# Representation of GD&T annotations in the context of STEP AP210 package models

#### James Stori, SFM Technology, Inc.

#### Kevin Brady, NIST

Allison Barnard Feeney, NIST

#### Thomas Thurman, Rockwell Collins, Inc. (retired) JUNE 14 2013

The ability to represent geometric dimension and tolerance (GD&T) annotations is critical to the successful application of digital product models to manufacturing industry needs for communication and archival of product and manufacturing information (PMI). The ISO-10303 (STEP) standard provides product data models that reflect the technical content of the relevant ASME and ISO standards. Industry groups including the Long Term Data Archiving and Retrieval Consortium (LOTAR) recognize this need and provide active support to continued development of the STEP standard as the relevant ASME and ISO standards are updated. ISO 10303-210:2011 (AP 210) is the application protocol part of STEP that addresses electronic design, focusing on requirements, assembly, interconnects and packaging. The AP 210 developers understood the central role of component library data to the electronic design process and provided a robust standard product model in the 2011 edition of the standard to support that role. The AP 210 package model provides a mechanical view of a component that is not an abstract geometric model but is electronic domain specific to support integration in a cohesive component library model. The AP 210 package model does support complete stand-alone implementations and the files created may be incorporated into a more comprehensive library model at a later time. The context for an AP 210package model is communication from a component supplier to a customer for the purpose of successfully including the component in the customer’s design. In order to be consistent with that context, an AP 210package model does not contain sufficient information for a manufacturer to fabricate the package.

To support the ongoing development of the AP210 standard, and promote its adoption for the neutral representation of electronic component data, recent updates to the STEP standard related to the representation of GD&T annotations have been exercised in the context of AP 210package models. This document describes the detailed representation of many of the most common GD&T annotations applied to AP 210package models. The largest family of standards for electronic component packages are those published by JEDEC. The context of the JEDEC standards matches the context of the AP 210package model. The JEDEC standards are electronic domain specific and map directly to the AP 210package model. The JEDEC standards are freely and publicly available (www.jedec.org), and contain detailed dimensioned and toleranced specifications for physical package configurations. JEDEC Publication 95 is a series of documents containing specifications for many common physical package configurations. A series of 10 detailed AP 210package models spanning 5 different JEDEC package families have been created. These representative package models contain detailed three-dimensional representations, explicit representation of critical package features such as individual terminals, and GD&T annotations consistent with the JEDEC specifications. Specific examples from these representative models are used in this document to illustrate the mapping of the various concepts to STEP entities and instances in physical files. The implementation and examples span both single-solid and multiple-solid geometric representations of packages, and the explicit links between the geometry and GD&T elements. Although the immediate focus of this document is the representation of GD&T concepts in AP 210package model representations in an electronic design context, the representation of GD&T annotations and three-dimensional geometry is common to other STEP application protocols, and the implementations and examples of those elements are applicable in the broader context of the STEP standard.

The implementation described here is based on a committee draft of STEP Part 47: Integrated generic resource: Shape variation tolerances [ISO 10303-47:2011-04(E)] dated April 22, 2011, and the MIM long form schema of the AP 210standard as of July 27, 2011. In addition, the GD&T representations described below are intended to be consistent with Version 3.6 of the CAX-IF Recommended Practices for the Representation and Presentation of Product Manufacturing Information (PMIRP). This document is intended to augment the PMIRP document with a specific focus on the representation of GD&T in the context of AP 210package models, as well as to illustrate a variety of specific GD&T concepts through detailed representative examples. Representative examples from two of the generated physical files are used in the discussion below to illustrate a number of the most commonly occurring GD&T concepts in the representation of JEDEC package models. Examples are taken from both a JEDEC transistor package family, TO-236 and a diode package family DO-214. The JEDEC drawing containing the GD&T annotations as well as a rendering of the corresponding 3D package models is contained in Appendix Figures A1 and A2.

# Defining the Dimensioning Standard

Because the STEP standards support both ASME and ISO PMI standards it is necessary to include information in the STEP product data that explicitly identifies which PMI standard is applicable. This information is included in the JEDEC specification. Clause 4, Defining the Dimensioning Standard, of the PMIRP provides guidance on data population necessary to represent the standard to be referenced for correct interpretation of the data by a post-processor. Because this document is concerned with representation, Figure 1 in the PMIRP applies. Because the JEDEC standard references ASME, the green strings and red strings in Figure 1 in the PMIRP represent the acceptable values for the data to be populated.

## Package Model Representation

Figure 1 illustrates some of the most important top-level elements in the AP 210package model. Because many of the JEDEC specifications include variations and each AP 210package model is explicit to a specific variation, the controlling document, document version and variation must all be specified in the AP 210package model instance data. The contents of the field labeled ‘PACKAGE DESIGNATOR’ (e.g., R-PDSO-G) should be copied into the product.id attribute. The contents of the field labeled ‘ISSUE’ (e.g., ‘H’) should be copied into the product\_definition\_formation.id attribute. The contents of the field labeled ‘VARIATIONS’ (e.g., ‘AA’, ‘BB’) should be copied into the product\_definition.id attribute. In the case of the JEDEC R-PDSO-G document, there would be two packages, one with product\_definition.id of ‘AA’ and one with product\_definition.id of ‘BB’. This would lead to two main shape representations, one for each of the variations. The GD&T annotations would be duplicated as necessary. While it is possible to build an instance model that contains a master product definition and two variational product definitions, that is not the approach taken in this recommended practice. The approach taken in this recommended practice simplifies the process requirements for the post-processor, as each package is a complete definition.

Each of the AP 210models contains one or more package. For each package, a detailed description of its physical representation is included. A package is a subtype of product\_definition and of product\_definition\_shape and thus may be directly referenced by a shape\_aspect. The package is referenced by a package\_body and typically multiple package\_terminal. Each package may contain multiple shape\_representations. For the purpose of the GD&T annotations, we are focused on the three-dimensional shape representation whose purpose is identified as ‘design’ while other shape representations, some of which are mandatory, are out of scope of this document. That shape\_representation will be related to the package as shown in Figure 1, and may be identified through the combination of the associated description\_attribute and the related representation with name ‘predefined shape purpose’. There are two common representations of 3D geometric models supported in the examples provided. The 3D geometry of a package may be represented as either a single solid or as an assembly-based representation. In either case, the geometric representation of the individual terminals and/or body of the package must be associated with the corresponding package\_terminals and/or package\_body. In the preferred representation, each terminal and/or body contains its own three-dimensional representation (most commonly an advanced\_brep\_shape\_representation) and these shape representations are mapped into the shape representation of the package with a usage\_concept\_usage\_relationship. In this scenario, many similar terminals of a package are defined by a single package\_terminal\_template\_definition, which in turn, has an associated advanced\_brep\_shape\_representation. That advanced\_brep\_shape\_representation is one of the advanced\_brep\_shape\_representation mentioned previously as being mapped into the shape representation of the package. Other shared properties of the terminals may be associated with the same package\_terminal\_template\_definition in order to more easily maintain a consistent set of information. Similarly, the body of the package can have its own associated three-dimensional shape representation, which is mapped into the shape representation of the package with a usage\_concept\_usage\_relationship. The resulting 3d shape representation of the package is analogous to a mechanical assembly composed of the terminals and body. This representation is preferred as it is more geometrically efficient, and more directly supports processing of the package model by downstream applications that must manipulate, use, and/or analyze the shapes of the individual package features.

Figures 2 and 3 illustrate the representation of a package\_terminal and package\_body respectively in the ‘assembly-style’ package structure, in which each has a 3d shape represented by an advanced\_brep\_shape\_representation. Note that in Figure 2, multiple instances of package\_terminal will often share a single package\_terminal\_template\_definition and each terminal template definition will have its own associated shape\_representation. Multiple terminals sharing a single geometric model would use a single terminal template definition, and other common attributes such as lead form. For example, if a given package has 64 similar gull wing terminals, and a single thermal pad, there would be a total of 65 instances of package\_terminal and two terminal template definitions, each with its own shape representation. For each individual terminal, the usage\_concept\_usage\_relationship (a subtype of mapped\_item) positions the terminal within the shape representation of the package. The mapping\_target and mapping\_source attributes specify the geometric placement operation for the terminal.

The alternative common geometric representation of a package model would be based on a single geometric model, most commonly a single manifold\_solid\_brep within a single shape\_representation.

Figure 4 illustrates that scenario. In that scenario, the relationship between the individual package terminals and package body and their associated geometry in the shape representation of the package would typically be expressed with the mechanism of the item\_identified\_representation\_usage (or a relevant subtype). This would most commonly result in a binding of groups of faces in the shape representation of the package to each of the terminals and body. This representation presents challenges for applications that must leverage and/or identify common geometry across similar terminals. Nevertheless, it does simplify several aspects of the geometric representation, and would be commonly encountered in 3D models provided by component suppliers, such as connector vendors. Each of the common package model representations (single vs. multiple solid) requires a specific mechanism in STEP for associating faces of the geometric model with the applicable GD&T elements.

## Shape\_aspects used in the representation of GD&T annotations

The shape\_aspect entity is used to represent a wide variety of features in STEP product models. Both package\_terminal and package\_body are subtypes of shape\_aspect. Additionally, many of the entities used to represent specific GD&T concepts, including datum, datum\_feature, centre\_of\_symmetry, etc., are subtypes of shape\_aspect. Multiple shape\_aspects will typically be used in the representation of an individual dimension or tolerance. At the lowest level, it is desired to link the GD&T elements to one or more faces in the applicable underlying solid model or models. There are two specific mechanisms that may be used to relate a shape aspect to a face in the model.

A geometric\_item\_specific\_usage may be used to relate a single face of a shape\_representation with a shape\_aspect. A shape\_aspect may be referenced by no more than one instance of geometric\_item\_specific\_usage. The result is that the shape\_aspect instance is a formal proxy for the geometric\_representation\_item because the combination of the shape\_aspect entity, geometric\_item\_specific\_usage entity and the geometric\_representation\_item, i.e. advanced\_face, are unique. Figure 4 illustrates the instantiation of a shape\_aspect used to identify a single face in the context of a single solid package model. It is important to note that in the single-solid package model, a face used in a GD&T annotation will also typically be an element of either the package\_body or a package\_terminal. If it is desired to identify the individual terminal or terminals, for example, referenced by a GD&T element, the receiving system must traverse the relationships from the shape aspects composing the GD&T element through the associated face or faces and back to any additional package terminals referencing the common face(s). When multiple shape\_aspects are to be related to the same face, only one of the identification relationships can be a geometric\_item\_specific\_usage. The other identification relationships must be instances of the supertype item\_identified\_specific\_usage.

In the context of a multiple-solid (assembly-style) package representation, the geometric\_item\_specific\_usage is insufficient to identify individual faces of individual terminals in the model. This is due to the fact that many terminal instances will often be sharing a single shape\_representation, and therefore a common collection of faces. In this case, a specialized subtype of geometric\_item\_specific\_usage must be employed. The chain\_based\_geometric\_item\_specific\_usage enables the identification of a single representation\_item, such as an advanced\_face, as the leaf node of a chain of mapped representations. The entities and relationships needed to uniquely identify a single face in the context of a multiple-solid package model representation are illustrated in Figures 2 and 3. Figure 2 illustrates a shape\_aspect (#1020) calling out a single face (#670) on a single instance of a package\_terminal. The chain\_based\_geometric\_item\_specific\_usage in Figure 2 contains two nodes – the first element in the list of nodes references the shape\_representation of the package (#816) while the second element references the shape\_representation of the terminal (#784). The undirected\_link list contains a single element, the usage\_concept\_usage\_relationship (a subtype of mapped\_item) that places an instance of the terminal shape representation into the package shape representation for one specific package\_terminal. Because the chain\_based\_geometric\_item\_specific\_usage is a subtype of geometric\_item\_specific\_usage the uniqueness restriction on geometric\_item\_specific\_usage applies to chain\_based\_geometric\_item\_specific\_usage. Per the PMIRP, if a single geometric feature consists of multiple faces, it is appropriate to have multiple item\_identified\_representation\_usage entities collecting the individual faces that are related to the single shape aspect. This will typically be the case for the representation of the package terminal or body, and would also be applicable for certain GD&T features, such as a hole consisting of two cylindrical faces.

#### Composite\_shape\_aspect

Often, it is either necessary or convenient to create a shape\_aspect that is a composition of other existing shape\_aspects. A composite\_group\_shape\_aspect and a composite\_unit\_shape\_aspect are two important subtypes of composite\_shape\_aspect. A composite\_unit\_shape\_aspect is used to aggregate multiple shape\_aspects that are to be treated as a unit. The composite\_group\_shape\_aspect is used when it is desired to apply a property to each of the constituent elements individually. The PMIRP specifies two options for the name attribute of composite\_group\_shape\_aspect: “multiple elements” or “pattern of features”. In the context of this document, “multiple elements” should always apply. When representing other specialized GD&T subtypes of shape\_aspect, a complex with composite\_shape\_aspect can also be instantiated. Figures 5 and 6 below illustrate several of the possible scenarios involving composite\_shape\_aspect.

Figure 5, below details the representation of what would commonly be referred to as a ‘regular feature of size’ (two opposed parallel surfaces). The source feature is identified as “datum B” in Figure A.2 This feature is a composite\_unit\_shape\_aspect, composed of the two shape aspects, each of which represents one of the opposed parallel surfaces. A length measure (a dimensional\_size) is applied to the feature that is independent of the location of the feature. In this case, the same feature of size serves as a datum\_feature for this particular package. As a result, a complex composite\_unit\_shape\_aspect+dimensional\_size\_with\_datum\_feature+composite\_shape\_aspect+shape\_aspect+datum\_feature+dimensional\_size (#1646) has been instantiated. Also important to note in this example is the fact that each of the two opposed parallel surfaces is itself an aggregate of two co-planar faces that are disjoint because of the presence of a wrap-around surface mount terminal. Therefore, each of the two opposed parallel surfaces is represented as a shape\_aspect with a shape\_aspect.description =‘unified geometric item’. In this particular example (single-solid package), each ‘unified geometric item’ would be referenced by two item\_identified\_representation\_usage instances, one per advanced\_face. For clarity those additional instances are not illustrated in Figure 5.

Figure 6 illustrates the instantiation of a composite\_unit\_shape\_aspect. In this case, a composite\_unit\_shape\_aspect that is also a datum\_feature is composed of the bottom faces of each of the three terminals in the package. The datum\_feature is also the basis for derivation of a tangent (#1090). The tangent is a subtype of derived\_shape\_aspect and is related to the plane by a geometric\_item\_specific\_usage.

The complex instance composite\_unit\_shape\_aspect+datum\_feature+composite\_shape\_aspect+shape\_aspect is appropriate in this scenario, as the three faces taken together are to be treated as a unit and form a datum feature. As this model is represented with an assembly-based structure, each of the three single-face shape aspects (#1018, #1020, #1022) is linked to a single face in the shape representation of the terminal template definition through a chain\_based\_geometric\_item\_specific\_usage. With reference to the sample physical file provided (TO-236\_AA\_with\_gdt.stp), each of these three shape aspects is actually calling out an individual placement of the same face (#670) – through the usage\_concept\_usage\_relationship referenced by the chain\_based\_geometric\_item\_specific\_usage.

#### Centre\_of\_symmetry

In some cases, a representation Item may not exist for that location in the model, e.g. the centre of a hole, so in that case a derived\_shape\_aspect will need to be created to provide an anchor point. Center plane is a commonly occurring element in GD&T that is represented as a subtype of derived\_shape\_aspect. A center plane is represented in STEP as a centre\_of\_symmetry that is ‘derived from’ a symmetric\_shape\_aspect. The geometry of the derived plane will be represented as a plane in a constructive\_geometry\_representation associated with the derived\_shape\_aspect. In the case of the center plane, the symmetric\_shape\_aspect will most commonly be two parallel opposing surfaces. If each of these is represented as a shape\_aspect, the symmetric\_shape\_aspect will also be a composite\_unit\_shape\_aspect. Figure 7 illustrates the representation of a center plane for a single terminal of the package, and the associated symmetric\_shape\_aspect representing the feature of size corresponding to the width of the terminal. Note the presence of the associated plane in a constructive\_geometry\_representation. This constructive\_geometry\_representation is related to the shape\_representation of the package. Note that the complex instance composite\_unit\_shape\_aspect in this figure does not include dimensional\_size\_with\_datum\_feature as that would require population of a datum and there is not a datum associated with this terminal, nor will a datum be associated for most terminals.

#### Tangent

Tangent planes are commonly occurring constructed elements in GD&T representation that are also represented as a subtype of derived\_shape\_aspect. A tangent may be derived from either a simple shape\_aspect or a composite\_shape\_aspect. As in the centre\_of\_symmetry case, the geometry of the derived plane will be represented as a plane in a constructive\_geometry\_representation associated with the derived\_shape\_aspect.

## Datum and Datum\_system

A datum in STEP is established by a datum\_feature or a set of datum\_targets. The datums in JEDEC package models have all been established based on a datum feature. Typically in design life cycle, the geometric item related to a datum is not provided, as that geometric item is usually created by inspection related applications. A package model however is an exception in that there will typically be a datum that is coincident with the ‘seating plane’ of the component. In the JEDEC specifications, by convention, this is usually Datum C. Figure 6 illustrates the representation of Datum C in the TO-236 package. The datum #1100 is related to #1086 that is a complex of composite\_unit\_shape\_aspect and datum\_feature and that consists of the faces tangent to the seating plane (one from each terminal). The tangent #1090 is ‘derived from’ #1086. The tangent #1090 is a complex instance with a seating\_plane. The associated plane in the constructive\_geometry\_representation expresses the orientation of the tangent. In the JEDEC specifications, datums A and B are typically orthogonal planes passing through the center of the package. The corresponding datum feature usually consists of opposing parallel sides of the package body. These datum features are expressed as a complex of dimensional\_size\_with\_datum\_feature and composite\_unit\_shape\_aspect. With reference to Figure A2, the datum feature for datum B is represented as in Figure 5. The composite\_unit\_shape\_aspect is composed of the opposing parallel surfaces. The fact that this same shape aspect is a dimensional\_size\_with\_datum\_feature enables simultaneous representation of the value of dimension E1. The representation of a datum system will be discussed below in the context of geometric tolerance.

## Dimensional\_size

A dimensional\_size is used to assign a length measure to a ‘feature of size’. Many of the common critical JEDEC package dimensions, such as toe-to-toe spans, body length and width, terminal width, etc. are best represented as a dimensional\_size. A dimensional\_size may only relate to a single shape\_aspect. Therefore it may be necessary to construct a composite\_unit\_shape\_aspect in order to apply the dimensional\_size. A dimensional\_size is associated with a shape\_dimension\_representation that contains the specific value or values of the dimension. The id attribute dimensional\_size inherits from shape\_aspect is required to be unique within a single product\_definition\_shape. In the generated examples, all shape\_dimension\_representation contain one or more values, qualified by a type\_qualifier. The implementation supports type qualifier values of ‘minimum’, ‘average’, and ‘maximum’, although additional qualifications are possible. There is no requirement or expectation that all qualified values will be provided for a given dimension. Often, the JEDEC specifications will provide all three values. In certain instances, however, only a minimum (or maximum) value may be available, or both minimum and maximum values are provided, but there is no nominal (or average) value available. Figure 8 illustrates the representation of the dimensional\_size and corresponding shape\_dimension\_representation for the value of dimension E1 in the DO-214\_AB package. There are three qualified values for this dimension (minimum, average, and maximum), each represented by a complex length\_measure\_with\_unit+measure\_representation\_item+ qualified\_representation\_item. Note that because there are two variations and each variation has a complete package model, there would be two instances of the value of E1 in the exchange file, one for each variation, even though the actual values are identical across the variations.

## Dimensional\_location

A dimensional\_location is used to associate a length measure with two shape\_aspects as a dimensional measure. The fundamental difference between dimensional\_size and dimensional\_location is that the user perceives the shape\_aspect to which the dimensional\_size is applied to be a single feature, whereas the dimensional\_location is applied between two features or between a feature and a datum. In the DO-214\_AB example, there are several dimensions populated as a dimensional\_location, such as dimensions A, A2, and A3 (see Figure A2). In these three examples, the dimension is a length measure between datum feature C and one of the parallel surfaces on the body of the package. The dimensional\_location is a subtype of a shape\_aspect\_relationship. The id attribute dimensional\_location inherits from shape\_aspect\_relationship is required to be unique within the product\_definition\_shape referenced through the path beginning with the relating attribute inherited from shape\_aspect\_relationship and ending at the relevant product\_definition\_shape. Figure 9 illustrates a dimensional\_location and corresponding shape\_dimension\_representation for dimension A (the overall package height) in the DO-214\_AB package. The representation of the value of the dimension, through the use of one or more qualified\_representation\_item is identical to the case of the dimensional\_size.

## Geometric\_tolerance

STEP supports the representation of specific tolerance types through subtypes of geometric\_tolerance. The JEDEC package models contain examples of several of these tolerance representations, including position\_tolerance, surface\_profile\_tolerance, and parallelism\_tolerance. Many times, the specific tolerance type will be instantiated as a complex with geometric\_tolerance\_with\_datum\_reference and/or modified\_geometric\_tolerance, or may be instantiated with geometric\_tolerance\_with\_modifiers as described in the PMIRP. Figures 10 and 11 demonstrate the population of a position tolerance with both a reference to a datum system and a tolerance modifier (least material condition). This particular tolerance controls the position of the center plane of each of the terminals independently, with respect to the established datum system. For this reason, the tolerance is associated with a composite\_group\_shape\_aspect, that is an aggregate of two centre\_of\_symmetry, one for each of the two terminals in the package. Note that only one of the terminals (#1711) and only one plane (#1715) is illustrated in Figure 11. A variety of other representational examples of geometric tolerances are contained in the populated physical files.

Future work:

NIST should work with JEDEC to develop a business case for providing an AP 210package model repository for the JEDEC specifications. This computer-interpretable repository could serve as a neutral digital reference model repository that includes both the current drawings and the related product models.

5/18/2012 change

1-General for all figures:

1a-change ‘single geometric feature’ to ‘unified geometric item’ and it goes on shape\_aspect.description.

1b-remove ‘single element’ string as a shape\_aspect by default is a single element.

2-For figure one:

2a-Please add the following 3 instances to figure one:

Package <-

product\_definition\_context\_association.definition

product\_definition\_context\_association <<<<new instance

{product\_definition\_context\_association.role ->

product\_definition\_context\_role<<<new instance

product\_definition\_context\_role.name = 'part definition type'}

product\_definition\_context\_association.frame\_of\_reference ->

product\_definition\_context<<<<new instance

product\_definition\_context.name = 'physical design usage'}

2b-Delete the product category with name of ‘package’.

**Figure 1. Top-level elements in the package model and the three-dimensional shape\_representation.**



**Figure 2. A terminal, associated template, and single face shape\_aspect in an ‘assembly’ style representation.**



**Figure 3. The package body and an associated single face shape\_aspect in an ‘assembly’ style representation.**



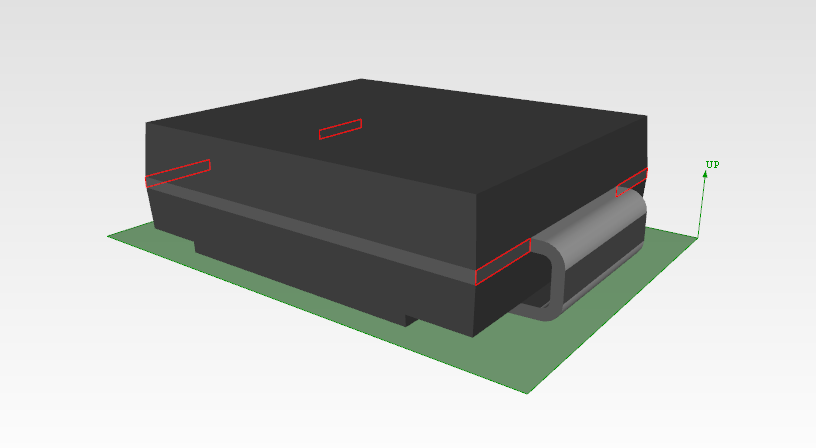
Figure 5 corrections: composite\_unit\_shape\_aspect+dimensional\_size\_with\_datum\_feature+composite\_shape\_aspect+shape\_aspect +symmetric\_shape\_aspect+datum\_feature+dimensional\_size(#1646);‘dimension E1 and datum feature B’ should be on shape\_aspect.id. ‘feature of size’ should be on description.

**Figure 4. Associations between a single face and a shape aspect of the package in a single-solid representation. A single face may be related to a terminal or the package body as well as a shape\_aspect contributing to a GD&T annotation.**



Figure 6 corrections: 5/18/2012

**Figure 5. Datum B and Dimension E1 – DO-214\_AB (single solid).**



#1594=SHAPE\_ASPECT('single geometric feature','gdt body length f2',#3,.T.);

#1628=SHAPE\_ASPECT('single geometric feature','gdt body length f1',#3,.T.);

#1646=(COMPOSITE\_SHAPE\_ASPECT()DATUM\_FEATURE()DIMENSIONAL\_SIZE(#1646,'length')DIMENSIONAL\_SIZE\_WITH\_DATUM\_FEATURE()SHAPE\_ASPECT('feature of size','dimension E1 and datum feature B',#3,.T.));



#1086 is a composite\_unit\_shape\_aspect+composite\_shape\_aspect+shape\_aspect+datum\_feature.

#1090 is a tangent derived from #1086

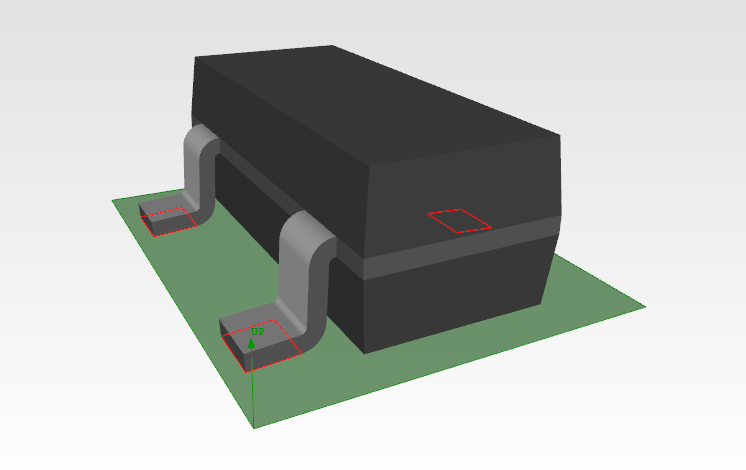
The tangent #1090 is a complex instance with a seating\_plane. <<<<New 5/18/2012>>>>

#1100 is a datum related to #1086.

(#1090 and #1100 are not directly related)

String names on #1086, 1018, 1020, 1022 probably should be removed, as they are not called out in mapping or PMI rec prac and may cause problems with complex instances created in other use cases.

**Figure 6. Datum C – TO-236\_AA (TO-236\_AA\_with\_gdt.stp).**



#1020

#1018

#1022=SHAPE\_ASPECT('single element','gdt terminal bottom pin 3',#3,.T.);

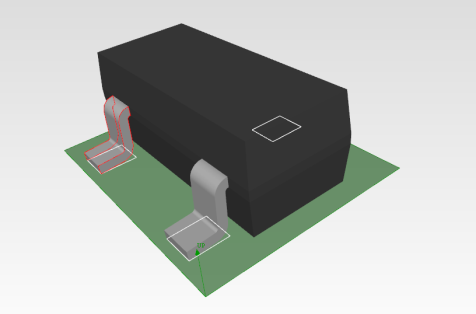
#1086=COMPOSITE\_UNIT\_SHAPE\_ASPECT('multiple elements','bottom all pins',#3,.T.);

Figure 7 corrections:5/18/2012

#1131 should be composite\_unit\_shape\_aspect+composite\_shape\_aspect+shape\_aspect+symmetric\_shape\_aspect.

The ‘pin 1…’ should be on the id attribute.

‘feature of size’ should be on description.



**Figure 7. Representation of a center plane – TO-236\_AA (TO-236\_AA\_with\_gdt.stp).**

#1032

#1040

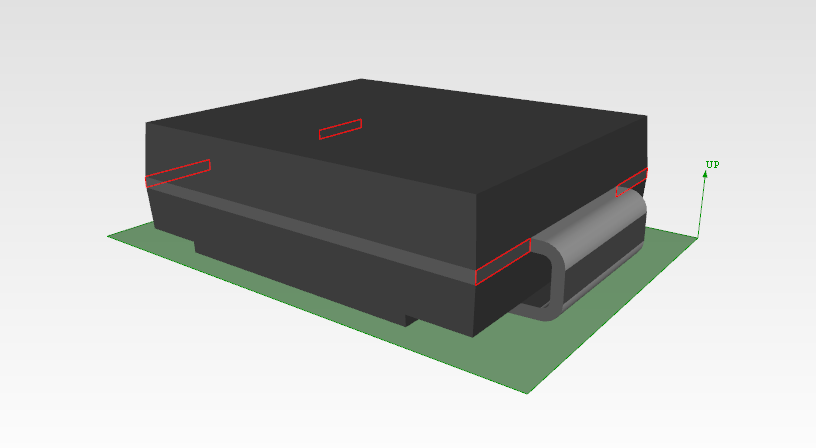
#1134=CENTRE\_OF\_SYMMETRY('centre plane','pin 1 centre\_of\_symmetry',#3,.F.);

#1131=(COMPOSITE\_SHAPE\_ASPECT()SHAPE\_ASPECT('feature of size','pin 1 width feature of size',#3,.T.) SYMMETRIC\_SHAPE\_ASPECT());



Figure 9 corrections:

**Figure 8. Dimension E1 – DO-214\_AB (single solid).**



#1594=SHAPE\_ASPECT('single geometric feature','gdt body length f2',#3,.T.);

#1628=SHAPE\_ASPECT('single geometric feature','gdt body length f1',#3,.T.);



#1646=(COMPOSITE\_SHAPE\_ASPECT()DATUM\_FEATURE()DIMENSIONAL\_SIZE(#1646,'length')DIMENSIONAL\_SIZE\_WITH\_DATUM\_FEATURE()SHAPE\_ASPECT('feature of size','dimension E1 and datum feature B',#3,.T.));

#1662 is not a tangent (doesn’t need to be.).#1771; use id = ‘A’

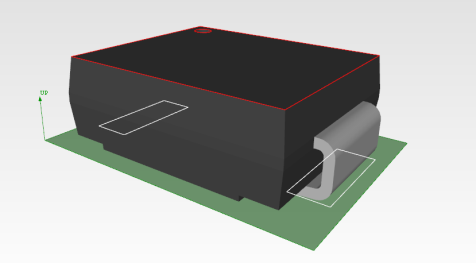
**#1626-remove ‘single element’**

**Figure 9. Dimension A – DO-214\_AB (single solid).**

#1662=(DATUM\_FEATURE()DERIVED\_SHAPE\_ASPECT()SHAPE\_ASPECT('tangent plane','datum feature C',#3,.T.)TANGENT());

#1626=SHAPE\_ASPECT('single element','gdt body top',#3,.T.);

#1769=DIMENSIONAL\_LOCATION('linear distance',$,#1626,#1662);B',#3,.T.));



**Figure 10. Position\_tolerance and datum\_system – DO-214\_AB (single solid).**



Figure 11 corrections:

Only a shape\_aspect\_relationship.relating\_shape\_aspect shall reference a composite\_shape\_aspect. In Figure 11 the sars between #1722, #1703, #1714 reverse ‘ing’, ‘ed’.

#1725 shows an attribute “value\_component” that specifies “.LEAST\_MATERIAL\_CONDITION” (missing trailing full stop); The attribute should be “modifier” and should have a value “.LEAST\_MATERIAL\_CONDITION.”

#1711 should be a composite\_unit\_shape\_aspect because it is representing a single thing, a terminal.

**Figure 11. Terminal center plane position\_tolerance – DO-214\_AB (single solid).**



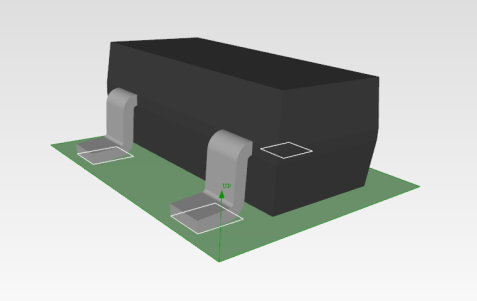
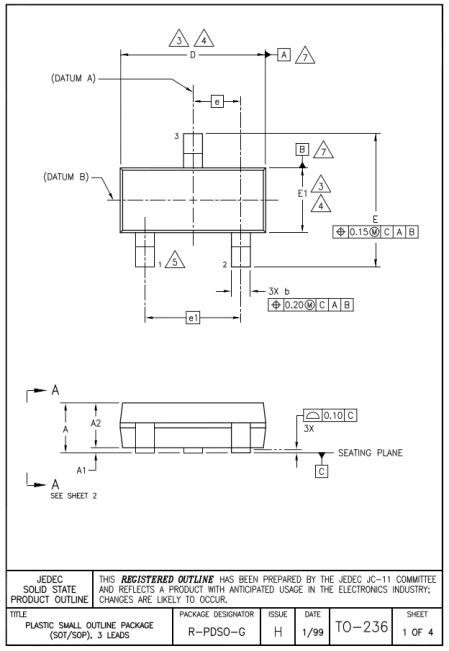
#### APPENDIX A– Generated Physical Files

AP 210package models have been generated and GD&T annotations have been added for instances from 5 separate package families. The examples were chosen to illustrate a broad range of package styles. Four of the five families are based on the JEDEC Publication 95 mechanical outlines. The fifth is a representative example of a common ‘chip’ style package not addressed by JEDEC. For each of the five package models, two separate representations were instantiated –both a single solid representation and a multiple-solid representation.

The table below summarizes the families and instances represented:

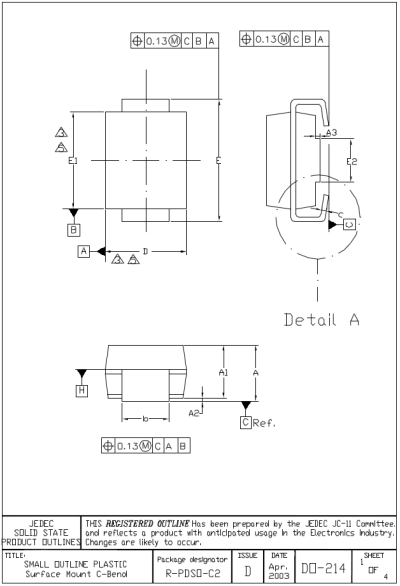
|  |  |  |  |
| --- | --- | --- | --- |
| Package | Category / Leadform | Image | AP 210models |
| DO-214\_AB | DO /  Ribbon L - Inward |  | DO-214\_AB\_with\_gdt.stp  DO-214\_AB\_with\_gdt\_single\_solid.stp |
| TO-236\_AA | TO /  Gull-Wing |  | TO-236\_AA\_with\_gdt.stp  TO-236\_AA\_with\_gdt\_single\_solid.stp |
| MO-211\_AA | BGA /  Ball |  | MO-211\_AA\_with\_gdt.stp  MO-211\_AA\_with\_gdt\_single\_solid.stp |
| MO-220\_VEEB | QFN /  Flat No-Lead Edge |  | MO-220\_VEEB\_with\_gdt.stp  MO-220\_VEEB\_with\_gdt\_single\_solid.stp |
| 0402 CHIP | RESC / Rectangular End Cap |  | KOA\_0402\_1E\_RK73B\_with\_gdt.stp  KOA\_0402\_1E\_RK73B\_with\_gdt\_single\_solid.stp |

Figure A1. JEDEC TO-236 Issue H.

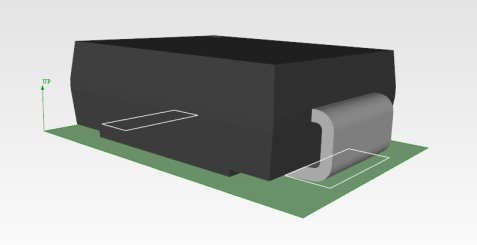


**TO-236\_AA\_with\_gdt.stp**

Figure A2. JEDEC DO-214 Issue D.



**DO-214\_AB\_with\_gdt\_single\_solid.stp**



## APPENDIX B – PackageGdtReporter

To demonstrate the extraction of GD&T annotations from the provided physical files, a sample Java utility is provided that uses the JSDAI query routines to traverse an AP 210package model and report the GD&T elements in a text-based format.

The following is the output obtained by executing the PackageGdtReporter on the sample model ‘TO-236\_AA\_with\_gdt.stp’. As may be seen in the output, the model contains three datums, numerous dimensions (dimensional\_size and dimensional\_location, as well as both position and surface profile tolerances. For all GD&T elements, the associated faces from the solid models are reported, as well as their role in the body and/or terminals of the package.

[Following not yet reviewed due to request for changes in text and instance figures. Review does need to be done.]

Importing.... C:\Users\James Stori\Workspace\EclipseWorkspace\JSDAIexpress\P21 Files\TO-236\_AA\_with\_gdt.stp

--- Exchange structure: C:\Users\James Stori\Workspace\EclipseWorkspace\JSDAIexpress\P21 Files\TO-236\_AA\_with\_gdt.stp

--- Imported to the repository: TO\_236\_AA\_annotation\_pin\_all\_stp

--- Reading time=0sec

Datum: 'A'

FEATUREOFSIZE

Face: #146 of body

Face: #228 of body

Datum: 'B'

FEATUREOFSIZE

Face: #115 of body

Face: #353 of body

Datum: 'C'

TANGENTPLANE

Face: #670 of terminal 1

Face: #670 of terminal 2

Face: #670 of terminal 3

Dimensional size: 'E' : [ 2.64 MM (MAXIMUM) ][ 2.1 MM (MINIMUM) ]

FEATUREOFSIZE

Face: #646 of terminal 1

Face: #646 of terminal 2

Face: #646 of terminal 3

Dimensional size: 'A2' : [ 0.85 MM (NOMINAL) ][ 1.02 MM (MAXIMUM) ][ 0.7 MM (MINIMUM) ]

FEATUREOFSIZE

Face: #290 of body

Face: #53 of body

Composite Dimensional size: 'b' : [ 0.3 MM (MINIMUM) ][ 0.5 MM (MAXIMUM) ]

FEATUREOFSIZE

Face: #469 of terminal 1

Face: #781 of terminal 1

FEATUREOFSIZE

Face: #469 of terminal 2

Face: #781 of terminal 2

FEATUREOFSIZE

Face: #469 of terminal 3

Face: #781 of terminal 3

Dimensional size: 'D' : [ 3.04 MM (MAXIMUM) ][ 2.9 MM (NOMINAL) ][ 2.8 MM (MINIMUM) ]

FEATUREOFSIZE

Face: #146 of body

Face: #228 of body

Dimensional size: 'E1' : [ 1.3 MM (NOMINAL) ][ 1.2 MM (MINIMUM) ][ 1.4 MM (MAXIMUM) ]

FEATUREOFSIZE

Face: #115 of body

Face: #353 of body

Dimensional location: 'e' (BASIC) : [ 0.95 MM ]

Feature 1:

CENTERPLANE

Face: #469 of terminal 3

Face: #781 of terminal 3

Feature 2:

CENTERPLANE

Face: #469 of terminal 2

Face: #781 of terminal 2

Dimensional location: 'e1' (BASIC) : [ 1.9 MM ]

Feature 1:

CENTERPLANE

Face: #469 of terminal 1

Face: #781 of terminal 1

Feature 2:

CENTERPLANE

Face: #469 of terminal 2

Face: #781 of terminal 2

Dimensional location: 'A' : [ 1.17 MM (MAXIMUM) ][ 0.7 MM (MINIMUM) ]

Feature 1:

GENERIC

Face: #290 of body

Feature 2:

TANGENTPLANE

Face: #670 of terminal 1

Face: #670 of terminal 2

Face: #670 of terminal 3

Dimensional location: 'A1' : [ 0.15 MM (MAXIMUM) ][ 0.05 MM (MINIMUM) ]

Feature 1:

GENERIC

Face: #53 of body

Feature 2:

TANGENTPLANE

Face: #670 of terminal 1

Face: #670 of terminal 2

Face: #670 of terminal 3

Tolerance (POSITION): 'dim E center position' 0.15 MM

Datum System [C(MMR) A B ]

CENTERPLANE

Face: #646 of terminal 1

Face: #646 of terminal 2

Face: #646 of terminal 3

Tolerance (SURFACEPROFILE): 'terminal bottom surface profile' 0.1 MM

Datum System [C ]

TANGENTPLANE

Face: #670 of terminal 1

Face: #670 of terminal 2

Face: #670 of terminal 3

Composite Tolerance (POSITION): 'terminal centre plane position' 0.2 MM

CENTERPLANE

Face: #469 of terminal 1

Face: #781 of terminal 1

CENTERPLANE

Face: #469 of terminal 2

Face: #781 of terminal 2

CENTERPLANE

Face: #469 of terminal 3

Face: #781 of terminal 3

Done