Recommended Practices for
AP203
Preliminary
April 23, 1998

<u>Contacts:</u>

Bill Anderson
ATI
5300 International Blvd
N. Charleston, SC 29418
843-760-3346
anderson@aticorp.org

Table of Contents

List of Figures

This page intentionally left blank

1 Introduction

STEP Application Protocol (AP) 203 (Configuration Controlled 3D Designs of Mechanical Parts and Assemblies) provides the data structures for the exchange of part and configuration identification with configuration control data either with or without associated 3D part model information. This international standard was developed under the auspices of the International Organization for Standardization (ISO) and is one of a series of parts comprising the full Standard for the Exchange of Product model data (STEP) standard known as ISO 10303.

This document has been prepared as a usage guide for industry and also to address some inconsistencies between the current state of the practice and the AP 203 data scope. This document assumes that the reader has at least a rudimentary knowledge of both AP 203 and its associated application domain. The figures in this document are intended to provide a navigational view of portions of the AP with boxes representing entities, lines being relationships, and arrow heads indicating the pointer direction.

As previously stated, AP 203 is but one part of the entire ISO 10303 product data standard. It was developed to represent one domain which is described in its scope section. AP 203 does not present itself as the data standard for configuration management of a product throughout its entire life cycle. The AP is centered on the design phase of mechanical parts. As STEP evolves, other APs (currently under development or proposed) will carry the data in AP 203 forward through the product life cycle.

This document will provide pre- and post-processor recommendations where attributes from the conceptual STEP data models may not actually have values in the AP 203 application domain. The terms pre-processor and post-processor refer to the applications which write and read the AP 203 data respectively. In these recommendations, the term "no standard mapping" means there is no mapping defined in the AP's ARM to AIM mapping table for the data.

Section 3 on page 47 of this document describes a recommended standard practice for AP 203 which actually extends beyond the scope of the AP itself. This special section describes how AP 203 can be used to configuration manage drawings with associated lists and relate them to parts. STEP will be dealing with this area through many APs (201, 202, 214, etc.) and AP 232, Technical Data Packaging, Core Information and Exchange.

The configuration management of drawings with associated lists and their relation to parts and specifications is an area widely utilized and addressed in industry today, but is currently not addressed by STEP. The part based configuration management in AP 203 is clearly where industry intends to head in the future, but some migration strategy is needed for legacy systems and their associated data. The practice in section 3 is put forth as a migration strategy which will be removed from this document once this void is adequately addressed in STEP.

2 Using AP 203

This section describes how AP 203 is intended to be used. The sole purpose of this section is to present the flow of the everyday engineering actions put forth in the AP 203 activity model using the application terms of the reference model tied to the constructs in the interpreted model. This section will establish limits on some of the data constructs which are not constrained in the Application Interpreted Model (AIM) of the AP.

2.1 General

In using AP 203, there are some constructs which have global applicability across all data in the exchange. These constructs relate to the file header for physical file exchanges, data definitions within the file related to the AP itself, and fundamental constructs which contain the information for people, organizations, dates, times, approvals, security classifications, and units of measure.

2.1.1 The STEP Physical File

At present, the standard way to externalize AP 203 data is via a physical file which is an ASCII encoding of the data based on the EXPRESS constructs in AP 203 as mapped through ISO 10303-21. This document will not deal with this in any detail. The user is referred to ISO parts 10303-11 for definitions of EXPRESS constructs and 10303-21 for information on how to map the EXPRESS constructs in AP 203 to the physical file.   (See G.1.10 and G.1.11 for implementors agreements that affect physical files.)

Post-processor Recommendations: When reading a physical file, post-processors should note all errors found during the reading of the file. It is recommended that post-processors provide options to the user on whether to continue when an error is encountered.   There are no recommendations on what a post-processor should do with erroneous data. This is left to the discretion of the implementor. If the implementor elects to correct erroneous data, the post-processor should inform the user (as above) of the bad data and what correction was made.

2.1.2 AP Identification and Contexts

STEP is an intelligent data standard and as such the representation of the data for an AP identifies the AP data structure through computer sensible data.   This is done through the entities **application_context** and **application_protocol_definition**. These entities as related sub-types are described pictorially in **Figure 1** on page 11.

The **application_context** entity identifies the application which defined the data.   The **application** attribute, based on its definition in ISO 10303-41, should have the value "configuration controlled 3D designs of mechanical parts and assemblies" as this is the application domain AP 203 is meant to cover.

The **application_protocol_definition** entity further identifies the AP. For AP 203, the **status** attribute, based on its definition in ISO 10303-41, should have the value "international standard" for AP 203. The **application_interpreted_model_schema_name** attribute should have the value "config_control_design" based on an AP 203 restriction. The **application_protocol_year** attribute should have the value "1994", based on its definition in ISO 10303-41.

The application identified by the **application_context** entity is broken down into elements in STEP. In AP 203, these elements are contexts. For AP 203, the valid context entities are **mechanical_context**, **product_definition_context**, **design_context**, and **product_concept_context**.

The **mechanical_context** entity is a sub-type of the **product_context** entity which identifies from what engineering discipline's point of view the data is being presented. For AP 203 which uses the **mechanical_context**, the value for the **discipline_type** is restricted to be "mechanical". This entity will establish the viewing perspective and therefore the requirements source for **product** entities which are defined in section 2.8.1.1 on page 12. It should be noted that this does <u>not</u> mean that 203 will only support purely mechanical parts. It actually means that any parts/ **product**s defined under AP 203 should have mechanical properties. It further means that these parts/ **product**s should be capable of being managed through the same configuration management processes that are used for mechanical parts.

The **product_definition_context** entity and its sub-type **design_context** identifies the life cycle stage or maturity of the data being presented. When **design_context** is used, the value of the **life_cycle_stage** attribute is restricted to be "design". The **design_context** entity shall be used for any **product**s which have shape related to them. This restriction is enforced through a rule in AP 203. The **product_definition_context** entities will establish the viewing perspective and therefore the requirements source for **product_definition** entities which are defined in section 2.8.1.3 on page 14.

The **product_concept_context** entity identifies what market segment or customers provided requirements for the data. This entity will establish the source of the requirements for **product_concept** entities which are defined in section 2.8.14 on page 42.

Pre-processor Recommendations: There is no standard mapping for the **name** attribute for **mechanical_context**, **product_definition_context**, **design_context**, and **product_concept_context**. Since there is no standard mapping in the AP 203 application domain for this attribute, it is recommended that this attribute contain a null string as minimal content or any appropriate/ mutually agreed upon string.

Post-processor Recommendations: Since there is no standard mapping for the **name** attribute for **mechanical_context**, **product_definition_context**, **design_context**, and **product_concept_context**, it is recommended that post-processors not assign any processing significance to this value.

2.2 People and Organizations

AP 203 represents people and organizations as they perform functions related to other data and data relationships. A **person** in AP 203 must exist in the context of some **organization**. A **person** in an **organization** is then associated to the data or data relationship in some role indicating the function being performed.

Both people and organizations <u>may</u> have addresses associated with them. This is entirely optional in AP 203 and is done through the **address** entity being related to the **person** (through **personal_address**) or **organization** (through **organizational_address**).

2.2.1 People

AP 203 specifies information about people through the **person** entity. A **person** is identified by an **id** with other data representing their name and optionally titles which may apply to them. In populating the data, the **id** must be unique. This is typically not a problem when the person is taken in the context of some specific group like a company or even country. In these instances, there are typically identifying numbers assigned to people. If the data being assembled is for worldwide consumption, the **id** must be unique in that domain.

Pre-processor Recommendations: All pre-processors should provide values for at least the **last_name** and **first_name** attributes for the **person** entity in order to provide a sense of meaning to the **id** attribute.

In cases where uniqueness of the **id** attribute may be a problem, pre-processors should prefix the **id** attribute with the **organization id** (as described in the following section) followed by a comma. For example, if the **organization id** value were "USA,93699" and the **person id** were "111111", the actual value of the **person id** would be "USA,93699,111111".

2.2.2 Organizations

AP 203 represents groups of people (e.g. companies, countries, etc.) through the **organization** entity. The identification or **id** data is optional. This information can be highly important in providing unique identification to the organization or company. It is recommended that this field always be populated with unique data. The **name** attribute must contain a short identifier or acronym for the **organization**. The **description** attribute may contain the full name of the organization or a textual explanation its reason for existence.

Pre-processor Recommendations: All pre-processors should provide a unique **organization id** to eliminate ambiguities where organizations may have the same names. If the intended domain for the data is large, the reader is referred to ISO/IEC 8824-1 which can provide some guidance on creating unique identifiers. A unique string obtained under ISO/IEC 8824-1 can be used as or

prefixed to the organization identifier. For example, if the organization typically used an identifier of "93699" and the unique string were "USA", the actual value of the **organization id** would be "USA,93699".

Post-processor Recommendations: All post-processors should make use of any provided information in the **id** attribute to eliminate ambiguities where **organization**s may have the same name.

2.2.3 Roles

The connection of people to organizations is accomplished through the **person_and_organization** entity. It is used to identify approvers for different aspects of the product data. It is also related to certain constructs to identify the people and organizations responsible for them and how they are responsible. This is done through the **cc_design_person_and_organization_assignment** entity which relates a person and organization in some role to an entity. The role is established in the **person_and_organization_role** entity **name** attribute. The data allowed in this attribute is constrained by the **restrict_person_and_organization_role** rule in AP 203. The sections which describe the use of the entity to which the **person_and_organization** is assigned will identify the allowed values for the **name** attribute of the **person_and_organization_role** entity. For convenience, these can also be found under the index entry in this document for **person_and_organization_role**.

2.3 Dates and Times

AP 203 represents dates and times to record when something occurred. In industry today, this is normally done with just a date. AP 203 requires both a date and a time for all events.

2.3.1 Dates

AP 203 provides three different ways to represent a **date**. All of these are documented in ISO 10303-41. This is problematic as it requires multiple conversions of the data depending on the date type received and the date type used by the organization.

Pre-processor Recommendations: It is recommended that pre-processors use **calendar_date** for date data.

Post-processor Recommendations: Post-processors must be able to process <u>all</u> forms of **date** in AP 203.

2.3.2 Time

AP 203 represents time through the entity **local_time**. As mentioned earlier, the requirement that

time be provided for every date may involve the invention or defaulting of data.

The **local_time** entity references a time zone identification through the **zone** attribute. The referred to **coordinated_universal_time_offset** entity identifies the delta from the current time zone to coordinated universal time. For AP 203's application domain, this should be considered the delta in hours and minutes between Greenwich Mean Time (GMT) and the local time zone.

> NOTE - Coordinated Universal Time is NOT exactly Greenwich Mean Time (GMT). The hour and minute offset is the same, but the second offset varies due to seasonal variations in the earth's axis orientation. The difference between GMT and coordinated universal time is on the order of .05 seconds which has essentially no effect in a configuration management (AP 203) application.

Pre-processor Recommendations: All pre-processors should use noon in the originating time zone as a default for **local_time** when this data is unavailable. All pre-processors should view Greenwich Mean Time and coordinated universal time as equal.

2.3.3 Roles

The connection of dates to times is accomplished through the **date_and_time** entity. It is used to identify when approval occurred for different aspects of the product data. It is also related to certain constructs to identify the date and time something started, stopped or occurred and what started, stopped or occurred. This is done through the **cc_design_date_and_time_assignment** entity which relates a date and time in some role to some construct. The role is established in the **date_time_role** entity **name** attribute. The data allowed in this attribute is constrained by the **restrict_date_time_role** rule in the data model. The sections which describe the use of the entity to which the **date_and_time** is assigned will identify the allowed values for the **name** attribute of the **date_time_role** entity. For convenience, these can also be found under the index entry in this document for **date_time_role**.

2.4 Approvals

There are many constructs in AP 203 which require approvals. Approving in AP 203 is accomplished by establishing an **approval** entity and relating it to some construct through a **cc_design_approval** entity. There are rules related to the use of the **approval** entity which require it to have an associated **approval_person_organization** and **approval_date_time**. This is sensible as an approval is normally given by someone at a certain time. Help on the creation of these entities is given in sections 2.2 on page 4 and 2.3 on page 5.

Every construct which requires an **approval** is allowed only one **approval**. This might lead to the misconception that only one person on one date/ time can approve something. This is not the case. The approval constructs in AP 203 actually designate that an approval cycle is required. This cycle may only need one signature. This explains the need for the **approval_status** entity

and the fact that it allows for a status of "not_yet_approved".

The above helps explain the requirement for an approval and the need for the **approval_status**. The **approval_date_time** records the date/ time the status was changed. It does <u>not</u> record (<u>necessarily</u>) when the approval was given by the **approval_person_organization** as there can be multiple **approval_person_organization**s related to an **approval** entity. If there is only one **approval_person_organization** and the **approval_status** is "approved", the **approval_date_time** indicates that this person/organization approved it on this date/time. When an approval event is a cycle which requires multiple people to concur on possibly differing dates/times, the dates/times are recorded through the relation of a **cc_design_date_and_time_assignment** entity with the **date_time_role** being "sign_off_date". This relation is explained in section 2.3 on page 5, but is not required in the AP. In the cycle case, the **approval_date_time** <u>only</u> indicates when the status of the **approval** was last changed.

The **approval_status name** attribute in AP 203 has a restriction on its possible values. The values shall only be "approved", "not_yet_approved", "disapproved" or "withdrawn". This restriction is enforced by the **restrict_approval_status** rule.

Pre-processor Recommendations: There is no standard mapping for the **approval level** attribute. Since there is no standard mapping in the AP 203 application domain for this attribute, it is recommended that this attribute contain a null string as minimal content or any appropriate/mutually agreed upon string. If no appropriate data for the **approval_role** attribute (why this person and organization is approving) is available it is recommended that this attribute contain the value "approver". It is recommended that <u>all</u> **approval_person_organization** instances have associated **cc_design_date_and_time_assignment** entities to provide complete clarity.

Post-processor Recommendations: Since there is no standard mapping for the **approval level** attribute, post-processors should not assign any processing significance to this value.

2.5 Security

AP 203 requires that certain constructs indicate their sensitivity to the owning organization. This is accomplished by establishing the **security_classification** entity and relating it to the construct via the **cc_design_security_classification** entity. The classification is given in **security_classification_level name** attribute. AP 203 restricts the values of this attribute to be "unclassified", "classified", "proprietary", "confidential", "secret", and "top_secret" through the **restrict_security_classification_level** rule. It should be noted that the value of "classified" only indicates that the data is <u>not</u> unclassified. This value is used when an organization has a security classification which does not <u>exactly</u> match any of the other values.

A **security_classification** in AP 203 requires an approval (see section 2.4 on page 6), a person and organization (see section 2.2 on page 4) in the role of **classification_officer** and a date and

time (see section 2.3 on page 5) in the role of **classification_date**. It should be noted that the AP provides for indication of an expiration date for the classification by relating a date/ time in the role of **declassification_date** (see section 2.3), but this is not required.

Pre-processor Recommendations: There is no standard mapping for the **security_classification purpose** attribute. Since there is no standard mapping in the AP 203 application domain for this attribute, it is recommended that this attribute contain a null string as minimal content or any appropriate/ mutually agreed upon string. There is no standard mapping for the **security_classification name** attribute. Since there is no standard mapping in the AP 203 application domain for this attribute, it is recommended that this attribute contain a null string as minimal content or any appropriate/ mutually agreed upon string.

If the **security_classification_level name** attribute is the value "classified", it is recommended that the organization's classification designation be placed in the **security_classification name** attribute. For example, if an organization had a security classification of "secret restricted", the **security_classification_level name** attribute value would have the value "classified", and the **security_classification name** attribute would have the value "secret restricted".

Post-processor Recommendations: There is no standard mapping for the **security_classification purpose** attribute. Since there is no standard mapping in the AP 203 application domain for this attribute, it is recommended that this attribute contain a null string as minimal content or any appropriate/ mutually agreed upon string. If the **security_classification_level name** attribute is the value "classified", it is recommended that post-processors regard the **security_classification name** data as the identification of a special or non-standard classification. If the **security_classification_level name** attribute has a value of other than "classified", it is recommended that post-processors not assign any processing significance to the **name** attributevalue.

2.6 Units of Measure

AP 203 provides for a number of units of measure which can be used for quantities or determining the dimensionality of a shape. The units of measure can be subdivided into explicit and context sensitive units of measure. The explicit units supported are: **length_unit**, **plane_angle_unit**, **solid_angle_unit**, **area_unit**, **mass_unit**, and **volume_unit**. The context sensitive units are defined through the use of the **context_dependent_unit name** attribute. Each of the units have a related measure. For the explicit units, these are: **length_measure**, **positive_length_measure**, **plane_angle_measure**, **positive_plane_angle_measure**, **solid_angle_measure**, **area_measure**, **mass_measure**, and **volume_measure**. For the context sensitive units, these are **context_dependent_measure**, **count_measure**, **descriptive_measure**.

2.7 Shape

AP 203 provides seven types of **shape_representation** which are grouped into five conformance classes as shown in table 15 of the AP. These classes are: geometrically bounded shape models

which are represented by **geometrically_bounded_wireframe_shape_representation** and **geometrically_bounded_surface_shape_representation** entities, wireframe with topology shape models which are represented by **edge_based_wireframe_shape_representation** and **shell_based_wireframe_shape_representation** entities, manifold surface with topology shape models which are represented by **manifold_surface_shape_representation** entities, faceted boundary representation shape models which are represented by **faceted_brep_shape_representation** entities, and boundary representation models which are represented by **advanced_brep_shape_representation** entities.

A **shape_representation** (see also G.1.18 on page 84) must be related to a **representation_context**. AP 203 has an additional rule which states that a **representation_context** must be a **geometric_representation_context**. The **representation_items** in the sub-types of **shape_representation** must have a dimensionality (as defined by the **dimension_of** function) of 3 if they are **geometric_representation_items**.

It should be noted that the **name** attribute of **representation_item** was intended to contain an identification or tag for the geometrical and topological entities. It should be noted that tag values from one CAD system are often not compatible with tag values of another system.

This document will not go into detail on **shape_representation**. It will only present clarifications and practices for the different types of shape as appropriate. There are a number of potential implementor agreements in regard to shape. These agreements are found in appendix G on page 81.

Pre-processor Recommendations: Since CAD system internal entity tag values are typically <u>not</u> compatible from one system to another, it is recommended that pre-processors either use the physical file entity number for the **name** attribute value of **representation_item** entities or a null string for minimal content.

2.7.1 Units for Shape

Units are defined for a type of **shape_representation** through the use of a complex instance of **global_unit_assigned_context** and **geometric_representation_context**. When global units are used, units must be defined for **length_unit**, **plane_angle_unit**, and **solid_angle_unit**. The base units for STEP are Standard International (SI) units which are represented through the **named_unit** sub-type **si_unit**. All other units (such as English units) are represented as **conversion_based_unit** entities which reference **si_units**. Physical file examples for SI and English units can be found in appendix D on page 68. (See G.1.9 on page 83 for an implementors agreement that affects units.)

In addition to the global measurement units described in the prior paragraph, AP 203 provides for the definition of a global gap tolerance for a shape model through the addition of **global_uncertainty_assigned_context** to the complex **representation_context** instance. This

entity defines a set of **uncertainty_measure_with_unit** entities to represent various gap type measurements. Physical file examples for SI and English units of uncertainty can be found in appendix D on page 68.

As a clarification to 10303-42, units on parametric representations are taken from the **global_unit_assigned_context** entity. They are not always degrees as might be extrapolated from reading the text of 10303-42. This is a consideration on choosing the global units for plane angles as radian units are irrational and potentially unstable.

Pre-processor Recommendations: If a pre-processor uses **global_uncertainty_assigned_context**, it should point to <u>one</u> **uncertainty_measure_with_unit** which should identify a **length_measure**. The value of the **name** attribute shall be "closure". The **length_measure** shall contain the value of the largest gap anticipated between elements that should be deemed coincident.

Pre-processors should use degree as the unit for **plane_angle_unit** as it is more stable than using a radian unit.

Post-processor Recommendations: Post-processors shall use the **uncertainty_measure_with_unit** value for error checking of the file where an error is a gap in the shape which is larger than the **length_measure** value.

2.7.2 Shape Aspects

Portions of a shape model can be designated as **shape_aspects**. This can be done just for internal model subdivisions or to attach specifications to portions of the shape. The **shape_aspect name** and **description** attributes have no standard mapping. The **product_definitional** flag must be ".T." if the portion of the shape identified is on the outer boundary of the shape model. The use of this construct will be dealt with in sections 2.8.5, 2.8.10, 2.8.3 and 2.8.11 on pages 20, 33, 18 and 36.

> NOTE - The IS edition of AP 203 does not list **shape_aspect** in the entity list for conformance class 1. This was an omission. **Shape_aspect** <u>should</u> be supported in conformance class 1 for qualification of specifications.

2.7.3 Boundary Representation Models

This sub-section will not provide detailed information on boundary representation models. It should be noted that most AP 203 conformance class 6 implementations are providing surface, seam and intersection curves in the data. This has been noted as an issue against AP 203 and is being considered for future editions. This practice may also be formalized through an implementors agreement (see G.1.1 on page 81) in the interim. Organizations implementing AP 203 should provide provisions for this data in their post-processors even though AP 203 has a

rule which precludes it.

Pre-processor Recommendations: Pre-processors should not use **face_outer_bound** designations on closed periodic surfaces (cylinder, sphere, torus) as this designation is ambiguous. Pre-processors

Post-processor Recommendations: Post-processors should ignore the **face_outer_bound** designations on closed periodic surfaces (cylinder, sphere, torus) as this designation is ambiguous.

2.8 Parts in AP 203

In order to define a part in AP 203, three basic AIM entities are used. The **product** entity establishes the part's identification (or part number), name (or nomenclature), and description. The **product_definition_formation_with_specified_source** entity identifies its version (or change level). The **product_definition** or **product_definition_with_associated_documents** entity identifies the engineering discipline view that all the data related to it represents (e.g. design engineering, manufacturing engineering, logistics, etc).

```
                                                    ┌──────────────┐
                                                    │ APPLICATION  │
                                                    │   CONTEXT    │
                                                    └──────────────┘

┌──────────┐      ┌──────────┐   ┌──────────┐            ┌──────────┐
│ PRODUCT  │      │          │   │MECHANICAL│            │APPLICATION│
│ RELATED  │─────▶│ PRODUCT  │──▶│ CONTEXT  │            │ PROTOCOL │
│ PRODUCT  │      │          │   │          │            │DEFINITION│
│ CATEGORY │      └──────────┘   └──────────┘            └──────────┘
└──────────┘

                  ┌──────────┐   ┌──────────────┐
┌──────────┐      │ PRODUCT  │   │  SECURITY    │
│ PERSON   │      │DEFINITION│◀──│CLASSIFICATION│
│  AND     │─────▶│FORMATION │   └──────────────┘
│ORGANIZA- │      │WITH SPEC-│
│  TION    │      │IFIED     │            ┌──────────┐
└──────────┘      │SOURCE    │            │ APPROVAL │
                  └──────────┘            └──────────┘

┌──────────┐      ┌ ─ ─ ─ ─ ┐   ┌──────────┐
│ PRODUCT  │      │ PRODUCT  │   │  DATE    │
│DEFINITION│◀─────│DEFINITION│◀──│   AND    │
│ CONTEXT  │      └ ─ ─ ─ ─ ┘   │  TIME    │
└──────────┘                    └──────────┘

NOTE: DASHED LINES INDICATE ENTITY IS NOT REQUIRED TO EXIST BY AP 203
```

Using the three entities (and EXPRESS sub-types), the part is identified, revision controlled, and life cycle stage insulated. **Figure 1** describes pictorially the relationships among the entities needed to define a part in AP 203 at a high level. These entities and relationships are necessary in order for AP 203 to support the various configuration control methodologies which affect parts. The reader must remember that AP 203 is a standard for all industry. As such, AP 203 represents the data (your data) as an abstraction from the way in which your organization actually does business.

2.8.1 Identifying Parts

2.8.1.1 The Product Entity

AP 203 deals with all parts as **product**s. The part number for a part is stored in the **id** attribute. The nomenclature or name of the part is stored in the **name** attribute. If there is an expanded name or description of the part this is stored in the **description** attribute. All STEP **product**s must be founded in some **product_context** which identifies the engineering discipline from which

the data is viewed. See section 2.1.2 on page 2 for guidance on creating this entity.

In populating the data, the **id** or part number must be unique. This is typically not a problem when the part is used only within a single company. If the data being assembled is for worldwide consumption, the **id** must be unique in that domain.

AP 203 requires that all **product**s exist in at least one **product_category**. This will be dealt with in detail in section 2.8.2 on page 16. This restriction (**product_requires_product_category**) forces all parts into one of the following categories: detail, assembly, inseparable assembly or customer_furnished_equipment.

Products in AP 203 require a **person_and_organization** in the role of "design_owner". This designation is applied to the person and organization or design authority who originally designed the part. More simply, this is the person and organization typically identified at the top of the title block on the drawing which defines the part. See section 2.2 on page 4 for guidance on how to create the person and organization entities.

Pre-processor Recommendations: All pre-processors should use non-defaulted data or user input for the values assigned to the design owner of a **product** as defaulting this data has a high probability of causing the data to be incorrect.

If the data is intended for external usage, the part number should be prefixed with the **organization id** value followed by a comma to ensure uniqueness. For example, if the **organization id** value were "USA,93699" and the **product id** were "999999", the actual value of the **product id** would be "USA,93699,999999".

2.8.1.2 The Product Definition Formation Entity

AP 203 has a rule (**product_requires_version**) which requires that all part **product**s be associated with a **product_definition_formation** entity. This relation is required as AP 203 is required to support the versioning of parts. This rule ensures that all information which typically varies from version to version is always related to the part. The **product_definition_formation** entity has a associated rule (**subtype_mandatory_product_definition_formation**) which forces all **product_definition_formation** entities to be **product_definition_formation_with_specified_source** entities. The net effect of these two rules is that a **product** must be associated with a **product_definition_formation_with_specified_source**.

There are many organizations which claim quite firmly (and possibly rightly so) that they do not version parts. All that is being done here is establishing a connection which may or may not have valuable data. There are some things to consider before any group claims that they do or do not version parts.

In AP 203, the connection being established is actually a connection to the data which comprises the body of a parts list for the part (if it is an assembly). If your organization versions parts, the **product_definition_formation_with_specified_source id** attribute should contain the value which represents this version. The **description** attribute should contain the reason for the creation of the version. The **source** attribute must contain a value of ".MADE.", ".BOUGHT." or ".NOT_KNOWN.". The value should be ".MADE." if the part is built within the company. The value should be ".BOUGHT." for vendor parts. Discretion on the value is left to the producer of the data with the above guidance given as this distinction can be unclear particularly when the data is exchanged to another party.

AP 203 requires that all **product_definition_formation** and therefore all **product_definition_formation_with_specified_source** entities be associated with a **person_and_organization** in the role of "creator". This person and organization is the one which created the change. The data for this person can be found by looking at the release or change paper work data and finding the initiator. For guidance on creating the entities associated with this data, see section 2.2 on page 4.

AP 203 requires that all **product_definition_formation** and therefore all **product_definition_formation_with_specified_source** entities be associated with at least one person and organization in the role of either "design_supplier" or "part_supplier". For guidance on creating the entities associated with this data, see section 2.2 on page 4. The person and organization in the role of "design_supplier" is the one which was the custodian of the master data when the version was created. The person and organization in the role of "part_supplier" is the one which had manufacturing cognizance (if the part is made internally to the organization) or the vendor who supplies the part if it is a vendor part.

AP 203 requires that all **product_definition_formation** and therefore all **product_definition_formation_with_specified_source** entities be associated with an **approval**. This is the person and organization that approved the version. The data for this person can be found by looking at the release or change paper work data and finding who approved the release or change. For guidance on creating the entities associated with an **approval**, see section 2.4 on page 6.

AP 203 requires that all **product_definition_formation** and therefore all **product_definition_formation_with_specified_source** entities be associated with a **security_classification**. For guidance on creating the entities associated with a **security_classification**, see section 2.5 on page 7.

Pre-processor Recommendations: If your organization does not version parts, the **id** attribute should contain a null string as minimal data content or any mutually agreed upon string. If the **id** attribute was a null string, the **description** value would also be a null string. All pre-processors should use non-defaulted data or user input for the values assigned to the creator, design and part suppliers, approvers and approval date for

product_definition_formation_with_specified_source entities as defaulting this data has a high probability of causing this data to be incorrect.

For the **security_classification** classification officer, classification date, approvers and approval dates, these can be extrapolated from the version creator and approval data if no appropriate data is available.

It is recommended that pre-processors use an **id** of "ANY" where they wish to indicate a generic revision of a part. This type of instancing would be used when the part with the revision of "ANY" is a component in an assembly to indicate that any existing revision of the component is valid for the assembly. This type of instancing reduces the amount of data to be sent in change packages. When this is used, it reduces the ability to track the actual contents of parts lists at a particular change level when the organization versions parts.

Post-processor Recommendations: When the value of the **id** and **description** attributes for **product_definition_formation_with_specified_source** is a null string, post-processors should use this as an indication that there is no version of the part.

It is recommended that post-processors recognize an **id** of "ANY" as indicating a generic revision of a part. This type of instancing would be used when the part with the revision of "ANY" is a component in an assembly to indicate that any existing revision of the component is valid for the assembly.

2.8.1.3 The Product Definition Entity

AP 203 and STEP use the **product_definition** entity to establish specific life cycle stage views of the product data. The use of **product_definition** entities is not required in AP 203, but this entity establishes many important relationships such as part to part and part to shape. If the **product_definition** entity is not used, all that can be done with AP 203 is identify individual parts with respect to their part number, name and version identification.

It is possible to have many **product_definition**s for a part/version combination. The **id** attribute should identify whose view of the **product** a particular instance represents. There are no standard mappings in the AP for this attribute or the **description** attribute (see pre-processor recommendations).

AP 203 requires that all **product_definition**s have a **person_and_organization** assigned in the role of "creator". This person and organization is the one which defined the view. If the **product_definition** is being used as solely a connection to shape, this would be the person who filed the CAD model of the shape. For guidance on creating these constructs, see section 2.2 on page 4.

AP 203 requires that all **product_definition**s have a **date_and_time** assigned in the role of "creation_date". This date and time is when the view was defined. If the **product_definition** is being used as solely a connection to shape, this would be the file date and time for the CAD model of the shape. If this is not the case, see the pre-processor recommendations. For guidance on creating the date and time constructs, see section 2.3 on page 5.

AP 203 requires that all **product_definition**s have an **approval**. This data is often difficult to obtain as those who approved the filing of the CAD model or creation of the **product_definition** are difficult to identify. If the information is available, see section 2.4 on page 6 for guidance on creating the approval.

AP 203 has an optional feature where a **product_definition** may be related to **document** entities through the sub-type **product_definition_with_associated_documents**. In AP 203, this usage was intended only for documents which identify associated Computer Aided Design (CAD) files (where the **document_type** attribute **product_data_type** has the value "cad_filename") and drawings (where the **document_type** attribute **product_data_type** has the value "drawing").

There is a rule in AP 203 restricting the **product_data_type** of **document**s, but the AP does not formally eliminate the possibility of relating specification type **document**s using the **product_definition_with_associated_documents** sub-type. This type of relationship indicates that the document is available through the design supplier's organization since it does not provide the **source** data for the specification. See section 2.8.3 on page 18 for more detail on creating and referencing specifications.

When using **product_definition_with_associated_documents** to reference CAD files, the **document id** attribute should contain the file name of the file with enough detail so that it is uniquely identified in the exchange. This means that the **id** attribute should identify the source and system (or standard (e.g. IGES, STEP, etc.)) together with the file name. A good method for this is to suffix the file name with the CAD file producer's Internet domain name with the unique system name whether or not the system is physically attached to the Internet. For example, if the Internet domain name for the company was "widget.com" and the system name was "sparky" and the file name was "mymodel.fle", the **document id** attribute value would be "sparky.widget.com/mymodel.fle". If the producer does not use the Internet, use the unique organization identification (see section 2.2 on page 4 for guidance). The **name** attribute should contain the simple file name. The **description** attribute should contain the description of the CAD file contents complete with that CAD (or other) package used to create the data.

Pre-processor Recommendations: There is no standard mapping for the **id** attribute of **product_definition**. It is recommended that this attribute contain possible values of "design", "analysis", "digital pre-assembly", "manufacturing", "as built", "as maintained". These values should be used to indicate which group owns the view for concurrent engineering purposes within a life cycle stage. There is no standard mapping for the **description** attribute. Since there is no standard mapping in the AP 203 application domain for this attribute, it is recommended that this

attribute contain a null string as minimal content or any appropriate/ mutually agreed upon string. Where values for the creator, creation date are not readily available, this information can be extrapolated from the creator and approval related to the **product_definition_formation_with_specified_source** as defined in section 2.8.1.2 on page 12. Pre-processors shall not use **product_definition_with_associated_documents** to relate specification type documents to the **product_definition**.

Pre-processors may use the / character as a delimiter to separate the sending system identification from the actual file name for the **document id** attribute if the receiving system does not have a uniqueness requirement on this value.

Post-processor Recommendations: All post-processors should utilize the values given above for pre-processors as computer sensible segregations of the **product_definition** data based on the **id** attribute. If a value other than those above is received, it should be regarded as "design". Since there is no standard mapping for the **description** attribute for the **product_definition** entity (and sub-type), it is recommended that post-processors not assign any processing significance to this value.

2.8.2 Categorizing Parts

AP 203 provides for assigning parts to categories and for creating hierarchical networks of categories. Categories can be extremely useful in adding intelligence to the data. Parts are assigned directly to categories though the **product_related_product_category** entity which is a sub-type of the **product_category** entity. Networks of categories can be created by relating super and subcategories through **product_category_relationship** entity. When a **product_related_product_category** participates in a **product_category_relationship** in the AP 203 domain, it should <u>always</u> be a subcategory.

Pre-processor Recommendations: There are no standard mappings for the **product_category_relationship name** or **description** attributes. Since there are no standard mappings in the AP 203 application domain for these attributes, it is recommended that these attributes contain a null string as minimal content or any appropriate/ mutually agreed upon string. Pre-processors should use lower case for **product_category name** values. Leading and trailing blanks in the **product_category name** value should be removed.

Post-processor Recommendations: Since there are no standard mappings for the **product_category_relationship name** or **description** attributes, it is recommended that post-processors not assign any processing significance to these values. Post-processors should attempt to store <u>all</u> categories and subcategories and category relationships received in an AP 203 exchange as this information adds meaning to the received data. If it is impossible to store the data, the user should be informed of all categories and relationships not processed. This would be best done by presenting the user with a report on the category structure in the file with subcategories indented. Post-processors should use non-case sensitive checking when

determining matches on processed category data. Leading and trailing blanks in the **product_category name** value should be removed.

2.8.2.1 AP 203 Standard Categories

AP 203 has a rule (**restrict_product_category_value**) which controls the categories which **product**s can be directly assigned to. The **name** attribute of **product_related_product_category** is restricted to only have the values of "assembly", "detail", "customer_furnished_equipment", "inseparable_assembly", "cast", "coined", "drawn", "extruded", "forged", "formed", "machined", "molded", "rolled", and "sheared".

There are no restrictions in AP 203 on the value of the **product_category name** attribute when the instance created is at the super-type level. This means that when hierarchical networks of categories are created there is no restriction on the names of the categories <u>not</u> directly related to a **product**.

2.8.2.1.1 Standard Parts

AP 203 defines via its mapping table a special mapping for standard parts. The defined mapping is that a **product_category** with a **name** attribute value of "standard_part" be created and that this category be related through a **product_category_relationship** to a **product_related_product_category** related to a part.

2.8.2.2 Recommended Categories

It is <u>strongly</u> recommended that all implementations of AP 203 establish an instance of **product_category** with a **name** attribute value of "part". This recommendation is made to facilitate interoperability and allow implementations to defend against other value assignments made in other APs which will result from the reuse of the resource part **product_category** entities.

It is further recommended that all implementations of AP 203 support the following high level categories which are not standardized in the AP, but will undoubtedly have common usage:

"commercial" - This category indicates that the **product** referenced is a general-purpose commercially-available part.

"customer furnished customer installed" - This category indicates that the **product** referenced is part of the system or unit for requirements definition, but is actually placed in the system or unit after some portion of delivery.

"government" - This category indicates that the **product** referenced is a part which has been developed or purchased to meet specialized government specifications.

18

"hazardous material" - This category indicates that the **product** referenced either is or contains hazardous material.

"interchangeable" - This category indicates that the **product** referenced is a part which requires no trimming or modification when replaced.

"material" - This category indicates that the **product** referenced is actually material or bulk material.

"replaceable" - This category indicates that the **product** referenced is a part which requires trimming or some modification (usually for fit) when replaced.

"serialized" - This category indicates that the **product** referenced is (or contains) a serialized part.

2.8.3 Relating Specifications to Parts

AP 203 relates specifications to entire parts by relating a **cc_design_specification_reference** entity to the **product_definition** of the part. If the specification only relates to a portion of the part, the **cc_design_specification_reference** entity is related to a **shape_aspect** which is in turn related to the **product_definition_shape** of the part. It should be noted that a specification related (through either method) to the **product_definition** of the part <u>must</u> be applicable in <u>every</u> usage. This relationship is shown pictorially in **Figure 2** on page 19.

The **cc_design_specification_reference** entity identifies the owner of the specification through the **source** attribute. This attribute should contain an unambiguous identification of where the receiver of the data could obtain a copy of the document. The **document** related to the **cc_design_specification_reference** must be uniquely identified in the exchange by the **id** attribute. This means that the **id** should contain any revision information needed to identify the document completely. The **name** attribute should contain the title of the document. The **description** attribute should contain an expanded explanation of the **document**'s contents.

```
        PRODUCT                    DOCUMENT          →      DOCUMENT
                                   USAGE
                                   CONSTRAINT

           ↑
   PRODUCT          PRODUCT              CC DESIGN
   DEFINITION   ←   DEFINITION   ←       SPECIFICATION
   FORMATION                            REFERENCE
   WITH SPECIFIED
   SOURCE

                      ↑
   DOCUMENT   ←   DOCUMENT      PRODUCT
                  USAGE         DEFINITION
                  CONSTRAINT    SHAPE

           ↑                      ↑
           CC DESIGN          SHAPE      ←      PROPERTY
           SPECIFICATION  →   ASPECT             DEFINITION
           REFERENCE

   NOTE: DASHED LINES INDICATE         SHAPE            SHAPE
   ENTITY IS NOT REQUIRED TO EXIST     REPRESENTATION ← DEFINITION
   BY AP 203                                            REPRESENTATION
```

Since many specifications cover a variety of subtopics and options on a given topic, it may be necessary to identify a particular subtopic of the specification and assign option values. In AP 203 this is accomplished by relating a **document_usage_constraint** to the document. The **subject_element** attribute identifies the particular section or topic being referenced in the specification. The **subject_element_value** identifies any option choices or restrictions placed on the section or subtopic.

The **document_usage_constraint** should <u>not</u> be used to reference classes defined in specifications such as process specifications. This should be done by using the **document** entity sub-type **document_with_class**. If classed documents require further restriction of the class, a **document_usage_constraint** may be related to the **document_with_class** entity.

AP 203 provides for **document**s related to a **product_definition** to be related to other **document**s in a network type relationship. This is accomplished through the **document_relationship** entity. There are no standard mappings for the **name** and **description** attributes in this entity.

Pre-processor Recommendations: There are no standard mappings for the **name** and **description** attributes in a **document_relationship**. Since there are no standard mappings in the AP 203 application domain for these attributes, it is recommended that these attributes contain a null string as minimal content or any appropriate/ mutually agreed upon string.

Post-processor Recommendations: Post-processors should store all data found in specification **document**s attached to **product_definition**s or **shape_aspect**s. If it is not possible to store all the data, the user must be informed of the data being omitted and its relationship to the **product_definition** or **shape_aspect**. Since there are no standard mappings for the **name** and **description** attributes in a **document_relationship**, it is recommended that post-processors not assign any processing significance to these values.

2.8.4 Relating Parts to Contracts

AP 203 provides an optional relationship of **product**s to contracts through the **cc_design_contract** entity which relates a **contract** to a **product_definition_formation_with_specified_source** or design version. In AP 203, a **contract** can be used to represent either an explicit contract which provides the requirements (and typically the funds) for the creation of designs for the **product** or some other agreement (such as a purchase order) which fulfills the same function. The **contract name** attribute should contain the contract or agreement identifying number or name if no number exists. The **purpose** attribute should contain the reason for the existence of the contract or agreement. The **contract_type description** attribute is restricted by AP 203 to contain only the values "fixed_price" or "cost_plus".

AP 203 requires that a **contract** have an associated **approval**. For guidance in creating the **approval** constructs, see section 2.4 on page 6 of this document. A **contract** must also have an associated **person_and_organization** in the role of "contractor". For guidance in creating the person and organization constructs, see section 2.2 on page 4 of this document. A **contract** <u>may</u> have an associated **date_and_time** in the role of "contract_date". For guidance in creating the date and time constructs, see section 2.3 on page 5 of this document.

Pre-processor Recommendations: It may be difficult to obtain the **approval** and **contractor** information. If this information is not available, it should be provided either through user input or from default data based on the **contract** name value.

2.8.5 Relating Shape to Parts

AP 203 uses two entities to form the link between the configuration management data for a part and the shape for a part. These two entities are **product_definition_shape** and **shape_definition_representation**. There are no standard mappings for the **product_definition_shape name** and **description** attributes. It should be noted that no link to shape is required. It is possible to use the **product_definition_shape** entity to indicate that a part

has (or will have) shape without relating a **shape_definition_representation**.

```
┌──────────┐       ┌──────────────┐       ┌──────────────┐
│          │       │ PRODUCT      │       │              │
│          │       │ DEFINITION   │       │ PRODUCT      │
│ PRODUCT  │◄──────│ FORMATION    │◄──────│ DEFINITION   │
│          │       │ WITH SPECIFIED│      │              │
│          │       │ SOURCE       │       │              │
└──────────┘       └──────────────┘       └──────────────┘
                                                 ▲
                                                 │
                                          ┌──────────────┐
                                          │ PRODUCT      │
                                          │ DEFINITION   │
                                          │ SHAPE        │
                                          └──────────────┘
                                            ▲         ▲
                               ┌────────────┘         └────────────┐
                    ┌──────────────┐                    ┌──────────────┐
                    │ SHAPE        │                    │ SHAPE        │
                    │ DEFINITION   │                    │ DEFINITION   │
                    │ REPRESENTATION│                   │ REPRESENTATION│
                    └──────────────┘                    └──────────────┘
                           │                                   │
                           ▼                                   ▼
                    ┌──────────────┐                    ┌──────────────┐
                    │ FACETED      │                    │ ADVANCED     │
                    │ BREP         │                    │ BREP         │
                    │ SHAPE        │                    │ SHAPE        │
                    │ REPRESENTATION│                   │ REPRESENTATION│
                    └──────────────┘                    └──────────────┘
```
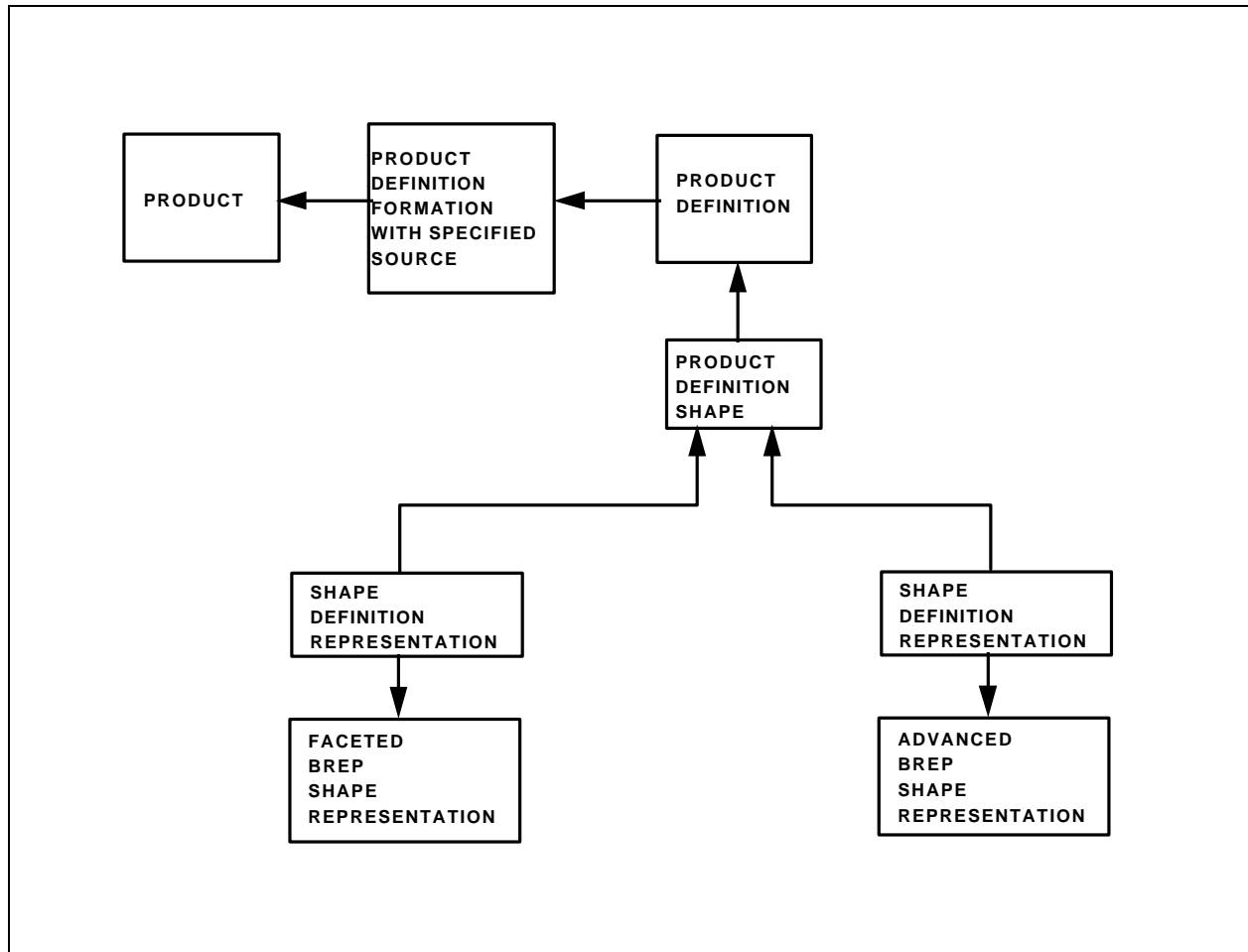
There must be only <u>one</u> **product_definition_shape** for each **product_definition** in an AP 203 exchange file. If there are multiple **shape_definition_representation** entities related to the **product_definition_shape**, these relationships describe alternate representations. This is depicted in **Figure 3**.
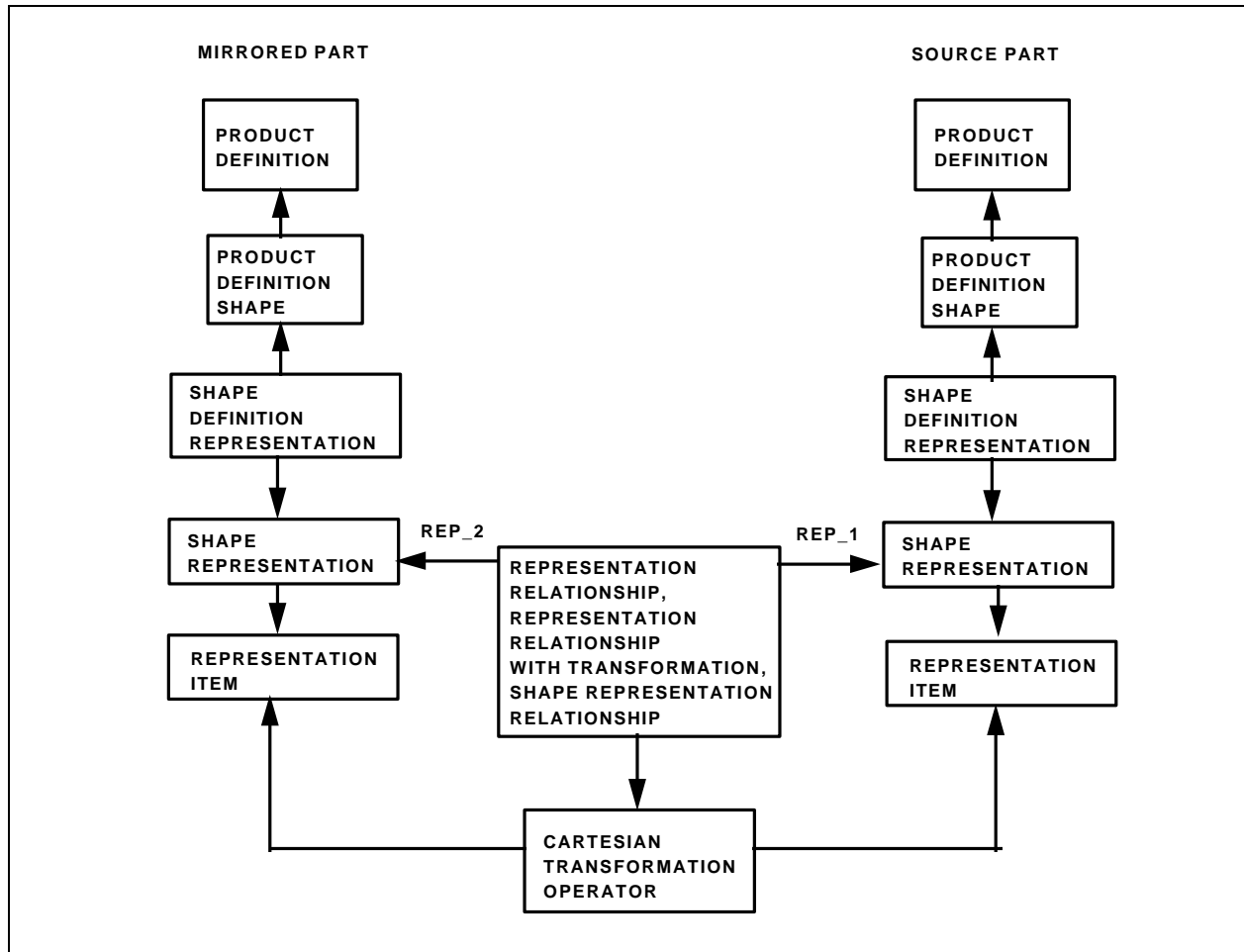
If the shape of the part is composed of shape constructs from multiple types of **shape_representation** to form the entire shape model, the main **shape_representation** shall be related to a **shape_definition_representation** which relates to the **product_definition_shape**. The other **shape_representations** are related to the main **shape_representation** through a **shape_representation_relationship**. This is depicted in **Figure 4**.

> NOTE - The above is a revised recommendation. See section I.1 on page 91 for the prior recommendation.

```
                    ┌──────────────────┐              ┌──────────────┐
                    │ PRODUCT          │              │ PRODUCT      │
  ┌──────────┐      │ DEFINITION       │              │ DEFINITION   │
  │ PRODUCT  │◄─────│ FORMATION        │◄─────────────│              │
  └──────────┘      │ WITH SPECIFIED   │              └──────────────┘
                    │ SOURCE           │                     ▲
                    └──────────────────┘                     │
                                                      ┌──────────────┐
                                                      │ PRODUCT      │
                                                      │ DEFINITION   │
                                                      │ SHAPE        │
                                                      └──────────────┘
                                                             ▲
                                                      ┌──────────────┐
                                                      │ SHAPE        │
                                                      │ DEFINITION   │
                                                      │ REPRESENTATION│
                                                      └──────────────┘
                                                             │
  ┌──────────────┐   REP_2  ┌──────────────┐  REP_1   ┌──────────────┐
  │ FACETED      │          │ SHAPE        │          │ ADVANCED     │
  │ BREP         │◄─────────│ REPRESENTATION│────────►│ BREP         │
  │ SHAPE        │          │ RELATIONSHIP │          │ SHAPE        │
  │ REPRESENTATION│         └──────────────┘          │ REPRESENTATION│
  └──────────────┘                                    └──────────────┘
```
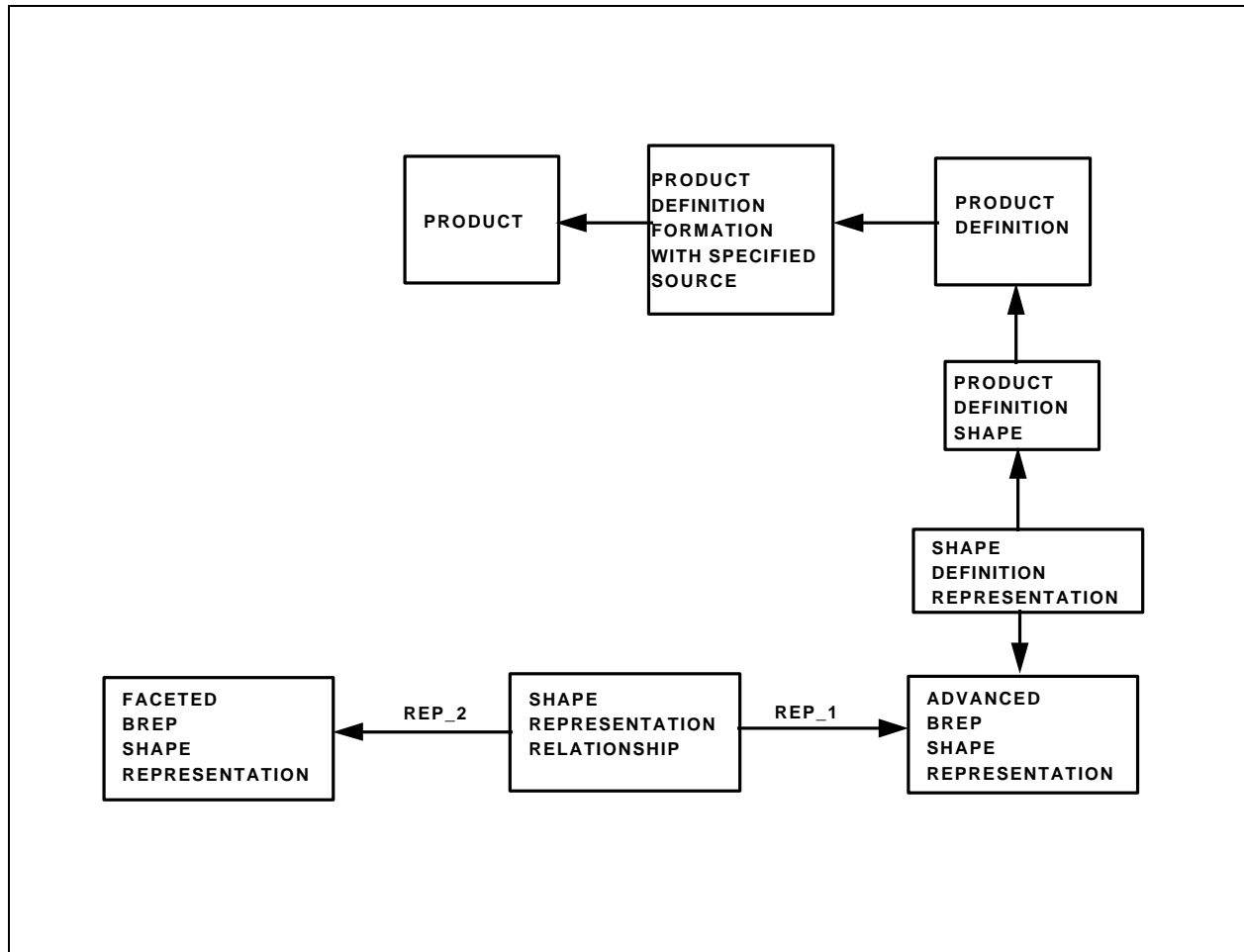
In some cases, the shape of a part is based on the shape of another part. This commonly occurs when the one part is the mirror image of the other. When this occurs, it is through a **representation_relationship_with_transformation**. This is structure is shown pictorially in **Figure 5**. The transformation is constructed based on a **functionally_defined_transformation**. It is presumed that the transformation would be applied to the coordinate system of the source part prior to it being mapped to that of the mirrored part.
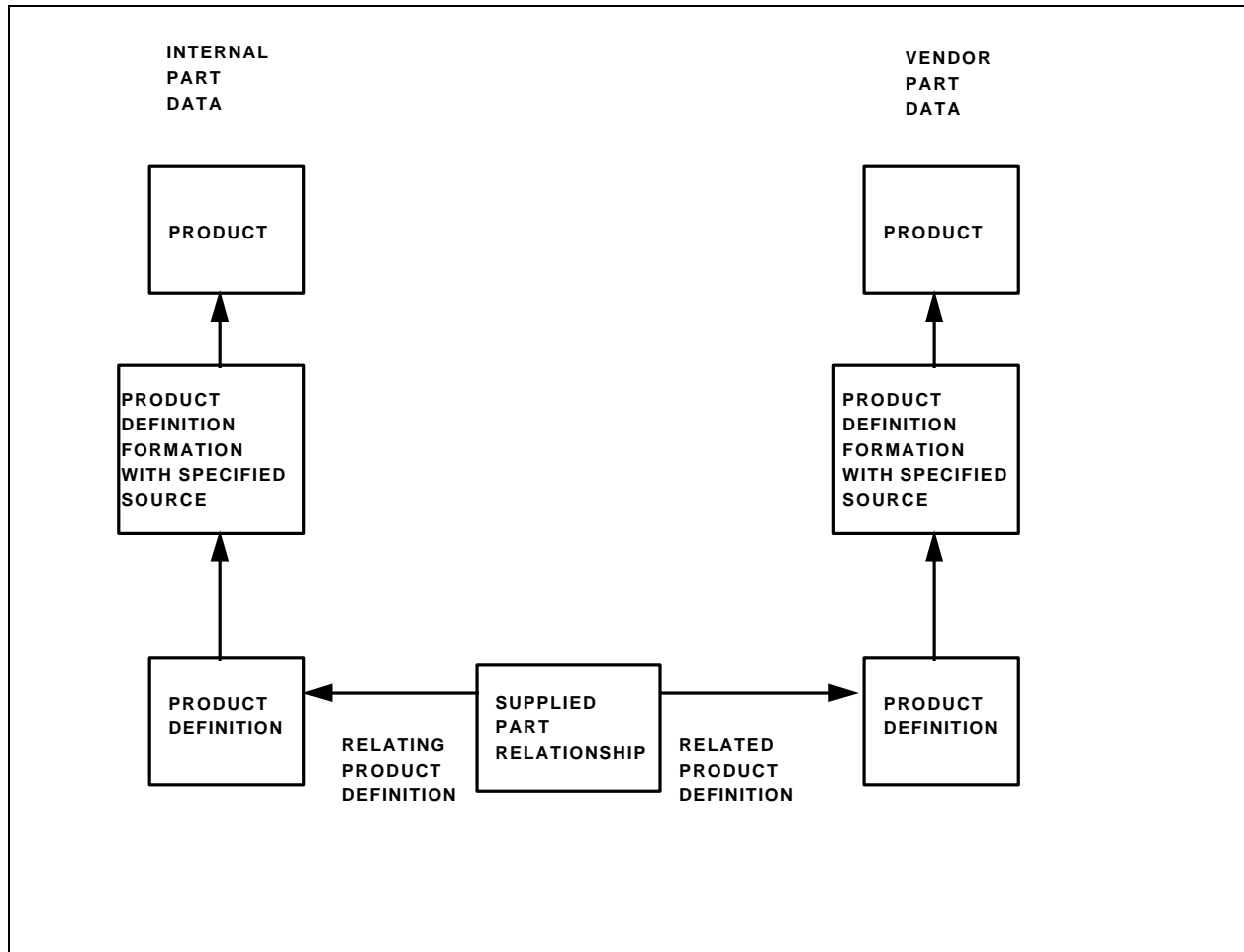
Pre-processor Recommendations: There are no standard mappings for the **name** and **description** attributes for **product_definition_shape**. Since there are no standard mappings in the AP 203 application domain for these attributes, it is recommended that these attributes contain a null string as minimal content or any appropriate/ mutually agreed upon string.

```
        MIRRORED PART                                          SOURCE PART

      ┌─────────────┐                                        ┌─────────────┐
      │  PRODUCT    │                                        │  PRODUCT    │
      │ DEFINITION  │                                        │ DEFINITION  │
      └─────────────┘                                        └─────────────┘
             ↑                                                      ↑
      ┌─────────────┐                                        ┌─────────────┐
      │  PRODUCT    │                                        │  PRODUCT    │
      │ DEFINITION  │                                        │ DEFINITION  │
      │   SHAPE     │                                        │   SHAPE     │
      └─────────────┘                                        └─────────────┘
             ↑                                                      ↑
      ┌─────────────┐                                        ┌─────────────┐
      │   SHAPE     │                                        │   SHAPE     │
      │ DEFINITION  │                                        │ DEFINITION  │
      │REPRESENTATION│                                       │REPRESENTATION│
      └─────────────┘                                        └─────────────┘
             ↓              REP_2                REP_1               ↓
      ┌─────────────┐              ┌──────────────────────┐  ┌─────────────┐
      │   SHAPE     │ ←─────────── │   REPRESENTATION     │ ─→│   SHAPE     │
      │REPRESENTATION│              │   RELATIONSHIP,      │   │REPRESENTATION│
      └─────────────┘              │   REPRESENTATION     │   └─────────────┘
             ↓                     │   RELATIONSHIP       │          ↓
      ┌─────────────┐              │   WITH TRANSFORMATION,│  ┌─────────────┐
      │REPRESENTATION│             │   SHAPE REPRESENTATION│  │REPRESENTATION│
      │    ITEM      │             │   RELATIONSHIP       │   │    ITEM      │
      └─────────────┘              └──────────────────────┘  └─────────────┘
             ↑                              ↓                        ↑
             │              ┌──────────────────────┐                │
             └───────────── │    CARTESIAN         │ ───────────────┘
                            │   TRANSFORMATION     │
                            │    OPERATOR          │
                            └──────────────────────┘
```

 Post-processor Recommendations: Since there are no standard mappings for the **name** and **description** attributes for **product_definition_shape**, it is recommended that post-processors not assign any processing significance to these values.

2.8.6 Renumbering Vendor Parts

In all realms of design and manufacturing business, it is common to buy parts from a vendor and renumber them under an internal numbering scheme. In today's practice, this is done through envelope, specification and source control drawings. An envelope drawing is used for a simple renumber of a part where the part is referenced on the envelope drawing and assigned a new part number via the associated parts list. A specification control drawing renumbers a part to show that it meets or exceeds the specifications defined on the drawing and to recommend sources for the part. A source control drawing renumbers a part and creates a restricted list of suppliers which are qualified to produce the part based on the specifications.

INTERNAL
PART
DATA

VENDOR
PART
DATA

PRODUCT

PRODUCT

PRODUCT
DEFINITION
FORMATION
WITH SPECIFIED
SOURCE

PRODUCT
DEFINITION
FORMATION
WITH SPECIFIED
SOURCE

PRODUCT
DEFINITION

SUPPLIED
PART
RELATIONSHIP

PRODUCT
DEFINITION

RELATING
PRODUCT
DEFINITION

RELATED
PRODUCT
DEFINITION

In AP 203, all of the above relationships are supported through the **supplied_part_relationship**. This relationship is used for the identification of "part_supplier"s and "design_supplier"s. The identification of "design_supplier" is actually redundant as this information can be obtained from the **person_and_organization** related to the **product_definition_formation_with_specified_source** in the role of "design_supplier". This document will only address the use of **supplied_part_relationship**s for renumbering of parts. The structure of a **supplied_part_relationship** is shown in **Figure 6**.

To renumber parts through a **supplied_part_relationship**, both parts must be defined (see section 2.8.1.1 on page 12 for guidance on how to create the constructs to do this). The **supplied_part_relationship** relates the "new" part number's **product_definition** in the **relating_product_definition** attribute to the "old" part number's **product_definition** in the **related_product_definition** attribute. There are no standard data or mappings for the **name** and **description** attributes. The **id** attribute must be unique, but there is, again, no standard mapping.

Certification of suppliers can be indicated through a **supplied_part_relationship**. This is accomplished by relating a **cc_design_certification** to the **supplied_part_relationship** which

relates a **certification** to the relationship. There are no standard mappings for the values of the **name** and **purpose** attributes for the **certification** entity. In AP 203, the **certification_type description** attribute is restricted in value to either "part_supplier" or "design_supplier".

If a **certification** is used, AP 203 requires that the **certification** be related to an **approval**. See section 2.4 on page 6 for guidance on creating the **approval** and related entities. It is further required that the **certification** be associated with a **date_and_time** in the role of "certification_date". See section 2.3 on page 5 for guidance on creating dates and time in AP 203.

> NOTE - This relationship is a type of **product_definition_relationship** and as such may have specifications related to it. The relationship of specifications to **product_definition_relationship**s is explained in section 2.8.11 on page 36 for the specific case of a **next_assembly_usage_occurrence**.

Pre-processor Recommendations: It may be difficult to obtain the data for the **certification**'s **approval** and "certification_date". Where this data is not immediately available, it can be extrapolated from the **approval** related to the **product_definition_formation_with_specified_source** found on the path referenced by the **relating_product_definition** attribute.

There are no standard mappings for the **name** and **description** attributes in a **supplied_part_relationship**. Since there are no standard mappings in the AP 203 application domain for these attributes, it is recommended that these attributes contain a null string as minimal content or any appropriate/ mutually agreed upon string. The **id** attribute must be constructed so as not to duplicate any assignments made to other entities which are sub-types of **product_definition_relationship.**

There are no standard mappings for the **name** and **purpose** attributes in a **certification**. Since there are no standard mappings in the AP 203 application domain for these attributes, it is recommended that these attributes contain a null string as minimal content or any appropriate/ mutually agreed upon string.

Post-processor Recommendations: Since there are no standard mappings for the **name** and **description** attributes for a **supplied_part_relationship**, it is recommended that post-processors not assign any processing significance to these values.

Since there are no standard mappings for the **name** and **purpose** attributes for a **certification**, it is recommended that post-processors not assign any processing significance to these values.

2.8.7 Alternate Parts

AP 203 designates alternate and substitute parts differently. Alternate parts are interchangeable in all occurrences whereas substitutes (see section 2.8.9.3 on page 33) are interchangeable only in a

particular usage. Alternate parts in AP 203 are defined through the **alternate_product_relationship** entity. This relationship is used in the definition of parts list data for alternate item designations. There are no standard mappings to the **name** and **description** attributes of this entity. The **basis** attribute should contain a rationale for the interchange (e.g. any use, first available, etc.).

Pre-processor Recommendations: There are no standard mappings for the **name** and **description** attributes of **alternate_product_relationship**. Since there are no standard mappings in the AP 203 application domain for these attributes, it is recommended that these attributes contain a null string as minimal content or any appropriate/ mutually agreed upon string.

Post-processor Recommendations: Since there are no standard mappings for the **name** and **description** attributes of **alternate_product_relationship**, it is recommended that post-processors not assign any processing significance to these values.

2.8.8 Make From Relationships

In AP 203, the fact that a part or the design for a part is made from another part or the design for another part is indicated by the **design_make_from_relationship**. To indicate either of the above, both parts must be defined (see section 2.8.1.1 on page 12 for guidance on how to create the constructs to do this). The **design_make_from_relationship** relates the source part number's **product_definition** in the **relating_product_definition** attribute to the resultant part number's **product_definition** in the **related_product_definition** attribute. The **id** attribute must be unique, but there is no standard mapping. There is no standard mapping for the **name** attribute. The **description** attribute should be set to whatever data is shown on the parts list of the resultant part as its material specification, if any. **Figure 7** shows this relationship pictorially.

> NOTE - This relationship is a type of **product_definition_relationship** and as such may have specifications related to it. The relationship of specifications to **product_definition_relationship**s is explained in section 2.8.11 on page 36 for the specific case of a **next_assembly_usage_occurrence**.

PRODUCT

PRODUCT

PRODUCT
DEFINITION
FORMATION
WITH SPECIFIED
SOURCE

PRODUCT
DEFINITION
FORMATION
WITH SPECIFIED
SOURCE

PRODUCT
DEFINITION

DESIGN
MAKE FROM
RELATIONSHIP

PRODUCT
DEFINITION

RELATING
PRODUCT
DEFINITION

RELATED
PRODUCT
DEFINITION

SOURCE
PART

RESULT
PART
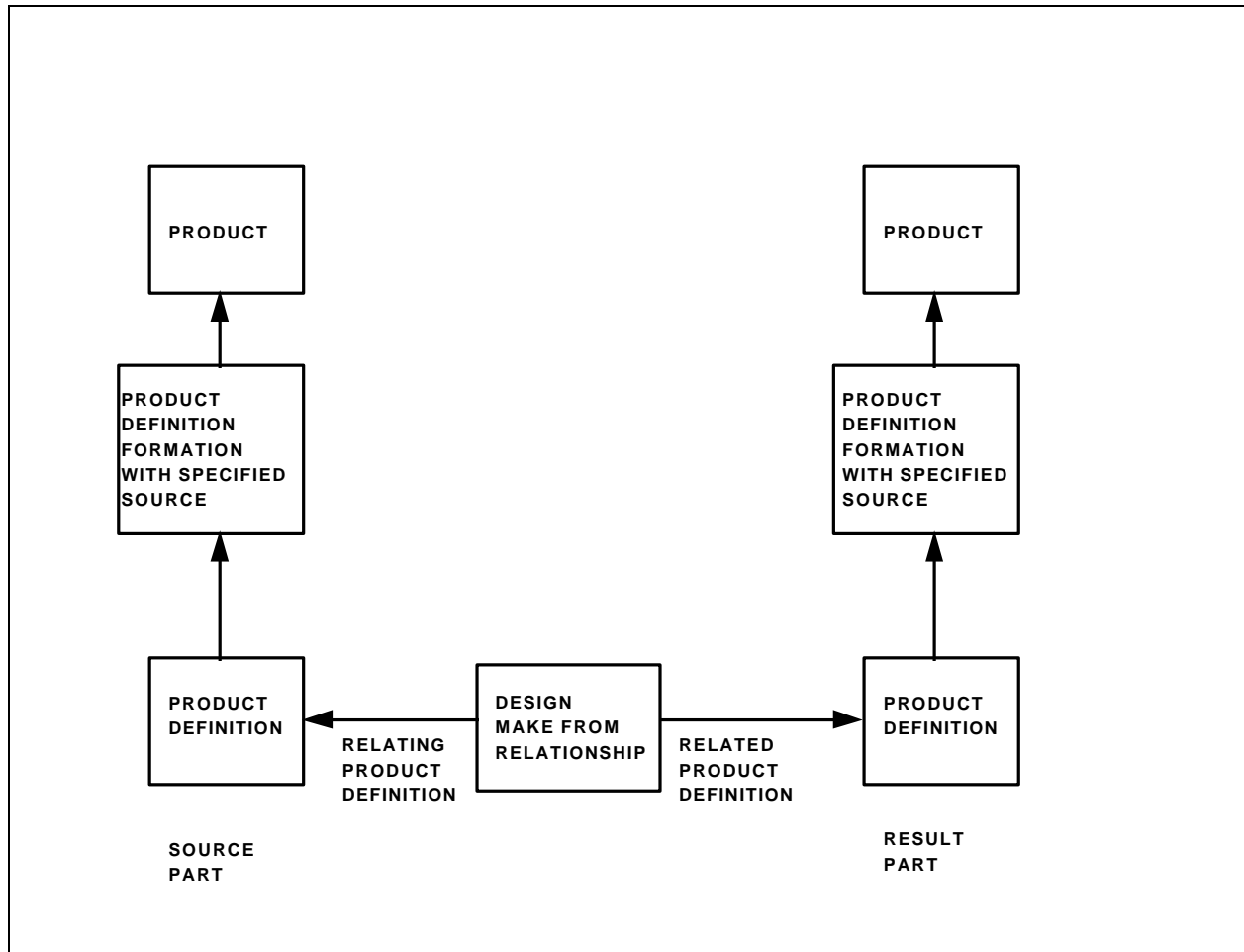
Pre-processor Recommendations: There is no standard value for the **name** attribute in a **design_make_from_relationship**. Since there is no standard mapping in the AP 203 application domain for this attribute, it is recommended that this attribute contain a null string as minimal content or any appropriate/ mutually agreed upon string. The **id** attribute must be constructed so as not to duplicate any assignments made to other entities which are sub-types of **product_definition_relationship.**

Post-processor Recommendations: Since there is no standard value for the **name** attribute for a **design_make_from_relationship**, it is recommended that post-processors not assign any processing significance to this value.

2.8.9 Assembling Parts

In AP 203, assemblies are defined as parts in the same way in which all parts are defined. The major difference is that an assembly has other parts related to it which represent the detail parts and sub-assemblies which comprise the assembly. This relationship of an assembly part to its components is defined through a **next_assembly_usage_occurrence** in AP 203. The structure of

this relationship is shown in **Figure 8**.
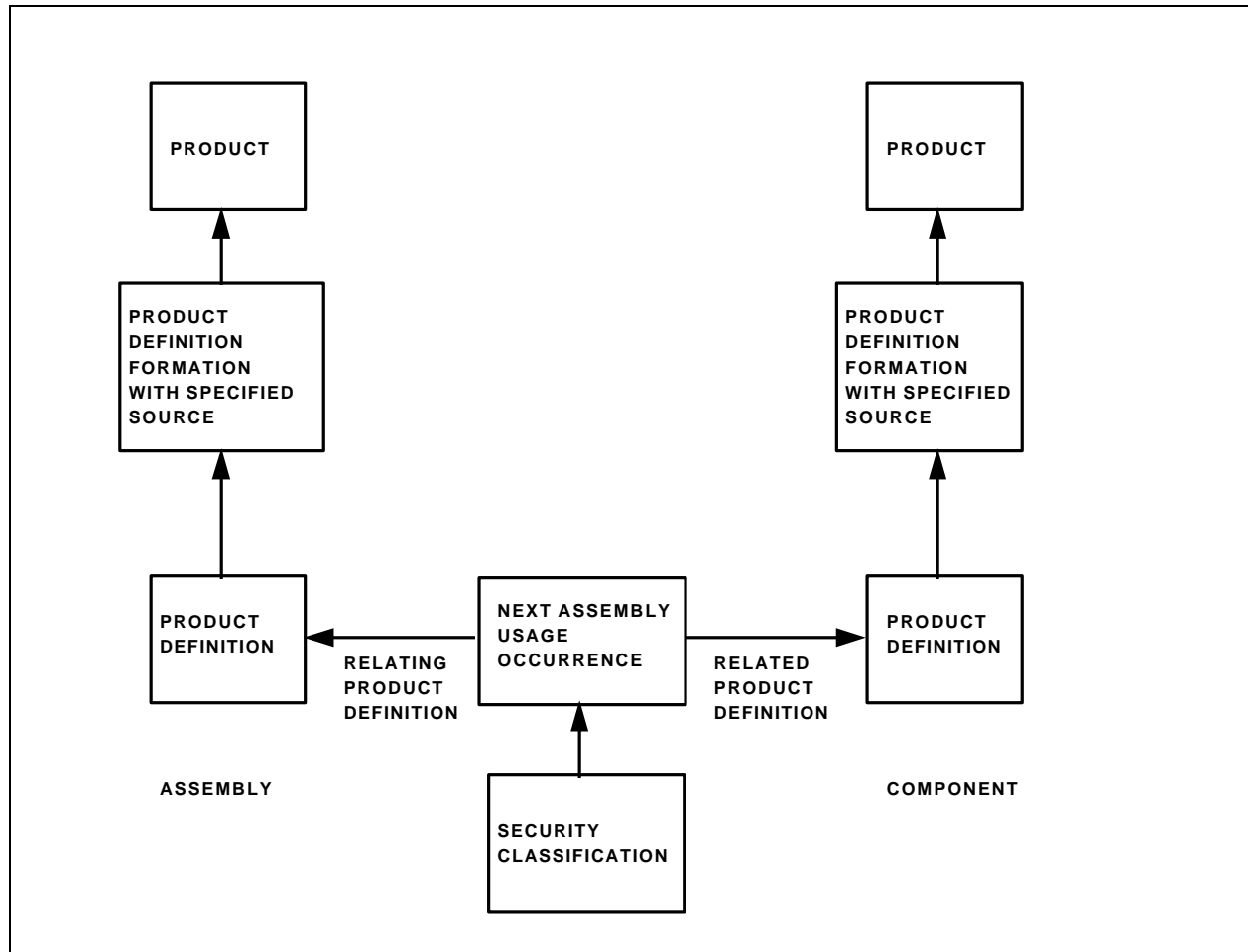
NOTE - Processors may use a version **id** of "ANY" where they wish to indicate a generic revision of a part when the part is a component in an assembly. This indicates that any existing revision of the component is valid for the assembly. This type of instancing reduces the amount of data to be sent in change packages. When this is used, it reduces the ability to track the actual contents of parts lists at a particular change level when the organization versions parts.

The **next_assembly_usage_occurrence id** attribute has no standard mapping, but must be unique for all instances of the entity. The **name** attribute should contain the item/find number from the parts list which identifies the usage. It is recommended that the **description** attribute contain an instance identifier for the usage, if one exists or contain a null string as minimal content. The **reference_designator** attribute is optional and when present designates a unique positional location.

It should be noted that since the usage is described by a **product_definition_relationship**, many different views of the usage can be established by varying the **relating_product_definition**. AP 203 can maintain one usage based on the "design" **product_definition** and another based on the "manufacturing" **product_definition**. The various **product_definition**s can move into other life cycle stages for the product as well. In this way, usages or parts lists can be defined for any of a number of views and life cycle stages of a design. See section 2.8.1.3 on page 14 for recommended values of the **product_definition id** attribute to support concurrent engineering during a particular life cycle stage.

AP 203 requires that all **assembly_component_usage**s and therefore all **next_assembly_usage_occurrence** entities be associated with a **security_classification**. For guidance on creating the entities associated with a **security_classification**, see section 2.5 on page 7.

Pre-processor Recommendations: The **id** attribute of the **next_assembly_usage_occurrence** must be constructed so as not to duplicate any assignments made to other entities which are sub-types of **product_definition_relationship.**

The **security_classification** classification officer, classification date, approvers and approval dates can be extrapolated from the version creator and approval data for the assembly part if no appropriate data is available.

2.8.9.1 Instances in Multi-Level Assemblies

AP 203 provides ability to identify individual occurrences of component in an multi-level assembly. This provides the ability to assign to each occurrence an identifier, a position in the assembly and, possibly, a geometrical representation that may be different from the one assigned to the product_definition of the component (for example, if a component is a flexible pipe, the geometrical representation assigned to the **product_definition** of the component may be a I shaped tube while the geometrical representation assigned to the occurrence of pipe positioned in the assembly may be an S shape constrained by the environment).

In order to distinguish the occurrences of component in an assembly of more than two hierarchical levels, the **specified_higher_usage_occurrence** entity is used. For example, in the case of the table, an instance of **specified_higher_usage_occurrence** with **description** attribute set to "Cap1" will correspond to the occurrence 1 of the cap. The **upper_usage** attribute of the **specified_higher_usage_occurrence** will identify that "Cap1" is a component mounted on the leg occurrence #1 which is used on the table (see figures 9 and 10).

NOTE - These figures and practice are courtesy of GOSET

2.8.9.2 Quantities in Assemblies

AP 203 provides for designating quantities of components in next assemblies and higher assemblies. The most common types of quantities are next assembly quantity and quantity for an end item. A next assembly quantity is the amount (count or other measure) of a part in its immediate parent part. The quantity for an end item is the amount (count or other measure) of a part in a finished manufactured item. The end item itself is designated by the organization and may be a configuration item (see 2.8.14 on page 42). These two types of quantity and their related data is typically what comprises the body of an application list.

2.8.9.2.1 Next Assembly Quantity

AP 203 provides two methods for specifying next assembly quantity. One method is to count the number of **next_assembly_usage_occurrence**s where the pair of the **relating_product_definition** and **related_product_definition** attributes are identical among multiple instances of the **next_assembly_usage_occurrence** entity. This type of quantity specification can only be used for items which are counted one piece at a time as there can be no unit of measure attached to this type of quantity. This method is extremely valuable where all instances of a component are specified geometrically as well as in the product structure.

The other method of specifying next assembly quantity in AP 203 is by creating a complex instance involving both **next_assembly_usage_occurrence** and **quantified_assembly_component_usage**. The quantity is explicitly stated in the **measure_with_unit** related to the **quantified_assembly_component_usage**.

> NOTE - Since these constructs are sub-types of **assembly_component_usage**, they will require a **security_classification**. See section 2.5 on page 7 for guidance on creating these constructs and section 2.8.9 on page 27 for processor recommendations.

Quantity designations are used on parts lists for **product**s. The AP 203 data structure is quite capable of providing the data for the body of a parts list. The information for each record in this list is generated for an assembly by obtaining the data for the **product**s related to it through **next_assembly_usage_occurrence**s. For a make from part, the same rationale is applied to the **design_make_from_relationship** with the resultant part from the make from also being called out. For a material callout, the parts list is determined from the **material_specification**s related to its **product_definition** unless the bulk material is assigned a part number internally by the organization or a quantity unit of measure other than a simple count is needed. If a bulk material is assigned an internal part number by an organization or a unit of measure other than a simple count is needed, the usage of the material becomes a **next_assembly_usage_occurrence** between the two.

2.8.9.2.2 End Item Quantity

End Item Quantity is the total quantity of a component in either the entire delivered unit or some

major subsection of a delivered unit. This quantity is designated in AP 203 by establishing a complex instance of **promissory_usage_occurrence** and **quantified_assembly_component_usage**. The quantity in the **measure_with_unit** related to the **quantified_assembly_component_usage** is the quantity of the part in the final article. This relationship is described pictorially in **Figure 11**.
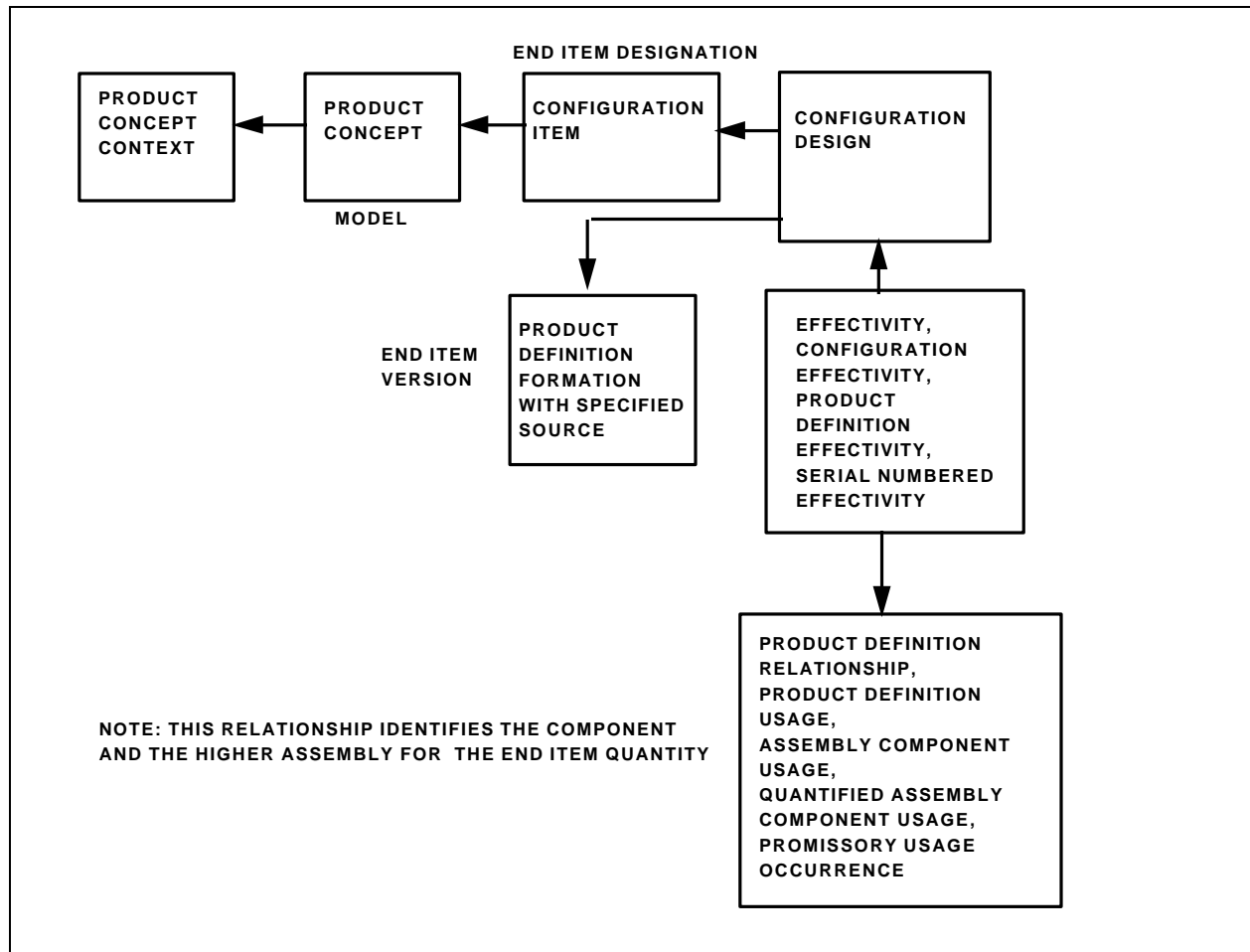
It should be noted that this could be a simple direct relationship or a more complex relationship. In the simple instance, the **relating_product_definition** will point to the **product_definition** of the **product** which is designated as the end item. In this case, the quantity is the total for the component (specified by the **related_product_definition**) in the end item for the indicated effectivity.   In the more complex instance, the **relating_product_definition** will point a **product_definition** of a higher assembly which is not the end item. In this case, the quantity is for the component (specified by the **related_product_definition**) in the assembly (specified by the **relating_product_definition**) as the assembly is used in the end item for the indicated effectivity.

Since **promissory_usage_occurrence** is a sub-type of **assembly_component_usage**, it will require a **security_classification**. See section 2.5 on page 7 for guidance on creating the **security_classification** constructs and section 2.8.9 on page 27 for processor recommendations for sub-types of **product_definition_relationship**. This relationship is a type of **product_definition_relationship** and as such may have specifications related to it. The relationship of specifications to **product_definition_relationship**s is explained in section 2.8.11 on page 36 for the specific case of a **next_assembly_usage_occurrence**.

> NOTE - The relationship described here is intended for part based systems and has been show to be problematic for drawing based systems. See appendix F section F.1.2 on page 76 for an alternative user defined entity which can be used as a migration technique in the interim.

2.8.9.3 Substituting Parts in Assemblies

AP 203 designates alternate and substitute parts differently. Alternate parts (described in section 2.8.7 on page 26) are interchangeable in all occurrences where as substitutes are interchangeable only in a particular usage. A substitute part is designated through the use of an **assembly_component_usage_substitute**. This relationship is used to define information which is represented on a parts list. There are no standard mappings for the **name** and **description** attributes in an **assembly_component_usage_substitute**. The **base** attribute points to the normal or preferred usage. The **substitute** attribute points to the surrogate usage.

> NOTE - AP 203 does not currently support substitutes in make from relationships. This is an issue against the current standard. See appendix F section F.1.4 on page 77 for the description of a user defined entity to be used to overcome this deficiency.

Pre-processor Recommendations: There are no standard mappings for the **name** and **description** attributes in an **assembly_component_usage_substitute**. Since there are no standard mappings in the AP 203 application domain for these attributes, it is recommended that these attributes contain a null string as minimal content or any appropriate/ mutually agreed upon string.

Post-processor Recommendations: Since there are no standard mappings for the **name** and **description** attributes in an **assembly_component_usage_substitute**, it is recommended that post-processors not assign any processing significance to these values.

2.8.10 Assemblies and Shape

The shape of an assembly is most often derived from the shape of its components. AP 203 provides three methods for dealing with the shape of an assembly. These are: replicating the shape of the components in the shape of the assembly, mapping the shape of the component into the shape of the assembly, and referencing the shape of a component. The following subsections

will deal with each of these methods.

2.8.10.1 Replicated Shape

One method for representing the shape of an assembly is to collect together all the elements of all the shapes of all the components explicitly in the shape of the assembly. This is the typical practice used in industry today, but it is highly inefficient. Using this method, the shape entities of the components of an assembly become shape entities in the assembly. The assembly shape becomes an conglomerate with no segregations of the various component shapes. In AP 203, this is represented by collecting all the geometry and topology of the components in an appropriate sub-type of **shape_representation**.

2.8.10.2 Mapped Shape

ASSEMBLY   COMPONENT

PRODUCT DEFINITION	NEXT ASSEMBLY USAGE OCCURRENCE	PRODUCT DEFINITION

RELATING PRODUCT DEFINITION   RELATED PRODUCT DEFINITION

PRODUCT DEFINITION SHAPE   PRODUCT DEFINITION SHAPE   PRODUCT DEFINITION SHAPE

SHAPE DEFINITION REPRESENTATION   SHAPE DEFINITION REPRESENTATION   SHAPE DEFINITION REPRESENTATION

SHAPE REPRESENTATION   SHAPE REPRESENTATION   SHAPE REPRESENTATION

REPRESENTATION ITEM   REPRESENTATION MAP   REPRESENTATION ITEM

MAPPED ITEM

In this method, the shape of the component is mapped into the shape of the assembly. This is done through the use of the **mapped_item** entity which can be used to map one **shape_representation** into another. (See G.1.15 on page 84 for an implementors agreement that affects this entity.) This method is the most efficient and versatile in AP 203. This method may only be used where the component and the assembly shapes are of the same type of **shape_representation.**

This entity is used for transformation without scaling on the component shape. Transformation without scaling is accomplished by relating an **axis2_placement** in the component identified as the **mapping_origin** in the **representation_map** to an **axis2_placement** in the assembly identified as the **mapping_target** in the **mapped_item.**

2.8.10.3 Referenced Shape

In this method, the shape of the component is referenced or related to the shape of the assembly. This is done through the use of the **context_dependent_shape_representation** entity.   This method is not dependent on both the component and assembly shapes being the same type of **shape_representation**.
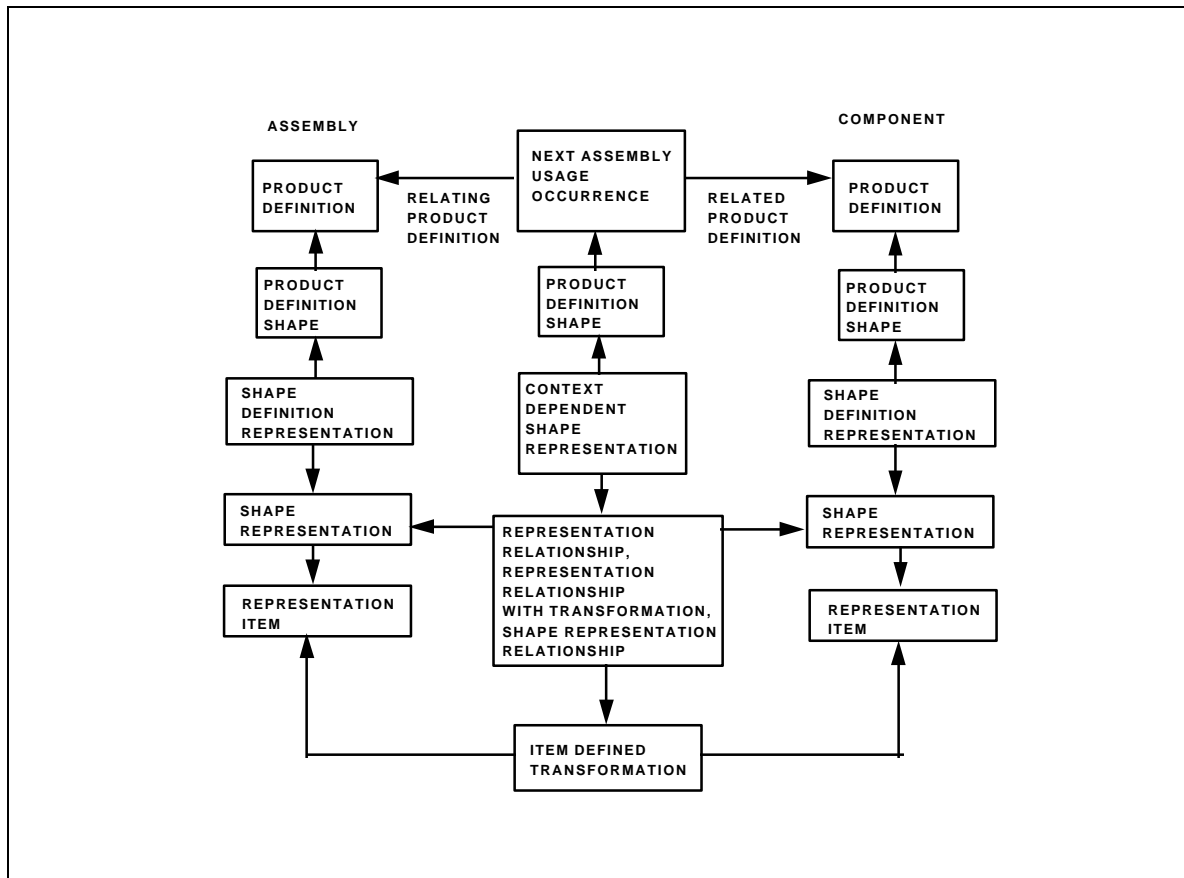
In this method, a complex instance of **representation_relationship**, **representation_relationship_with_transformation**, and **shape_representation_relationship** relates the **shape_representation**s for the component and assembly together and relates an **item_defined_transformation** to the relationship. The **context_dependent_shape_representation** entity relates the complex relationship with the transformation to a **product_definition_shape** which is related to the

next_assembly_usage_occurrence. This is described pictorially in **Figure 13**.

2.8.10.4 Component Shape Aspects and Assembly Shape

When a component part in an assembly has **shape_aspect**s in its **shape_representation**, the aspect **shape_representation** should be brought into the assembly through **context_dependent_shape_representation**. The reason for this is that the aspect is only in the assembly by virtue of its relationship to the component which is also in the assembly. If the component is removed from the assembly, the aspect representation should be removed as well. If the aspect needs to be altered, it should be altered in the context of the component not in the context of the assembly.

2.8.11 Assemblies and Specifications

The reference of a specification to an assembly itself without respect to any particular component in the assembly is done in the same manner it is for parts (see section 2.8.3 on page 18). An assembly is peculiar since specifications may need to be related to the usage of a particular component in an assembly or the interface between the component and the assembly. AP 203 relates specifications to assembly/ component relationships by relating a **cc_design_specification_reference** entity to a **shape_aspect** which references a **product_definition_shape** that is related to the **next_assembly_usage_occurrence** of the part. The shape related to the **shape_aspect** describes the actual area of application for the specification. This relationship is shown pictorially in **Figure 14**.

It should be noted that in a conformance class 1 AP 203 implementation there is no shape data to specify the actual area the specification is applicable to. In this type of implementation, a **document_usage_constraint** should be used to clarify the application of the specification. This entity is explained later in this section.

The **cc_design_specification_reference** entity identifies the owner of the specification through the **source** attribute. This attribute should contain an unambiguous identification of where the receiver of the data could obtain a copy of the document. The **document** related to the **cc_design_specification_reference** must be uniquely identified in the exchange by the **id** attribute. This means that the **id** should contain any revision information needed to identify the document completely. The **name** attribute should contain the title of the document. The **description** attribute should contain an expanded explanation of the **document**'s contents.

ASSEMBLY

COMPONENT

PRODUCT DEFINITION

PRODUCT DEFINITION

NEXT ASSEMBLY USAGE OCCURRENCE

DOCUMENT

DOCUMENT USAGE CONSTRAINT

PRODUCT DEFINITION SHAPE

CC DESIGN SPECIFICATION REFERENCE

SHAPE ASPECT

PROPERTY DEFINITION

NOTE: DASHED LINES INDICATE ENTITY IS NOT REQUIRED TO EXIST BY AP 203

SHAPE REPRESENTATION

SHAPE DEFINITION REPRESENTATION

Since many specifications cover a variety of subtopics and options on a given topic, it may be necessary to identify a particular subtopic of the specification and assign option values. In AP 203 this is accomplished by relating a **document_usage_constraint** to the document. The **subject_element** attribute identifies the particular section or topic being referenced in the specification. The **subject_element_value** identifies any option choices or restrictions placed on the section or subtopic.

The above should <u>not</u> be used to reference classes defined in specifications such as process specifications. This should be done by using the **document** entity sub-type **document_with_class**. If classed documents require further restriction of the class, a **document_usage_constraint** may be related to the **document_with_class** entity.

Pre-processor Recommendations: The IS edition of AP 203 does not list **shape_aspect** in the entity list for conformance class 1. This was an omission. **Shape_aspect** <u>should</u> be supported in

conformance class 1 for qualification of specifications.

Post-processor Recommendations: Post-processors should store all data found in specification **document**s attached to **shape_aspect**s. If it is not possible to store all the data, the user must be informed of the data being omitted and its relationship to the **shape_aspect**.

2.8.12 Engineering Release/ Change Data

AP 203 provides data structures for representation of the data used in the engineering release and change process. The structures are based on a request and action process where a request is established documenting the need for a potential release or change which may or may not ever be incorporated. If the request is incorporated, it is done through some action being taken on the request which results in either a new release of a design or a change to a existing design.

It should be noted that these constructs have been designed to represent <u>all</u> request and incorporation structures in the AP 203 application domain. All release and change proposals and requests (Engineering Change Proposals, Requests for Engineering Action, etc.) are represented by the request portion of the structure. All release and change incorporations are represented by the action portion of the structure. Differentiation between types of requests and actions can be done structurally based on the guidance in this section, by its identification (**id** for requests, **name** for actions), or by the originator. Differentiation by identification or originator is very process dependent but can be necessary particularly for preliminary requests and proposals.

Some types of releases and changes in organizations may not involve a two step process. In this case, both data structures are implemented simultaneously and reference the same release or change documentation. Since these constructs in AP 203 are intended to support many different release and change processes/documentation, in some cases, some of the required data may not exist.

The release process is initiated through an AP 203 **versioned_action_request** which is related to the design being released through a **start_request**. The **versioned_action_request** has a related **action_method**. In this case, both the **versioned_action_request** and the **action_method** would indicate that the respective **purpose**s were to initially release the design or create the design for the initial release. This request process is followed (in the data) by an **action_directive** which is related to the design to be released through a **start_work**. The **action_directive** also identifies the **start_request** as the request being satisfied/ incorporated. A **directed_action** relates the **action_method** to the **action_directive** which in the case of initial release may be moot. The structure of these relationships (at a high level) is shown in **Figure 15**.
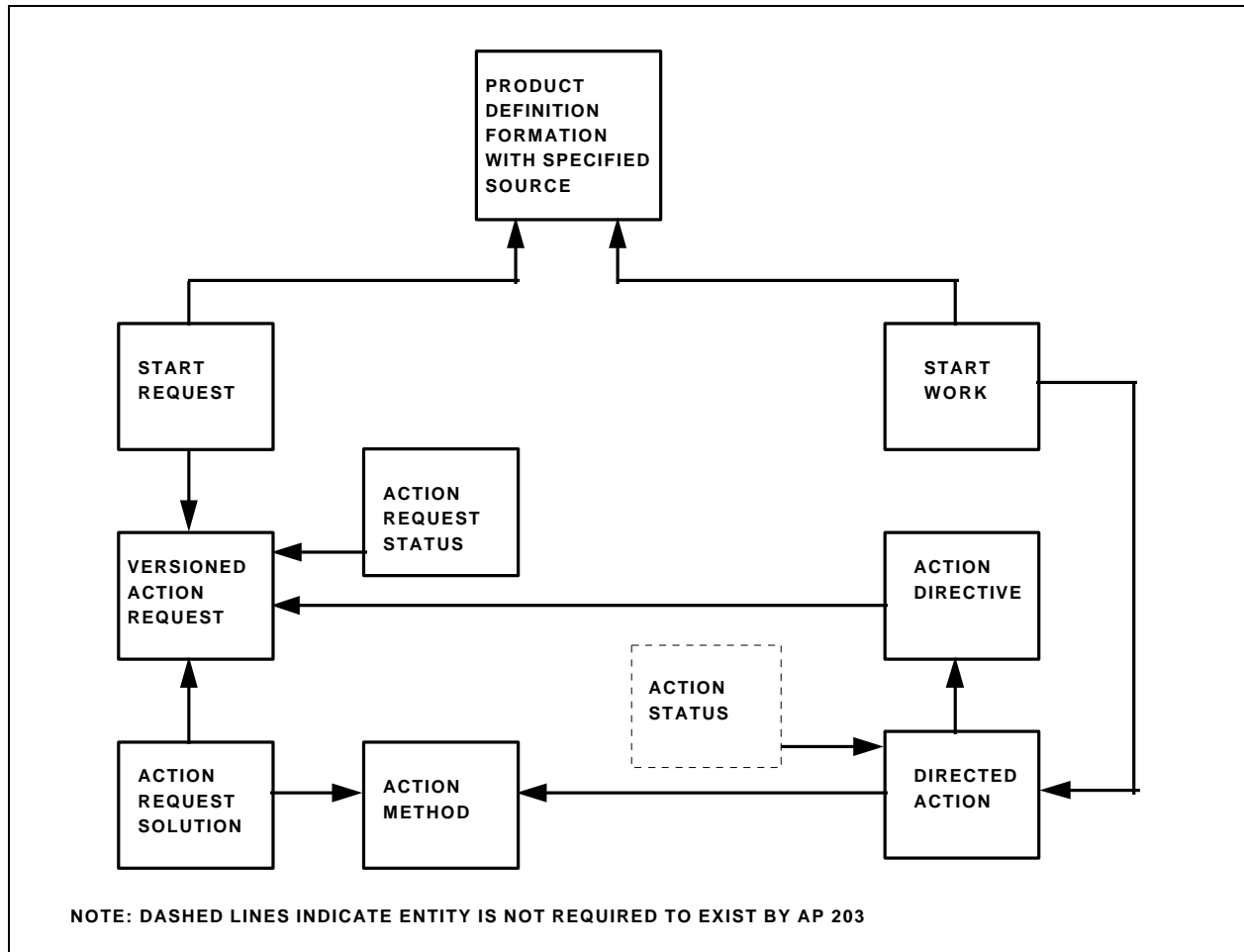
```
                    ┌─────────────────┐
                    │ PRODUCT         │
                    │ DEFINITION      │
                    │ FORMATION       │
                    │ WITH SPECIFIED  │
                    │ SOURCE          │
                    └─────────────────┘
```

NOTE: DASHED LINES INDICATE ENTITY IS NOT REQUIRED TO EXIST BY AP 203

The change process is initiated through an AP 203 **versioned_action_request**, as well, which is related to the design proposed to be changed through a **change_request**. The **versioned_action_request** has a related **action_method**. In this case, there may be many **action_method**s or ways to solve the problem. This request process is followed (in the data) by an **action_directive** which is related to the new design or version to be released through a **change**. The **action_directive** also identifies the **change_request**(s) as the request(s) being satisfied/ incorporated. A **directed_action** relates the **action_method** to the **action_directive** indicating which of possibly many methods for the request or requests incorporated was chosen. The structure of these relationships (at a high level) is shown in **Figure 15**.

2.8.12.1 Requests for Release/Change

Requests for release or change are created in AP 203 by relating a **versioned_action_request** to a **product_definition_formation_with_specified_source** through a **start_request** or **change_request**. The **start_request** or **change_request** identifies through the **items** attribute the **product_definition_formation_with_specified_source** to be released or changed. In the

case of a **start_request**, this is a bit odd as this structure in AP 203 requires an identification of a version at request time which will in fact result from the request. This may be changed in future editions of the AP.

NOTE: DASHED LINES INDICATE ENTITY IS NOT REQUIRED TO EXIST BY AP 203
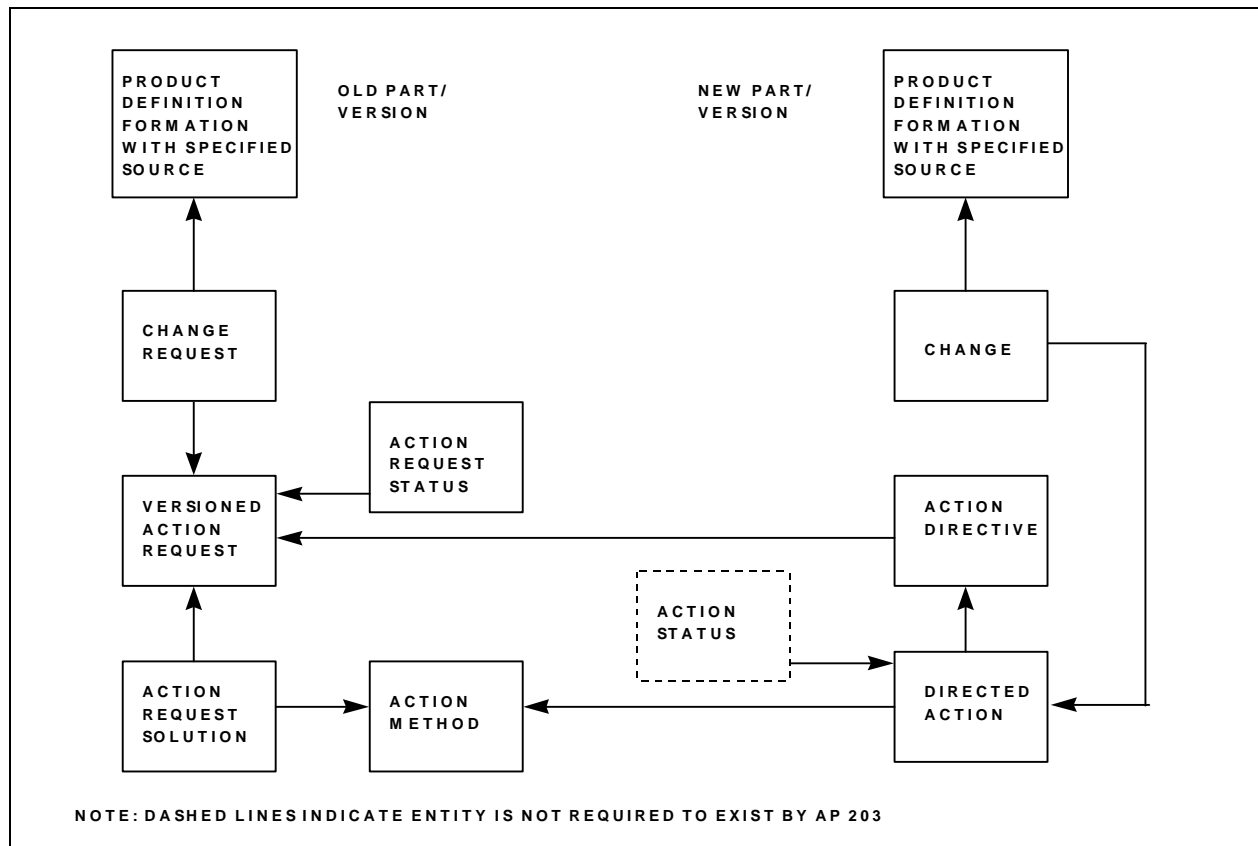
The **versioned_action_request id** attribute contains the identification of the request. This information is the document or request number. The **version** attribute is the version of the request itself. This attribute is used to identify actual versioning of the request or reissues of the request. The **purpose** attribute should contain text identifying the end result anticipated from this version of this request. The **description** attribute should contain a general description of the request. In AP 203, a **versioned_action_request** is required to have an associated **action_request_status**. The AP restricts the values for the **status** attribute to "proposed", "in_work", "issued", or "hold".

A request for release or change may have many possible ways it can be resolved. This is more common for changes than releases, but the AP 203 data structure supports the documentation of the engineering thought process gone through in either case. This is accomplished through a combination of the **action_request_solution** and **action_method** entities. **Action_request_solution** relates an **action_method** to a **versioned_action_request**. The **action_method name** attribute should contain a reference to any formal documentation for a

proposed solution to the release/ change request. The **description** attribute should contain a detailed description of the method through which the request is to be satisfied. The **consequence** attribute should contain any determined or perceived consequence to using this method to satisfy this request. The **purpose** attribute should contain the intention of the method as a single method may be used to satisfy many requests.

AP 203 requires that a **start_request** and a **change_request** have a related **approval**. As these requests normally have a number of signatories, there should be no problem obtaining this data if it is stored in electronic form. For guidance in creating the **approval** constructs, see section 2.4 on page 6 of this document. A **start_request** or **change_request** is required to be associated with a date and time in the role of "request_date" which indicates when the request was created. For guidance in creating the date and time constructs, see section 2.3 on page 5 of this document. Lastly, a **start_request** or **change_request** is required to be associated with at least one person and organization in the role of "initiator" or "request_recipient". For guidance in creating the person and organization constructs, see section 2.2 on page 4 of this document.

2.8.12.2 Release/Change Incorporation

Release of a design or change incorporation into a design is accomplished in AP 203 through the **start_work** and **change** constructs which relate an **action_directive** to the new design or new design version by pointing to the **product_definition_formation_with_specified_source** which results from the release or change. A **directed_action** related to the **action_directive** identifies the **action_method** actually used to satisfy the requests related to the **action_directive**. In the case where many requests are being incorporated, there many be many **directed_action**s to indicate the appropriate methods.

The **action_directive name** attribute is the identification of the formal documentation to incorporate the change or release the design. In cases where there is no second set of paper work or documentation (i.e. there is a one to one correspondence between **versioned_action_request** and **action_directive**), the **action_directive name** value is the same as the **versioned_action_request id** value. The **description** attribute should contain a phrase or group of phrases indicating the final result of the release or change. The **analysis** attribute should identify any investigative results which support the release or change. Likewise, the **comment** attribute should contain any textual commentary which supports the release or change. An **action_directive** may be associated with an **action_status** which serves the same function as **action_request_status** in the previous request section. It is not required, in AP 203, that the **action_directive** be related to an **action_status** as the two sets of data may represent one or two documents.
The **directed_action name** attribute should contain the identification of the formal documentation as to why the method identified was chosen. The **description** attribute should contain a textual description, either in summary or detail, supporting the chosen method.

In AP 203, a **start_work** or **change** is required to have an associated **approval**. If these

42

constructs are representing the same document, they could share the **approval**. For guidance in creating the **approval** constructs, see section 2.4 on page 6 of this document. A **start_work** or **change** is required to have a date and time associated with it in the role of "start_date" which is when the work to satisfy the request or requests began. Once completed, a **start_work** or **change** may have a date and time associated with it in the role of "release_date". For guidance on creating these date constructs, see section 2.3 on page 5 of this document.

2.8.13 Release/Change Reissues

Engineering release and changes are reissued. This can be caused by an error or omission in the change package. It may also be done to signify changes in effectivity which have no effect on the version of the part.

AP 203 supports the reissue of releases and changes. To reissue a release or a change, a **versioned_action_request** is created with an **id** attribute value equal to the **action_directive name** being reissued. The **versioned_action_request version** attribute contains the reissue identifier. This new **versioned_action_request** is added to the set of **requests** in the original **action_directive** that was issued.

2.8.14 Configuration Identification

Configuration identification in AP 203 is done through the **configuration_item** entity. This entity identifies **product**s as end items or items which are sold or delivered. As in industry, this designation can be applied to full systems or spares (which are also referred to as the lowest level replaceable units).

The **configuration_item id** attribute is a unique identification of the item which may be a part number but more probably a moniker. The **name** attribute is a short description of the item. The **description** attribute is optional and would be the expanded name or description of the item. The **purpose** attribute is also optional and would contain a description of the item's intended use.

A **configuration_item** is related to a **product_concept**. The **product_concept id** attribute is more commonly known as the model designation. The **product_concept** taken together with the **configuration_item** describe a model series or configured production run. The **name** attribute is a short description of the model. The **description** attribute is the expanded name or description of the model. The **product_concept** is related to a **product_concept_context** where the **market_segment_type** attribute identifies what customer or group of customers provided the requirements for the model.
AP 203 requires that a **configuration_item** have an **approval**. For guidance on creating the **approval** constructs, see section 2.4 on page 6. A **configuration_item** must also be associated with a **person_and_organization** in the role of "configuration_manager". For guidance on creating these constructs, see section 2.2 on page 4.

43

Pre-processor Recommendations: In some cases, it may be difficult to determine the **approval** and "configuration_manager" for a **configuration_item**. If the item has effectivity (see next section), this information may be extrapolated from the **approval** and "creator" information for the **product_definition_formation_with_specified_source** for that **product**. If not, this information should be obtained from user input or a default based on the **configuration_item id** attribute.

2.8.15 Effectivity

Effectivity is the designation that something or a relationship between two things is used or planned to be used in some **configuration_item**. In AP 203, effectivity is designated on relationships between **product_definition**s by either ranges of serial numbers, ranges of dates or a lot. This is accomplished through a complex instance of the entities **effectivity**, **configuration_effectivity**, **product_definition_effectivity** and one of either **serial_numbered_effectivity**, **dated_effectivity** or **lot_effectivity**.

A **serial_numbered_effectivity** specifies an **effectivity_start_id** with an optional **effectivity_end_id**. If the **effectivity_end_id** does not exist, the effectivity is good for the starting serial number and all following serial numbers. A **dated_effectivity** follows the same pattern using dates rather than serial numbers. A **lot_effectivity** indicates an **effectivity_lot_id** and an **effectivity_lot_size**.

The above entities specify the effectivity identifiers. These entities are related to a **product_definition_relationship** through the **usage** attribute in the **product_definition_effectivity** entity. The **effectivity** entity **id** attribute has no standard mapping. The **configuration_effectivity** entity relates these relationships to a **configuration_design** which relates a **configuration_item** to a **product_definition_formation_with_specified_source** or design version. **Figure 15** shows this relationship pictorially for a **serial_numbered_effectivity**.

The whole relationship here can be simply stated as a range of serial numbers, dates or a lot number related to a **product_definition_formation_with_specified_source** or design version which is designated as a **configuration_item.** This does mean that all **configuration_items** must be associated to a design version in order to have **effectivity**. In AP 203, an **effectivity** requires an **approval**. This information may be difficult to find in some instances (see pre-processor recommendations for guidance. For guidance in creating the **approval** constructs, see section 2.4 on page 6.
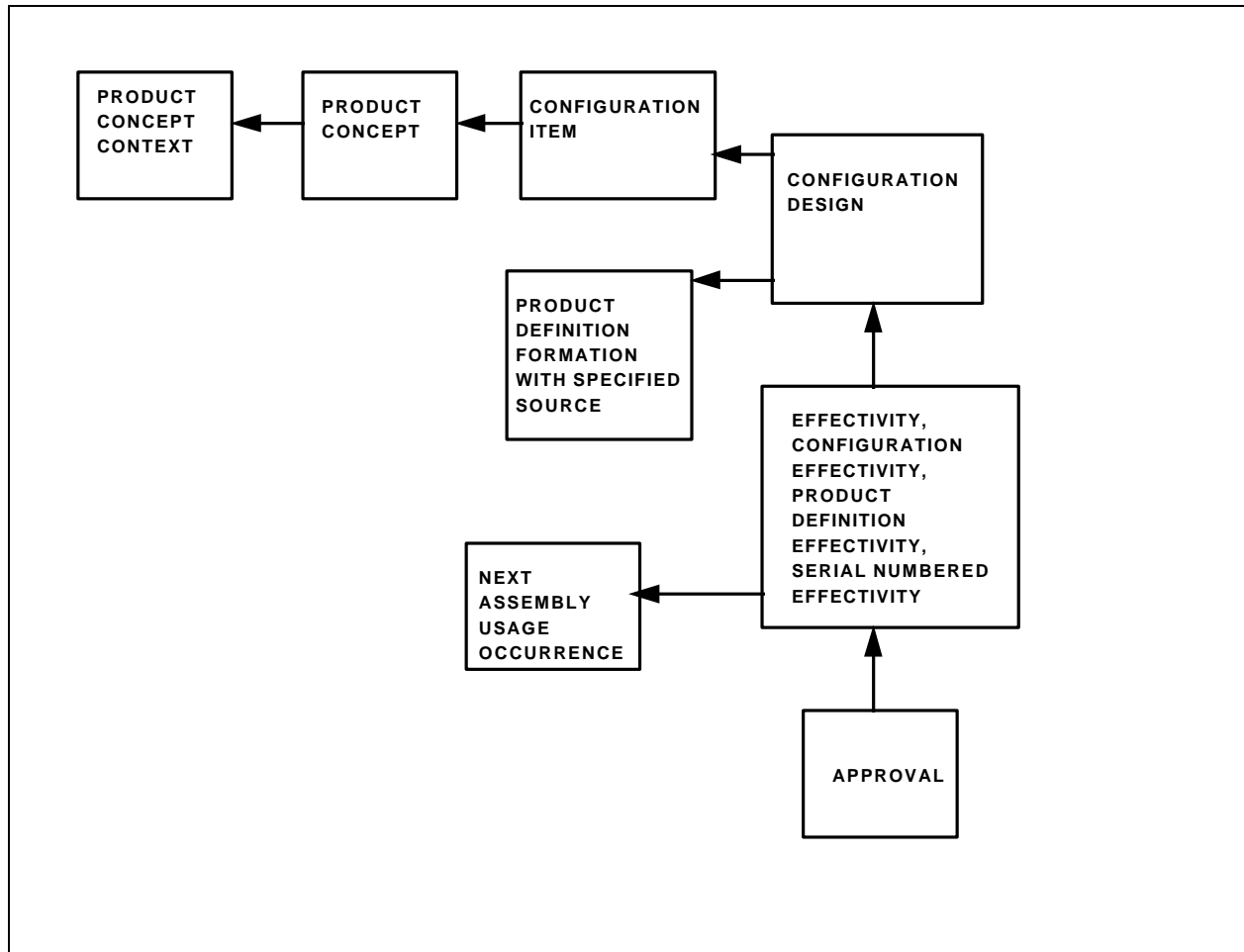
It should be noted that since the effectivity is related to a **product_definition_relationship**, many different views of the effectivity can be established by varying the **relating_product_definition**. AP 203 can maintain one effectivity based on the "design" **product_definition** and another based on the "manufacturing" **product_definition**. The various **product_definition**s can move into other life cycle stages for the design as well. In this way, effectivities can be defined for any of a number of views and life cycle stages of the design. See section 2.8.1.3 on page 14 for recommended values of the **product_definition id** attribute to support concurrent engineering during a particular life cycle stage.

The conformance classes in AP 203 do not require that effectivity relationships be instantiated. The reason for this is that there are occasions where data needs to be exchanged or shared prior to an effectivity being defined. This tends to occur early in a new design.

All effectivities in AP 203 are explicit effectivities and there are no assumed effectivities. Some systems in existence today assume a part is effective for all planned or actual instances of a product model if the effectivity is not explicitly defined. This is not the intent in AP 203. If a part

has no effectivity in the AP 203 data structures, it has <u>no</u> effectivity. If a part is effective for all instances of a product model, the data should explicitly state all the effective instances. The effectivities in AP 203 contain open ranges for serial numbers and dates to allow for open or full effectivities. Using these constructs, all that is required is a start point. If there is a desire for full effectivity and the start point is not defined, the value "1" should be used for the **serial_numbered_effectivity effectivity_start_id** or the equivalent date of January 1st year 1 should used for **dated_effectivity effectivity_start_date**.

> NOTE - Open effectivity does not make sense for a lot effectivity as it is inherently closed (other than lot size). Lot effectivity is typically an effectivity designated in the manufacturing view of a product or part.

The exchange or sharing of effectivity information creates the need for optional processing capability in at least pre-processors to allow for perspective. It is typically desirable for the lead contractor in a partnership or team to provide effectivity definitions to sub-contractors. It is usually undesirable for the lead contractor to utilize effectivities echoed back by sub-contractors as they reflect what was originally sent but not necessarily the most current data (in some cases).

The above is a simple case. Most cases involve even more variables such as who in the exchange or sharing arrangement is the defining body for the effectivity of a particular part or usage. One way to deal with this situation is for pre-processors to provide options for ignoring effectivity entirely, loading it or either ignoring or loading it based on externally defined criteria such as the part's design owner, design supplier or part number and for post-processors to provide a switch for a user choice on whether or not defined effectivity information in the system should be used in the interchange.

Pre-processor Recommendations: There is no standard mapping for the **id** attribute of the **effectivity** entity. Since there is no standard mapping in the AP 203 application domain for this attribute, it is recommended that this attribute contain a null string as minimal content or any appropriate/ mutually agreed upon string. If the effectivity approval information is not readily available, it can be extrapolated from the engineering change which designated the effectivity. Pre-processors should interpret the value "1" for the **serial_numbered_effectivity effectivity_start_id** or the equivalent date of January 1st year 1 for **dated_effectivity effectivity_start_date** as full or open effectivity when the values are specified with no ending range value. It is recommended that pre-processors provide options for ignoring effectivity entirely, loading it, or either ignoring or loading it based on externally defined criteria such as the part's design owner or part number to allow for a user choice as to whether the data is utilized or not depending on the source.

Post-processor Recommendations: There is no standard mapping for the **id** attribute of the **effectivity** entity. Since there is no standard mapping in the AP 203 application domain for this attribute, it is recommended that post-processors assign no processing significance to this value. When there is a need for full effectivity and the start point is not defined, post-processors should

use the value "1" for the **serial_numbered_effectivity effectivity_start_id** or the equivalent date of January 1st year 1 for **dated_effectivity effectivity_start_date**. It is recommended that post-processors provide a switch for a user choice on whether or not defined effectivity information in the system should be used in the interchange.

3 Drawings and Associated Lists in AP 203

Mechanical drawings are essentially two dimensional renderings of a three dimensional object with requirements for the finished object. Most drawings are intended to convey the design intent to manufacturing engineering. On defense contracts, drawings are deliverable as part of a technical data package. Drawings also have a financial connotation in that they are a cost estimating and accumulating item.

STEP currently addresses drawings through a number of proposed APs. Some of these are Explicit Draughting (AP 201), Associative Draughting (AP 202), Core Data for Automotive Mechanical Design Processes (AP 214) and Technical Data Packaging Core Information and exchange (AP 232). These APs document the requirements and STEP constructs used in the definition of data in STEP related to mechanical design drawings and technical data packages.

The TDP AP will also address the representation of list data which is associated to the drawing. Lists are typically computer generated information which, at one time, was presented on the drawing format. These listings describe information on effectivity, parts defined and used by the drawing, and references to other drawings and specifications made on the drawing just to name a few. This information was migrated off the drawing to an associated list to reduce drawing maintenance costs.

STEP AP 203 is scoped to address the configuration control of three dimensional models of mechanical parts. The parts could be either piece parts (detail parts) or assemblages of detail parts and subassemblies. This AP does not specifically address the way in which any of these parts is currently documented. This is currently done via the mechanical drawing. The IS release of AP 203 does include the ability to relate drawing documents to parts. This new capability in AP 203 now allows a part to reference a drawing. This new capability also allows for the representation of drawing trees through the use of the document_relationship entity. Unfortunately, this new capability does not allow for the identification of the revision of the drawing except by appending it to the drawing number in the **document id** attribute. It also does not allow for relating the drawing document to engineering release and change documentation. This leaves the IS release of AP 203 and the first release of STEP still falling short of supporting a significant current business process in many organizations. It is therefore recommended that this method <u>not</u> be used.

STEP is an integrated data standard. The current STEP standard would relate the drawing to the part through the part version. This is where the currently proposed APs (201, 202, 203, 214, 232, etc.) currently integrate. Once completed, the TDP AP should completely address this current void.

This section of this document proposes that nothing be changed in STEP immediately. It further proposes that drawings and associated lists be recorded in AP 203 as **product**s to close this gap in the near term. This paradigm shift allows for release and change documentation to be related

to the drawing **product**, list **product** and/ or the part **product** depending on which is effected by the change. This shift also allows for the representation of parallel drawing, list and part trees, and the creation of relationships among drawings, lists, parts and their related specifications. Subsequent sections will explore this in detail.

This section will be removed from this document or revised once STEP or some other standard has an AP to deal with this current mismatch between what is available in the standard and industry's current needs.

3.1 General

Given the premise that a drawing or list is a **product**, the data required in AP 203 is straight forward and given in the rules. A drawing/ list **product** is governed by the same rules as any other **product**. The one unique stipulation would be that a drawing or list **product** entity never have a **product_definition_shape**. The reason for this is that the 'shape' for the drawing **product** is actually contained in the AP 201 or 202 data structure for drawings in STEP. List **product**s never have any shape as they are additional information related to the shape of the part as depicted on the drawing.

The sections that follow discuss various types of drawings and lists and how they can be related to the parts depicted on or documented by them. In general, parts, drawings and lists are related using the **product_definition_relationship** entity. When this entity is used to relate drawings and parts, the **product_definition_relationship name** attribute is set to the item number (balloon, find number) on the drawing for the part and the attribute **description** is used to identify the drawing zone in which the part is shown. If multiple zones are designated, these should be separated by commas. Drawings may be related to other drawings and lists in the same manner with the exception that the **name** and **description** attributes have no values (i.e. contain a null string or some mutually agreed upon value).

For simplicity's sake and to increase the usability of the data, the **product_category** information for drawing and list **product**s should include a special intervening **product_category** entities which defines the things under it as drawings or lists. The remainder of this section will provide some rudimentary definitions for the types of drawings and lists typically encountered.

3.1.1 Mono-Detail Drawings

Mono-detail drawings are drawings where only one part is detailed or defined. This is the simplest case of the drawing to part relationship since each drawing defines one part and vice versa. In a mono-detail detail part drawing, a single part made from raw material or made from another single part made from raw material is documented on a single drawing. Mono-detail assembly drawings bring together a number of mono-detail piece parts or mono-detail sub-assemblies into a larger single assembly documented on one drawing.

3.1.2 Multi-Detail Drawings

Multi-detail drawings define the shape of many parts on one drawing. Multi-detail piece part drawings bring together a number of piece parts on a single drawing. This is somewhat odd since multi-detail usually occurs in an assembly context. Multi-detail assembly drawings show an assembly of parts with some or all of the sub-parts or multiple assemblies are defined on the same drawing.

3.1.3 Tabulated Drawings

Tabulated drawings are a very common type of multi-detail drawing. In this case, multiple parts are defined on a single drawing using the same picture, but using variables for some unique feature. For example, a drawing for a #8 sheet metal screw would show a picture of the screw. The length dimension would be shown as a variable, usually denoted by a letter, which is defined in a table. The table would show the various part numbers of the various length screws along with the appropriate value for the dimension.

3.1.4 Lists

Associated lists to drawings come in many varieties. The most common types are parts lists, notes lists, application lists, and data lists. Each of these drawing related documents has a header which identifies the list, its revision (or what revision of the drawing it is for), etc. The body of the list is a tabular listing of records with some fixed structure which may or may not use indenting of the data to indicate nesting of the list records (sometimes referred to as an indentured list). The list body may be preceded or followed by notes, revision history, and/ or reference documents.

This recommended practice will not deal with any list in detail. It will instead provide guidance on using the AP 203 structures with independently versioned lists. References to the tabular data found in particular list can be found through index entries in this document for the particular list.

3.2 Drawings in AP 203

In order to define a drawing in AP 203, the **product** entity establishes the drawing's identification (or drawing number), name (or title), and description. The **product_definition_formation_with_specified_source** entity identifies its version (or change level). The **product_definition** or **product_definition_with_associated_documents** entity is used to relate the drawing to the parts defined on it and the CAD file that contains computer model of the drawing. Appendices B and C on pages 63 and 65 contain a physical file examples of the recommendations in this section.

3.2.1 Identifying Drawings

3.2.1.1 The Product Entity

This section recommends that drawings be treated as **product**s in AP 203. The drawing number for a drawing is stored in the **id** attribute. The title of the drawing is stored in the **name** attribute. If there is an expanded title or description of the drawing this is stored in the **description** attribute. All STEP **product**s must be founded in some **product_context** which identifies the engineering discipline from which the data is viewed. See section 2.1.2 on page 2 for guidance on creating this entity.

AP 203 requires that all **product**s exist in at least one **product_category** (see section 2.8.2 on page 16). This rule (**product_requires_product_category**) will force all drawings created under this recommended practice into either the detail or assembly categories as the other possibilities make no sense for drawings.

Products in AP 203 require a **person_and_organization** in the role of "design_owner". This designation is applied to the those who drew the initial drawing. More simply, this is the person and organization typically identified at the top of the title block on the drawing. See section 2.2 on page 4 for guidance on how to create the person and organization entities.

The reader is referred to section 2.8.1.1 on page 12 for processor recommendations and general guidance in creating the **product** entity and its related constructs as they are the same as the guidance for parts.

3.2.1.2 The Product Definition Formation Entity

The **product_definition_formation_with_specified_source id** attribute should contain the value which represents the version or revision for the drawing. The **description** attribute should contain the reason for the creation of the version. The **source** attribute must contain a value of ".MADE.", ".BOUGHT." or ".NOT_KNOWN.". The value should be ".MADE." if the drawing was drawn within the company. The value should be ".BOUGHT." for drawings which are subcontracted. Discretion on the value is left to the producer of the data with the above guidance given as this distinction can be unclear particularly when the data is exchanged to another party.

AP 203 requires that all **product_definition_formation**s and therefore all **product_definition_formation_with_specified_source** entities be associated with a person and organization in the role of "creator". This **person_and_organization** is the one which created or prepared the drawing. The data for this person can be found either on the face of the drawing or on the change paperwork (typically the initiator). For guidance on creating the entities associated with this data see section 2.2 on page 4.

AP 203 requires that all **product_definition_formation** and therefore all **product_definition_formation_with_specified_source** entities be associated with at least one person and organization in the role of either "design_supplier" or "part_supplier". Only

"design_supplier" makes sense for drawings. The person and organization in the role of "design_supplier" is the one which drew the version of the drawing. For guidance on creating the entities associated with this data, see section 2.2 on page 4.

The reader is referred to section 2.8.1.2 on page 12 for processor recommendations and general guidance in creating this entity and its related constructs as they are the same as the guidance for parts.

3.2.1.3 The Product Definition Entity

Product_definition practices for drawings and parts are virtually identical. For the optional feature where a **product_definition** may be related to **document** entities through the sub-type **product_definition_with_associated_documents**, this entity should only be used for documents which identify associated Computer Aided Design (CAD) file (where the **document_type** attribute **product_data_type** has the value "cad_filename") or documents which represent computer identifiable note data (see section 3.2.2 on page 51).

The reader is referred to section 2.8.1.3 on page 14 for processor recommendations and general guidance in creating the **product_definition** entity and its related constructs as they are the same as the guidance for parts.

3.2.2 Drawing Notes

Drawings typically contain notes which are either generally applicable or are flagged to parts and their relationships. This information is represented on most drawings as a string of text and symbols with a note identifier or number. In many cases, these notes are not only shown on the drawing but also are maintained in configuration management systems. This sub-section addresses how the computer identifiable note data can be represented in AP 203 when it is available.

For computer identifiable note data, the **product_definition_with_associated_documents** will reference **document** entities where the **document_type** **product_data_type** is "design_specification", "material_specification", "surface_finish_specification", or "process_specification". In these instances, these are <u>not</u> specification references but are notes which specify design requirements. If the **product_definition_with_associated_documents** references notes as specified here and the notes do not <u>also</u> relate to a **product_definition_relationship** to a part product, they are notes with general applicability to the entire drawing. If the **product_definition_with_associated_documents** references notes as specified here and the notes do relate to a **product_definition_relationship**, they are notes with specific applicability to the referenced part within the context of the drawing. **Figure 18** describes these relationships pictorially. When this note data is recorded on a document separate from the drawing, this document is referred to as a notes list.
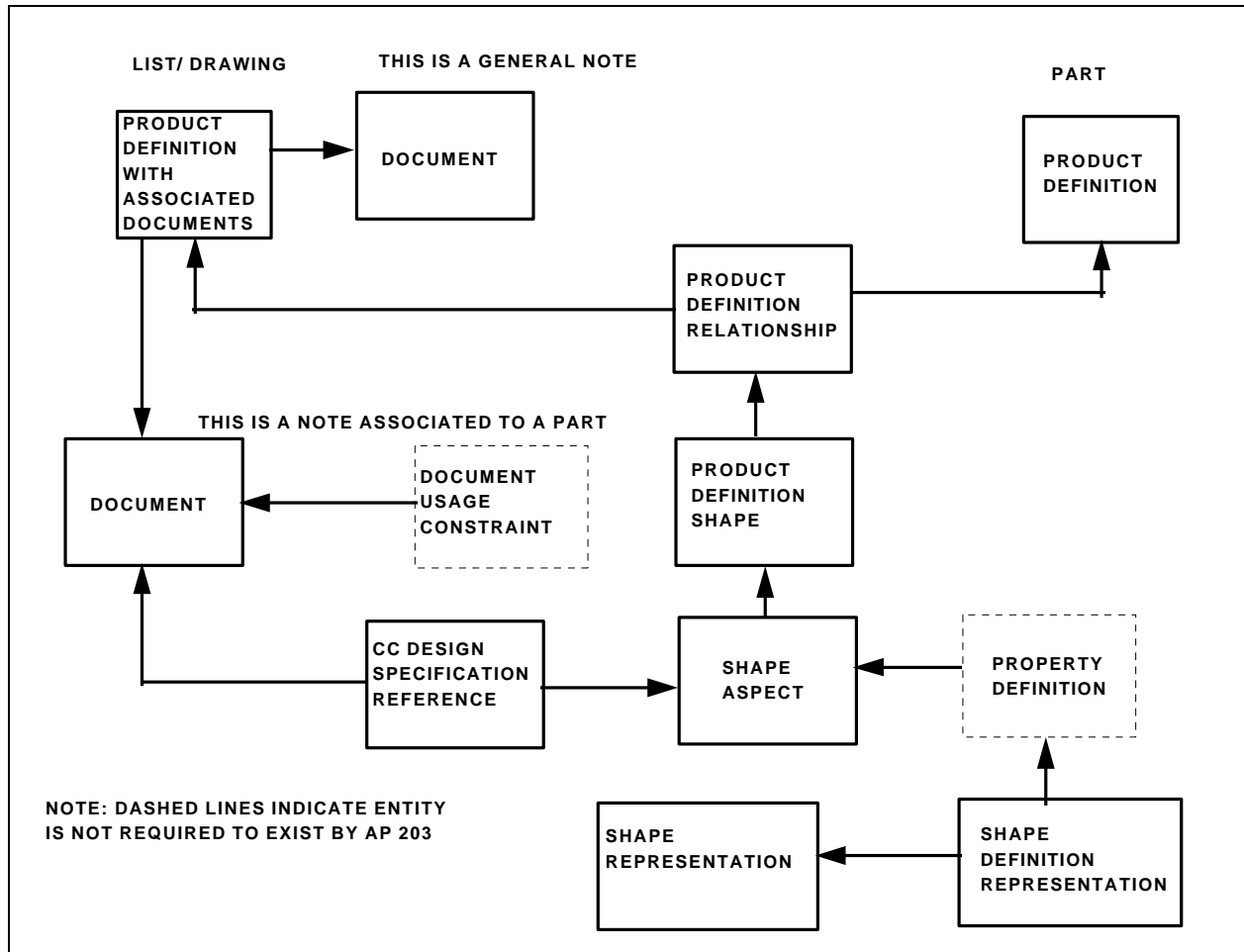
52

LIST/ DRAWING   THIS IS A GENERAL NOTE   PART

PRODUCT DEFINITION WITH ASSOCIATED DOCUMENTS

DOCUMENT

PRODUCT DEFINITION

PRODUCT DEFINITION RELATIONSHIP

THIS IS A NOTE ASSOCIATED TO A PART

DOCUMENT

DOCUMENT USAGE CONSTRAINT

PRODUCT DEFINITION SHAPE

CC DESIGN SPECIFICATION REFERENCE

SHAPE ASPECT

PROPERTY DEFINITION

NOTE: DASHED LINES INDICATE ENTITY IS NOT REQUIRED TO EXIST BY AP 203

SHAPE REPRESENTATION

SHAPE DEFINITION REPRESENTATION

It should be noted that AP 203 attaches notes to **product_definition_relationship** (and its sub-types) through a **shape_aspect**. This attachment is intended to use the shape for the **shape_aspect** to identify the exact location of the applicability. In an AP 203 conformance class 1 implementation, there is no shape data. In this case, a **document_usage_constraint** should be used to qualify the applicability of the note.

When computer identifiable note representation is used, the **document id** attribute shall be unique. If your organization does not uniquely identify notes, one possible alternative would be using a string composed of the drawing number, a comma, the drawing revision, a comma, and the note number. For example, if we are representing note 1 for drawing 222222 at rev A, the **document id** attribute would contain "222222,A,1". The **name** attribute would contain the note number. The **description** attribute would contain the note text.

The reader is referred to section 2.8.3 on page 18 for processor recommendations and any additional guidance needed in creating the **document** entity and its related constructs as they are

53

the same as the guidance for parts.

3.2.3 Categorizing Drawings

The reader is referred to section 2.8.2 on page 16 for processor recommendations and general guidance in creating the **product_category** and related entities and constructs as they are the same as the guidance for parts with the exception noted below.

All pre-processors should designate drawings as either "assembly" or "detail". Detail drawings are drawings which do not document any assembly parts. All drawing **product**s must have the category of "assembly" or "detail" related to a **product_category** whose **name** attribute value is "drawing".

3.2.4 Relating Specifications to Drawings

AP 203 relates specifications to entire **product**s by relating a **cc_design_specification_reference** entity to the **product_definition** of the **product**. This relationship shall be allowed by this recommended practice for drawings defined as **product**s where the specification is referenced or shown on the drawing. It should be noted that a data list could be constructed from this information if all the references exist and are made accurately.

The reader is referred to section 2.8.3 on page 18 for processor recommendations and general guidance in creating the **document** entity and its related constructs as they are the same as the guidance for parts.

3.2.5 Relating Drawings to Contracts

The reader is referred to section 2.8.4 on page 20 for processor recommendations and general guidance in creating the **contract** entity and its related constructs as they are the same as the guidance for parts.

3.2.6 Relating Drawings to Parts
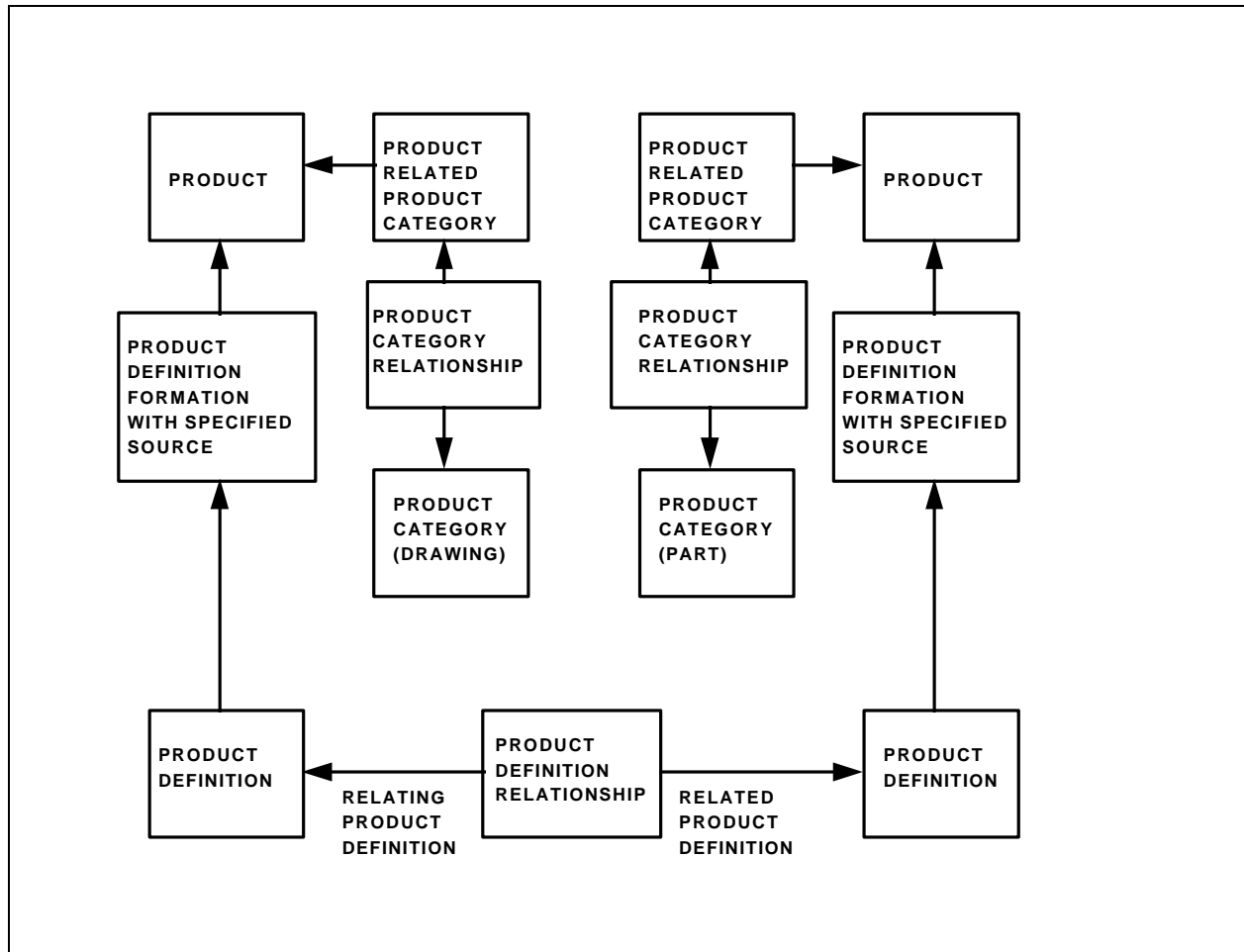
For the purposes of this recommended practice, drawings shall be related to the parts which are defined on them (sometimes known as domestic parts) through the use of a **product_definition_relationship** where the **relating_product_definition** attribute contains the drawing's **product_definition** and the **related_product_definition** attribute contains the **product_definition** of the part defined on the drawing. The **id** attribute must be unique. Since there is no standard mapping, the data should be generated in a manner to ensure that the value does not replicate any other value used for this entity or its sub-types. The **name** attribute shall contain the find number data for the part on the drawing. The **description** attribute shall contain the drawing zone data where the part is referenced. If the part is shown in multiple zones, they shall be listed in the attribute separated by commas. The structure of this relationship is shown in **Figure 19**.

3.2.7 Engineering Release/ Change Data

The engineering release and change structures in AP 203 are documented in section 2.8.12 on page 38 as they relate to parts. This section of this document will address how these structures

55

are used in relation to drawings and parts for the purposes of this recommended practice.

In the drawing based realm of change control, requests for change against the drawing typically result in changes to existing designs and the creation of new designs before they are ever incorporated into the drawing. This is commonly known as the red-line change process. This section of this document will describe how to deal with this situation in AP 203. It will also describe how to use the AP 203 data structures to represent the initial release of the drawing and its related parts as well as changes which affect both drawing and its related parts.

The initial release process is initiated through an AP 203 **versioned_action_request** which is related to the drawing **product_definition_formation_with_specified_source** (or version) being created along with the **product_definition_formation_with_specified_source** for any parts defined on the drawing through a **start_request**. This request process is followed (in the data) by an **action_directive** which is related to the drawing and parts to be released through a **start_work**. When this data is instantiated fully, both the drawing and associated parts referenced through the **items** attributes of the **start_request** and **start_work** have been released.

The change process is initiated through an AP 203 **versioned_action_request** which is related to the drawing proposed to be changed and any parts which are to be affected or created through a **change_request**. At this point, the change process may continue for the part (as defined in section 2.8.12.2 on page 41) while the request is left pending against the drawing. The part, in this case, will move through the **action_directive** phase causing a new part or part version to be created.

At some point in the change process, all the requested changes to the drawing will be incorporated. This incorporation may or may not affect parts. All of the affected items will be referenced by a **change** and one definite result will be a new version or **product_definition_formation_with_specified_source** for the drawing. This process in AP 203 will use the exact same data structures as used in section 2.8.12.2.

3.3 Lists in AP 203

In order to define a list in AP 203, the **product** entity establishes the list's identification (or list/ drawing number), name (or title), and description. The **product_definition_formation_with_specified_source** entity identifies its version (or change level). The **product_definition** or **product_definition_with_associated_documents** entity is used to relate the list to the drawing to which it is associated. Appendix C on page 65 contains a physical file example of the recommendations in this section.

3.3.1 Identifying Lists

3.3.1.1 The Product Entity

This section recommends that lists be treated as **product**s in AP 203. The list identification number (sometimes also designated as a drawing number on the lists's format) is stored in the **id** attribute (e.g. PL222222, NL222222, AL222222, etc.). The title or nomenclature of the list is stored in the **name** attribute. The **description** contains the official description of the list such as "parts list". All STEP **product**s must be founded in some **product_context** which identifies the engineering discipline from which the data is viewed. See section 2.1.2 on page 2 for guidance on creating this entity.

AP 203 requires that all **product**s exist in at least one **product_category** (see section 2.8.2 on page 16). This rule (**product_requires_product_category**) will force all lists created under this recommended practice into either the detail or assembly categories as the other possibilities make no sense for lists. The list's **product_related_product_category name** value should be the same as its associated drawing (assembly or detail). Additional guidance on categories for lists is provided in section 3.3.3 on page 58.

Products in AP 203 require a person and organization in the role of "design_owner". The data for this will be the same as for the associated drawing. The reader is referred to section 3.2.1.1 on page 50 for processor recommendations and general guidance in creating this entity and its related constructs as they are the same as the guidance for drawings.

3.3.1.2 The Product Definition Formation Entity

The **product_definition_formation_with_specified_source id** attribute should contain the value which represents the version or revision for the list. The **description** attribute should contain the reason for the creation of the version. The **source** attribute must contain a value of ".MADE.", ".BOUGHT." or ".NOT_KNOWN.". The value should be ".MADE." if the list was created within the company. The value should be ".BOUGHT." for lists which are subcontracted. Discretion on the value is left to the producer of the data with the above guidance given as this distinction can be unclear particularly when the data is exchanged to another party.

AP 203 requires that all **product_definition_formation**s and therefore all **product_definition_formation_with_specified_source** entities be associated with a person and organization in the role of "creator". This person and organization is the one which created or prepared the list. The data for this person can be found on the change paperwork. For guidance on creating the entities associated with this data see section 2.2 on page 4.

AP 203 requires that all **product_definition_formation** and therefore all **product_definition_formation_with_specified_source** entities be associated with at least one person and organization in the role of either "design_supplier" or "part_supplier". For lists, this data will be the same as for the "creator" in the prior paragraph.

The reader is referred to section 2.8.1.2 on page 12 for processor recommendations and general guidance in creating this entity and its related constructs as they are the same as the guidance for

parts.

3.3.1.3 The Product Definition Entity

Product_definition practices for lists, drawings and parts are virtually identical. For the optional feature where a **product_definition** may be related to **document** entities through the sub-type **product_definition_with_associated_documents**, this entity may be used for documents which identify associated Computer Aided Design (CAD) file (where the **document_type** attribute **product_data_type** has the value "cad_filename"). In the case of lists, this may also be an ASCII file of the list's textual data. See section 3.3.2 on page 57 for additional information on using the **product_definition_with_associated_documents** entity with lists.

The reader is referred to section 2.8.1.3 on page 14 for processor recommendations and general guidance in creating the **product_definition** entity and its related constructs as they are the same as the guidance for parts.

3.3.2 List Notes

Lists can contain notes which are either generally applicable or are flagged to parts and their relationships. This information is represented on lists as a string of text with a note identifier or number. In most cases, these notes are maintained in configuration management systems. This sub-section addresses how the computer identifiable note data can be represented in AP 203 when it is available.

For computer identifiable note data, the **product_definition_with_associated_documents** will reference **document** entities where the **document_type** **product_data_type** is "design_specification", "material_specification", "surface_finish_specification", or "process_specification". In these instances, these are <u>not</u> specification references but are notes which specify design requirements. If the **product_definition_with_associated_documents** references notes as specified here and the notes do not <u>also</u> relate to a **product_definition_relationship** to a part product, they are notes with general applicability to the entire list. If the **product_definition_with_associated_documents** references notes as specified here and the notes do relate to a **product_definition_relationship**, they are notes with specific applicability to the referenced part within the context of the list. **Figure 18** on page 52 describes these relationships pictorially.

It should be noted that AP 203 attaches notes to **product_definition_relationship** (and its sub-types) through a **shape_aspect**. This attachment is intended to use the shape for the **shape_aspect** to identify the exact location of the applicability. In an AP 203 conformance class 1 implementation, there is no shape data. In this case, a **document_usage_constraint** should be used to qualify the applicability of the note.

When computer identifiable note representation is used, the **document id** attribute is required to

be unique. If your organization does not uniquely identify notes (e.g. in an external database), one possible alternative would be using a string composed of the list number, a comma, the list revision, a comma, and the note number. For example, if we are representing note 1 for notes list NL222222 at rev A, the **document id** attribute would contain "NL222222,A,1". The **name** attribute would contain the note number. The **description** attribute would contain the note text.

The reader is referred to section 2.8.3 on page 18 for processor recommendations and any additional guidance needed in creating the **document** entity and its related constructs as they are the same as the guidance for parts.

3.3.3 Categorizing Lists

The reader is referred to section 2.8.2 on page 16 for processor recommendations and general guidance in creating the **product_category** and related entities and constructs as they are the same as the guidance for parts with the exception noted below.

All pre-processors should designate lists as either "assembly" or "detail" in the **product_related_product_category name** value. This value shall be the same as the **product_related_product_category name** for the drawing to which the list is associated. All list **product**s must have the category of "assembly" or "detail" related as a **subcategory** in a **product_category_relationship** to a **product_category** whose **name** attribute value describes the type of list. The value of this attribute should be "parts list", "data list", "application list", "notes list", "running list" or, for a list which is outside these delineated values, a value which describes the content of the list. The prior **product_category** shall related as a **subcategory** in a **product_category_relationship** to a **product_category** whose **name** attribute value is "list".

For example, a parts list numbered PL999999 would have a **product** whose **id** attribute is "PL999999". This product would be related to a **product_related_product_category** whose **name** attribute value is "assembly". The **product_related_product_category** would be related as a **sub_category** in a **product_category_relationship** to a **product_category** whose **name** attribute value is "parts list". This **product_category** would in turn be related as a **sub_category** in a **product_category_relationship** to a **product_category** whose **name** attribute value is "list".

3.3.4 Relating Specifications to Lists

AP 203 relates specifications to entire **product**s by relating a **cc_design_specification_reference** entity to the **product_definition** of the **product**. This relationship shall be allowed by this recommended practice for lists defined as **product**s where the specification is referenced on the list as a reference document.
The reader is referred to section 2.8.3 on page 18 for processor recommendations and general guidance in creating the **document** entity and its related constructs as they are the same as the guidance for parts.

3.3.5 Relating Lists to Contracts

The reader is referred to section 2.8.4 on page 20 for processor recommendations and general guidance in creating the **contract** entity and its related constructs as they are the same as the guidance for parts.

3.3.6 Relating Lists to Drawings

For the purposes of this recommended practice, lists shall be related to their associated drawing through the use of a **product_definition_relationship** where the **relating_product_definition** attribute contains the drawing's **product_definition** and the **related_product_definition** attribute contains the **product_definition** of the list. This is described pictorially in **Figure 20**.
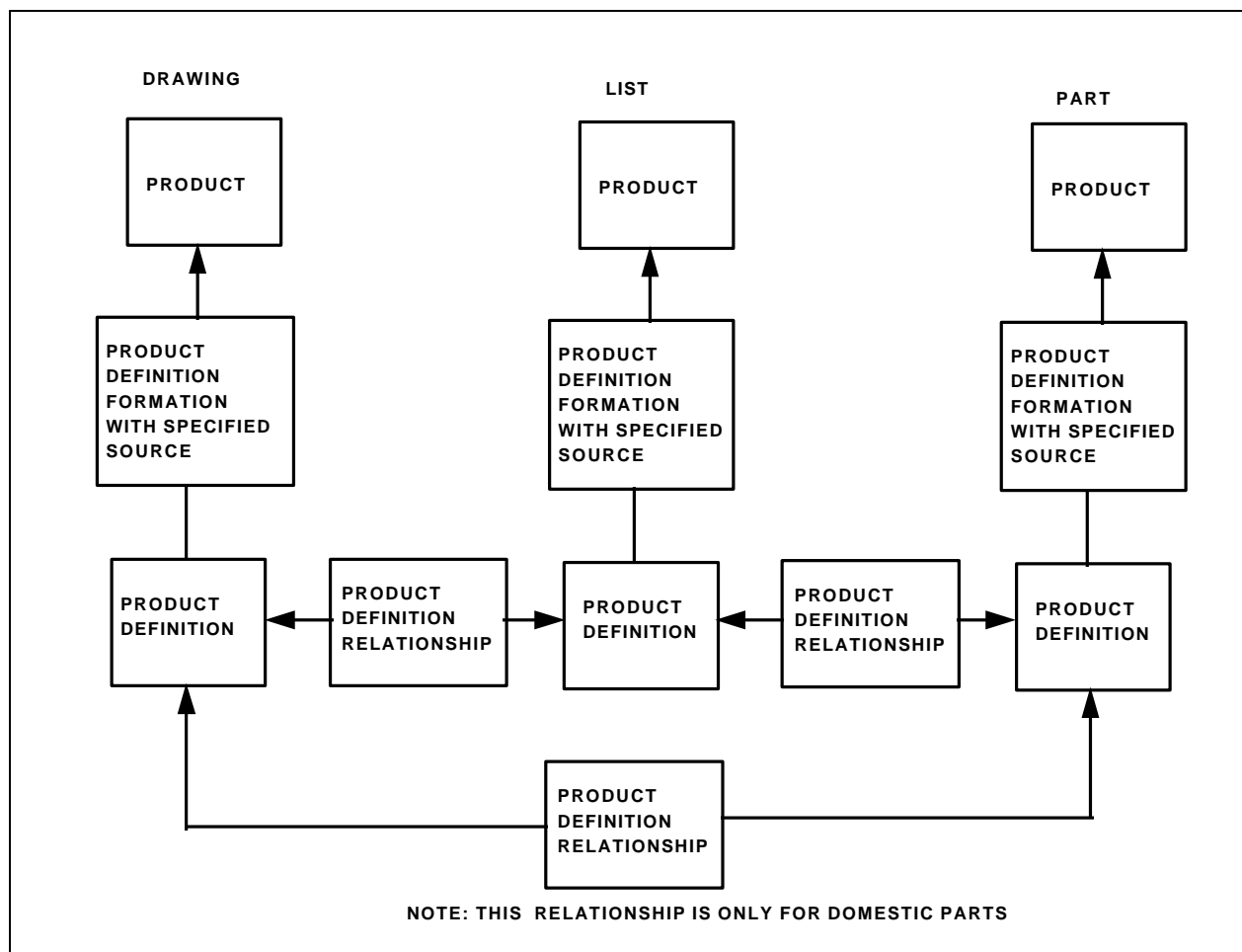
All pre-processors should ensure the **id** attribute of the **product_definition_relationship** is

unique. Since there is no standard mapping, the data should be generated in a manner to ensure that the value does not replicate any other value used for this entity or its sub-types. The **name** and the **description** attribute shall contain a null value for minimal data content as there is no standard mapping for these attributes.

3.3.7 Relating Lists to Parts

A list shall be related to all parts or part versions which it references. This relationship is established through the use of a **product_definition_relationship** where the **relating_product_definition** attribute contains the list's **product_definition** and the **related_product_definition** attribute contains the **product_definition** of the part. This is described pictorially in **Figure 20**.

All pre-processors should ensure the **id** attribute of the **product_definition_relationship** is unique. Since there is no standard mapping, the data should be generated in a manner to ensure that the value does not replicate any other value used for this entity or its sub-types. The **name** and the **description** attribute shall contain a null value for minimal data content or any mutually agreed upon string.

3.3.8 Engineering Release/ Change Data

The engineering release and change structures in AP 203 are documented in section 2.8.12 on page 38 as they relate to parts and in section 3.2.7 on page 55 as they relate to drawings. This section of this document will address how these structures are used in relation to lists for the purposes of this recommended practice.

In the drawing based realm of change control, requests for change against the drawing typically result in changes to lists before they are ever incorporated into the drawing. This is commonly known as the red-line change process. This section of this document will describe how to deal with this situation in AP 203 as it relates to lists. It will also describe how to use the AP 203 data structures to represent the initial release of the drawing, lists and related parts and changes which affect the drawing, lists and related parts.

The initial release process is initiated through an AP 203 **versioned_action_request** which is related to the drawing and list **product_definition_formation_with_specified_source** entities (or versions) being created along with the **product_definition_formation_with_specified_source** for any parts defined on the drawing through a **start_request**. This request process is followed (in the data) by an **action_directive** which is related to the drawing, list and parts to be released through a **start_work**. When this data is instantiated fully, the drawing, list and associated parts referenced through the **items** attributes of the **start_request** and **start_work** have been released.

The change process is initiated through an AP 203 **versioned_action_request**, as well, which is

related to the drawing proposed to be changed and any parts or lists which are to be affected or created through a **change_request**. At this point, the change process may continue for the part (as defined in section 2.8.12.2 on page 41) and list while the request is left pending against the drawing. The part and list, in this case, will move through the **action_directive** phase causing a new part or part version to be created with a new version for the list.

At some point in the change process, all the requested changes to the drawing will be incorporated. This incorporation may or may not affect parts or lists. All of the affected items will be referenced by a **change** and one definite result will be a new version or **product_definition_formation_with_specified_source** for the drawing. This process in AP 203 will use the exact same data structures as used in section 2.8.12.2 on page 41.

Appendix B - Physical File Example

This physical file shows the minimum number of entities instances required in an AP 203 class 1 file. This file uses entity long names and is formatted for readability. The actual number of characters needed to represent the data would be considerably less if short names were used and was not formatted for readability.

```
ISO-10303-21;
HEADER;
FILE_DESCRIPTION(('THIS IS A SAMPLE AP203 STEP MODEL'),'1');
FILE_NAME('CONCEPTUAL PART EXAMPLE',
'1995-03-24 T15:30:00',('LORI BRINDLE'),('PDES, Inc.'),'NO VERSION',
'HAND POPULATED','APPROVED BY LARRY MCKEE');
FILE_SCHEMA(('CONFIG_CONTROL_DESIGN'));
ENDSEC;
DATA;
#1078=COORDINATED_UNIVERSAL_TIME_OFFSET(5,$,.BEHIND.);
#1079=LOCAL_TIME(12,0,$,#1078);
#1083=PERSON('1111111','MCKEE','LARRY',('J'),$,$);
#1084=ORGANIZATION('PDESI','PDES, Inc.','STEP INDUSTRY CONSORTIUM');
#1085=PERSON_AND_ORGANIZATION(#1083,#1084);
#1086=CALENDAR_DATE(1994,2,1);
#1087=DATE_AND_TIME(#1086,#1079);
#1102=APPLICATION_CONTEXT('configuration controlled 3d designs of
      mechanical parts and assemblies');
#1103=APPLICATION_PROTOCOL_DEFINITION('international standard',
      'config_control_design',1994,#1102);
#1104=MECHANICAL_CONTEXT('',#1102,'mechanical');
#1105=PRODUCT('2865000-1','REAR PANEL','REAR PANEL FOR BOX',(#1104));
#1106=PRODUCT_RELATED_PRODUCT_CATEGORY('detail','',(#1105));
#1107=PERSON_AND_ORGANIZATION_ROLE('design_owner');
#1108=CC_DESIGN_PERSON_AND_ORGANIZATION_ASSIGNMENT(#1085,#1107,(#1105));
#1109=PRODUCT_CATEGORY('part','');
#1110=PRODUCT_CATEGORY_RELATIONSHIP('','',#1109,#1106);
#1119=PRODUCT_DEFINITION_FORMATION_WITH_SPECIFIED_SOURCE('-',
     'PRE-RELEASE',#1105,.MADE.);
#1120=PERSON_AND_ORGANIZATION_ROLE('design_supplier');
#1121=CC_DESIGN_PERSON_AND_ORGANIZATION_ASSIGNMENT(#1085,#1120,(#1119));
#1128=SECURITY_CLASSIFICATION_LEVEL('unclassified');
#1129=SECURITY_CLASSIFICATION('','',#1128);
#1130=CC_DESIGN_SECURITY_CLASSIFICATION(#1129,(#1119));
#1147=PERSON_AND_ORGANIZATION_ROLE('classification_officer');
#1148=CC_DESIGN_PERSON_AND_ORGANIZATION_ASSIGNMENT(#1085,#1147,(#1129));
#1153=DATE_TIME_ROLE('classification_date');
#1154=CC_DESIGN_DATE_AND_TIME_ASSIGNMENT(#1087,#1153,(#1129));
#1155=DESIGN_CONTEXT('',#1102,'design');
#1156=PRODUCT_DEFINITION('design','',#1119,#1155);
#1175=PERSON_AND_ORGANIZATION_ROLE('creator');
#1176=CC_DESIGN_PERSON_AND_ORGANIZATION_ASSIGNMENT(#1085,#1175,(#1119,#1156));
#1181=DATE_TIME_ROLE('creation_date');
#1182=CC_DESIGN_DATE_AND_TIME_ASSIGNMENT(#1087,#1181,(#1156));
#1185=APPROVAL_ROLE('approver');
#1203=APPROVAL_STATUS('approved');
#1205=APPROVAL(#1203,'');
#1206=CC_DESIGN_APPROVAL(#1205,(#1119,#1129,#1156));
#1207=APPROVAL_PERSON_ORGANIZATION(#1085,#1205,#1185);
```

```
#1208=APPROVAL_DATE_TIME(#1087,#1205);
ENDSEC;
END-ISO-10303-21;
```

Appendix C - Drawing as Product Physical File Example

This physical file shows the minimum number of entities instances required in an AP 203 class 1 file using drawing as product. This file uses entity long names and is formatted for readability. The actual number of characters needed to represent the data would be considerably less if short names were used and was not formatted for readability.

```
ISO-10303-21;
HEADER;
FILE_DESCRIPTION(('THIS IS A SAMPLE AP203 STEP MODEL'),'1');
FILE_NAME('CONCEPTUAL PART EXAMPLE',
'1995-03-24 T15:30:00',
('LORI BRINDLE'),
('PDES, Inc.'),
'NO VERSION',
'HAND POPULATED',
'APPROVED BY LARRY MCKEE');
FILE_SCHEMA(('CONFIG_CONTROL_DESIGN'));
ENDSEC;
DATA;
#1078=COORDINATED_UNIVERSAL_TIME_OFFSET(5,$,.BEHIND.);
#1079=LOCAL_TIME(12,0,$,#1078);
#1083=PERSON('1111111','MCKEE','LARRY',('J'),$,$);
#1084=ORGANIZATION('PDESI','PDES, Inc.','STEP INDUSTRY CONSORTIUM');
#1085=PERSON_AND_ORGANIZATION(#1083,#1084);
#1086=CALENDAR_DATE(1994,2,1);
#1087=DATE_AND_TIME(#1086,#1079);
#1102=APPLICATION_CONTEXT('configuration controlled 3d designs of
      mechanical parts and assemblies');
#1103=APPLICATION_PROTOCOL_DEFINITION('international standard',
      'config_control_design',1994,#1102);
#1104=MECHANICAL_CONTEXT('',#1102,'mechanical');
#1105=PRODUCT('2865000-1','REAR PANEL','REAR PANEL FOR BOX',(#1104));
#1119=PRODUCT_DEFINITION_FORMATION_WITH_SPECIFIED_SOURCE('-',
      'PRE-RELEASE',#1105,.MADE.);
#1155=DESIGN_CONTEXT('',#1102,'design');
#1156=PRODUCT_DEFINITION('design','',#1119,#1155);
#2105=PRODUCT('2865000','REAR PANEL','REAR PANEL FOR BOX',(#1104));
#2119=PRODUCT_DEFINITION_FORMATION_WITH_SPECIFIED_SOURCE('-',
      'PRE-RELEASE',#2105,.MADE.);
#2156=PRODUCT_DEFINITION('design','',#2119,#1155);
#2200=PRODUCT_DEFINITION_RELATIONSHIP('ID1','1-1','A2',#2156,#1156);
#9105=PRODUCT_RELATED_PRODUCT_CATEGORY('detail','',(#1105));
#9106=PRODUCT_RELATED_PRODUCT_CATEGORY('detail','',(#2105));
#9107=PERSON_AND_ORGANIZATION_ROLE('design_owner');
#9108=CC_DESIGN_PERSON_AND_ORGANIZATION_ASSIGNMENT(#1085,#9107,(#1105,#2105));
#9109=PRODUCT_CATEGORY('part','');
#9110=PRODUCT_CATEGORY_RELATIONSHIP('','',#9109,#9105);
#9111=PRODUCT_CATEGORY('drawing','');
#9112=PRODUCT_CATEGORY_RELATIONSHIP('','',#9111,#9106);
#9120=PERSON_AND_ORGANIZATION_ROLE('design_supplier');
#9121=CC_DESIGN_PERSON_AND_ORGANIZATION_ASSIGNMENT(#1085,#9120,(#1119,#2119));
#9128=SECURITY_CLASSIFICATION_LEVEL('unclassified');
#9129=SECURITY_CLASSIFICATION('','',#9128);
#9130=CC_DESIGN_SECURITY_CLASSIFICATION(#9129,(#1119,#2119,#2200));
```

```
#9147=PERSON_AND_ORGANIZATION_ROLE('classification_officer');
#9148=CC_DESIGN_PERSON_AND_ORGANIZATION_ASSIGNMENT(#1085,#9147,(#9129));
#9153=DATE_TIME_ROLE('classification_date');
#9154=CC_DESIGN_DATE_AND_TIME_ASSIGNMENT(#1087,#9153,(#9129));
#9175=PERSON_AND_ORGANIZATION_ROLE('creator');
#9176=CC_DESIGN_PERSON_AND_ORGANIZATION_ASSIGNMENT(#1085,#9175,(#1119,#2119,
      #1156,#2156));
#9181=DATE_TIME_ROLE('creation_date');
#9182=CC_DESIGN_DATE_AND_TIME_ASSIGNMENT(#1087,#9181,(#1156,#2156));
#9185=APPROVAL_ROLE('approver');
#9203=APPROVAL_STATUS('approved');
#9205=APPROVAL(#9203,'');
#9206=CC_DESIGN_APPROVAL(#9205,(#1119,#9129,#1156,#2119,#2156));
#9207=APPROVAL_PERSON_ORGANIZATION(#1085,#9205,#9185);
#9208=DATE_TIME_ROLE('sign_off_date');
#9209=CC_DESIGN_DATE_AND_TIME_ASSIGNMENT(#1087,#9208,(#9207));
#9209=APPROVAL_DATE_TIME(#1087,#9205);
ENDSEC;
END-ISO-10303-21;
```

Appendix D - Drawing and List Physical File Example

This physical file an AP 203 class 1 file using drawing as product with two lists. The file has a single part which is made from raw material with a parts list and a notes list product. The parts list has a general note. The notes list has a note related to a part. The file also has engineering release documentation information.   This file uses entity long names and is formatted for readability. The actual number of characters needed to represent the data would be considerably less if short names were used and was not formatted for readability.

```
ISO-10303-21;
HEADER;
FILE_DESCRIPTION(('THIS IS A SAMPLE AP203 STEP MODEL'),'1');
FILE_NAME('CONCEPTUAL PART EXAMPLE',
'1995-09-24 T15:30:00',
('LORI BRINDLE'),
('PDES, Inc.'),
'NO VERSION',
'HAND POPULATED',
'APPROVED BY LARRY MCKEE');
FILE_SCHEMA(('CONFIG_CONTROL_DESIGN'));
ENDSEC;
DATA;
#1078=COORDINATED_UNIVERSAL_TIME_OFFSET(5,$,.BEHIND.);
#1079=LOCAL_TIME(12,0,$,#1078);
#1083=PERSON('1111111','MCKEE','LARRY',('J'),$,$);
#1084=ORGANIZATION('PDESI','PDES, Inc.','STEP INDUSTRY CONSORTIUM');
#1085=PERSON_AND_ORGANIZATION(#1083,#1084);
#1086=CALENDAR_DATE(1994,2,1);
#1087=DATE_AND_TIME(#1086,#1079);
#1102=APPLICATION_CONTEXT('configuration controlled 3d designs of
      mechanical parts and assemblies');
#1103=APPLICATION_PROTOCOL_DEFINITION('international standard',
      'config_control_design',1994,#1102);
#1104=MECHANICAL_CONTEXT('',#1102,'mechanical');
#1105=PRODUCT('2865000-1','A PART','A PART EXAMPLE',(#1104));
#1119=PRODUCT_DEFINITION_FORMATION_WITH_SPECIFIED_SOURCE('-',
     'PRE-RELEASE',#1105,.MADE.);
#1155=DESIGN_CONTEXT('',#1102,'design');
#1156=PRODUCT_DEFINITION('design','',#1119,#1155);
#2105=PRODUCT('2865000','A DRAWING','A DRAWING EXAMPLE',(#1104));
#2119=PRODUCT_DEFINITION_FORMATION_WITH_SPECIFIED_SOURCE('-',
     'PRE-RELEASE',#2105,.MADE.);
#2156=PRODUCT_DEFINITION_WITH_ASSOCIATED_DOCUMENTS('design','',#2119,#1155,
     (#2184,#2284));
#2183=DOCUMENT_TYPE('design_specification');
#2184=DOCUMENT('2865000,-,1','1',
     'THIS IS GENERAL NOTE 1 ON THE DRAWING',#2183);
#2200=PRODUCT_DEFINITION_RELATIONSHIP('IDDR1','1-1','A2',#2156,#1156);
#2284=DOCUMENT('2865000,-,2','2',
     'THIS IS NOTE 2 ON THE DRAWING TIED TO A PART',#2183);
#2285=PRODUCT_DEFINITION_SHAPE('','',#2200);
#2286=SHAPE_ASPECT('','',#2285,.F.);
#2287=CC_DESIGN_SPECIFICATION_REFERENCE(#2284,'PDES,INC.',(#2286));
#3105=PRODUCT('PL2865000','A PARTS LIST','PARTS LIST',(#1104));
```

```
#3119=PRODUCT_DEFINITION_FORMATION_WITH_SPECIFIED_SOURCE('-',
      'PRE-RELEASE',#3105,.MADE.);
#3156=PRODUCT_DEFINITION_WITH_ASSOCIATED_DOCUMENTS('design','',#3119,#1155,
      (#3184));
#3184=DOCUMENT('PL2865000,-,1','1',
      'THIS IS GENERAL NOTE 1 ON THE PARTS LIST',#2183);
#3200=PRODUCT_DEFINITION_RELATIONSHIP('IDPL1','','',#2156,#3156);
#3201=PRODUCT_DEFINITION_RELATIONSHIP('IDPL2','','',#3156,#1156);
#3284=DOCUMENT('2865000,-,2','2',
      'THIS IS NOTE 2 ON THE PARTS LIST TIED TO A PART',#2183);
#3285=PRODUCT_DEFINITION_SHAPE('','',#3201);
#3286=SHAPE_ASPECT('','',#2285,.F.);
#3287=CC_DESIGN_SPECIFICATION_REFERENCE(#3284,'PDES,INC.',(#3286));
#4105=PRODUCT('NL2865000','A NOTES LIST','NOTES LIST',(#1104));
#4119=PRODUCT_DEFINITION_FORMATION_WITH_SPECIFIED_SOURCE('-',
      'PRE-RELEASE',#4105,.MADE.);
#4156=PRODUCT_DEFINITION_WITH_ASSOCIATED_DOCUMENTS('design','',#4119,#1155,
      (#4184,#4284));
#4184=DOCUMENT('NL2865000,-,1','1',
      'THIS IS GENERAL NOTE 1 ON THE NOTES LIST',#2183);
#4200=PRODUCT_DEFINITION_RELATIONSHIP('IDNL1','','',#2156,#4156);
#4201=PRODUCT_DEFINITION_RELATIONSHIP('IDNL2','','',#4156,#1156);
#4284=DOCUMENT('NL2865000,-,1','1',
      'THIS IS NOTE 1 ON THE NOTES LIST TIED TO A PART',#2183);
#4285=PRODUCT_DEFINITION_SHAPE('','',#4201);
#4286=SHAPE_ASPECT('','',#4285,.F.);
#4287=CC_DESIGN_SPECIFICATION_REFERENCE(#4284,'PDES,INC.',(#4286));
#5105=PRODUCT('1X1X1X1','BULK MATERIAL','A PART MARKED MATERIAL EXAMPLE',
      (#1104));
#5119=PRODUCT_DEFINITION_FORMATION_WITH_SPECIFIED_SOURCE('',
      '',#5105,.BOUGHT.);
#5156=PRODUCT_DEFINITION('design','',#5119,#1155);
#5200=DESIGN_MAKE_FROM_RELATIONSHIP('DMR1','','BULK MATERIAL',#5156,#1156);
#5201=PRODUCT_DEFINITION_RELATIONSHIP('IDDR2','1-2','A2',#2156,#5156);
#9103=PRODUCT_RELATED_PRODUCT_CATEGORY('detail','',(#1105));
#9104=PRODUCT_RELATED_PRODUCT_CATEGORY('detail','',(#2105));
#9105=PRODUCT_RELATED_PRODUCT_CATEGORY('detail','',(#3105));
#9106=PRODUCT_RELATED_PRODUCT_CATEGORY('detail','',(#4105));
#9107=PERSON_AND_ORGANIZATION_ROLE('design_owner');
#9108=CC_DESIGN_PERSON_AND_ORGANIZATION_ASSIGNMENT(#1085,#9107,
      (#1105,#2105,#3105,#4105));
#9109=PRODUCT_CATEGORY('part','');
#9110=PRODUCT_CATEGORY_RELATIONSHIP('','',#9109,#9103);
#9111=PRODUCT_CATEGORY('drawing','');
#9112=PRODUCT_CATEGORY_RELATIONSHIP('','',#9111,#9104);
#9113=PRODUCT_CATEGORY('list','');
#9114=PRODUCT_CATEGORY_RELATIONSHIP('','',#9113,#9105);
#9115=PRODUCT_CATEGORY_RELATIONSHIP('','',#9113,#9106);
#9116=PRODUCT_CATEGORY('parts list','');
#9117=PRODUCT_CATEGORY_RELATIONSHIP('','',#9116,#9105);
#9118=PRODUCT_CATEGORY('notes list','');
#9119=PRODUCT_CATEGORY_RELATIONSHIP('','',#9118,#9106);
#9120=PERSON_AND_ORGANIZATION_ROLE('design_supplier');
#9121=CC_DESIGN_PERSON_AND_ORGANIZATION_ASSIGNMENT(#1085,#9120,
      (#1119,#2119,#3119,#4119));
#9128=SECURITY_CLASSIFICATION_LEVEL('unclassified');
#9129=SECURITY_CLASSIFICATION('','',#9128);
#9130=CC_DESIGN_SECURITY_CLASSIFICATION(#9129,
```

```
        (#1119,#2119,#3119,#4119));
#9147=PERSON_AND_ORGANIZATION_ROLE('classification_officer');
#9148=CC_DESIGN_PERSON_AND_ORGANIZATION_ASSIGNMENT(#1085,#9147,(#9129));
#9153=DATE_TIME_ROLE('classification_date');
#9154=CC_DESIGN_DATE_AND_TIME_ASSIGNMENT(#1087,#9153,(#9129));
#9175=PERSON_AND_ORGANIZATION_ROLE('creator');
#9176=CC_DESIGN_PERSON_AND_ORGANIZATION_ASSIGNMENT(#1085,#9175,
        (#1119,#2119,#3119,#4119,#1156,#2156,#3156,#4156));
#9181=DATE_TIME_ROLE('creation_date');
#9182=CC_DESIGN_DATE_AND_TIME_ASSIGNMENT(#1087,#9181,
        (#1156,#2156,#3156,#4156));
#9200=VERSIONED_ACTION_REQUEST('ENXXXXXX','','INITIAL RELEASE',
        'RELEASE PART');
#9201=ACTION_REQUEST_STATUS('issued',#9200);
#9202=ACTION_METHOD('','RELEASE PART','','');
#9203=ACTION_REQUEST_SOLUTION(#9202,#9200);
#9204=START_REQUEST(#9200,(#1119,#2119,#3119,#4119));
#9205=ACTION_DIRECTIVE('ENXXXXXX','RELEASE PART','','',
        (#9200));
#9206=DIRECTED_ACTION('ENXXXXXX','RELEASE PART',#9202,#9205);
#9207=ACTION_STATUS('issued',#9206);
#9208=START_WORK(#9206,(#1119,#2119,#3119,#4119));
#9275=PERSON_AND_ORGANIZATION_ROLE('request_recipient');
#9276=CC_DESIGN_PERSON_AND_ORGANIZATION_ASSIGNMENT(#1085,#9275,
        (#9204));
#9281=DATE_TIME_ROLE('request_date');
#9282=CC_DESIGN_DATE_AND_TIME_ASSIGNMENT(#1087,#9281,
        (#9204));
#9283=DATE_TIME_ROLE('start_date');
#9284=CC_DESIGN_DATE_AND_TIME_ASSIGNMENT(#1087,#9283,
        (#9208));
#9300=APPROVAL_ROLE('approver');
#9303=APPROVAL_STATUS('approved');
#9305=APPROVAL(#9303,'');
#9306=CC_DESIGN_APPROVAL(#9305,
        (#9129,#1119,#1156,#2119,#2156,#3119,#3156,#4119,#4156,#9204,#9208));
#9307=APPROVAL_PERSON_ORGANIZATION(#1085,#9305,#9300);
#9308=DATE_TIME_ROLE('sign_off_date');
#9309=CC_DESIGN_DATE_AND_TIME_ASSIGNMENT(#1087,#9308,(#9207));
#9310=APPROVAL_DATE_TIME(#1087,#9305);
ENDSEC;
END-ISO-10303-21;
```

Appendix E - Units Examples

This appendix is provided to give examples of instantiations of AP 203 global units for both standard international (SI) and English units.

When instantiated in a physical file, global SI units where length is expressed in centimeters with an overall model closure tolerance of .000001 of a centimeter would appear as:

```
#1=DIMENSIONAL_EXPONENTS(0.0,0.0,0.0,0.0,0.0,0.0,0.0);
#13=(LENGTH_UNIT()NAMED_UNIT(*)SI_UNIT(.CENTI.,.METRE.));
#14=(NAMED_UNIT(*)PLANE_ANGLE_UNIT()SI_UNIT($,.RADIAN.));
#15=(NAMED_UNIT(*)SI_UNIT($,.STERADIAN.)SOLID_ANGLE_UNIT());
#16=UNCERTAINTY_MEASURE_WITH_UNIT(LENGTH_MEASURE(0.000001),#13,
     '','');
#17=(GEOMETRIC_REPRESENTATION_CONTEXT(3)
     GLOBAL_UNCERTAINTY_ASSIGNED_CONTEXT((#16))
     GLOBAL_UNIT_ASSIGNED_CONTEXT((#13,#14,#15))
     REPRESENTATION_CONTEXT('',''));
```

When instantiated in a physical file, global English units where length is expressed in inches with an overall model closure tolerance of .000001 of an inch would appear as:

```
#1=DIMENSIONAL_EXPONENTS(0.0,0.0,0.0,0.0,0.0,0.0,0.0);
#13=(LENGTH_UNIT()NAMED_UNIT(*)SI_UNIT(.CENTI.,.METRE.));
#14=(NAMED_UNIT(*)PLANE_ANGLE_UNIT()SI_UNIT($,.RADIAN.));
#15=(NAMED_UNIT(*)SI_UNIT($,.STERADIAN.)SOLID_ANGLE_UNIT());
#16=LENGTH_MEASURE_WITH_UNIT(LENGTH_MEASURE(2.54),#13);
#17=DIMENSIONAL_EXPONENTS(1.0,0.0,0.0,0.0,0.0,0.0,0.0);
#18=(CONVERSION_BASED_UNIT('INCH',#16)LENGTH_UNIT()NAMED_UNIT(#17));
#19=UNCERTAINTY_MEASURE_WITH_UNIT(LENGTH_MEASURE(0.000001),#18,
     '','');
#20=(GEOMETRIC_REPRESENTATION_CONTEXT(3)
     GLOBAL_UNCERTAINTY_ASSIGNED_CONTEXT((#19))
     GLOBAL_UNIT_ASSIGNED_CONTEXT((#18,#14,#15))
     REPRESENTATION_CONTEXT('',''));
```

Appendix F - Conformance Class 1 Implementations and Subsets

AP 203 conformance class 1 implementations involve only configuration management structures. This can be problematic in certain instances in that some of the complex relationships in AP 203 actually need shape in order to be defined clearly. This is particularly apparent with regard to specifications which can be applied to a portion of a part or to a relationship through a **shape_aspect**. This document provides guidance in sections 2.8.3 and 2.8.11 on pages 18 and 36 which can be used by pre-processors to overcome or bypass these problems. Post-processors will be forced to deal with these situations by informing the user of conformance class 1 data which involves shape through messages.

Conformance class 1 of AP 203 encompasses <u>all</u> the configuration management data and relationships in AP 203. This was done intentionally in AP 203 to limit the number of conformance classes in the AP. At the time that AP 203 was released to ISO, it was felt that a large number of conformance classes would impede implementation of the AP.

In the recent past, a need to subset class 1 of AP 203 has become apparent. The current content of class 1 can be overkill for small suppliers and early design phase exchanges of engineering mockups which only need a portion of the data and functionality in conformance class 1. This appendix will describe four logical subsets of AP 203 conformance class 1.

In AP 203, there are hidden proper subsets in conformance class 1. This paper will propose four subclasses of conformance class 1 which are proper subsets and could be used as a basis for an implementors agreement or future Application Interpreted Constructs (AICs).

F.1 Product Identification (Minimal)

This subset is for use in identifying parts with no product structure, engineering change information, or effectivity. It should be noted that this subset has been structured so that <u>no</u> external mapping (or complex entity instances) is required. In other words, the entities in this subset all result in a simple instance in a STEP physical file The list following this description includes super-types (e.g. **product_definition_formation**) only for reference as they will be used in the other subsets which will involve complex instances (for convenience, the super-types are marked with an asterisk). If this subset is adopted in an implementors agreement, it should allow for relatively inexpensive translators to be developed for small suppliers. The entities in this subset are:

application_context
*application_context_element
application_protocol_definition
approval
*approval_assignment

approval_date_time
approval_person_organization
approval_role
approval_status
calendar_date (or ordinal_date or week_of_year_and_day_date)
cc_design_approval
cc_design_date_and_time_assignment
cc_design_person_and_organization_assignment
cc_design_security_classification
coordinated_universal_time_offset
*date
date_and_time
*date_and_time_assignment
date_time_role
local_time
mechanical_context
organization
person
person_and_organization
*person_and_organization_assignment
person_and_organization_role
product
product_category
product_category_relationship
*product_context
product_definition
product_definition_context (or design_context)
*product_definition_formation
product_definition_formation_with_specified_source
product_definition_shape
product_related_product_category
*property_definition
security_classification
*security_classification_assignment
security_classification_level

F.2 Product Identification, Structure and Effectivity

This subset is for use in identifying parts and assembly component structures with no engineering change data that involve a fully developed parts list with design perspective application data (effectivity). This subset provides all the various types of AP 203 product relationships with fully represented quantities. This subset would include all the entities listed in section E.1 on page 69 with the following additions:

alternate_product_relationship
area_measure_with_unit
area_unit
*assembly_component_usage
assembly_component_usage_substitute
cc_design_certification
certification
*certification_assignment
certification_type
configuration_design
configuration_effectivity
configuration_item
context_dependent_unit
conversion_based_unit
dated_effectivity
design_make_from_relationship
dimensional_exponents
*effectivity
length_measure_with_unit
length_unit
lot_effectivity
mass_measure_with_unit
mass_unit
*measure_with_unit
*named_unit
next_assembly_usage_occurrence
product_concept
product_concept_context
product_definition_effectivity
product_definition_relationship
*product_definition_usage
promissory_usage_occurrence
quantified_assembly_component_usage
serial_numbered_effectivity
*si_unit
specified_higher_usage_occurrence
supplied_part_relationship
volume_measure_with_unit
volume_unit

F.3 Engineering Change Identification

This subset is for use in identifying parts versions and their relationships. If it is used, it must be used in concert with either the rudimentary or effectivity subsets from the prior sections. This subset would involve the entities from sections E.1 and/or E.2 with:

*action
*action_assignment
action_directive
action_method
*action_request_assignment
action_request_solution
action_request_status
action_status
change
change_request
directed_action
*executed_action
start_request
start_work
versioned_action_request

F.4 Conformance Class 1 Entities

This section includes the conformance class 1 entities not in the previous subsets. The entities listed here when taken with the entities in sections E.1, E.2, and E.3 comprise all of conformance class 1. The entities in this section are, generally speaking, optional in the AP. They are used for addresses, sundry relationships, contract and document references.

*address
approval_relationship
cc_design_contract
cc_design_specification_reference
contract
*contract_assignment
contract_type
document
*document_reference
document_relationship
document_type
document_usage_constraint
document_with_class
organization_relationship
organizational_address
organizational_project

personal_address
product_definition_with_associated_documents
shape_aspect
shape_aspect_relationship

Appendix G - Schema Extensions

AP 203 is an International Standard STEP Application Protocol. It provides a method to represent data in a standard form for exchange and sharing. The schema in AP 203 is designed to meet the scope and requirements described in the AP document. Individual companies and teaming partners may have either internal data or mutually agreed upon team specific data which is non-standard, but required for day to day operations and archival.

Obviously AP 203, being a standard, will not always be capable of representing unique operational data for a company or team. The unique operational data can be added as extensions to the AP schema. This appendix of this document will describe how extensions can be made to any STEP Application Protocol to provide extended data representation capabilities.

One example of company or team unique data is codes. These pieces of information are short hand representations for data which typically can be represented in AP 203. An example would be a material code. AP 203 provides for attaching material specifications to parts and aspects of parts. This reference is computer identifiable, but requires that the material designation be made to some externally referenced specification either with or without a usage constraint. Materials can in fact be referenced this way, but in most material systems, this is accomplished by a code to save data space. These codes also drive downstream processes and therefore are critical pieces of data.

The code information could be derived by interpreting the material specification and associated usage constraints and matching it to the proper code. This is highly inefficient if we know that the user of our AP 203 data will need the code data. The easiest way to maintain this data is to extend the schema so that the codes can be maintained. The only problem with doing this is that the extension is non-standard by definition.

STEP does provide a method for making these extensions within the framework of the standard. This method is partially described in ISO 10303-21. ISO 10303-21 allows for the creation of user defined entities which are outside the EXPRESS schemas specified in the header section of the file. When user defined entities are instanced in a physical file, the entity name is preceded by an exclamation mark (!). This delimiter is a signal to receiving processors that this is a user defined entity and should be ignored unless recognized by the receiving processor.

The description in ISO 10303-21 is only partial as it does not provide any guidance on how to structure these entities so that they do not accidentally limit the standard exchange specified by the EXPRESS schemas that <u>ARE</u> specified in the header section.

This appendix of this document will describe how to structure schema extensions so that they are only extensions. This is important as there are many ways to create entities in EXPRESS. For example, if we use AP 203 as our base and create and entity named **product_with_code** with the

attribute **code** which is a sub-type of **product** and use it in all our exchanges, we will have problems in exchanges with standard systems due to the mapping of EXPRESS to the physical file since the **product** entity instance in the physical file will disappear and be replaced by the **product_with_code** entity instance.

To eliminate this problem, extensions to the schema should be entity instances which point <u>at</u> standard entity instances. With this methodology, the prior example would be changed so that **product_with_code** is a stand alone entity (i.e. not a sub-type) and would reference the **product** entity. By doing this, we now would end up with an extra entity in the physical file which would reference the product entity with the **code**. This extra entity can be ignored by standard AP 203 translators with no effect on the standard exchange.

Even with the above methodology extensions should <u>NOT</u> be used simply as a matter of convenience. For example, if the native system has a material code for material and the description of the material is in another system, the processor developer should not use dummy information for the material specification and put the code in a user defined entity.

Schema extensions should be carefully evaluated since the information will be lost in all standard exchanges. Schema extensions for information which a processor developer has determined not to exist in AP 203 should <u>ONLY</u> be considered after an issue is raised against the AP, and the appropriate AP authority has determined that the proposed information should be representable but has been omitted. If the information in question is out of the scope of AP 203 but is in scope for another STEP AP, the processor developer should seriously consider using the second AP to represent the information before implementing an extension to the AP 203 schema to represent the data.

G.1 Schema Extensions Currently in Use

This section lists known AP 203 schema extensions that are in use. These schema extensions are user defined entities which are being used as pure extensions to the AP 203 schemas or are being used as a work around to problems found in AP 203. The reason for their existence is noted in the entity definition.

G.1.1 stocksize

The **stocksize** entity defines the size of the raw stock (input material) needed to create a part. This entity has been defined to record data which is stored in many configuration management systems, but is beyond the current scope of AP 203 since it is a manufacturing shape definition.

```
*)
ENTITY stocksize;
  part_on_parts_list  : product_definition_relationship;
  size : STRING;
END_ENTITY;
(*
```

Attribute definitions:

part_on_parts_list: identifies the particular part to parts list relationship (see section 3.3.7 on page 60) to which the **size** applies.

size: identifies the manufacturing requirements information needed for the proper formation of the part in the **part_on_parts_list** relationship.

G.1.2 qfa

A **qfa** entity identifies a final assembly quantity for a part in relation to its usage on a given drawing for a sub-section of a product model in a specified effectivity range. This entity has been defined as a work around technique for drawing based effectivities since the drawing instance on which the quantity is based may not be totally defined (as and AP 203 **product**) when the relationship is established.

The **next_assembly_drawing**, **section**, **model_id**, **from_eff**, and **thru_eff** values form a unique identification for the part **quantity**.

```
*)
ENTITY qfa;
  definition              : product_definition;
  next_assembly_drawing   : STRING;
  section                 : STRING;
  model_id                : STRING;
  from_eff                : STRING;
  thru_eff                : STRING;
  quantity                : INTEGER;
END_ENTITY;
(*
```

Attribute definitions:

definition: identifies the discipline view definition of the part to which the final assembly quantity applies.

next_assembly_drawing: identifies a drawing number on which the part is or is intended to be used.

section: identifies a sub-section of the product model which utilizes the part or parts defined on the **next_assembly_drawing**.

model_id: identifies the product model which contains the **section**.

from_eff: identifies a starting serial number for the **model_id**.

thru_eff: identifies an ending serial number for the **model_id**.
G.1.3 drawing_sheets

A **drawing_sheet** entity identifies a single formatted page of a drawing which can be used on many revisions of a drawing. This entity has been defined to capture drawing unique data which is outside the scope of AP 203.

```
*)
ENTITY drawing_sheet;
  sheet_size         :    OPTIONAL STRING;
  sheet_number       :    STRING;
  sheet_rev          :    STRING;
  assigned_dwg_revs  :    SET [1:?] OF product_definition_formation;
  frames             :    OPTIONAL LIST [1:?] OF UNIQUE STRING;
  cad_filename       :    OPTIONAL STRING;
  sec_class          :    OPTIONAL security_classification;
END_ENTITY;
(*
```

Attribute definitions:

sheet_size: identifies the length and width dimensions of the sheet based on a code designation described in the drafting standard identified on the drawing.

sheet_number: identifies the sheet.

sheet_rev: identifies the change level of the sheet.

assigned_dwg_revs: identifies revisions of the drawing to which the sheet applies.

frames: identifies the aperture cards on which the drawing sheet was microfilmed.

cad_filename: identifies the computer aided design model file which contains the drawing sheet data.

sec_class: identifies the **security_classification** of the drawing sheet.

G.1.4 design_make_from_relationship_substitute

A **design_make_from_relationship_substitute** relates two **design_make_from_relationship**s where the **substitute** may be used in place of the **base**. A **design_make_from_relationship_substitute** instance indicates that a part can be made from two different source parts. This entity has been defined to provide a work around to a problem in the current release of AP 203 which does not support substitutions for make from relationships.

```
*)
```

```
ENTITY design_make_from_relationship_substitute;
  name        : label;
  description : text;
  base        : design_make_from_relationship;
  substitute  : design_make_from_relationship;
UNIQUE
  UR1 : base, substitute;
WHERE
  WR1: base.related_product_definition :=:
       substitute.related_product_definition;
  WR2: base :<>: substitute;
END_ENTITY;
(*
```

Attribute definitions:

name: identifies a designation for the **design_make_from_relationship_substitute**.

description: identifies the characterization of the **design_make_from_relationship_substitute**.

base: identifies the **design_make_from_relationship** which represents the preferred source part for the make from operation.

substitute: identifies the **design_make_from_relationship** which may be used in place of the **base**.

Formal propositions:

UR1: The combination of **base** and **substitute** shall be unique.

WR1: The same **product_definition** instance shall be identified by the **related_product_definition** attributes of the **base** and **substitute** **design_make_from_relationship**s.

WR2: The **base** and **substitute** shall not be the same **design_make_from_relationship** instance.

G.1.5 type_affect

A **type_affect** identifies the kind of a release or change.

```
*)
TYPE type_affect= ENUMERATION OF (
  add,
  revise,
  delete);
END_TYPE;
(*
```

Enumerated item definitions:

add: The indication that the item affected by the release or change is being created.

revise: The indication that the item affected by the release or change is being modified.
delete: The indication that the item affected by the release or change is being removed.

G.1.6 affected_item_select

An **affected_item_select** identifies an **alternate_product_relationship**, **assembly_component_usage_substitute**, **cc_design_specification_reference**, **design_make_from_relationship**, **document**, **drawing_sheet**, **design_make_from_relationship_substitute**, **product_definition_formation_with_specified_source**, **product_definition**, **product_definition_relationship**, **qfa**,or a **stocksize** as being added, revised or deleted due to an engineering change .

```
*)
TYPE  affected_item_select = SELECT (
  alternate_product_relationship,
  assembly_component_usage_substitute,
  cc_design_specification_reference,
  design_make_from_relationship,
  directed_action,
  document,
  drawing_sheet,
  design_make_from_relationship_substitute,
  product_definition_formation_with_specified_source,
  product_definition,
  product_definition_relationship,
  supplied_part_relationship
  serial_numbered_effectivity,
  qfa,
  stocksize,
  versioned_action_request);
END_TYPE;
(*
```

G.1.7 engineering_change_affect

An **engineering_change_affect** relates a **versioned_action_request** to an item it affects and assigns a modifier to determine whether the action is and add, change or delete and identifies any residual affects.

```
*)
ENTITY engineering_change_affect;
  affect_type : type_affect;
  affect_cause : versioned_action_request;
  affected_item : affected_item_select;
```

```
  description: OPTIONAL STRING;
  related_affects : LIST [0:?] OF engineering_change_affect;
END_ENTITY;
(*
```

Attribute definitions:

affect_type: An indicator as to what kind of action is occurring.

affect_cause: The **versioned_action_request** which initiated the action.
affected_item: The data for the product which is affected by the **versioned_action_request**.

description: A phrase characterizing the action being taken.

related_affects: A group of product data which is residually affected by the action.

G.1.8 constrained_document_reference

A **constrained_document_reference** a constraint to the usage of a document.

```
*)
ENTITY constrained_document_reference;
  source : document_reference;
  subject_element : label;
  subject_element_value : text;
END_ENTITY;
(*
```

Attribute definitions:

source: The **document_reference** being constrained.

subject_element: The section or portion of the referenced document that is being constrained.

subject_element_value: The value for the constraint.

Appendix H - Implementors Agreements

This appendix lists known agreements being proposed/used in the implementation of AP 203 for data exchange/sharing. The agreements listed in this appendix are being employed on a test basis by some number of software developers. They are provided for the reader's information. The agreements are currently authored by the roundtables of the PDES, Inc. and ProSTEP consortia. All agreements listed here, if proven viable, will be proposed as formal implementors agreements and/or issues against the standard through the SC4 Enhancement/Discrepancy System (SEDS) which provides formal tracking of issues against the standard.

The reader should also monitor SEDS for status on implementors agreements and issues against STEP and AP 203. SEDS reports can be accessed via the Internet by anonymous FTP to 129.6.32.54. The current reports are maintained in the directory /pub/step/sc4docs/seds/reports. In this directory, issue reports take the form of syynnnnn.txt where s is the status which is "O" for open and "C" for closed, yy is the last two digits of the year, nnnnn is a sequential number.

H.1 Potential Agreements

H.1.1 Surface, Seam, and Intersection Curves for Advanced_face

Description: I) Surface, seam, and intersection curves are not allowed as edge basis curves (WR5) in an advanced_face; ii) Pcurves are allowed but cannot store all of the information needed for an edge basis curve.

Agreement: Add surface, seam, and intersection curves to the set of allowed edge basis curves. Receiving system chooses whether to use the 3D curves or recalculate them. Always use a surface curve between an edge and pcurve. Trimmed curves should never be referenced, even indirectly through a surface, seam, or intersections curve.

H.1.2 Handling of Line

Description: Some systems allow a bspline curve with degree 1 instead of explicitly defining a line.

Agreement: A line entity must be explicitly defined as 1) an unbounded straight line or 2) a trimmed curve whose underlying curve is a line trimmed by curve parameters.

H.1.3 Handling of Bounded Pcurve

Description: There are three ways to define a bounded pcurve:

1. a pcurve which references a bounded curve, 2. a pcurve AND bounded curve, and 3. a bounded pcurve.

Agreement: Use the first option and reference a bounded curve. Never use the second option.

H.1.4 Handling of Bounded Surface Curve

Description: There are three ways to define a bounded surface curve:
1. a surface curve whose 3D curve is a bounded curve, 2. a surface curve AND bounded curve, and 3. a bounded surface curve.

Agreement: Use the first option and reference a bounded curve. Never use the second option.

H.1.5 Conical Surface Parametrisation

Description: The parametrisation given in Part 42 makes use of Tan(semi-angle). This becomes less stable as the semi-angle approaches 90 degrees.

Agreement: Use a surface_of_revolution for cones with large semi-angles.

H.1.6 Trimming of Cylinder by Plane

Description: If a cylinder is trimmed by a plane more or less paralleled to the axis of the cylinder, the trimming may be a line or an ellipse depending on the post-processor's interpretation.

Agreement: Use a line if the axis and plane are parallel within the uncertainty_measure_with_unit.

H.1.7 Face_Outer_Bound Designation for Closed Periodic Surfaces

Description: The face_outer_bound designation is ambiguous for edges of closed periodic surfaces used in cylinders, spheres and torii.

Agreement: Ignore this information when reading entities with closed periodic surfaces from STEP files and do not include it when writing these same entities. Continue using the designation for unambiguous entities such as open or planar surfaces.

H.1.8 Discontinuity within a Face

Description: The standard does not restrict the geometry of a face from being represented by a surface with tangency discontinuities such as a surface of revolution based on triangle segments.

Agreement: Whenever a topology-based surface model or a boundary representation contains tangency discontinuities, these discontinuities shall be reflected in the topology part of the model by corresponding vertices or edges.

H.1.9 Global_Unit_Assigned_Context and Entity Parameterisations

Description: I)   It is unclear whether global_unit_assigned_context applies to entity parameterisations; ii) Part 42 angle parameter examples suggest they must always be in degrees; iii) Radian units are irrational and therefore potentially unstable for angle parameterisations.

Agreement: Consistently use the plane_angle_unit (degree or radian) referenced by global_unit_assigned_context for all entity model angles and parameterisations when importing STEP models. It is preferred to use degrees for plane_angle_unit when exporting. Clarify the Part 42 text.

H.1.10 Use of Scope in Part 21

Description: Semantics of the scope construct are unclear and optional.

Agreement: Processors should be able to read this construct, but should not use it when writing STEP files.

H.1.11 External Mapping in All Classes

Description: The IS version of Part 21 includes a second implementation class which requires the use of the external mapping for entities involving sub/supertypes.

Agreement: Do not use this implementation class.

H.1.12 Use of Optional Attributes in Geometry

Description: Some geometric entities include optional attributes. How should these be used when writing and reading STEP files?

Agreement: When writing STEP files (exporting) provide values unless there is a defined default. Use these attributes as information only when reading (importing).

H.1.13 Minimum Topology of a Sphere

Description: Part 42 does not specify a fixed topology for a sphere.

Agreement: The recommended minimum topology is a single vertex loop located at the North pole of the sphere as determined by the parametrisation of the underlying spherical surface. Those systems that add a seam to the sphere may continue to do so.

H.1.14 Drawings and Associated Lists

Description: AP 203 does not address drawings and APs 201/202 do not address drawing change control.

Agreement: Identify drawings and lists as products and use 203 configuration management data to control them.

H.1.15 Mapped_Item

Description: The mapped_item construct allows for any representation_item type entity to be used in defining the mapping transformation. This is an unnecessary generalization for rigid body transformations and creates problems in determining the overall structure of the transformation.

Agreement: When using mapped_item, all references to representation_items shall be to axis2_placement_3d entities.

H.1.16 Definitional_Representation

Description: The global rule SUBTYPE_MANDATORY_SHAPE_REPRESENTATION is viewed as too restrictive in the case of the use of a PCURVE construct. In the event that the SHAPE_REPRESENTATION is a DEFINITIONAL_REPRESENTATION, violations of this rule will be passed.

H.1.17 Geometric_Representation_Item_3D

Description: The global rule GEOMETRIC_REPRESENTATION_ITEM_3D is also viewed as too restrictive in the case of the use of a PCURVE construct. It is the considered majority opinion of vendors that in the event of SHAPE_REPRESENTATION that is a DEFINITIONAL_REPRESENTATION, violations of this rule will be passed on those instances of GEOMETRIC_REPRESENTATION_ITEM making up the underlying geometry of the curve referenced by : pcurve.reference_to_curve\representation.items.

H.1.18 Shape_Representation

Description: In implementations of AP 203, shape_representation instances may be used to represent the shape of a part where the items in the representation do NOT satisfy the rules of any of the other SUBTYPEs. This technique has been introduced as a harmonization with AP 214 and is intended for a future edition of AP 203.

Appendix I - Entity/Type to STEP Parts

This appendix lists all the entities and types used in AP 203 and cross references them to the STEP part in which they are defined. The first column is the entity or type name. The second column defines whether it is an entity or type. The last column lists the STEP part in which the full textual definition of the item exists. If the entry in the last column end in an asterisk (*), the description of the entity has been modified by AP 203 in clause 5.2.3.2 in the AP. If the entry in the last column ends in /AIC, the item is a proposed Application Interpreted Construct (AIC).

NAME	KIND	PART
ACTION	ENTITY	41*
ACTION_ASSIGNMENT	ENTITY	41
ACTION_DIRECTIVE	ENTITY	41*
ACTION_METHOD	ENTITY	41
ACTION_REQUEST_ASSIGNMENT	ENTITY	41
ACTION_REQUEST_SOLUTION	ENTITY	41*
ACTION_REQUEST_STATUS	ENTITY	41*
ACTION_STATUS	ENTITY	41
ADDRESS	ENTITY	41
ADVANCED_BREP_SHAPE_REPRESENTATION	ENTITY	203/AIC
ADVANCED_FACE	ENTITY	203/AIC
AHEAD_OR_BEHIND	TYPE	41
ALTERNATE_PRODUCT_RELATIONSHIP	ENTITY	44
APPLICATION_CONTEXT	ENTITY	41*
APPLICATION_CONTEXT_ELEMENT	ENTITY	41
APPLICATION_PROTOCOL_DEFINITION	ENTITY	41*
APPROVAL	ENTITY	41*
APPROVAL_ASSIGNMENT	ENTITY	41*
APPROVAL_DATE_TIME	ENTITY	41*
APPROVAL_PERSON_ORGANIZATION	ENTITY	41*
APPROVAL_RELATIONSHIP	ENTITY	41
APPROVAL_ROLE	ENTITY	41
APPROVAL_STATUS	ENTITY	41*
APPROVED_ITEM	TYPE	203
AREA_MEASURE	TYPE	41
AREA_MEASURE_WITH_UNIT	ENTITY	41
AREA_UNIT	ENTITY	41
ASSEMBLY_COMPONENT_USAGE	ENTITY	44
ASSEMBLY_COMPONENT_USAGE_SUBSTITUTE	ENTITY	44
AXIS1_PLACEMENT	ENTITY	42
AXIS2_PLACEMENT	TYPE	42
AXIS2_PLACEMENT_2D	ENTITY	42
AXIS2_PLACEMENT_3D	ENTITY	42
BEZIER_CURVE	ENTITY	42
BEZIER_SURFACE	ENTITY	42
BOOLEAN_OPERAND	TYPE	42
BOUNDARY_CURVE	ENTITY	42
BOUNDED_CURVE	ENTITY	42
BOUNDED_SURFACE	ENTITY	42
BREP_WITH_VOIDS	ENTITY	42

```
B_SPLINE_CURVE                                          ENTITY   42
B_SPLINE_CURVE_FORM                                     TYPE     42
B_SPLINE_CURVE_WITH_KNOTS                               ENTITY   42
B_SPLINE_SURFACE                                        ENTITY   42
B_SPLINE_SURFACE_FORM                                   TYPE     42
B_SPLINE_SURFACE_WITH_KNOTS                             ENTITY   42
CALENDAR_DATE                                           ENTITY   41
CARTESIAN_POINT                                         ENTITY   42
CARTESIAN_TRANSFORMATION_OPERATOR                       ENTITY   42
CARTESIAN_TRANSFORMATION_OPERATOR_2D                    ENTITY   42
CARTESIAN_TRANSFORMATION_OPERATOR_3D                    ENTITY   42
CC_DESIGN_APPROVAL                                      ENTITY   203
CC_DESIGN_CERTIFICATION                                 ENTITY   203
CC_DESIGN_CONTRACT                                      ENTITY   203
CC_DESIGN_DATE_AND_TIME_ASSIGNMENT                      ENTITY   203
CC_DESIGN_PERSON_AND_ORGANIZATION_ASSIGNMENT            ENTITY   203
CC_DESIGN_SECURITY_CLASSIFICATION                       ENTITY   203
CC_DESIGN_SPECIFICATION_REFERENCE                       ENTITY   203
CERTIFICATION                                           ENTITY   41*
CERTIFICATION_ASSIGNMENT                                ENTITY   41
CERTIFICATION_TYPE                                      ENTITY   41*
CERTIFIED_ITEM                                          TYPE     203
CHANGE                                                  ENTITY   203
CHANGE_REQUEST                                          ENTITY   203
CHANGE_REQUEST_ITEM                                     TYPE     203
CHARACTERIZED_DEFINITION                                TYPE     41
CHARACTERIZED_PRODUCT_DEFINITION                        TYPE     41
CIRCLE                                                  ENTITY   42
CLASSIFIED_ITEM                                         TYPE     203
CLOSED_SHELL                                            ENTITY   42
COMPOSITE_CURVE                                         ENTITY   42
COMPOSITE_CURVE_ON_SURFACE                              ENTITY   42
COMPOSITE_CURVE_SEGMENT                                 ENTITY   42
CONFIGURATION_DESIGN                                    ENTITY   44
CONFIGURATION_EFFECTIVITY                               ENTITY   44
CONFIGURATION_ITEM                                      ENTITY   44*
CONIC                                                   ENTITY   42
CONICAL_SURFACE                                         ENTITY   42
CONNECTED_EDGE_SET                                      ENTITY   42
CONNECTED_FACE_SET                                      ENTITY   42
CONTEXT_DEPENDENT_MEASURE                               TYPE     41
CONTEXT_DEPENDENT_SHAPE_REPRESENTATION                  ENTITY   41*
CONTEXT_DEPENDENT_UNIT                                  ENTITY   41
CONTRACT                                                ENTITY   41*
CONTRACTED_ITEM                                         TYPE     203
CONTRACT_ASSIGNMENT                                     ENTITY   41
CONTRACT_TYPE                                           ENTITY   41*
CONVERSION_BASED_UNIT                                   ENTITY   41
COORDINATED_UNIVERSAL_TIME_OFFSET                       ENTITY   41
COUNT_MEASURE                                           TYPE     41
CURVE                                                   ENTITY   42
CURVE_BOUNDED_SURFACE                                   ENTITY   42
CURVE_ON_SURFACE                                        TYPE     42
CURVE_REPLICA                                           ENTITY   42
CYLINDRICAL_SURFACE                                     ENTITY   42
DATE                                                    ENTITY   41*
DATED_EFFECTIVITY                                       ENTITY   41
DATE_AND_TIME                                           ENTITY   41
```

NAME	KIND	PART
DATE_AND_TIME_ASSIGNMENT	ENTITY	41
DATE_TIME_ITEM	TYPE	203
DATE_TIME_ROLE	ENTITY	41*
DATE_TIME_SELECT	TYPE	41
DAY_IN_MONTH_NUMBER	TYPE	41
DAY_IN_WEEK_NUMBER	TYPE	41
DAY_IN_YEAR_NUMBER	TYPE	41
DEFINITIONAL_REPRESENTATION	ENTITY	43
DEGENERATE_PCURVE	ENTITY	42
DESCRIPTIVE_MEASURE	TYPE	41
DESIGN_CONTEXT	ENTITY	203/AIC
DESIGN_MAKE_FROM_RELATIONSHIP	ENTITY	203
DIMENSIONAL_EXPONENTS	ENTITY	41
DIMENSION_COUNT	TYPE	42
DIRECTED_ACTION	ENTITY	41
DIRECTION	ENTITY	42
DOCUMENT	ENTITY	41
DOCUMENT_REFERENCE	ENTITY	41
DOCUMENT_RELATIONSHIP	ENTITY	41
DOCUMENT_TYPE	ENTITY	41*
DOCUMENT_USAGE_CONSTRAINT	ENTITY	41
DOCUMENT_WITH_CLASS	ENTITY	41
EDGE	ENTITY	42
EDGE_BASED_WIREFRAME_MODEL	ENTITY	42
EDGE_BASED_WIREFRAME_SHAPE_REPRESENTATION	ENTITY	203/AIC
EDGE_CURVE	ENTITY	42
EDGE_LOOP	ENTITY	42
EFFECTIVITY	ENTITY	41*
ELEMENTARY_SURFACE	ENTITY	42
ELLIPSE	ENTITY	42
EVALUATED_DEGENERATE_PCURVE	ENTITY	42
EXECUTED_ACTION	ENTITY	41
FACE	ENTITY	42
FACETED_BREP	ENTITY	42
FACETED_BREP_SHAPE_REPRESENTATION	ENTITY	203/AIC
FACE_BOUND	ENTITY	42
FACE_OUTER_BOUND	ENTITY	42
FACE_SURFACE	ENTITY	42
FUNCTIONALLY_DEFINED_TRANSFORMATION	ENTITY	43
GEOMETRICALLY_BOUNDED_SURFACE_SHAPE_REPRESENTATION	ENTITY	203/AIC
GEOMETRICALLY_BOUNDED_WIREFRAME_SHAPE_REPRESENTATION	ENTITY	203/AIC
GEOMETRIC_CURVE_SET	ENTITY	42
GEOMETRIC_REPRESENTATION_CONTEXT	ENTITY	42
GEOMETRIC_REPRESENTATION_ITEM	ENTITY	42*
GEOMETRIC_SET	ENTITY	42
GEOMETRIC_SET_SELECT	TYPE	42
GLOBAL_UNCERTAINTY_ASSIGNED_CONTEXT	ENTITY	43
GLOBAL_UNIT_ASSIGNED_CONTEXT	ENTITY	41*
HOUR_IN_DAY	TYPE	41
HYPERBOLA	ENTITY	42
IDENTIFIER	TYPE	41

90

NAME	KIND	PART
PLANE	ENTITY	42
PLANE_ANGLE_MEASURE	TYPE	41
PLANE_ANGLE_MEASURE_WITH_UNIT	ENTITY	41
PLANE_ANGLE_UNIT	ENTITY	41
POINT	ENTITY	42
POINT_ON_CURVE	ENTITY	42
POINT_ON_SURFACE	ENTITY	42
POINT_REPLICA	ENTITY	42
POLYLINE	ENTITY	42
POLY_LOOP	ENTITY	42
POSITIVE_LENGTH_MEASURE	TYPE	41
POSITIVE_PLANE_ANGLE_MEASURE	TYPE	41
PREFERRED_SURFACE_CURVE_REPRESENTATION	TYPE	42
PRODUCT	ENTITY	41*
PRODUCT_CATEGORY	ENTITY	41*
PRODUCT_CATEGORY_RELATIONSHIP	ENTITY	41
PRODUCT_CONCEPT	ENTITY	44*
PRODUCT_CONCEPT_CONTEXT	ENTITY	41
PRODUCT_CONTEXT	ENTITY	41*
PRODUCT_DEFINITION	ENTITY	41*
PRODUCT_DEFINITION_CONTEXT	ENTITY	41*
PRODUCT_DEFINITION_EFFECTIVITY	ENTITY	41
PRODUCT_DEFINITION_FORMATION	ENTITY	41*
PRODUCT_DEFINITION_FORMATION_WITH_SPECIFIED_SOURCE	ENTITY	41
PRODUCT_DEFINITION_RELATIONSHIP	ENTITY	41
PRODUCT_DEFINITION_SHAPE	ENTITY	41
PRODUCT_DEFINITION_USAGE	ENTITY	44*
PRODUCT_DEFINITION_WITH_ASSOCIATED_DOCUMENTS	ENTITY	41
PRODUCT_RELATED_PRODUCT_CATEGORY	ENTITY	41*
PROMISSORY_USAGE_OCCURRENCE	ENTITY	44
PROPERTY_DEFINITION	ENTITY	41
PROPERTY_DEFINITION_REPRESENTATION	ENTITY	41
QUANTIFIED_ASSEMBLY_COMPONENT_USAGE	ENTITY	44
QUASI_UNIFORM_CURVE	ENTITY	42
QUASI_UNIFORM_SURFACE	ENTITY	42
RATIONAL_B_SPLINE_CURVE	ENTITY	42
RATIONAL_B_SPLINE_SURFACE	ENTITY	42
RECTANGULAR_COMPOSITE_SURFACE	ENTITY	42
RECTANGULAR_TRIMMED_SURFACE	ENTITY	42
REPARAMETRISED_COMPOSITE_CURVE_SEGMENT	ENTITY	42
REPRESENTATION	ENTITY	43*
REPRESENTATION_CONTEXT	ENTITY	43*
REPRESENTATION_ITEM	ENTITY	43*
REPRESENTATION_MAP	ENTITY	43
REPRESENTATION_RELATIONSHIP	ENTITY	43
REPRESENTATION_RELATIONSHIP_WITH_TRANSFORMATION	ENTITY	43
REVERSIBLE_TOPOLOGY	TYPE	42
REVERSIBLE_TOPOLOGY_ITEM	TYPE	43
SEAM_CURVE	ENTITY	43
SECOND_IN_MINUTE	TYPE	41
SECURITY_CLASSIFICATION	ENTITY	41
SECURITY_CLASSIFICATION_ASSIGNMENT	ENTITY	41*
SECURITY_CLASSIFICATION_LEVEL	ENTITY	41*
SERIAL_NUMBERED_EFFECTIVITY	ENTITY	41
SET_OF_REVERSIBLE_TOPOLOGY_ITEM	TYPE	42
SHAPE_ASPECT	ENTITY	41

NAME	KIND	PART
SHAPE_ASPECT_RELATIONSHIP	ENTITY	41
SHAPE_DEFINITION	TYPE	41
SHAPE_DEFINITION_REPRESENTATION	ENTITY	41
SHAPE_REPRESENTATION	ENTITY	41*
SHAPE_REPRESENTATION_RELATIONSHIP	ENTITY	41
SHELL	TYPE	42
SHELL_BASED_SURFACE_MODEL	ENTITY	42
SHELL_BASED_WIREFRAME_MODEL	ENTITY	42
SHELL_BASED_WIREFRAME_SHAPE_REPRESENTATION	ENTITY	203/AIC
SI_PREFIX	TYPE	41
SI_UNIT	ENTITY	41
SI_UNIT_NAME	TYPE	41
SOLID_ANGLE_MEASURE	TYPE	41
SOLID_ANGLE_MEASURE_WITH_UNIT	ENTITY	41
SOLID_ANGLE_UNIT	ENTITY	41
SOLID_MODEL	ENTITY	42
SOURCE	TYPE	41
SPECIFIED_HIGHER_USAGE_OCCURRENCE	ENTITY	44
SPECIFIED_ITEM	TYPE	203
SPHERICAL_SURFACE	ENTITY	42
START_REQUEST	ENTITY	203
START_REQUEST_ITEM	TYPE	203
START_WORK	ENTITY	203
SUPPLIED_PART_RELATIONSHIP	ENTITY	203
SUPPORTED_ITEM	TYPE	41
SURFACE	ENTITY	42
SURFACE_CURVE	ENTITY	42
SURFACE_MODEL	TYPE	42
SURFACE_OF_LINEAR_EXTRUSION	ENTITY	42
SURFACE_OF_REVOLUTION	ENTITY	42
SURFACE_PATCH	ENTITY	42
SURFACE_REPLICA	ENTITY	42
SWEPT_SURFACE	ENTITY	42
TEXT	TYPE	41
TOPOLOGICAL_REPRESENTATION_ITEM	ENTITY	42
TOROIDAL_SURFACE	ENTITY	42
TRANSFORMATION	TYPE	43
TRANSITION_CODE	TYPE	42
TRIMMED_CURVE	ENTITY	42
TRIMMING_PREFERENCE	TYPE	42
TRIMMING_SELECT	TYPE	42
UNCERTAINTY_MEASURE_WITH_UNIT	ENTITY	43
UNIFORM_CURVE	ENTITY	42
UNIFORM_SURFACE	ENTITY	43
UNIT	TYPE	41
VECTOR	ENTITY	42
VECTOR_OR_DIRECTION	TYPE	42
VERSIONED_ACTION_REQUEST	ENTITY	41*
VERTEX	ENTITY	42
VERTEX_LOOP	ENTITY	42
VERTEX_POINT	ENTITY	42
VERTEX_SHELL	ENTITY	42
VOLUME_MEASURE	TYPE	41
VOLUME_MEASURE_WITH_UNIT	ENTITY	41
VOLUME_UNIT	ENTITY	41

```
WEEK_IN_YEAR_NUMBER                              TYPE     41
WEEK_OF_YEAR_AND_DAY_DATE                         ENTITY   41
WIREFRAME_MODEL                                   TYPE     42
WIRE_SHELL                                        ENTITY   42
WORK_ITEM                                         TYPE     203
YEAR_NUMBER                                       TYPE     41
```

Appendix J - Recommendation Compatibility

This appendix list any recommendations that have been changed for the purposes of upward compatibility. If a recommendation existed in the past it is probable that processors may have utilized that recommendation. Current processors should be aware of these recommendations as data of these types may occur in files.

J.1 Parts Composed of Multiple Shape Representations

NOTE - From editions prior to July 7, 1997

If the shape of the part is composed of shape constructs from multiple types of **shape_representation** to form the entire shape model, the main **shape_representation** shall be related to a **shape_definition_representation** which relates to the **product_definition_shape**. The other **shape_representations** are related to the **product_definition_shape** through **shape_aspect**s. This is shown in Figure 21.

Index