feature shift by 2: 2.6/2 = ±1.3 mm.
6. This is the equivalent ± symmetric tolerance value for the tolerance analysis.

Datum feature shift definition is developed in EXPRESS-G as shown in Figure 53. The symmetric bilateral tolerance value for datum feature shift is represented by `tolerance_value` entity defined in STEP. Datum feature shift definition collects any information that contributes the calculation of it: assigned geometric tolerance, specified datum modifier, size dimension assigned to datum feature and geometric tolerance assigned to this datum feature. The last element is defined as optional attribute in Figure

53 since datum feature might or might not have a geometric tolerance specified. Datum feature shift is also related to shape aspect to be associated with a loop element.

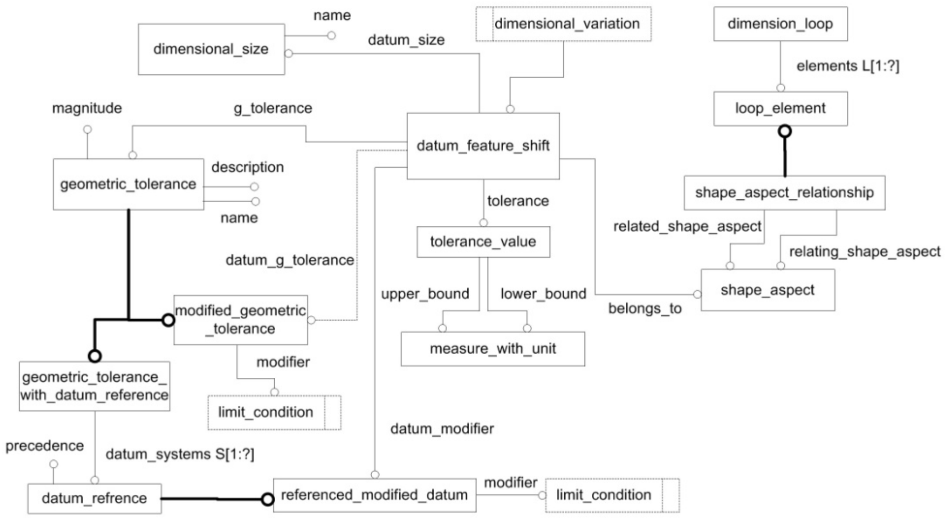


Figure 53: Detailed representation of the `datum_feature_shift` in EXPRESS-G

4.4.3.6 Assembly Shift

Assembly shift (Fischer, 2004) is the maximum misalignment between components of an assembly caused by clearance between an as-produced hole feature and a fastener. For example, specifying a positional tolerance at MMC for holes leads to the greatest possible dislocation when the hole is produced at LMC (the largest hole). As a result, the difference between the hole diameter at LMC and the fastener diameter is the assembly shift. The fastener nominal values (e.g., 8 mm diameter for a M8 bolt) are assumed as the fastener diameters in this assembly shift calculation. Each assembly shift value has to be converted to symmetric bilateral tolerances for tolerance analysis.

The representation of assembly shift in EXPRESS-G is shown in Figure 54. Assembly shift definition captures the following information that contributes to the calculation of it: size dimensions assigned to internal and external features (i.e., hole and shaft, respectively), association of assembly shift with shape aspects of internal and external feature and geometric tolerance assigned to internal feature if any. Plus/minus tolerance values of assembly shift are represented by `tolerance_value`.

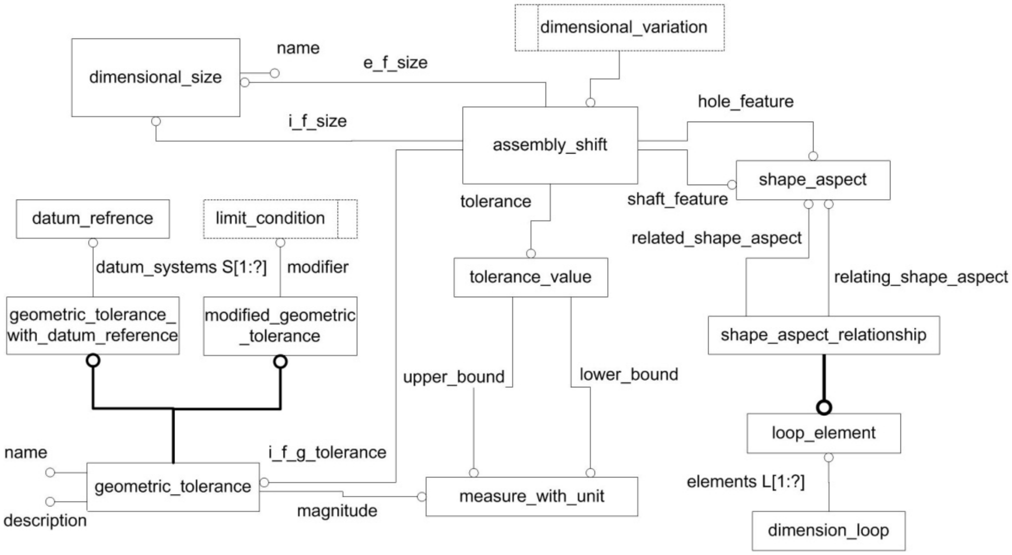


Figure 54: Detailed representation of the `assembly_shift` in EXPRESS-G

4.4.4 Tolerance Analysis Process

In this section, we represent how to aggregate all the information required for 1-D tolerance analysis. To carry out the 1-D tolerance analysis in a CAD package, in addition to the information requirements from product modeling and GD & T representation, information regarding the analysis objectives and the dimension loop is needed. The 1-D tolerance analysis can be represented by the following information requirements: nominal dimensions, bonus tolerances, converted geometric tolerances as used in the analysis (i.e., plus/minus tolerances converted from geometric tolerances), datum feature shifts and assembly shifts. The EXPRESS-G representation of tolerance analysis is given in Figure 55.

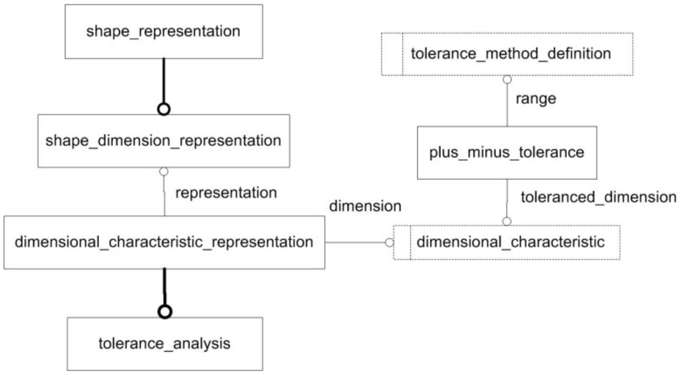

shape_representation
shape_dimension_representation
representation
dimensional_characteristic_representation
tolerance_analysis
tolerance_method_definition
range
plus_minus_tolerance
toleranced_dimension
dimension
dimensional_characteristic

Figure 55: Detailed representation of the `tolerance_analysis` in EXPRESS-G

The common properties of all different types of tolerance analysis are represented by `tolerance_analysis` entity. The analysis objective of the problem (i.e., the gap/distance/size to be analyzed for dimensional variation) is needed to start the analysis. In order to use available GD & T specifications in STEP for defining the analysis objective, `tolerance_analysis` is characterized as a subset of `dimensional_characteristic_representation`, which is given in Part 47 (Figure 55). This entity has two attributes to identify size or location dimension implicitly and to define value of the dimension. The tolerances are associated with the implicit size or location dimensions.

The linear_tol_analysis entity is represented as a subset of the entity tolerance_analysis. The EXPRESS-G representation is shown in Figure 56. This entity collects all the information required for 1-D tolerance analysis: loop attribute represents the dimension loop used for the tolerance analysis; components attribute represent any dimensional variation that has an effect on the tolerance analysis.

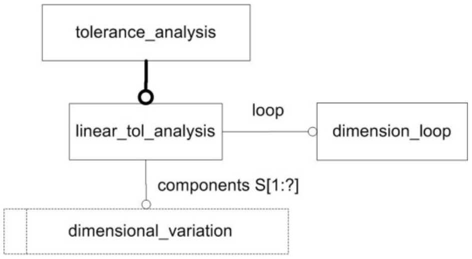


Figure 56: Detailed representation of the linear_tol_analysis in EXPRESS-G

Hierarchically grouping the information content is still not helpful for users because the data content needed is much smaller than these information groups. Therefore, the development of the templates is discussed in the next section.

4.5 Development of Templates for FCCs

The last step of the development of FCCs for 1-D tolerance analysis is to break down the information groups into small, repeatedly instantiated information groups that are called as templates. A template explicitly identifies the entities to be instantiated and attributes to be selected. Templates might use (or consist of) other templates. The developed templates are discussed in detail in the next chapter.

As an example, developed *shape aspect template* is shown in Figure 57. The shape aspect entity is used to identify a part of the product shape definition to attach GD & T specifications. The part of the product shape is defined by *representation template*. The association between shape aspect and product shape is realized by property definition template. The product definition template identifies the product for which shape aspect is defined.

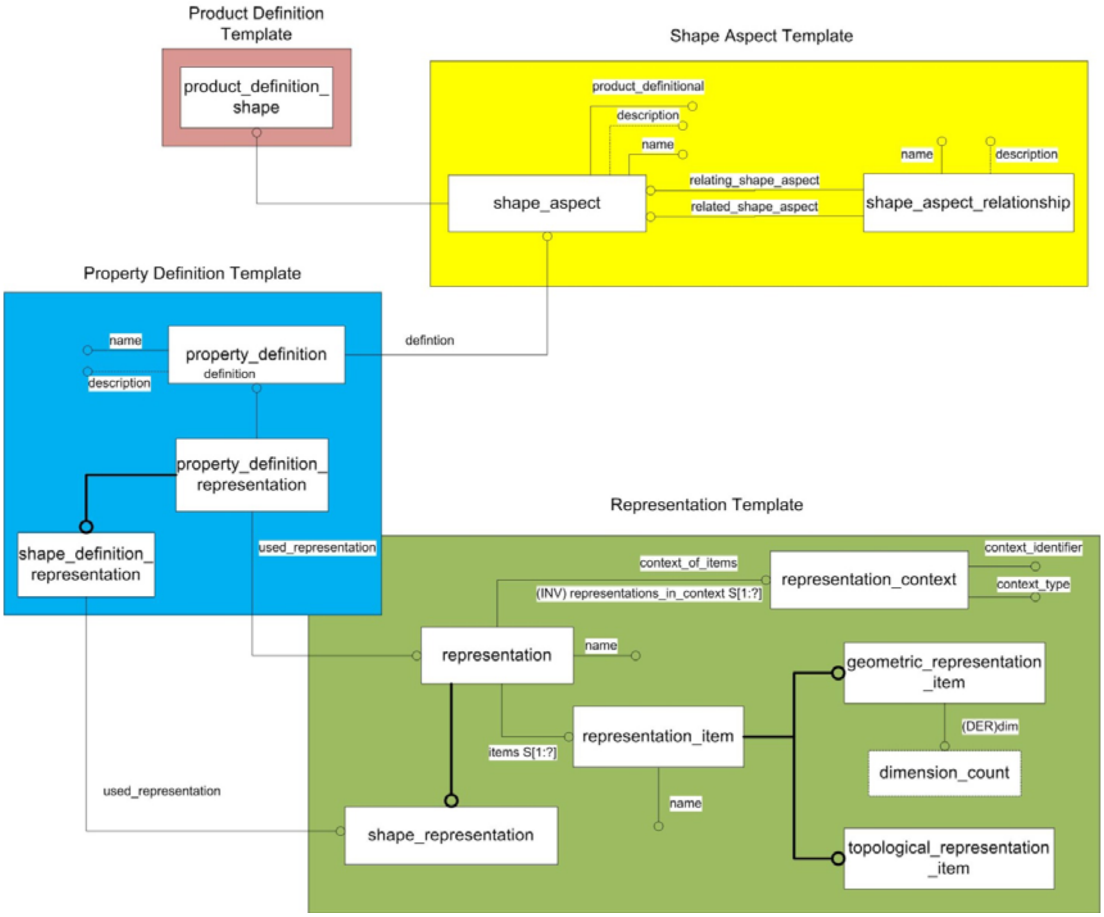
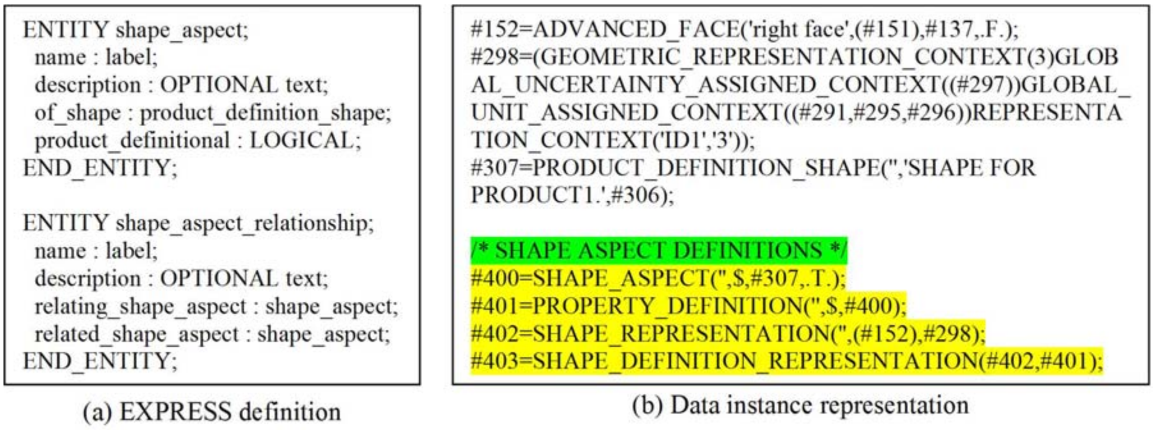


Figure 57: Shape aspect template

Example instantiation of the shape aspect template is shown in Figure 58-b. EXPRESS definitions for this template is also given in Figure 58-a, for ready reference. The instantiation of shape aspect template requires four entity instantiations (i.e., #400 - #403) which are highlighted in yellow. Other entity instances are defined by other templates and needed for the shape aspect template.



```
ENTITY shape_aspect;
  name : label;
  description : OPTIONAL text;
  of_shape : product_definition_shape;
  product_definitional : LOGICAL;
END_ENTITY;

ENTITY shape_aspect_relationship;
  name : label;
  description : OPTIONAL text;
  relating_shape_aspect : shape_aspect;
  related_shape_aspect : shape_aspect;
END_ENTITY;
```

(a) EXPRESS definition

```
#152=ADVANCED_FACE('right face',(#151),#137,.F.);
#298=(GEOMETRIC_REPRESENTATION_CONTEXT(3)GLOB
AL_UNCERTAINTY_ASSIGNED_CONTEXT((#297))GLOBAL_
UNIT_ASSIGNED_CONTEXT((#291,#295,#296))REPRESENTA
TION_CONTEXT('ID1','3'));
#307=PRODUCT_DEFINITION_SHAPE('','SHAPE FOR
PRODUCT1.',#306);

/* SHAPE ASPECT DEFINITIONS */
#400=SHAPE_ASPECT('',$,#307,.T.);
#401=PROPERTY_DEFINITION('',$,#400);
#402=SHAPE_REPRESENTATION('',(#152),#298);
#403=SHAPE_DEFINITION_REPRESENTATION(#402,#401);
```

(b) Data instance representation

Figure 58: Shape aspect EXPRESS definition and data instance representation

4.6 Case Study

In this section, the sample tolerance analysis problem which was discussed in section 4.1 (Figure 38) is presented as a case study. The implementation framework is shown in Figure 59. In this figure, the assembly is modeled in a CAD package. All GD & T specifications are defined for products. Since the current CAD systems do not translate GD & T specifications assigned to products, these specifications are manually added to the STEP file. The enriched STEP file then can be used by another CAD system to carry out tolerance analysis. In Figure 60, part of the enriched STEP file for our bolted

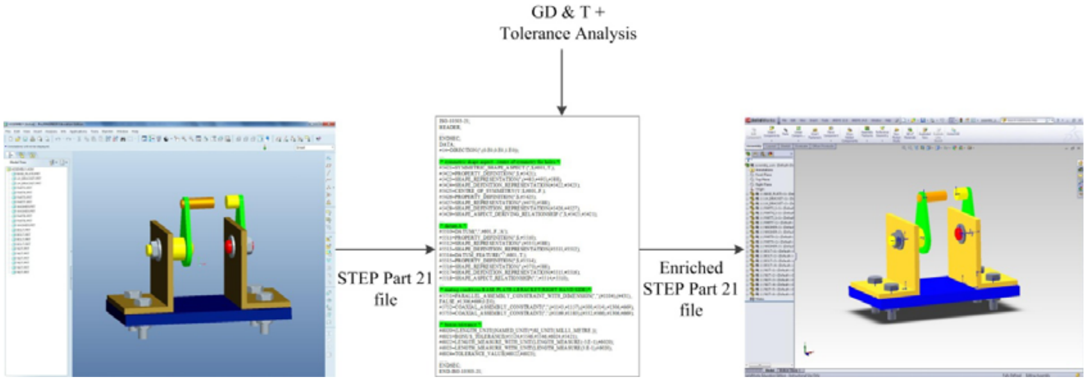
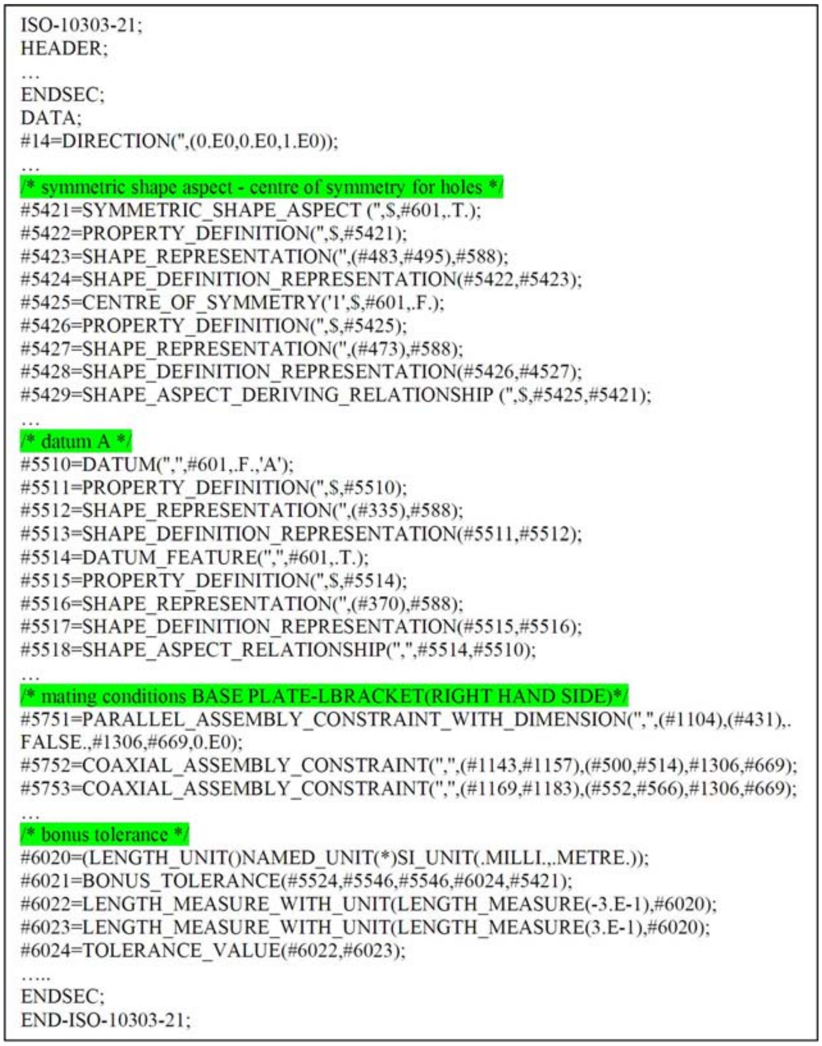


Figure 59: Framework for implementation of FCC - tolerance analysis



```
ISO-10303-21;
HEADER;
…
ENDSEC;
DATA;
#14=DIRECTION(",(0.E0,0.E0,1.E0));
…
/* symmetric shape aspect - centre of symmetry for holes */
#5421=SYMMETRIC_SHAPE_ASPECT (",$,#601,.T.);
#5422=PROPERTY_DEFINITION(",$,#5421);
#5423=SHAPE_REPRESENTATION(",(#483,#495),#588);
#5424=SHAPE_DEFINITION_REPRESENTATION(#5422,#5423);
#5425=CENTRE_OF_SYMMETRY('1',$,#601,.F.);
#5426=PROPERTY_DEFINITION(",$,#5425);
#5427=SHAPE_REPRESENTATION(",(#473),#588);
#5428=SHAPE_DEFINITION_REPRESENTATION(#5426,#4527);
#5429=SHAPE_ASPECT_DERIVING_RELATIONSHIP (",$,#5425,#5421);
…
/* datum A */
#5510=DATUM(",",#601,.F.,'A');
#5511=PROPERTY_DEFINITION(",$,#5510);
#5512=SHAPE_REPRESENTATION(",(#335),#588);
#5513=SHAPE_DEFINITION_REPRESENTATION(#5511,#5512);
#5514=DATUM_FEATURE(",",#601,.T.);
#5515=PROPERTY_DEFINITION(",$,#5514);
#5516=SHAPE_REPRESENTATION(",(#370),#588);
#5517=SHAPE_DEFINITION_REPRESENTATION(#5515,#5516);
#5518=SHAPE_ASPECT_RELATIONSHIP(",",#5514,#5510);
…
/* mating conditions BASE PLATE-LBRACKET(RIGHT HAND SIDE)*/
#5751=PARALLEL_ASSEMBLY_CONSTRAINT_WITH_DIMENSION(",",(#1104),(#431),.
FALSE.,#1306,#669,0.E0);
#5752=COAXIAL_ASSEMBLY_CONSTRAINT(",",(#1143,#1157),(#500,#514),#1306,#669);
#5753=COAXIAL_ASSEMBLY_CONSTRAINT(",",(#1169,#1183),(#552,#566),#1306,#669);
…
/* bonus tolerance */
#6020=(LENGTH_UNIT()NAMED_UNIT(*)SI_UNIT(.MILLI.,.METRE.));
#6021=BONUS_TOLERANCE(#5524,#5546,#5546,#6024,#5421);
#6022=LENGTH_MEASURE_WITH_UNIT(LENGTH_MEASURE(-3.E-1),#6020);
#6023=LENGTH_MEASURE_WITH_UNIT(LENGTH_MEASURE(3.E-1),#6020);
#6024=TOLERANCE_VALUE(#6022,#6023);
…..
ENDSEC;
END-ISO-10303-21;
```

Figure 60: Part 21 STEP file for the bolted assembly

assembly is shown. It is evident that, the data instances are organized according to our templates. In the figure, instantiations of symmetric shape aspect template, datum template, assembly constraint template and bonus tolerance template are shown.

Implementation of FCCs for 1-D tolerance analysis will help identify the missing information that is needed for 1-D tolerance analysis.

Chapter 5

Development of Templates for Part 47 Information Models

In this section, templates for representing GD & T (geometric dimensions and tolerances) information defined in Part 47 of STEP are developed. These templates then can be used for constructing functionality-based conformance classes (FCC) proposed in the previous chapter.

5.1 Introduction

Integrated resources provide abstract and highly usable product models for the development of APs in STEP. Part 47 is an integrated resource of STEP; it is called *shape variation* tolerance and is divided into three distinct information models: (1) The shape_aspect_definition schema provides the definitions for the spatial characteristics of a shape, which are required for dimensioning and tolerancing. (2) Representation of location and size dimensions is provided by the shape_dimension_schema. (3) The shape_tolerance_schema provides the constructs for describing tolerances, and includes two types of tolerance representation: plus-minus tolerance and geometrical tolerance.

Hierarchical relationships among the layers of information needed for functionality-based CCs for tolerance analysis, as shown in Figure 61, is also used for representing GD & T. In the figure, the information models of Part 47 are represented in three layers: shape aspect, dimensions and tolerances.

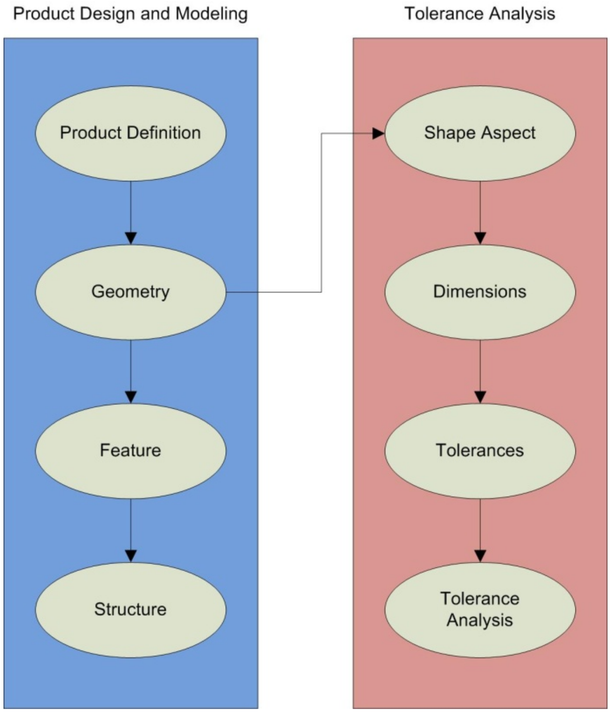


Figure 61: Information layers for tolerance analysis

In the following three subsections, templates are created for repeating patterns in the information models of Part 47: for shape aspect, for dimensions and for tolerances. In these template definitions, entities that are to be instantiated are identified with "^instantiate." In addition, the GD & T specifications for a sample part, shown in Figure 62, are used to represent the instantiations of some templates.

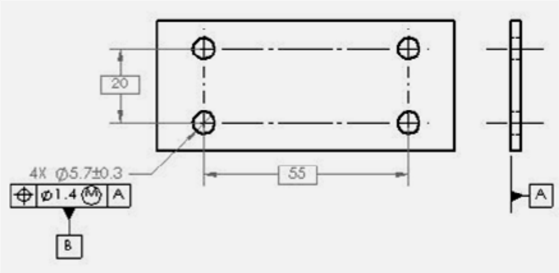


Figure 62: Dimensions and tolerances of a sample part

5.1.1 Templates for the Shape Aspect Information Model

Representing any dimension or tolerance requires identifying and associating the dimension and/or tolerance specifications with a product shape. Identifying process can be accomplished by using the template representing_shape_aspect, shown in Figure 63.

It should be noted that only the following entities are instantiated with this template: `shape_aspect`, `property_definition`, `shape_definition_representation` and `shape_representation`. The `shape_representation` is a subset of the `representation`. Therefore, the `items` attribute collects the geometric definitions for the `shape_aspect`.

The relationship between two `shape_aspect` definitions is represented by the `shape_aspect_relationship` entity. This relationship entails that the `related_shape_aspect` be dependent on the `relating_shape_aspect`. The template for this representation is shown in Figure 64.

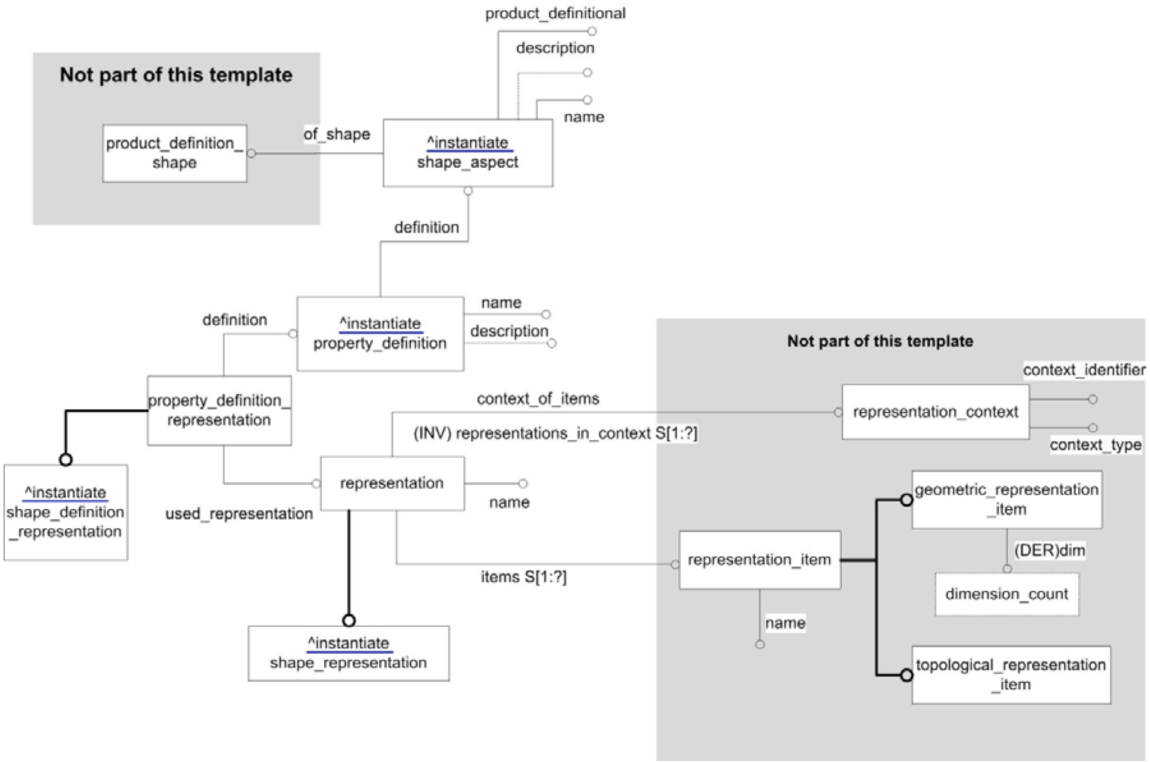

Not part of this template
product_definitional
description
name
product_definition_shape
of_shape
^instantiate shape_aspect
definition
name
description
definition
^instantiate property_definition
Not part of this template
context_identifier
property_definition_representation
context_of_items
representation_context
(INV) representations_in_context S[1:?]
context_type
^instantiate shape_definition_representation
representation
name
used_representation
geometric_representation_item
(DER)dim
representation_item
items S[1:?]
dimension_count
name
^instantiate shape_representation
topological_representation_item

Figure 63: The template representing_shape_aspect

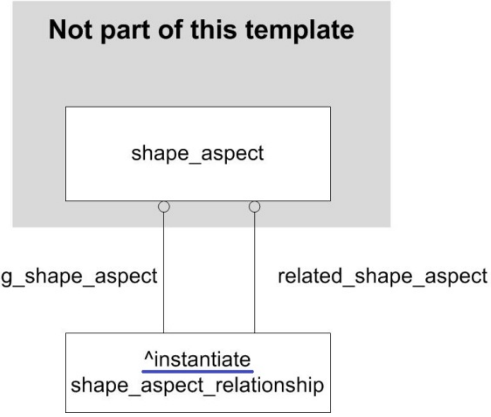

Not part of this template
shape_aspect
relating_shape_aspect
related_shape_aspect
^instantiate shape_aspect_relationship

Figure 64: The template representing_shape_aspect_relationship

Any `shape_aspect` whose existence depends on other `shape_aspects` is represented by `derived_shape_aspect` in Part 47. This entity has eight sub-classes.

Under certain conditions, a specialized subtype of derived_shape_aspect should be used. Under certain other conditions, the plain derived_shape_aspect entity should be used. For example, a cylindrical hole is symmetrical about an axis, which exists only when the hole exists. For this reason, the axis is defined as a derived_shape_aspect, i.e., the centre_of_symmetry. The data instance of the centre_of_symmetry will have one symmetric_shape_aspect, i.e., any shape_aspect of a product that is symmetrical about a geometric element. The symmetric_shape_aspect entity can also be used to represent the shape_aspect defined by a group of identified shape_aspects of a product that is symmetrical overall. The template for this representation is shown in Figure 65. It should be noted that both symmetric_shape_aspect and centre_of_symmetry are subsets of the shape_aspect, and have the same requirements. In addition, in the template, the inverse attribute basis_relationships of the shape_aspect_relationship identifies relationships with one or more features that are symmetrical about centers of symmetry, e.g., a point, axis, or median plane. The relating_shape_aspect attribute of the shape_aspect_relationship has to be the symmetric_shape_aspect. The related_shape_aspect attribute should refer to shape_aspects that are symmetric. The symmetry definition is represented by the centre_of_symmetry. The relationship between the symmetric_shape_aspect and the centre_of_symmetry is described by the derived_shape_aspect_relationship, a subset of the shape_aspect_relationship. For this entity, the related_shape_aspect is the symmetric_shape_aspect and the relating_shape_aspect is the centre_of_symmetry. For example, the central axis of a cylindrical hole feature can be represented as a centre_of_symmetry. The cylindrical face of the internal surface of the hole can be represented as a symmetrical_shape_aspect.

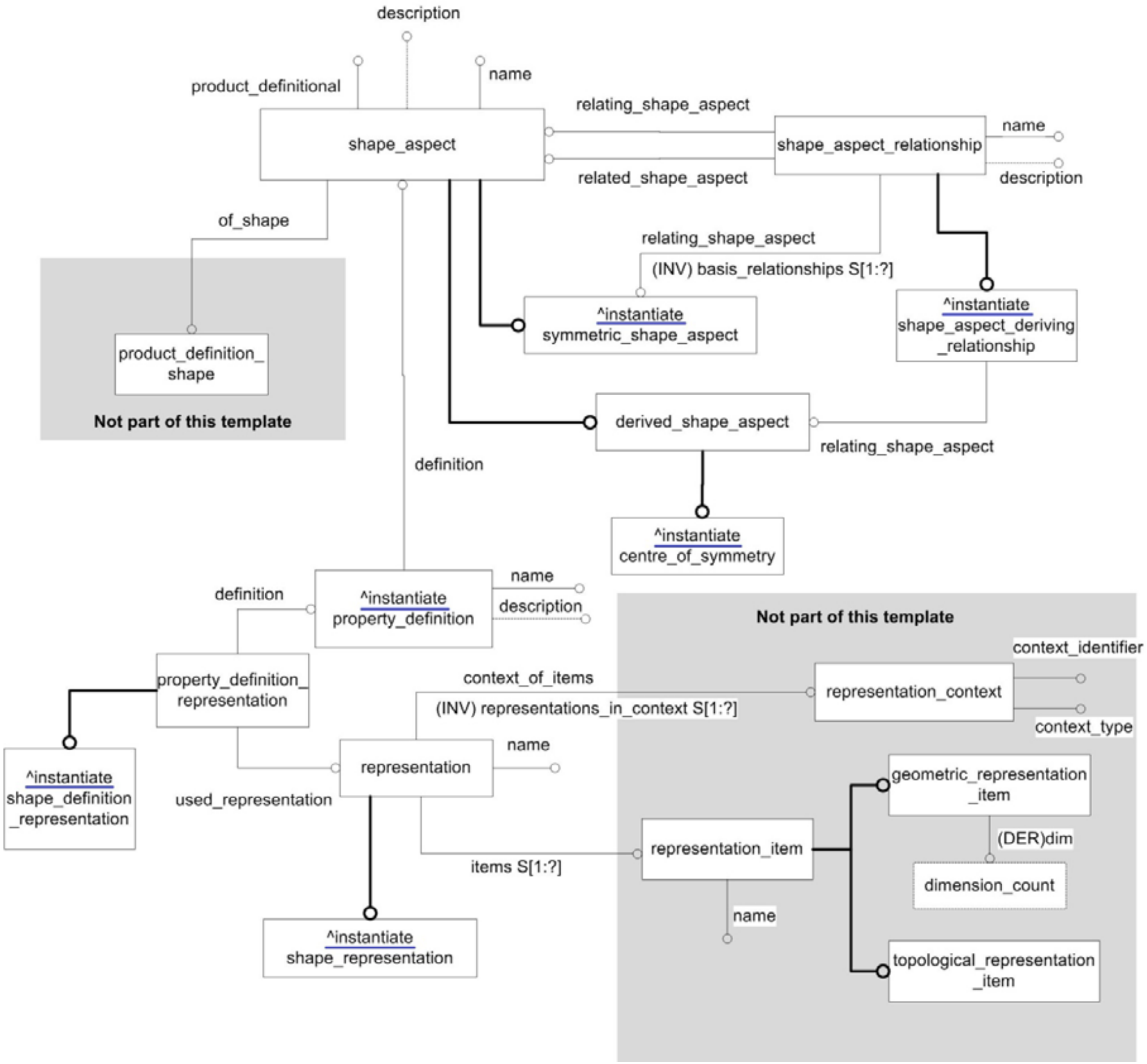


Figure 65: The template representing_symmetric_shape_aspect

As an example, the template shown in Figure 65 is used to identify one of the hole features in our example part (Figure 62) as symmetric_shape_aspect, shown in

Figure 66. Templates for other subclasses of the derived_shape_aspect can be created in a similar way.

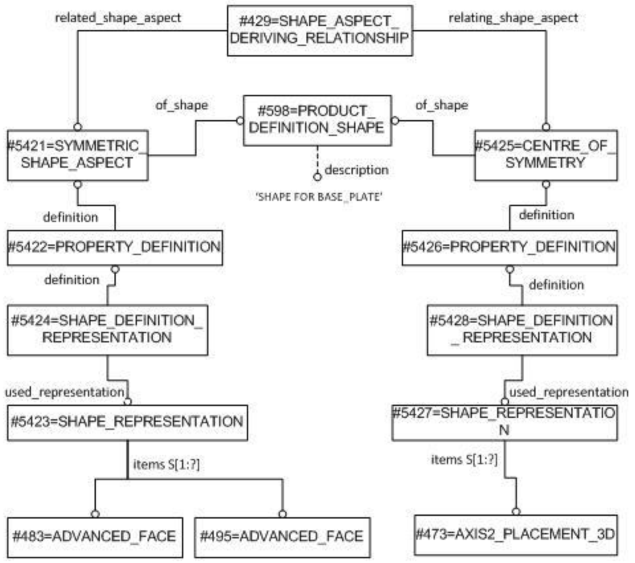


Figure 66: Instantiation of the representing_symmetric_shape_aspect

Defining GD & T specifications for discrete features of a product necessitates associating shape_aspect definitions for each feature with a composite_shape_aspect definition. The template for this representation is given in Figure 67. In order to define GD & T specifications for a pattern of features, the name attribute of the composite_shape_aspect should be set as “pattern of features.” For disjointed features that are not part of a pattern, the name attribute should be set as “multiple elements.” It should be noted that for each shape_aspect definition, there should be one shape_aspect_relationship with its relating_shape_aspect attribute connected to the composite_shape_aspect. For example, the four holes in

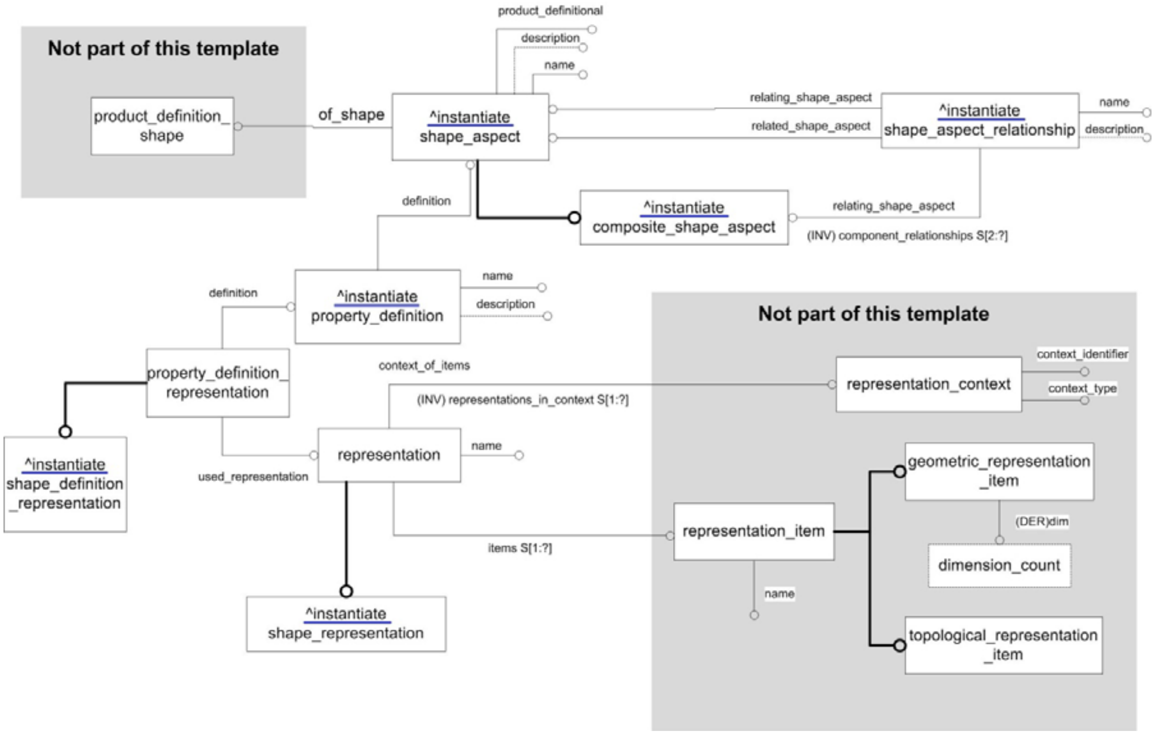

Not part of this template
product_definition_shape
of_shape
product_definitional
description
name
^instantiate shape_aspect
relating_shape_aspect
related_shape_aspect
^instantiate shape_aspect_relationship
name
description
definition
^instantiate composite_shape_aspect
relating_shape_aspect
(INV) component_relationships S[2:?]
definition
^instantiate property_definition
name
description
Not part of this template
property_definition_representation
context_of_items
representation_context
context_identifier
context_type
(INV) representations_in_context S[1:?]
^instantiate shape_definition_representation
used_representation
representation
name
geometric_representation_item
(DER)dim
representation_item
dimension_count
items S[1:?]
name
^instantiate shape_representation
topological_representation_item

Figure 67: The template representing_composite_shape_aspect

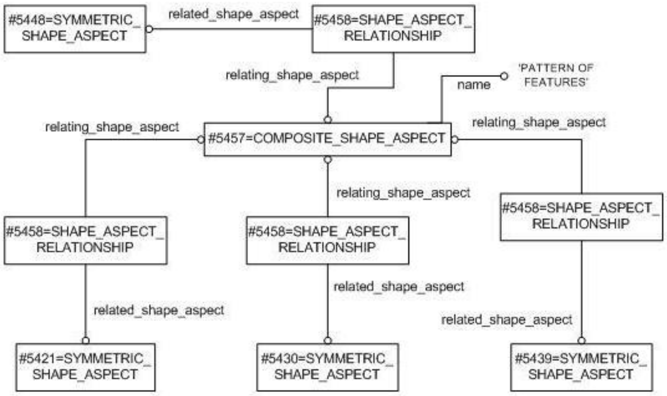

#5448=SYMMETRIC_SHAPE_ASPECT
related_shape_aspect
#5458=SHAPE_ASPECT_RELATIONSHIP
relating_shape_aspect
name
'PATTERN OF FEATURES'
relating_shape_aspect
#5457=COMPOSITE_SHAPE_ASPECT
relating_shape_aspect
relating_shape_aspect
#5458=SHAPE_ASPECT_RELATIONSHIP
#5458=SHAPE_ASPECT_RELATIONSHIP
#5458=SHAPE_ASPECT_RELATIONSHIP
related_shape_aspect
related_shape_aspect
related_shape_aspect
#5421=SYMMETRIC_SHAPE_ASPECT
#5430=SYMMETRIC_SHAPE_ASPECT
#5439=SYMMETRIC_SHAPE_ASPECT

Figure 68: Instances for representing pattern of holes

our sample part (see Figure 62) are identified as “pattern of features” in the template (see Figure 68).

For representing geometric dimensions and tolerances, *datum* is a referenced perfect theoretical entity which is simulated by datum features or datum targets. Any datum which is simulated by a datum feature that has a size can have a material modifier. These concepts appear repeatedly in the representation of geometric tolerances.

Figure 69 shows the template for representing a datum with a datum feature. As shown, the instantiation of the `datum` requires a `datum_feature` in order to define which part of the geometry is being treated as a datum. It should be noted that both entities (i.e., `datum` and `datum_feature`) are subsets of the `shape_aspect`. Hence, the instantiation of these two entities requires a minimum set of information for the `shape_aspect`. Then, a `datum_feature` is associated with a corresponding `datum` definition using the `shape_aspect_relationship`**,** as shown in Figure 69.

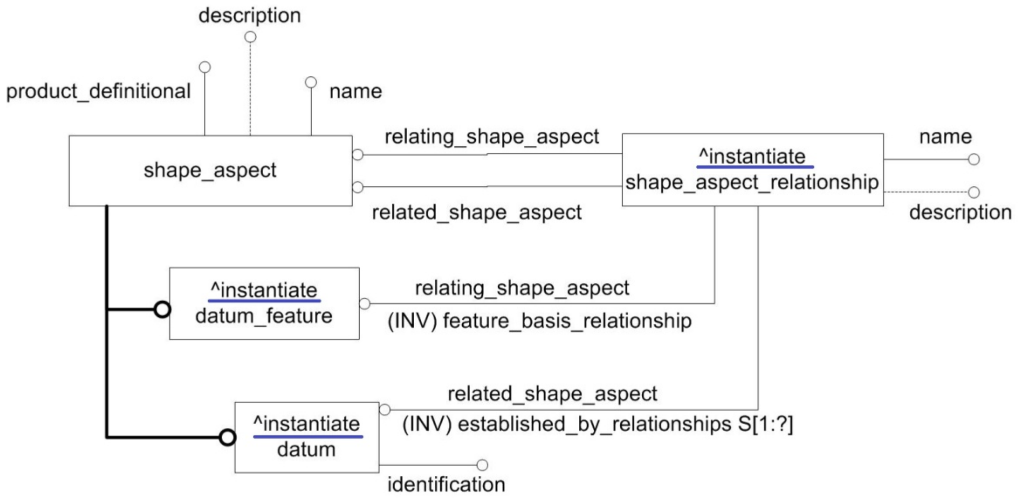


Figure 69: The template representing_datum_with_datum_feature

Another important point is that both `datum` and `datum_feature`, as `shape_aspects`, should have `representation_items`: the `datum` should be associated with a plane or axis definition, defined as a `geometric_representation_item`, whereas the corresponding `datum_feature` should be associated with a planar surface or cylindrical surface, defined as a `topological_representation_item` (since a datum can be simulated by a simple planar or cylindrical datum feature). For example, the template shown in Figure 70 is used to identify datum A for our sample part (Figure 62).

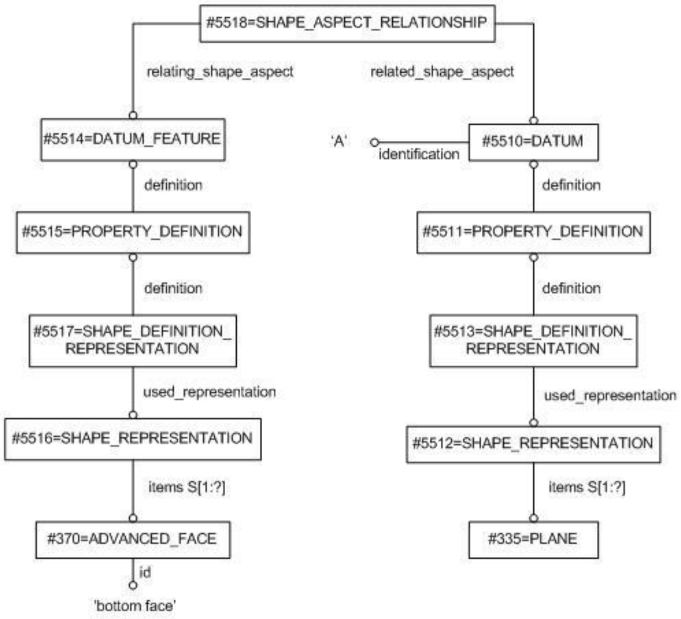


Figure 70: Instances for datum A

A template for representing a datum with a datum target is presented in Figure 71. This is similar to the template for representing a datum with a datum feature.

Referencing a datum for a geometric tolerance is defined by the entity `datum_reference` in part 47. The template for representing `datum_reference` is shown in Figure 72. This entity has two attributes: the first—`precedence`—shows the priority assigned to a datum; the second— referenced_datum—refers to the datum entity. The precedence attribute is used to show the order of the datum in the feature control frame.

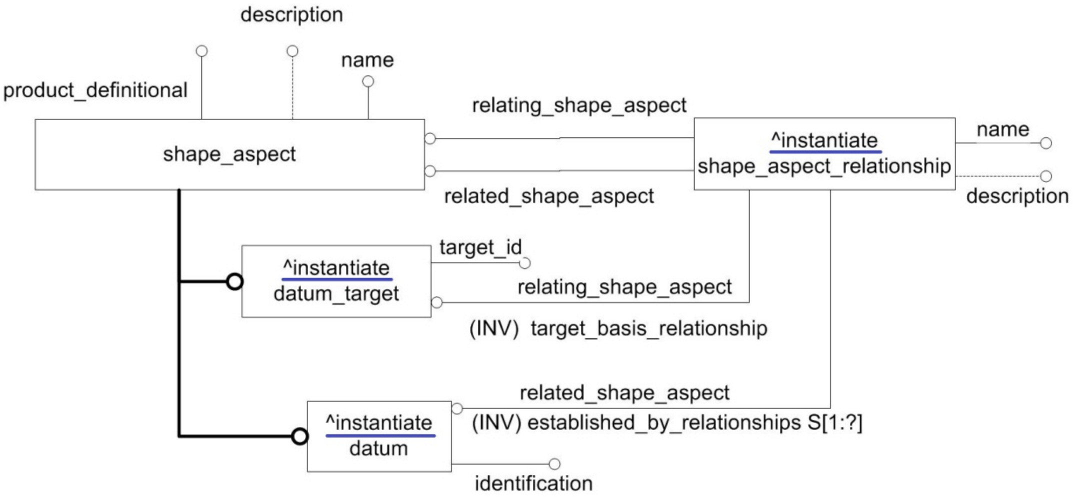


Figure 71: The template representing_datum_with_datum_target

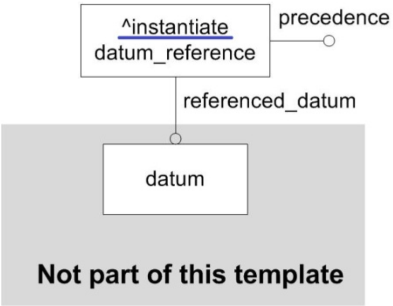


Figure 72: The template representing_datum_reference

Referencing a datum with a material modifier condition for a geometric tolerance is represented by referenced_modified_datum, which is presented in Figure 73. In this case, the datum feature should be a feature of size, and the geometric tolerance should be defined with a modified datum. The referenced_modified_datum is a subset of the datum_reference, and in addition to superset attributes, it has another attribute called modifier. The following modifiers can be defined by the enumeration type limit_condition:

- MAXIMUM_MATERIAL_CONDITION
- LEAST_MATERIAL_CONDITION
- REGARDLESS_OF_FEATURE_SIZE.

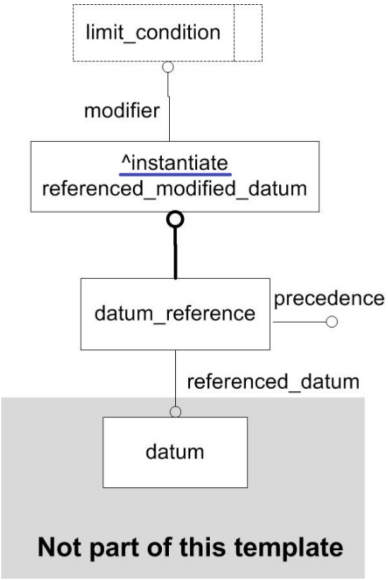


Figure 73: The template representing_referenced_modified_datum

5.1.2 Templates for the Dimensions Information Model

The description of location and size dimensions is defined in Part 47. In order to define a dimension between two geometric items, the template for representing dimensional location can be used (see Figure 74). The entity dimensional_location is a subtype of the shape_aspect_relationship; it implicitly describes the location dimension between two shape_aspects. The dimensions that are mapped to name attributes of dimensional_locations are given in Table 7. The dimensional_location has two subsets: angular_location and dimensional_location_with_path. The angular_location is instantiated to describe the measure of the angle defined by the two shape_aspect elements (see Figure 75). Its angle selection attribute is an angle_relator, and the allowable values for angle_relator are "equal," "large" and "small." The dimensional_location_with_path, as a specialized subtype of dimensional_location, uses the same structure as the template in Figure 74. This subtype entity provides a path for the measurement to follow by means of a shape_aspect (Figure 76).

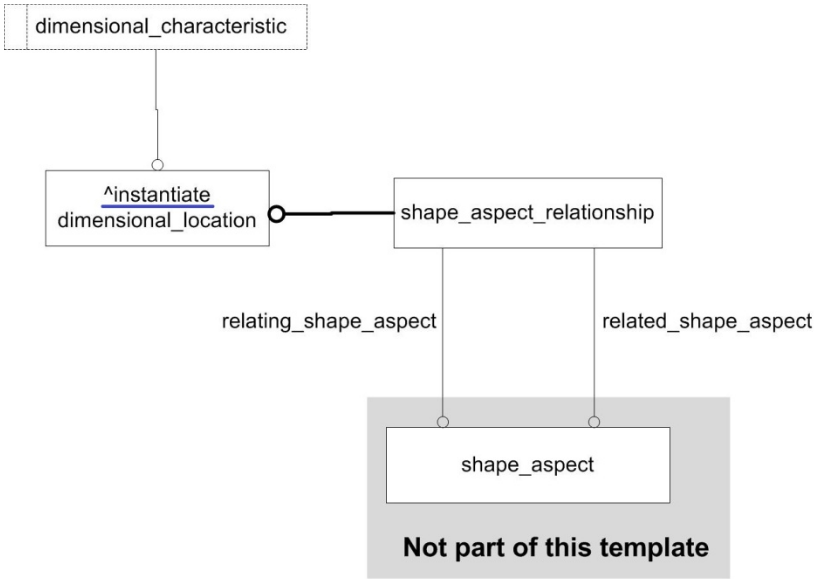


Figure 74: The template representing_dimensional_location

Table 7: Name Attributes of Dimensional_location

Dimensional location	*Dimensional_location.name*
Angular location dimension	N/A
Curved distance dimension	'curved distance'
Linear distance dimension	'linear distance'

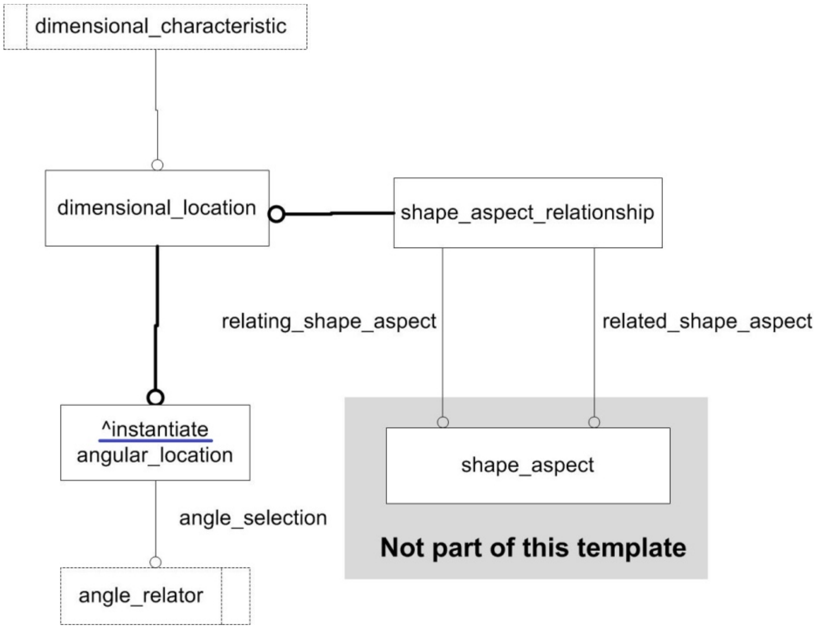


Figure 75: The template representing_angular_location

Size dimensions in GD & T specifications can be defined by the template representing_dimensional_size, shown in Figure 77. The dimensional_size is used when the measurement applies to only one feature of size, rather than involving a relationship between two distinct geometric or topological features. Note that this "one feature of size" can, under certain circumstances, be a composite of several shape_aspects. Also, the shape_aspect used for dimensional_size should lie on

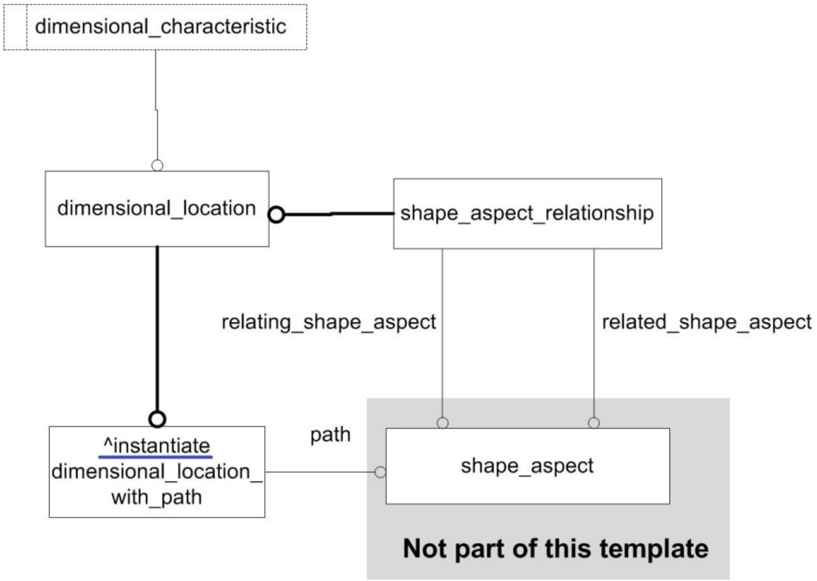


Figure 76: The template representing_dimensional_location_with_path

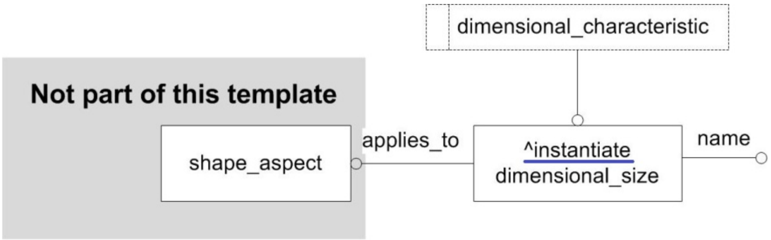


Figure 77: The template representing_dimensional_size

the physical boundary of the shape, which means the product_definitional attribute should be set to “true.” The dimensions that are mapped to name attributes of dimensional_sizes are given in Table 8. The dimensional_size has two subsets: angular_size and dimensional_size_with_path. The angular_size is instantiated to describe an angular measure between two boundaries of a shape_aspect (Figure 78). Its angle selection attribute is an angle_relator select type. A template for representing dimensional_size_with_path is shown in Figure 79. Its structure is similar to that of the template for representing_dimensional_size. In Figure 79, a path for the measurement to follow by means of a shape_aspect is provided.

Table 8: Name Attributes of Dimensional_size

Dimensional size	*Name* attribute
Angular size dimension	N/A
Curved size dimension	'curve length'
Diameter size dimension	'diameter'
Height size dimension	'height'
Length size dimension	'length'
Radial size dimension	'radius'
Thickness size dimension	'thickness'
Width size dimension	'width'

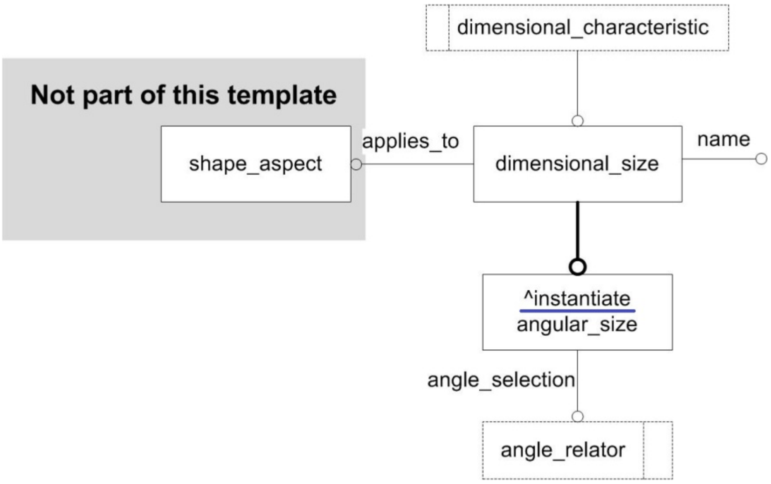


Figure 78: The template representing_angular_size

The dimensional_size and dimensional_location entities both define implicit dimensions which can be derived from geometric definitions of

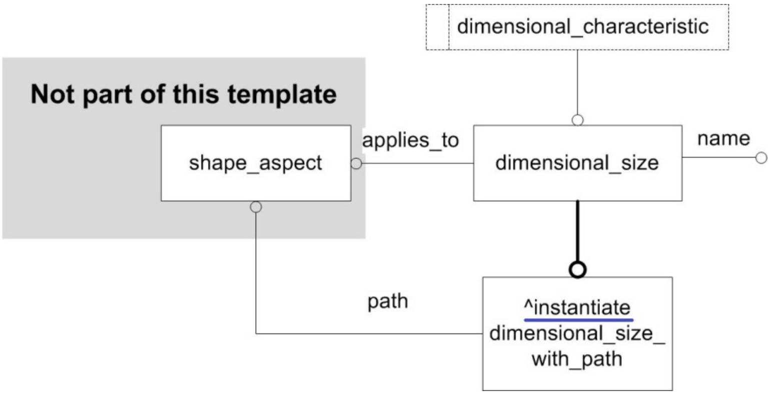


Figure 79: The template representing_dimensional_size_with_path

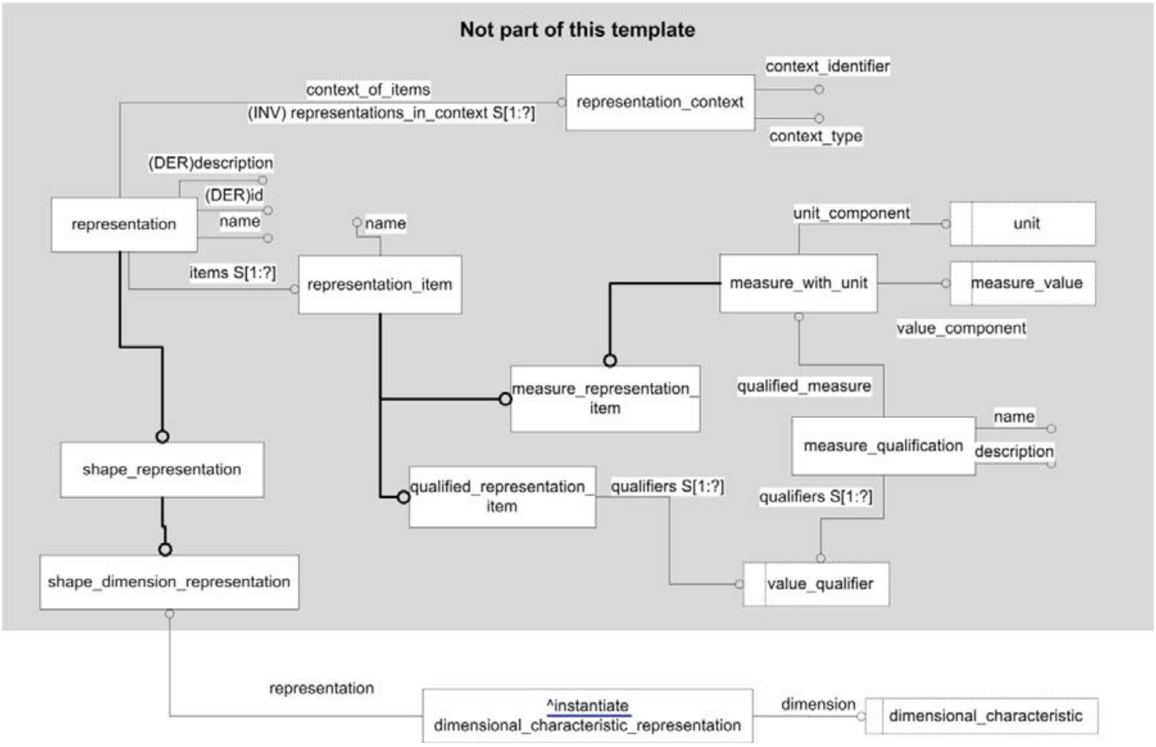


Figure 80: The template representing_dimensional_characteristic_representation

shape_aspects. In order to associate explicit dimensions with these implicit dimension definitions, the entity dimensional_characteristic_representation is defined in Part 47, and the template for this representation is shown in Figure 80. As is evident from the figure, the representation attribute is associated with the explicit non-geometric dimension representation, and the dimension attribute is associated with the implicit dimension definition through select types of the entity dimensional_characteristic. In this figure, the items attribute of shape_dimension_representation should have only measure_representation_items. On the other hand, dimensional_characteristic can be either dimensional_location, dimensional_size or one of their subtypes. For example, the size dimension for the pattern of holes in our sample part is shown in Figure 81, using the templates representing_dimensional_size and representing_dimensional_characteristic_representation.

5.1.3 Templates for the Tolerances Information Model

In part 47, dimensional tolerances are defined by the entities tolerance_value and limits_and_fits. The tolerance_value is used to define plus-minus tolerances, and the tolerances specified by this entity consist of numeric values added to the nominal dimension of a shape_aspect. A template for this representation is shown in Figure 82. There are two constraints for the tolerance_value: the upper_bound value should be bigger than the lower_bound value, and the values for both the upper and lower bounds should be the same unit of measure. The entity limits_and_fits, on the other hand, is used to specify tolerances within a pre-defined fit system (Figure 83).

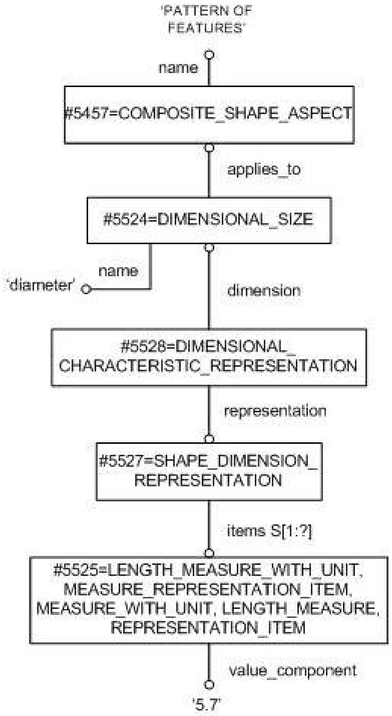

'PATTERN OF FEATURES'
name
#5457=COMPOSITE_SHAPE_ASPECT
applies_to
#5524=DIMENSIONAL_SIZE
name
'diameter'
dimension
#5528=DIMENSIONAL_ CHARACTERISTIC_REPRESENTATION
representation
#5527=SHAPE_DIMENSION_ REPRESENTATION
items S[1:?]
#5525=LENGTH_MEASURE_WITH_UNIT, MEASURE_REPRESENTATION_ITEM, MEASURE_WITH_UNIT, LENGTH_MEASURE, REPRESENTATION_ITEM
value_component
'5.7'

Figure 81: Instances for the dimensional size of the pattern of holes

The association between a dimensional tolerance and a dimension is defined by the entity `plus_minus_tolerance`. A template for this representation is shown in Figure 84. For this template, `dimensional_characteristic` can have values of either `dimensional_location`, `dimensional_size` or one of their subtypes. The `tolerance_method_definition` can have only one of two values: `tolerance_value` or `limits_and_fits`. For example, Figure 85 shows the plus/minus tolerances for the size of the hole feature in our sample, using the templates representing_tolerance_value, representing_plus_minus_tolerance and representing_dimensional_size.

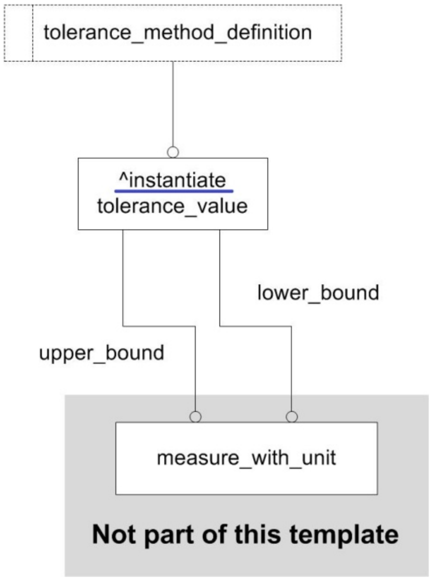


Figure 82: The template representing_tolerance_value

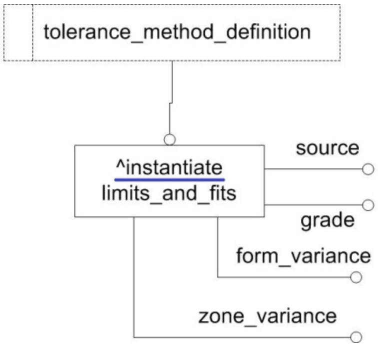


Figure 83: The template representing_limits_and_fits

Geometric tolerance definitions can be categorized into six types: (1) a geometric tolerance without any datum or any material modifier, (2) a geometric tolerance with a material modifier, (3) a geometric tolerance with a datum, (4) a geometric tolerance with a modified datum, (5) a geometric tolerance specified on a per unit basis of the toleranced feature, and (6) a composite tolerance (i.e., a feature control frame with at least two geometric tolerances).

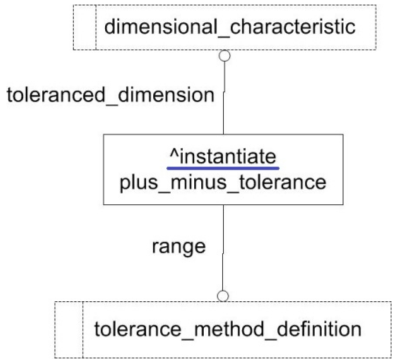


Figure 84: The template representing_plus_minus_tolerance

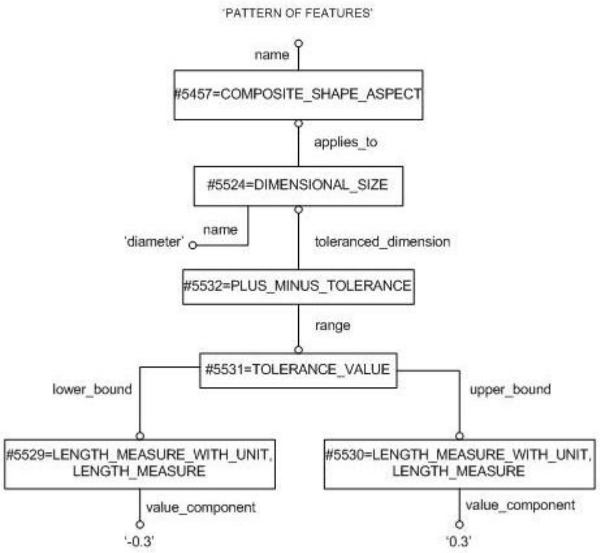


Figure 85: Instances for dimensional tolerance for the pattern of holes

(1) The specification of a geometric tolerance without a datum and material modifier can be created using the template representing_geometric_tolerance (Figure 86). The `name` attribute of the `geometric_tolerance` describes the type of tolerance (e.g., flatness tolerance). The `description` attribute is used as a supplementary note. The `magnitude` attribute states the size of the tolerance and

should be equal to or greater than 0. The toleranced_shape_aspect attribute refers to the shape_aspect for which a geometric tolerance is being defined.

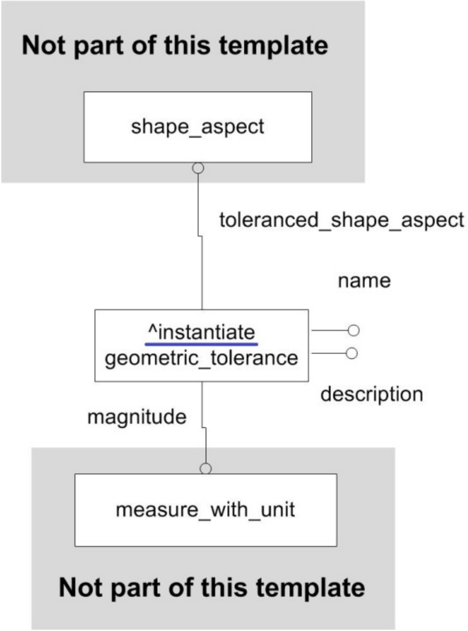


Figure 86: The template representing_geometric_tolerance

(2) Representation of a geometric tolerance with a material modifier is shown in Figure 87— the template representing_modified_geometric_tolerance. Both the geometric_tolerance and modified_geometric_tolerance entities are instantiated in this template. The modifier attribute of the modified_geometric_tolerance entity is a limit_condition enumeration type. It can assign one of the following values:

- MAXIMUM_MATERIAL_CONDITION
- LEAST_MATERIAL_CONDITION
- REGARDLESS_OF_FEATURE_SIZE.

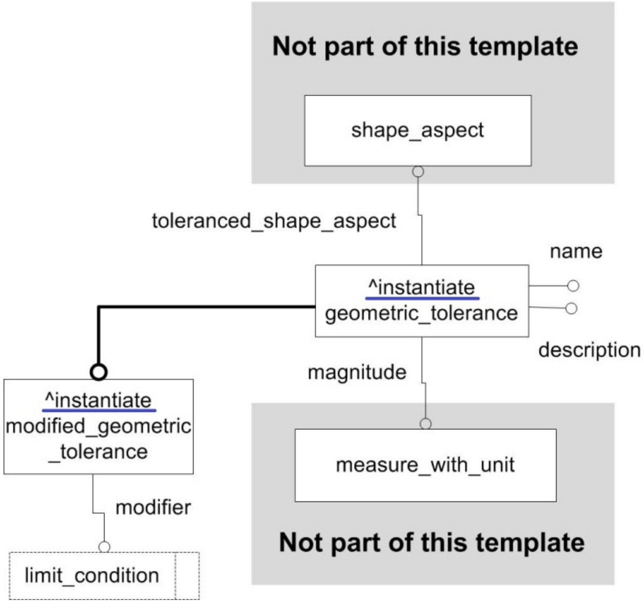


Figure 87: The template representing_modified_geometric_tolerance

(3) A geometric tolerance with a datum can be represented by using the template representing_geometric_tolerance_with_datum_reference (Figure 88). Both the `geometric_tolerance` and `geometric_tolerance_with_datum_reference` entities are instantiated in this template. The `geometric_tolerance_with_datum_reference` has only one attribute, which is `datum_systems`, associated with a set of `datum_references`. In the feature control frame, the datum, if present, appears in a specific order, i.e. primary, secondary or tertiary. However, the STEP implementation of the datum systems is an unordered list. The precedence of the datum in the feature control frame is, therefore, given by the `precedence` attribute in the `datum_reference`. This is important to note, as there is no guarantee that the order in the `geometric_tolerance_with_datum` is correct.

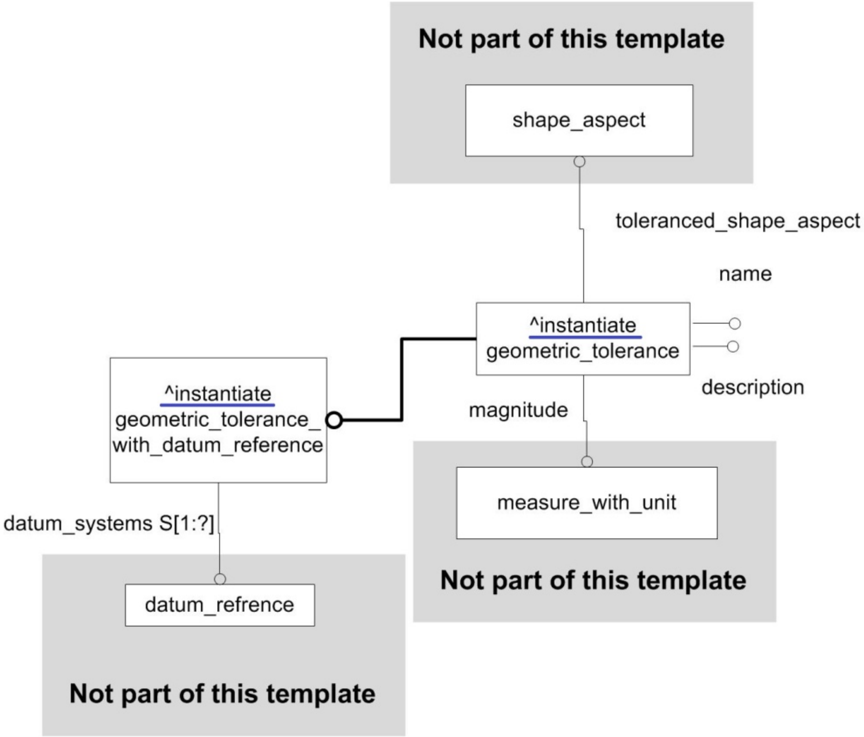


Figure 88: The template representing_geometric_tolerance_with_datum_reference

(4) A geometric tolerance with a modified datum is instantiated by using the same template from Figure 88. However, in this case, the `referenced_modified_datum` (Figure 73) is instantiated instead of a `datum_reference` for the `datum_systems` attribute. For example, the geometric tolerance specified for the pattern of holes in our sample part is shown in Figure 89; it was created using the following templates: representing_geometric_tolerance, representing_modified_geometric_tolerance, representing_geometric_tolerance_with_datum_reference, representing_datum_with_datum_feature and representing_datum_reference.

(5) A geometric tolerance specified on a per unit basis of the toleranced feature is represented by the `geometric_tolerance_with_defined_unit`. A template for this representation is shown in Figure 90. This entity has an additional attribute, `unit_size`, to represent the unit measure applied to a tolerance.

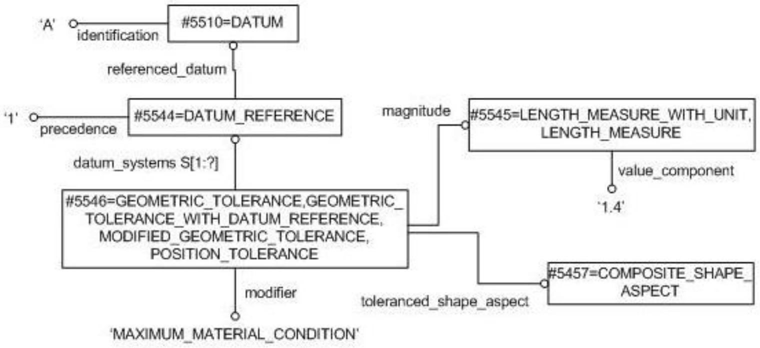

'A'
identification
#5510=DATUM
referenced_datum
'1'
precedence
#5544=DATUM_REFERENCE
datum_systems S[1:?]
magnitude
#5545=LENGTH_MEASURE_WITH_UNIT, LENGTH_MEASURE
value_component
'1.4'
#5546=GEOMETRIC_TOLERANCE,GEOMETRIC_TOLERANCE_WITH_DATUM_REFERENCE, MODIFIED_GEOMETRIC_TOLERANCE, POSITION_TOLERANCE
#5457=COMPOSITE_SHAPE_ASPECT
modifier
toleranced_shape_aspect
'MAXIMUM_MATERIAL_CONDITION'

Figure 89: Instances for geometric tolerance

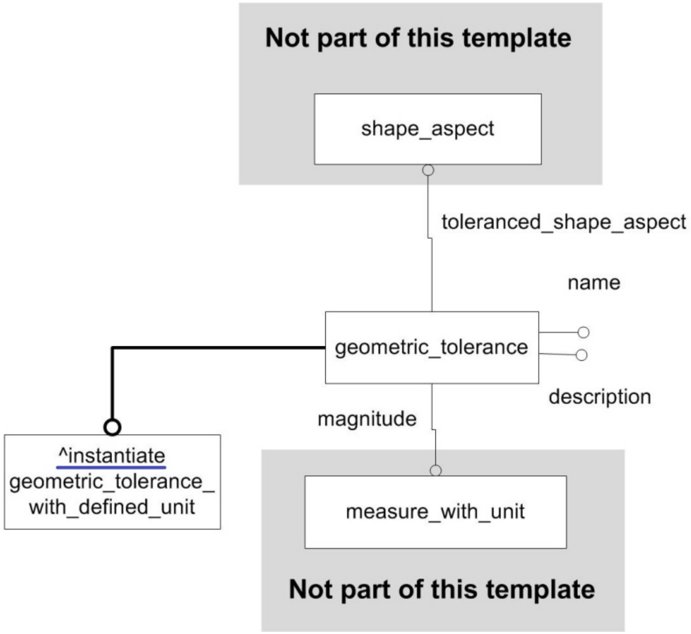

Not part of this template
shape_aspect
toleranced_shape_aspect
name
geometric_tolerance
description
magnitude
^instantiate geometric_tolerance_with_defined_unit
measure_with_unit
Not part of this template

Figure 90: The template representing_geometric_tolerance_with_defined_unit

(6) A composite geometric tolerance, assuming it has two feature control frames, can be instantiated by first instantiating the `geometric_tolerance` for each control frame, and then these two `geometric_tolerance` instances can be connected by the `geometric_tolerance_relationship`. A template for this representation is presented in Figure 91. The `description` attribute of the `geometric_tolerance_relationship` should be set as "composite tolerance." In the `geometric_tolerance_relationship`, the upper feature control frame is the `relating_geometric_tolerance`, and the lower one is the `related_geometric_tolerance`. If there are more than two feature control frames for a composite geometric tolerance, more than one `geometric_tolerance_relationship` has to be instantiated. For example, a composite tolerance with three feature control frames would require two `geometric_tolerance_relationship` instances: the first one relating to the top and middle feature control frames, and the second to the middle and bottom feature control frames.

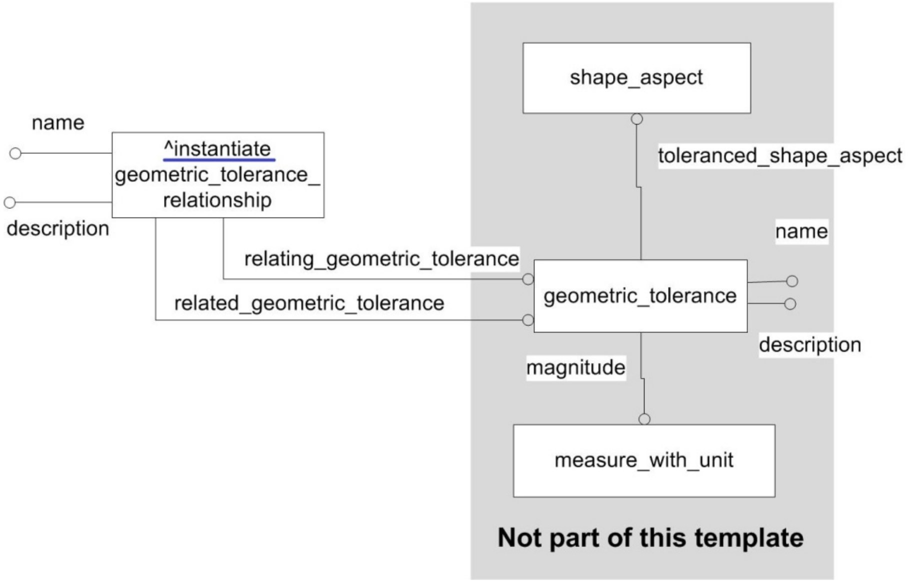


Figure 91: Template representing_geometric_tolerance_relationship

Chapter 6

Interpreting the Semantics of GD & T Specifications of a Product for Tolerance Analysis

6.1 The Proposed Methodology

In this chapter, the mechanism to interpret the semantics of GD & T specifications for tolerance analysis is described. The schematic representation of the mechanism is shown in Figure 92. The first step is to develop the information model for tolerance analysis in EXPRESS so that this model can be integrated with the AP 214 EXPRESS model (because the product, geometry, structure and GD & T information is defined in AP 214). The second step is to translate this EXPRESS model into OWL 2 through the OntoSTEP Protégé plug-in. Then, the SWRL (2012) rules are developed to map the GD & T specifications into the tolerance analysis application domain. The obtained ontology in OWL is used for developing the complete product data instances.

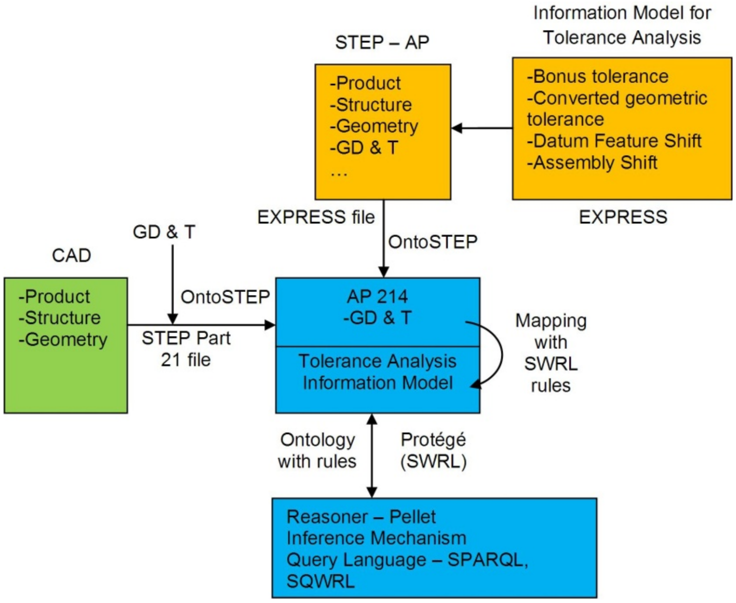


Figure 92: The schematic representation of the interpretation process

3-D product models are created in CAD systems. These models are then translated into STEP formats. Since the current STEP translators do not create the GD & T specifications, these specifications have to be added manually to the STEP file. The final STEP file is translated to the developed OWL schema to take advantage of inferencing and reasoning mechanisms. The details of these steps are discussed in the following sub-sections: (1) the development of the tolerance analysis information model, (2) the translation of AP 214 and tolerance analysis EXPRESS models into OWL 2, (3) Mapping the GD & T specifications to the specifications needed for tolerance analysis.

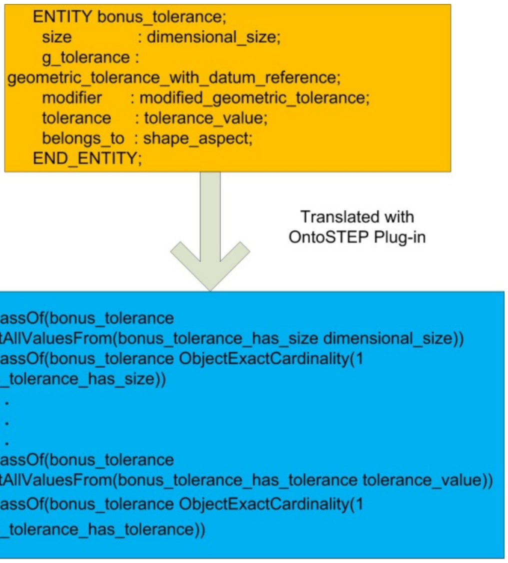
6.1.1 Development of the Tolerance Analysis Information Model

The tolerance analysis helps evaluate the effects of assigned tolerances on a product. The GD & T assigned to a product leads to five main types of variation that needs to be interpreted in a tolerance analysis: (1) dimensional tolerance, (2) converted geometric tolerance, (3) bonus tolerance, (4) datum feature shift and (5) assembly shift. In section 4, an information model in EXPRESS for the representation of these variation items (except the dimensional tolerance) has been developed. The dimensional tolerance representation is already available in AP 214.

6.1.2 Translation of the EXPRESS Models to OWL Schema

In this section, we would like to discuss how to translate and import EXPRESS models into OWL schema using OntoSTEP plug-in. In this translation process, entities and instances in EXPRESS are mapped to classes and individuals in OWL, respectively. In Figure 93, the translation process is shown schematically. The EXPRESS definition of the `bonus_tolerance` is translated into the functional syntax representation of OWL in the figure. The `bonus_tolerance` entity is mapped to the corresponding class with the same name in OWL. The attributes of this entity is mapped to ObjectProperties in OWL. For example, the `size` attribute is mapped to the *bonus_tolerance_has_size* ObjectProperty. In OWL, the domain and range of the properties are constrained explicitly. For the translation, the domain is the class name representing the translated entity (e.g., the `bonus_tolerance`) and range is the class name that represents entity type of attribute (e.g., `dimensional_size`). Unless the properties are restricted in OWL, they can be used to connect each individual of their domain to many individuals of their range. Hence, the usage of the *bonus_tolerance_has_size* property has to be restricted with the cardinality definition of “ObjectExactCardinality” construct in OWL

(Figure 93). The range of this property has to be restricted with the "ObjectAllValuesFrom".



```
    ENTITY bonus_tolerance;
      size            : dimensional_size;
      g_tolerance :
geometric_tolerance_with_datum_reference;
      modifier     : modified_geometric_tolerance;
      tolerance    : tolerance_value;
      belongs_to  : shape_aspect;
    END_ENTITY;
```

```
SubClassOf(bonus_tolerance
ObjectAllValuesFrom(bonus_tolerance_has_size dimensional_size))
SubClassOf(bonus_tolerance ObjectExactCardinality(1
bonus_tolerance_has_size))
        .
        .
        .
SubClassOf(bonus_tolerance
ObjectAllValuesFrom(bonus_tolerance_has_tolerance tolerance_value))
SubClassOf(bonus_tolerance ObjectExactCardinality(1
bonus_tolerance_has_tolerance))
```

Figure 93: Mapping EXPRESS models to OWL schema

Further details of translating the STEP models into OWL schema can be found in (Barbau et al., 2012). In our study, the EXPRESS definitions of the `converted_geometric_tolerance`, the `bonus_tolerance`, the `datum_feature_shift` and the `assembly_shift` are combined with AP 214 EXPRESS schema. Then, this schema is imported into OWL schema using OntoSTEP plug-in.

6.1.3 Mapping GD & T Specifications to the Specifications Needed for Tolerance Analysis

In this study, the rule-based reasoning is used for the interpretation of the GD & T specifications for tolerance analysis. The rules are developed in the SWRL language. Before developing the rules, it should be noted that Protégé and associated SWRL language have limited capabilities for reasoning. The followings are some of the limitations that we encountered during the mapping process:

1. The SWRL rules cannot be used to create new individuals in ontology. These rules can only be used for reclassifying an individual's class definition and/or add new relationships between the available individuals.
2. Another issue with Protégé and SWRL is that Protégé is not a computational application program. The calculations necessary for the creation of the concepts of the tolerance analysis from the GD & T require computational tools. With the SWRL rules, only one of the simple addition, subtraction, multiplication and division can be done once. It is not possible to make calculations of the several mathematical operations to infer a fact. This prevents the calculation of the values of the tolerance analysis concepts directly from the GD & T in Protégé.

In order to cater these limitations, the values of datum feature shift and assembly shift are calculated beforehand and corresponding individuals in the OWL schema are created manually. The Pellet-based (Pellet, 2012) reasoning mechanism is then used to infer the additional relationships required for the complete class definitions of `converted_geometric_tolerance`, `bonus_tolerance`, `datum_feature_shift`, etc. In the following sub-sections, the development of the SWRL rules for (1) the converted geometric tolerance, (2) the bonus tolerance and (3) the datum feature shift are discussed.

6.1.3.1 The SWRL Rules for the Converted Geometric Tolerance

Since any geometric tolerance has to be converted to plus/minus tolerances in a tolerance analysis process, we would like to infer the values of these plus/minus tolerances as well as the relationships required for the `converted_geometric_tolerance`. It should be noted that the structure of the information is defined in Section 4.4.3.2 for this concept.

The SWRL rules developed for inferring the instances of the `converted_geometric_tolerance` are given in Table 9. The difference between these two rules is that in the first rule, the positive tolerance value of the plus/minus tolerances is inferred for the `converted_geometric_tolerance` whereas in the second rule, the negative tolerance value is inferred.

In Table 9, the antecedent column shows the requirements of the rules for the application of reasoning whereas the consequent column shows the results that will be inferred after reasoning. Both of the rules in Table 9 require the following conditions:

- A geometric tolerance with a unit and a value component should be defined.
- This geometric tolerance should be assigned to a shape aspect.
- An instance of the `converted_geometric_tolerance` should be available.
- This instance should be connected to the geometric tolerance as well as the instance for the plus/minus tolerance (i.e., `tolerance_value`).

Table 9: The SWRL rules for the `converted_geometric_tolerance`

Rule	Antecedent	Consequent
1	geometric_tolerance(?y), shape_aspect(?x), tolerance_value(?a), length_measure(?v), converted_geometric_tolerance_has_g_tolerance(?b,?y), converted_geometric_tolerance_has_tolerance(?b,?a), geometric_tolerance_has_toleranced_shape_aspect(?y,?x), geometric_tolerance_has_magnitude(?y,?m), measure_with_unit_has_unit_component (?m,?u), measure_with_unit_has_value_component(?m,?v), to_decimal(?v, ?w), divide(?r, ?w, 2), tolerance_value_has_upper_bound(?a,?u1), measure_with_unit_has_value_component(?u1,?v1)	converted_geometric_tolerance_has_belongs_to(?b,?x), measure_with_unit_has_unit_component (?u1, ?u), to_decimal(?v1,?r)
2	geometric_tolerance(?y), shape_aspect(?x), tolerance_value(?a), length_measure(?v), converted_geometric_tolerance_has_g_tolerance(?b,?y), converted_geometric_tolerance_has_tolerance(?b,?a), geometric_tolerance_has_toleranced_shape_aspect(?y,?x), geometric_tolerance_has_magnitude(?y,?m), measure_with_unit_has_unit_component(?m,?u), measure_with_unit_has_value_component(?m,?v), to_decimal(?v,?w), divide(?r,?w,-2), tolerance_value_has_lower_bound(?a,?u1), measure_with_unit_has_value_component(?u1,?v1)	converted_geometric_tolerance_has_belongs_to(?b,?x), measure_with_unit_has_unit_component (?u1,?u), to_decimal(?v1,?r)

If these conditions are satisfied the reasoning mechanism will infer automatically the values and units of the plus/minus tolerances for the instance of the `converted_geometric_tolerance`. The relationship between the instance of the `converted_geometric_tolerance` and the `shape_aspect` will be inferred as well.

6.1.3.2 The SWRL Rules for the Bonus Tolerance

In this section, the development of the SWRL rules for inferring the bonus tolerance from the available GD & T information is discussed. The information structure needed for the development of the SWRL rules has been given in section 4.4.3.4.

Since a bonus tolerance is calculated for a feature of size that has a dimension and a geometric tolerance with a material modifier, there are two rules developed for the bonus tolerance (Table 10). Rule-1 is considered for a feature of size that has a geometric tolerance with the least material condition as the material modifier; whereas the Rule-2 is considered when the geometric tolerance has the material modifier of the maximum material condition. Other than this distinction, the conditions required for both rules are given as follows:

Table 10: The SWRL rules for the `bonus_tolerance`

Rule	Antecedent	Consequent
1	dimensional_size(?z), modified_geometric_tolerance(?y), shape_aspect(?x), plus_minus_tolerance(?a), tolerance_value(?r), dimensional_size_has_applies_to(?z,?x), geometric_tolerance_has_toleranced_shape_aspect (?y,?x), modified_geometric_tolerance_has_modifier (?y,least_material_condition), plus_minus_tolerance_has_toleranced_dimension (?a,?z), plus_minus_tolerance_has_range(?a,?r), bonus_tolerance_has_belongs_to(?b,?x)	bonus_tolerance_has_tolerance (?b, ?r), bonus_tolerance_has_size(?b,?z), bonus_tolerance_has_g_tolerance (?b,?y),bonus_tolerance_has_modifier (?b,least_material_condition)
2	dimensional_size(?z), modified_geometric_tolerance(?y), shape_aspect(?x), plus_minus_tolerance(?a), tolerance_value(?r), dimensional_size_has_applies_to(?z,?x), geometric_tolerance_has_toleranced_shape_aspect (?y,?x),modified_geometric_tolerance_has_modifier (?y,maximum_material_condition), plus_minus_tolerance_has_toleranced_dimension (?a,?z), plus_minus_tolerance_has_range(?a,?r), bonus_tolerance_has_belongs_to(?b,?x)	bonus_tolerance_has_tolerance(?b,?r), bonus_tolerance_has_size(?b,?z), bonus_tolerance_has_g_tolerance (?b,?y),bonus_tolerance_has_modifier (?b,maximum_material_condition)

- A dimension and a geometric tolerance should be assigned to a feature of size.
- The value and unit components of the plus/minus tolerances for the dimension should be defined.
- The assigned geometric tolerance should have a material modifier.

- The instance of the bonus_tolerance should be connected to the feature of size (i.e., shape_aspect).

The inferencing mechanism will infer the following information once the conditions described above is true:

- the plus/minus tolerances for the bonus tolerance,
- the dimensional size used for the bonus tolerance,
- the geometric tolerance used for the bonus tolerance,
- the material modifier used for the bonus tolerance.

6.1.3.3 The SWRL Rules for the Datum Feature Shift

The datum feature shift in a tolerance analysis process is defined for any geometric tolerance with a modified datum. The conversion of the datum feature shift from the GD & T is dependent on the type of datum feature (i.e., internal feature or external feature), the geometric tolerance assigned to the datum feature and the material modifier defined for this geometric tolerance. Different use cases for the datum feature shift are presented in Table 11. It is divided into three main sections. The first row shows that there is no datum feature shift for a geometric tolerance definition without a modified datum. The second row is used when a geometric tolerance has a modified datum but the datum feature does not have any geometric tolerance. The last group in the table is used for any geometric tolerance with a modified datum and when the datum feature has a geometric tolerance.

Table 11: Datum Feature Shift

Referred datum	Modifier	Internal Datum Feature	External Datum Feature
Datum referenced w/o modifier		0	0
Modified datum and datum feature w/o geometric tolerance		Upper limit – lower limit	Upper limit – lower limit
Modified datum and datum feature with geometric tolerance	RFS	N/A	N/A
	MMC	(Nominal size + upper limit) – (Nominal size – lower limit – geometric tolerance)	(Nominal size + upper limit + geometric tolerance) – (Nominal size – lower limit)
	LMC	(Nominal size + upper limit + geometric tolerance) – (Nominal size – lower limit)	(Nominal size + upper limit) – (Nominal size – lower limit – geometric tolerance)

For the first use case, there is no datum feature shift and there is no need to define any rule. The rules defined for the second case are subsets of the rules defined for the third case however; this relationship between the rules creates a problem for the reasoning purposes.

The reasoning mechanism will infer both of the rules (i.e., the second case and the third case rules) are true and generate the new inferred information. The cause of this problem is that in the second rule, we should be able to define the rule for a geometric tolerance with modified datum and the corresponding datum feature should not have any geometric tolerance. This is not possible to define in ontology because of the open world assumption. In the ontology, a datum feature without a geometric tolerance does not mean it will not have any geometric tolerance. It only means this geometric tolerance information is not available yet. Therefore, we only concentrate on the last row which is showing how to calculate the datum feature shift for a geometric tolerance with modified datum and when the corresponding datum feature has a geometric tolerance with material modifier.

The geometric tolerance assigned to the datum feature can have three different material modifiers: the regardless of feature of size (RFS), the maximum material condition (MMC) and the least material condition (LMC). The RFS condition does not yield any datum feature shift. Because of that reason, only two cases are considered for the development of the SWRL rules in this study: the MMC and the LMC.

The SWRL rules developed for inferring the instances of the `datum_feature_shift` are given in Table 12. The information structure needed for the development of the SWRL rules has been given in Section 4.4.3.5. The first rule is developed for the MMC condition in Table 11 whereas the second rule is developed for the LMC condition. Both of the rules defined in Table 12 require the following conditions:

- A geometric tolerance with a modified datum should be defined for a feature (i.e., `shape_aspect`).
- The corresponding datum feature should have a geometric tolerance with a material modifier.
- The datum feature has to be a feature of size.
- The instance of the `datum_feature_shift` should be connected to the `geometric_tolerance`.

If these conditions are satisfied the reasoning mechanism will infer automatically the relationship between datum feature shift and the followings:

Table 12: The SWRL rules for the `datum_feature_shift`

Rule	Antecedent	Consequent
1	geometric_tolerance_with_datum_reference(?y), shape_aspect(?x), geometric_tolerance_has_toleranced_shape_aspect (?y,?x),geometric_tolerance_with_datum_reference_has_ datum_ system(?y,?d), set_of_datum_reference_has_content(?d,?d1), referenced_modified_datum(?d1), referenced_modified_datum_has_modifier (?d1,maximum_material_condition), datum_reference_has_referenced_datum(?d1,?d1d), shape_aspect_relationship_has_related_shape_aspect (?sr,?d1d), shape_aspect_relationship_has_relating_shape _aspect(?sr,?d1df), datum_feature(?d1df), datum(?d1d), shape_aspect_relationship_has_relating_shape_aspect (?sr2,?d1df), shape_aspect_relationship_has_related_shape_aspect (?sr2,?s1), dimensional_size(?ds), dimensional_size_has_applies_to(?ds,?s1), datum_feature_shift_has_g_tolerance(?dfs,?y), modified_geometric_tolerance(?y1), geometric_tolerance_has_toleranced_shape_aspect(?y1,?s1), modified_geometric_tolerance_has_modifier(?y1,maximum_material_condition)	datum_feature_shift_has_datum_modifier (?dfs,?d1), datum_feature_shift_has_datum_size (?dfs,?ds), datum_feature_shift_has_belongs_to (?dfs,?x)
2	geometric_tolerance_with_datum_reference(?y), shape_aspect(?x), geometric_tolerance_has_toleranced_shape_aspect (?y,?x),geometric_tolerance_with_datum_reference_has_ datum_ system(?y,?d), set_of_datum_reference_has_content(?d,?d1), referenced_modified_datum(?d1), referenced_modified_datum_has_modifier (?d1,least_material_condition), datum_reference_has_referenced_datum(?d1,?d1d), shape_aspect_relationship_has_related_shape_aspect (?sr,?d1d), shape_aspect_relationship_has_relating_shape _aspect(?sr,?d1df), datum_feature(?d1df), datum(?d1d), shape_aspect_relationship_has_relating_shape_aspect (?sr2,?d1df), shape_aspect_relationship_has_related_shape_aspect (?sr2,?s1), dimensional_size(?ds), dimensional_size_has_applies_to(?ds,?s1), datum_feature_shift_has_g_tolerance(?dfs,?y), modified_geometric_tolerance(?y1), geometric_tolerance_has_toleranced_shape_aspect(?y1,?s1), modified_geometric_tolerance_has_modifier(?y1,maximum_material_condition)	datum_feature_shift_has_datum_modifier (?dfs,?d1), datum_feature_shift_has_datum_size (?dfs,?ds), datum_feature_shift_has_belongs_to (?dfs,?x)

- the `referenced_modified_datum`,
- the `dimensional_size` defined for the `datum_feature`,
- the toleranced feature (i.e, `shape_aspect`) that is associated with the datum feature shift.

6.2 Case Study

In this section, the example tolerance analysis problem given in chapter 4 is discussed to verify the interpretation mechanism developed in this study. The details of the problem have been described in section 4.1.

The tolerance analysis solution (see Figure 94) is presented here with L-bracket (figure) and base (figure) for ready reference. To be able to carry out the tolerance analysis, the specifications needed for this analysis have to be created. These specifications are as follows:

- Any geometric tolerance which has an effect on the gap variation has to be converted to the plus/minus tolerance. In Figure 95, the positions of holes on the base plate have to be converted to the respective plus/minus tolerances. In Figure 96, the profile of the flange face and the positions of holes on the L-bracket have to be converted to plus/minus tolerance.
- Any applicable bonus tolerance associated with the specified geometric tolerance has to be defined. In Figure 95, there is a bonus tolerance for the positions of the holes on the base plate. In Figure 96, there is no bonus tolerance to be considered for the profile tolerances because they are specified at RFS. There is also no bonus tolerance for the positions of the holes on the L-brackets because the pattern of holes is the secondary datum and this pattern locates other features.

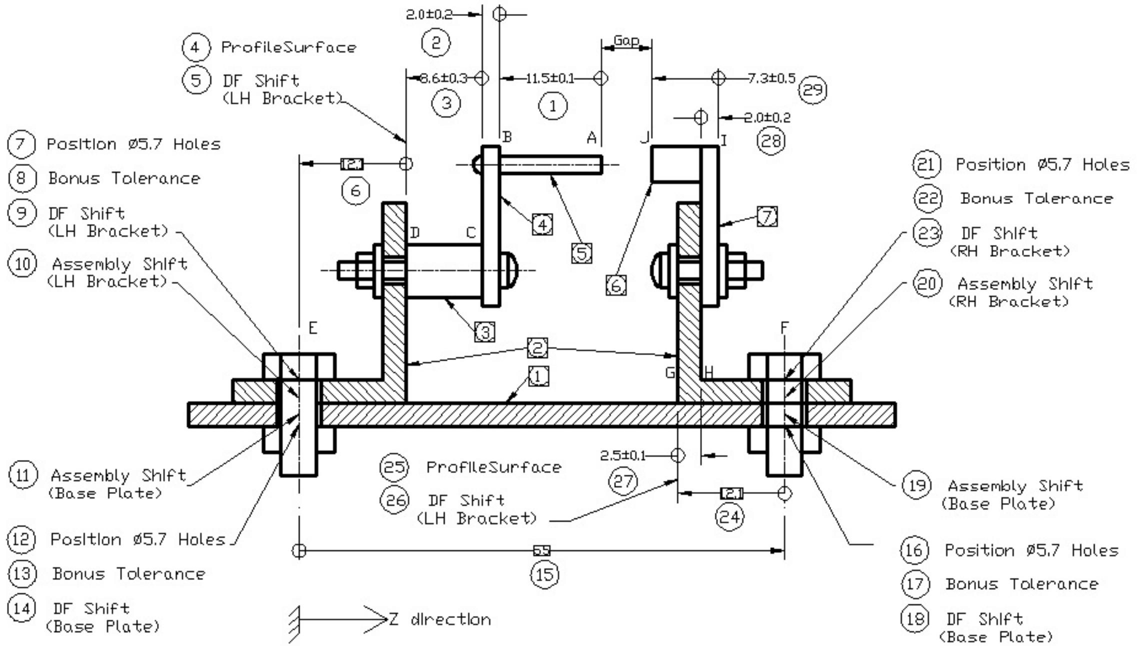


Figure 94: The stack-up analysis details for complex assembly (Fischer, 2004)

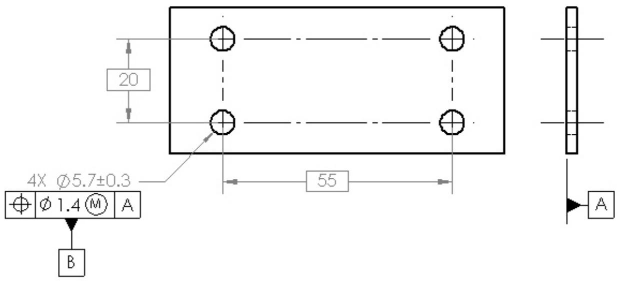


Figure 95: Details of the Base Plate (Fischer, 2004)

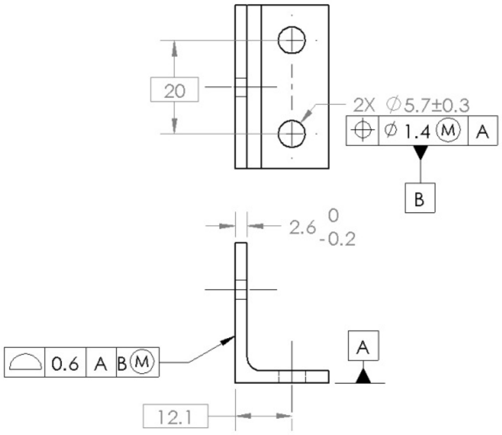


Figure 96: Details of the L-bracket (Fischer, 2004)

- Any applicable datum feature shift of the specified geometric tolerance has to be defined. A datum feature shift for the positions of the holes on both the L-bracket (Figure 96) and the base plate (Figure 95) is not applicable, because the datum features are not features of size; they are planar features. In Figure 96, the datum feature shift for the profile tolerance of the L-bracket has to be calculated.

In Figure 94, the assembly shifts at points E and F for possible position changes of the parts have to be calculated. These shifts are the maximum assembly shifts.

6.2.1 Implementation

In this section, implementation of the case study is presented. Pro/E (2012) is used to model the assembly defined in the previous section. Then, all product information for the assembly is translated into STEP format with Pro/E translator. The STEP format is set to AP214. Unfortunately, Pro/E translator cannot translate the GD & T specifications. Because of that reason, the STEP instances for the GD & T specifications are added manually to the STEP file.

In order to test the model, the obtained STEP file is imported to the ontology in OWL using OntoSTEP plug-in. This process directly creates all the OWL individuals defined in the STEP file. In the following two sections, first the individuals required for the specifications needed for tolerance analysis are defined and then, the inferred new information created after inferencing is discussed.

6.2.1.1 Asserted Instances and Properties

In this section, the instances for the classes in addition to the instances provided in the STEP file are presented with their properties needed for the ontology. These instances are the members of classes: the `converted_geometric_tolerance`, the `tolerance_value`, the `bonus_tolerance` and the `datum_feature_shift`.

There are two geometric tolerances converted to plus/minus tolerances for our tolerance analysis problem: the profile tolerance on the L-bracket and the positional tolerance on the base plate. In Table 13, the instances created for the `converted_geometric_tolerance` are given. The instances of the classes that are connected to the instances of the `converted_geometric_tolerance` through the properties are defined in rows. The letter “D” represents the domain of the property where as the letter “R” represents the range of the property.

The instances of the `tolerance_value` required for the instances of the `converted_geometric_tolerance` are created in the ontology as well. They are shown in Table 14.

Table 13: Asserted instances for the `converted_geometric_tolerance`

		Instances	
Converted_geometric_tolerance		i6031	i6036
Asserted Properties	converted_geometric_tolerance_has_g_tolerance D: converted_geometric_tolerance R: geometric_tolerance	i5639	i5546
	converted_geometric_tolerance_has_tolerance D: converted_geometric_tolerance R: tolerance_value	i6034	i6039

Table 14: Asserted instances for the `tolerance_value` of the `converted_geometric_tolerance`

		Instances	
Tolerance_value		i6034	i6039
Asserted Properties	tolerance_value_has_upper_bound D: tolerance_value R: measure_with_unit	i6032	i6037
	tolerance_value_has_lower_bound D: tolerance_value R: measure_with_unit	i6033	i6038

There is only one bonus tolerance for our tolerance analysis problem: it is defined for the pattern of holes on the base plate. In Table 15, the instance created for the `bonus_tolerance` is given. This instance is connected to the instance of the `shape_aspect` through the `bonus_tolerance_has_belongs_to` property. This instance of the `shape_aspect` is used for the definition of the positional tolerance on the base plate.

Table 15: Asserted instances for the `bonus_tolerance`

		Instances
Bonus_tolerance		i6021
Asserted Properties	bonus_tolerance_has_belongs_to D: bonus_tolerance R: shape_aspect	i5421

In our tolerance analysis problem, there is only one datum feature shift which is due to the profile tolerance on the L-bracket. In Table 16, the instance created for the `datum_feature_shift` is given. This instance is connected to an instance of the `geometric_tolerance` through the `datum_feature_shift_has_g_tolerance` property. The instance of the `geometric_tolerance` in Table 16 is the positional tolerance on the L-bracket which is used to find out the datum feature shift. Other than this property, the instance of the `datum_feature_shift` is also connected to the `tolerance_value` through the `datum_feature_shift_has_tolerance`.

Table 16: Asserted instances for the `datum_feature_shift`

		Instances
Datum_feature_shift		i6051
Asserted Properties	datum_feature_shift_has_g_tolerance D: datum_feature_shift R: geometric_tolerance	i5639
	datum_feature_shift_has_tolerance D: datum_feature_shift R: tolerance_value	i6054

6.2.1.2 Inferred Instances and Properties

In this section, the new knowledge inferred by the reasoner (i.e., Pellet) and the SWRL rules are discussed. As a result of reasoning, the new properties are inferred for the instances of the `converted_geometric_tolerance`, the `bonus_tolerance` and the `datum_feature_shift`.

The instances of the `converted_geometric_tolerance` are connected to the instances of the `shape_aspect` by the reasoner. These instances of the `shape_aspect` are used for the definition of the profile tolerance on the L-bracket and the positional tolerance on the base plate. The inferred instances are given in Table 17. Additionally, the following information is inferred by the reasoner for the instances of the `tolerance_value` that are connected to the instances of the `converted_geometric_tolerance`:

Table 17: Inferred instances for the `converted_geometric_tolerance`

		Instances	
Converted_geometric_tolerance		i6031	i6036
Inferred Properties	converted_geometric_tolerance_has_belongs_to D: converted_geometric_tolerance R: shape_aspect	i5596	i5421

- the units assigned to the instances of the geometric tolerances are connected to the instances of the `tolerance_value`
- division of the geometric tolerance value to “2” is assigned as the upper bound of the `tolerance_value`

- division of the geometric tolerance value to “-2” is assigned as the lower bound of the tolerance_value

The reasoner has created four new properties for the instance of the bonus_tolerance (Table 18). These properties define the relationship between the instance of the bonus_tolerance and the instances of any of the tolerance_value, the dimensional_size, the geometric_tolerance and the modified_geometric_tolerance. The inferred information is needed for the calculation of the instance of the bonus_tolerance. A snapshot of the Protégé is given to show the inferred new information in Figure 97.

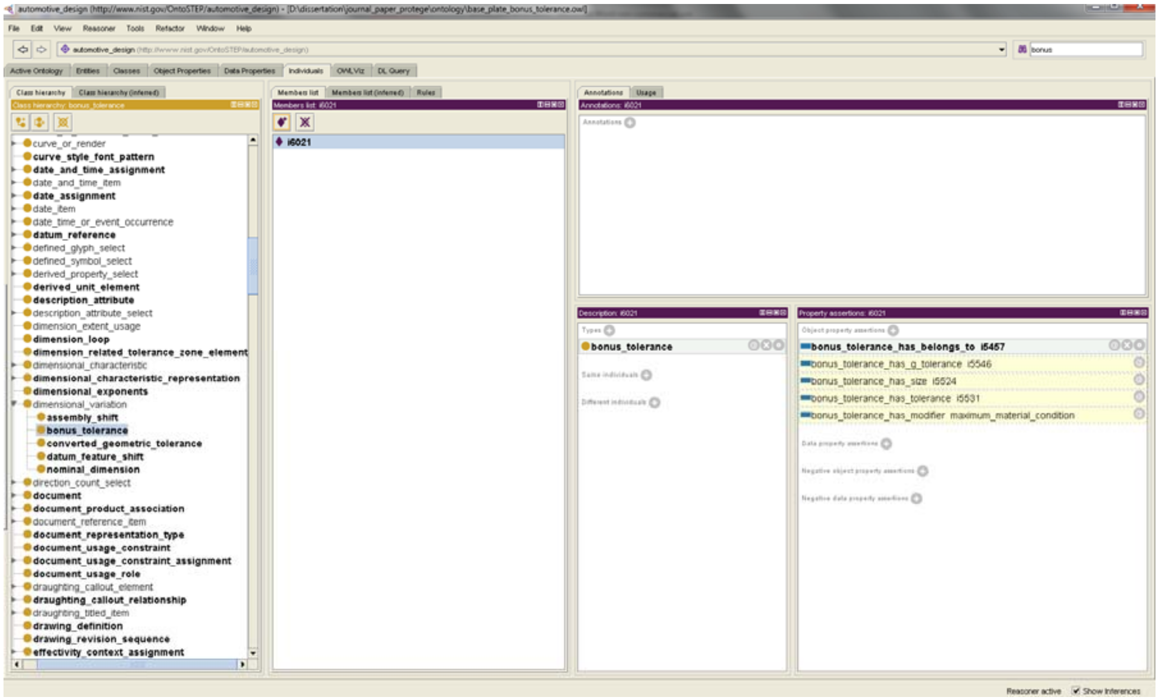
Table 18: Inferred instances for the bonus_tolerance

		Instances
Bonus_tolerance		i6021
Inferred Properties	bonus_tolerance_has_tolerance D: bonus_tolerance R: tolerance_value	i6024
	bonus_tolerance_has_size D: bonus_tolerance R: dimensional_size	i5524
	bonus_tolerance_has_g_tolerance D: bonus_tolerance R: geometric_tolerance	i5546
	Bonus_tolerance_has_modifier D: bonus_tolerance R: modified_geometric_tolerance	maximum_material_condition

For the instance of the datum_feature_shift, we have three new inferred properties. They connect the instance of the datum_feature_shift to the instances of the `referenced_modified_datum`, the `dimensional_size` and the `geometric_tolerance` as shown in Table 19.

Table 19: Inferred instances for the `datum_feature_shift`

		Instances
Datum_feature_shift		i6051
Inferred Properties	Datum_feature_shift_has_datum_ modifier D: datum_feature_shift R: referenced_modified_datum	i6037
	datum_feature_shift_has_datum_ size D: datum_feature_shift R: dimensional_size	i5604
	Datum_feature_shift_has_datum_ g_tolerance D: datum_feature_shift R: geometric_tolerance	i5635



automotive_design (http://www.nist.gov/OntoSTEP/automotive_design) - [D:\dissertation\journal_paper_protege\ontology\base_plate_bonus_tolerance.owl]

automotive_design (http://www.nist.gov/OntoSTEP/automotive_design)

Class hierarchy: bonus_tolerance

- curve_or_render
- curve_style_font_pattern
- date_and_time_assignment
- date_and_time_item
- date_assignment
- date_item
- date_time_or_event_occurrence
- datum_reference
- defined_glyph_select
- defined_symbol_select
- derived_property_select
- derived_unit_element
- description_attribute
- description_attribute_select
- dimension_extent_usage
- dimension_loop
- dimension_related_tolerance_zone_element
- dimensional_characteristic
- dimensional_characteristic_representation
- dimensional_exponents
- dimensional_variation
    - assembly_shift
    - bonus_tolerance
    - converted_geometric_tolerance
    - datum_feature_shift
    - nominal_dimension
- direction_count_select
- document
- document_product_association
- document_reference_item
- document_representation_type
- document_usage_constraint
- document_usage_constraint_assignment
- document_usage_role
- draughting_callout_element
- draughting_callout_relationship
- draughting_titled_item
- drawing_definition
- drawing_revision_sequence
- effectivity_context_assignment

Members list: i6021

i6021

Annotations: i6021

Description: i6021

Types

bonus_tolerance

Property assertions: i6021

Object property assertions

bonus_tolerance_has_belongs_to i5457
bonus_tolerance_has_g_tolerance i5546
bonus_tolerance_has_size i5524
bonus_tolerance_has_tolerance i5531
bonus_tolerance_has_modifier maximum_material_condition

Figure 97: The inferred new information for the `bonus_tolerance`

Chapter 7

Conclusion and Future Studies

Historically, the creation of products has been influenced by three primary drivers: customer needs, cost to produce, and time to market. Today a new driver is emerging: sustainability. Sustainability requirements add another dimension to the already complex demands for product life-cycle information management. The companies are required to develop sustainable manufacturing and sustainable products due to depletion of resources, environmental regulations, environmental consciousness and destroying effect that we have on Earth. This requires a fast, unambiguous and complete product information exchange among all stakeholders in a product life-cycle. The STEP standards have an ambitious scope to support complete product life-cycle data however; in order to support the development of sustainable manufacturing, STEP standards have to be enriched for many other product life-cycle activities. In this research work, only two of the most important activities that have a direct effect on manufacturing processes, product performance and end-of-life cycle activities have been studied to show how STEP-based product models can be used to support sustainable manufacturing: (1) the substance and materials information declaration as required by environmental regulations and (2) 1-D tolerance analysis. It should be noted that geometric dimension and tolerance

(GD & T) specifications are dependent on materials specifications. Also, both GD & T and materials information directly affect manufacturing processes, product quality and end-of-life cycle activities like recycling, remanufacturing, recovering and reusing.

The substance and materials information in products are screened closely by manufacturing industries due to environmental regulations. The industry standards like IPC-1752 help industry declare the substance and materials information for the purpose of material tracking and declaration in a supply chain. However, the implementation of these standards is not directly connected to the product life-cycle and needs manual or proprietary mechanisms to extract (or integrate) the required information. The representation and extraction of these information requirements from an international standard, which covers complete product lifecycle data, is a need of the hour. By doing so, companies can increase the capabilities to identify right away the substance level problems of their products related to sustainability requirements and take necessary actions faster. Therefore in this dissertation, the information requirements of IPC-1752 has been identified and extracted from integrated resources of ISO 10303- STEP. It has been shown that even though STEP integrated resources have the capabilities to represent the information requirements of materials declaration standards, implementable parts of STEP (i.e., application protocols) do not inherit full capability from these resources. Hence, additional mechanisms are needed to extract information from application protocols.

Another scope of this research was to improve the structure of the STEP information models so that a user would know exactly what needs to be implemented to carry out any particular engineering analysis (e.g., tolerance analysis). The proposed structure should also provide the information in small, repeatedly needed patterns. In this thesis, the functionality based conformance classes for 1-D tolerance analysis is developed. To achieve that a schema for tolerance analysis has been developed and integrated with STEP product models. A case study has been shown as well.

Final objective of this research was to extract any semantic information from available STEP information models. In this research, the semantic interpretations of GD & T for tolerance analysis in a computer system have been represented successfully with a case study.

7.1 Contribution

The following are the contributions of this thesis:

i. The information requirements for IPC-175X standards have been extracted from STEP standard by developing the necessary EXPRESS schema.

ii. The evaluation of representation capabilities of two (2) application protocols (AP 203 and AP 214) has been given.

iii. The functionality-based CCs (FCCs) are proposed. FCCs suggest organizing the required information for an engineering activity into hierarchical layers. It helps users to retrieve product data instances in layers. For each layer, the small repeatedly used information is grouped into templates. Hence, in a Part 21 STEP file, the data instances are organized according to first templates and then FCCs. The structure of FCCs helps to identify the available and unavailable information content in the Part 21 STEP file.

iv. FCCs for 1-D tolerance analysis have been developed. In this process, an information model (i.e. EXPRESS schema) for tolerance analysis is developed by enriching STEP product model with GD & T information, and the assembly constraint definitions available in STEP have been modified. The necessary templates have also been developed by grouping the information into small, repeatedly used patterns to facilitate populating product data.

v. How to map STEP-based product information model into OWL-based ontological environment has been discussed so that appropriate semantic reasoning could be facilitated. The mapping of product's GD & T information content from STEP to OWL-based models in Protégé using NIST's OntoSTEP plug-in has been shown. It is also shown how this ontology based model could be used for interpreting semantics of GD & T needed for tolerance analysis by developing SWRL rules (i.e., description logic rules).

7.2 Future Work

The proposed work can be improved further by extending the work as listed below:

i. It is evident from this study that AP203 2nd Ed. or AP 214 3rd Ed. cannot fully represent the required information for IPC-1751 and IPC-1752 standards. Some additional mechanisms might have to be developed by the PLM system implementer to represent the required information.

ii. In order to have full control over the product material information and take a proactive action for the evolving regulations, the companies should have access to the following information:

- Materials usage,
- Materials deterioration history
- Materials handling constraints
- Materials usage constraints
- Materials maintenance requirements
- Materials disposal requirements
- Safety requirements for material handling
- Material reuse information
- Materials recovery suggestions
- Materials recycling directives

Unfortunately, this information is not available either in IPC-1752 or STEP standards. The materials information representation can be improved by adding the aforementioned information.

iii. The potentials of the developed FCCs can be shown thoroughly by developing web services for templates of these FCCs in service oriented architecture that can be used to provide data exchange mechanisms in the form of interoperable services available on the internet.

iv. The proposed methodology for semantic interpretation can be followed in a similar way to reason about inspection activities that are needed for product inspection purposes. For other product life-cycle phases, we may need different interpretation of the specified GD & T and we can still apply the same methodology as required. The wider application of this method is mainly restricted by the inherent limitations of Protégé, SWRL, and reasoning capabilities of their inference mechanisms.

v. How designers can use the enriched STEP product models to develop sustainable products would provide much more competitive advantage to the companies.

vi. It should be also further investigated how restricted materials information can be used to develop strategies for end-of-life activities (i.e., recycling, reusing, remanufacturing, etc.). For example, identifying the restricted materials present in products, their amount and concentration is very useful to make a decision on reusing, remanufacturing, recycling, etc.

vii. Finally, companies would benefit a lot if the effect of GD & T specifications and product performance on the end-of-life activities could be investigated.

References

Al-Ashaab, A., Rodriguez, K., Molina, A., Cardenas, M., Aca, J., Saeed, M., and Abdalla, H., 2003, Internet-based collaborative design for an injection-moulding system, Concurrent Engineering Research and Applications, 11(4):289–299.

ASME, 1994, ASME Dimensioning and Tolerancing, ASME Y14.5M-1994 Standard, The American Society of Mechanical Engineers, New York.

Barbau, R., Krima, S., Rachuri, S., Narayanan, A., Fiorentini, X., Foufou, S., Sriram, R.D., 2012, OntoSTEP: Enriching product model data using ontologies, Computer Aided Design 2012, 44(6):575-590.

Barreiro, J., Labarga, J.E., Vizan, A., Rios, J., 2003, Information model for the integration of inspection activity in a concurrent engineering framework, International Journal of Machine Tools & Manufacture, 43(8):797–809.

Baysal, M.M., Roy, U., Sudarsan, R., Sriram, R.D. and Lyons, K.W., 2004, The open assembly model for the exchange of assembly and tolerance information: overview and example, In: Proceedings of the ASME Design Engineering Technical Conference, Salt Lake City, Utah.

Chin, K.-S., Zhao, Y. and Mok, C. K., 2002, STEP-Based Multiview Integrated Product Modelling for Concurrent Engineering. International Journal of Advanced Manufacturing Technology, 20(12):896–906.

Cogorno, G., 2006, Geometric Dimensioning and Tolerancing for Mechanical Design: A Self-Teaching Guide to ANSI Y 14.5M1982 and ASME Y 14.5M1994 Standards, McGraw-Hill Professional.

Drake, P. J., 1999, Dimensioning and Tolerancing Handbook, McGraw-Hill.

DEX, 2012, Data Exchange Specifications technology overview, http://www.plcs-resources.org/plcs/dexlib/dex_index.htm.

Enovia – SolidWorks, 2012, http://www.3ds.com/products/enovia.

Feeney, A. B., 2002, The STEP Modular Architecture, Journal for Computing and Information Science in Engineering, 2(2):132-135.

Fenves, S.J., 2001, A core product model for representing design information, National Institute of Standards and Technology, NISTIR 6736, Gaithersburg, MD 20899.

Fiorentini, X., Gambino, I., Liang, V., Rachuri, S., Mahesh, M. and Bock, C., 2007, An ontology for assembly representation, National Institute of Standards and Technology, NISTIR 7436, Gaithersburg, MD 20899.

Fischer, B. R., 2004, Mechanical Tolerance Stackup and Analysis, CRC Press.

Geng, H., 2004, Manufacturing Engineering Handbook, McGraw-Hill.

Gong, D.-C. and Wang, Y.-T., 2011, UML presentation of a conceptual green design control system to react to environmental requirements, International Journal of Advanced Manufacturing Technology, 52(5-8):463–476.

IEC 62264, 2002, Enterprise-Control System Integration: Part 1-Models and Terminology, Part 2-Model Object Attributes, International Electrotechnical Commission, ISO/IEC, Geneva.

IPC, 2012, Association Connecting Electronics Industries, http://www.ipc.org.

IPC-1751A, 2012, Generic Requirements for Declaration Process Management, http://www.ipc.org.

IPC-1752A, 2012, Materials Declaration Management. http://www.ipc.org.

IPC-1756, 2012, Manufacturing Process Data Management. http://www.ipc.org.

ISO 10303-1:1994, Industrial automation systems and integration - Product data representation and exchange - Part 1: Overview and fundamental principles, International Organization for Standardization (ISO), Geneva, Switzerland.

ISO 10303-11: 1994, Product data representation and exchange - Part 11: Description methods: The EXPRESS language reference, International Organization for Standardization (ISO), Geneva, Switzerland.

ISO 10303-21: 2002, Product data representation and exchange - Part 21: Implementation methods: Clear text encoding of the exchange structure, International Organization for Standardization (ISO), Geneva, Switzerland.

ISO 10303-28: 2003, Product data representation and exchange - Part 28: Implementation methods: XML representations of EXPRESS schemas and data, International Organization for Standardization (ISO), Geneva, Switzerland.

ISO 10303-41: 1994, Product data representation and exchange - Part 41: Integrated Generic Resources: Fundamentals of product description and support, International Organization for Standardization (ISO), Geneva, Switzerland.

ISO 10303-42: 1994, Product data representation and exchange - Part 44: Integrated Generic Resources: Geometric and topological representation, International Organization for Standardization (ISO), Geneva, Switzerland.

ISO 10303-43:2004, Product data representation and exchange - Part 43: Integrated generic resource: Representation structures, International Organization for Standardization (ISO), Geneva, Switzerland.

ISO 10303-44: 1994, Product data representation and exchange - Part 44: Integrated Generic Resources: Product structure configuration, International Organization for Standardization (ISO), Geneva, Switzerland.

ISO 10303-45: 1997, Product data representation and exchange - Part 45: Integrated Generic Resources: Materials, International Organization for Standardization (ISO), Geneva, Switzerland.

ISO 10303-47: 1994, Product data representation and exchange - Part 47: Integrated Generic Resources: Shape Variation Tolerances, International Organization for Standardization (ISO), Geneva, Switzerland.

ISO 10303-108: 2005, Product data representation and exchange - Part 109: Integrated application resource: Parameterization and constraints for explicit geometric product models, International Organization for Standardization (ISO), Geneva, Switzerland.

ISO 10303-109: 2004, Product data representation and exchange - Part 109: Integrated application resource: Kinematic and geometric constraints for assembly models, International Organization for Standardization (ISO), Geneva, Switzerland.

ISO 10303-203:1994. Industrial automation systems and integration - Product data representation and exchange - Part 203: Application protocol: Configuration controlled 3D designs of mechanical parts and assemblies, International Organization for Standardization (ISO), Geneva, Switzerland.

ISO 10303-203:2005, Industrial automation systems and integration - Product data representation and exchange - Part 203: Application protocol: Configuration controlled 3D designs of mechanical parts and assemblies (modular version), International Organization for Standardization (ISO), Geneva, Switzerland.

ISO 10303-209: 2001, Product Data Representation and exchange - AP 209: Composite and metallic structural analysis and related design, International Organization for Standardization (ISO), Geneva, Switzerland.

ISO 10303-214:2003, Industrial automation systems and integration - Product data representation and exchange - Part 214: Application protocol: Core data for automotive mechanical design processes, International Organization for Standardization (ISO), Geneva, Switzerland.

ISO 10303-214:2010, Industrial automation systems and integration - Product data representation and exchange - Part 214: Application protocol: Core data for automotive mechanical design processes, International Organization for Standardization (ISO), Geneva, Switzerland.

ISO 10303-224:2006, Industrial automation systems and integration - Product data representation and exchange - Part 224: Application protocol: Mechanical product definition for process planning using machining features, International Organization for Standardization (ISO), Geneva, Switzerland.

ISO 10303-239: 2005, Industrial automation systems and integration - Product data representation and exchange -AP 239: Application protocol: Product life-cycle support, International Organization for Standardization (ISO), Geneva, Switzerland.

ISO 10303-1022:2010, Industrial automation systems and integration - Product data representation and exchange - Part 1022: Application module: Part and version identification, International Organization for Standardization (ISO), Geneva, Switzerland.

ISO 10303-1023:2004, Industrial automation systems and integration - Product data representation and exchange - Part 1023: Application module: Part view definition, International Organization for Standardization (ISO), Geneva, Switzerland.

ISO TC 184/SC4 N1161, 2001, Guidelines for the Content of Application Modules.

JIG-101:2012, Material Composition Declaration for Electrotechnical Products, www.eia.org/jig, Joint Industry Guide.

Kemmerer, S., 1999, STEP: The Grand Experience, (Editor), NIST Special Publication 939, National Institute of Standards and Technology, Gaithersburg, MD 20899, USA.

Kim, K-Y., Manley, D.G. and Yang, H., 2006, Ontology-based assembly design and information sharing for collaborative product development, Computer Aided Design, 38(12):1233-1250.

Liang, J., Shah, J.J., D'Souza, R., Urban, S.D., Ayyaswamy, K., Harter, E. and Bluhm, T., 1999, Synthesis of consolidated data schema for engineering analysis from multiple STEP application protocols, Computer-Aided Design, 31(7):429–447.

Matsokis, A. and Kiritsis, D., 2011, Ontology applications in PLM, International Journal of Product Lifecycle Management, 5(1):84-97.

Ming, X.G., Mak, K.L., Yan, J.Q., 1998, A PDES/STEP-based Information model For Computer-Aided Process Planning, Robotics and Computer-Integrated Manufacturing, Volume 14(5-6):347-361.

Neumann, A., 2003, Geometric Tolerancing Stacks and Analysis Workbook, Technical Consultants Inc.

OASIS Product Life Cycle Support (PLCS) Technical Committee, 2008, http://www.oasis-open.org/.

OFFICIAL JOURNAL OF THE EUROPEAN UNION. Directive 2002/95/EC of the European Parliament and of the Council of 27 January 2003 on the restriction of the use of certain hazardous substances in electrical and electronic equipment. 2002.

OFFICIAL JOURNAL OF THE EUROPEAN UNION. Regulation (EC) No 1907/2006 of The European Parliament and of The Council of 18 December 2006 concerning the Registration, Evaluation, Authorisation and Restriction of Chemicals (REACH), establishing a European Chemicals Agency, amending Directive 1999/45/EC and repealing Council Regulation (EEC) No 793/93 and Commission Regulation (EC) No 1488/94 as well as Council Directive 76/769/EEC and Commission Directives 91/155/EEC, 93/67/EEC, 93/105/EC and 2000/21/EC. 2006.

OntoSTEP, 2012, Plug-in, http://www.nist.gov/el/msid/ontostep.cfm.

OWL, 2012, Web Ontology Language Overview, http://www.w3.org/TR/owlfeatures/.

Panetto, H., Dassisti, M. and Tursi, A., 2012, ONTO-PDM: Product-driven ONTOlogy for Product Data Management interoperability within manufacturing process environment, Advanced Engineering Informatics, 26(2):334-348.

Pellet, 2012, OWL 2 Reasoner for Java, http://clarkparsia.com/pellet/.

PLCS template, 2012, assigning_reference_data, http://www.plcs-resources.org/plcs/dexlib/data/templates/assigning_reference_data/sys/section.htm#description.

Pro/Engineer, 2012, www.ptc.com/.

PROMISE, 2012, PROMISE FP6 project, www.promise.no.

Sarigecili, M.I., Roy, U., Rachuri, S., Sriram, R.D., 2009, Development of Functionality-Based Conformance Classes for ISO 10303: Conformance Classes for Tolerancing Functionality, Computer-Aided Design & Applications, 6(2):167-179.

Sarigecili, M.I., Roy, U., Rachuri, S., 2012, Enriching STEP Product Model with GD & T Information for 1-D Tolerance Analysis, submitted to International Journal of Production Research.

SCRIBA, 2012, Free implementation tool, Software and Systems Division, National Institute of Standards and Technology (NIST), http://www.nist.gov/itl/ssd/ei/scriba.cfm.

Shaharoun, A.M., Ab Razak, J. Alam, M.R., 1998, A STEP-based geometrical representation as part of product data model of a plastics part, Journal of Materials Processing Technology, 76(1-3):115-119.

Sharma, R. and Gao, J.X., 2002, Implementation of STEP 224 in an automatic manufacturing planning system, In Proceedings of the Institute of Mechanical Engineers, Part B: Journal of Engineering Manufacture, 216(9):1277–1289.

SOAP, 2012, Simple Object Access Protocol, http://www.w3.org/TR/soap/.

SWRL, 2012, Semantic Web Rule Language. http://www.w3.org/Submission/SWRL/.

Tang, D., Zheng, L., Li, Z. and Chin, K.S., 2001, STEP-based product modelling for concurrent stamped part and die development, Computers in Industry, 46(1):75–94.

Teamcenter–Siemens, 2012, http://www.plm.automation.siemens.com/en_us/products/ teamcenter/.

Windchill – PTC, 2012, http://www.ptc.com/product/windchill/.

Xie, S.Q., Yang, W.Z., Tu, Y.L., 2008, Towards a generic product modelling framework, International Journal of Production Research, 46(8):2229-2254.

XUL, 2012, XML User Interface Language, https://developer.mozilla.org/en/XUL.

Yang, W. Z., Xie, S. Q., Ai, Q. S., Zhou, Z. D., 2008, Recent development on product modelling: a review, International Journal of Production Research, 46(21):6055–6085.

Zha, X.F. and Du, H., 2002, A PDES/STEP-based model and system for concurrent integrated design and assembly planning, Computer Aided Design, 34(14):1087–1110.

Zhang, Hong-Chao, 1997, Advanced Tolerancing Techniques, John Willey & Sons, Inc.

Zhou, C-H., Eynard, B., Roucoules, L. and Ducellier, G., 2008, RoHS Compliance Declaration Based on RCP and XML Database, In: Global Design to Gain a Competitive Edge An Holistic and Collaborative Design Approach based on Computational Tools, Springer-Verlag.

Zhou, Z. D., Ai, Q. S., Liu, Q., Yang, W. Z., Xie, S. Q., 2009(a), A STEP-compliant product data model for injection moulding products, International Journal of Production Research 47(16):4497–4520.

Zhou, C.H., Eynard, B. and Roucoules, L., 2009(b), Interoperability between PLM and RoHS compliance management based on XML and Smart Client, Journal of Computing and Information Science in Engineering, 9(3):1-6.

Vita

NAME OF AUTHOR:	Mehmet Ilteris Sarigecili
PLACE OF BIRTH:	Adana, Turkey
DATE OF BIRTH:	March 20, 1979

GRADUATE AND UNDERGRADUATE SCHOOLS ATTENDED:

Mechanical Engineering Department, Cukurova University, Adana, Turkey

Institute of Applied and Natural Sciences, Cukurova University, Adana, Turkey

DEGREES AWARDED:

Master of Science (Mechanical Engineering), 2004, Cukurova University

Bachelor of Science (Mechanical Engineering), 2001, Cukurova University

AWARDS AND HONORS:

Graduate Assistantships, Syracuse University 2006-2012

"HONOR DEGREE**,**" Board of Directors of Mechanical Engineering Department of Çukurova University (2001).

PROFESSIONAL EXPERIENCE:

Teaching Assistant, Syracuse University, 2009-2012

Graduate Assistant, Syracuse University, 2006-2009

Visiting Research Scientist, Lehigh University, 2005-2006

Lecturer, Cukurova University, 2003-2005