NSRP

SHIP STRUCTURE

APPLICATION PROTOCOL

Annex A

Arrangements

Version 2.0
30 September 1999

Table of Contents

TABLE OF CONTENTS **2**

TABLE OF FIGURES **7**

A. SHIP ARRANGEMENTS **8**

A.1 TECHNICAL DISCUSSION 8
A.1.1 Arrangement (Internal Subdivision) *8*
A.1.2 Compartments *9*
A.1.3 Zones *10*
A.1.3.1 Design Zones 10
A.1.3.2 Fire Zones 11
A.1.3.3 Collective Protection System (CPS) Zones 11
A.1.3.4 Arrangement Zones 11
A.2 INFORMATION CONTENT 13
A.2.1 Application Objects *13*
A.2.1.1 address 13
A.2.1.2 advanced_face 14
A.2.1.3 area_unit 14
A.2.1.4 axis2_placement_2d 14
A.2.1.5 axis2_placement_3d 14
A.2.1.6 b_spline_curve 15
A.2.1.7 b_spline_curve_with_knots 16
A.2.1.8 b_spline_surface 17
A.2.1.9 b_spline_surface_with_knots 18
A.2.1.10 bounded_curve 19
A.2.1.11 bounded_pcurve 19
A.2.1.12 bounded_surface 20
A.2.1.13 bounded_surface_curve 20
A.2.1.14 calendar_date 20
A.2.1.15 Capacity_properties 20
A.2.1.16 Cargo_bay_definition 21
A.2.1.17 Cargo_compartment_design_definition 22
A.2.1.18 Cargo_compartment_functional_definition 22
A.2.1.19 Cargo_compartment_property_set 22
A.2.1.20 cartesian_point 23
A.2.1.21 circle 23
A.2.1.22 closed_shell 23
A.2.1.23 Compartment 23
A.2.1.24 Compartment_areas 23
A.2.1.25 Compartment_design_definition 24
A.2.1.26 Compartment_design_requirement 25
A.2.1.27 Compartment_design_to_functional_definition_relationship 26
A.2.1.28 Compartment_extents 26
A.2.1.29 Compartment_functional_definition 27
A.2.1.30 Compartment_naval_administrative_property_set 28
A.2.1.31 Compartment_permeability 32
A.2.1.32 Compartment_property_set 32
A.2.1.33 Compartment_property_to_design_definition_relationship 33
A.2.1.34 Compartment_volume 33
A.2.1.35 composite_curve 33
A.2.1.36 composite_curve_on_surface 34
A.2.1.37 composite_curve_segment 34
A.2.1.38 conic 35
A.2.1.39 connected_face_set 35
A.2.1.40 coordinated_universal_time_offset 35
A.2.1.41 curve 36
A.2.1.42 date 36
A.2.1.43 date_and_time 36

A.2.1.44 Definable_object 36
A.2.1.45 Definition 37
A.2.1.46 Definition_relationship 37
A.2.1.47 Derived_Unit 37
A.2.1.48 Derived_Unit_Element 38
A.2.1.49 Design_definition 38
A.2.1.50 Design_requirement 38
A.2.1.51 dimensional_exponents 38
A.2.1.52 direction 39
A.2.1.53 Document 39
A.2.1.54 Document_reference 39
A.2.1.55 edge 39
A.2.1.56 edge_curve 39
A.2.1.57 edge_loop 40
A.2.1.58 elementary_surface 40
A.2.1.59 ellipse 40
A.2.1.60 Extended_conversion_based_unit 40
A.2.1.61 Extended_measure_with_unit 41
A.2.1.62 External_instance_reference 41
A.2.1.63 External_reference 42
A.2.1.64 face 42
A.2.1.65 face_based_surface_model 42
A.2.1.66 face_bound 42
A.2.1.67 face_outer_bound 43
A.2.1.68 face_surface 43
A.2.1.69 Functional_definition 43
A.2.1.70 General_characteristics_definition 43
A.2.1.71 General_compartment_property_set 44
A.2.1.72 geometric_representation_context 45
A.2.1.73 geometric_representation_item 45
A.2.1.74 global_axis_placement 45
A.2.1.75 Global_id 46
A.2.1.76 Habitable_compartment_design_definition 46
A.2.1.77 Habitable_compartment_functional_definition 46
A.2.1.78 Habitable_compartment_property_set 47
A.2.1.79 intersection_curve 48
A.2.1.80 Item 48
A.2.1.81 Length_Unit 48
A.2.1.82 line 48
A.2.1.83 local_time 49
A.2.1.84 Longitudinal_position 49
A.2.1.85 loop 49
A.2.1.86 Machinery_compartment_design_definition 49
A.2.1.87 Machinery_compartment_functional_definition 49
A.2.1.88 Moments_of_inertia 50
A.2.1.89 Named_Unit 50
A.2.1.90 non_manifold_surface_shape_representation 51
A.2.1.91 Non_structural_moulded_form_design_definition 51
A.2.1.92 open_shell 52
A.2.1.93 Organisation 52
A.2.1.94 organization 52
A.2.1.95 oriented_closed_shell 52
A.2.1.96 oriented_edge 52
A.2.1.97 oriented_face 53
A.2.1.98 oriented_open_shell 53
A.2.1.99 oriented_path 54
A.2.1.100 parametric_representation_context 54
A.2.1.101 path 54
A.2.1.102 pcurve 54
A.2.1.103 person 55
A.2.1.104 Person_and_organisation 55
A.2.1.105 person_and_organization 55

A.2.1.106 placement 56
A.2.1.107 plane 56
A.2.1.108 Plane_Angle_Unit 56
A.2.1.109 point 56
A.2.1.110 polyline 56
A.2.1.111 rational_b_spline_curve 56
A.2.1.112 rational_b_spline_surface 56
A.2.1.113 representation 57
A.2.1.114 representation_context 57
A.2.1.115 representation_item 57
A.2.1.116 representation_relationship 58
A.2.1.117 seam_curve 58
A.2.1.118 shape_representation 58
A.2.1.119 Ship 58
A.2.1.120 Si_Unit 58
A.2.1.121 Simple_definition_version 62
A.2.1.122 Space 62
A.2.1.123 Space_product_structure_definition 62
A.2.1.124 Spacing_grid 63
A.2.1.125 Spacing_position 63
A.2.1.126 Spacing_table 63
A.2.1.127 spherical_surface 65
A.2.1.128 surface 65
A.2.1.129 surface_curve 65
A.2.1.130 Tank_compartment_design_definition 66
A.2.1.131 Tank_compartment_functional_definition 66
A.2.1.132 Tank_compartment_property_set 66
A.2.1.133 Tank_geometric_parameters 67
A.2.1.134 Tank_piping_design_properties 67
A.2.1.135 Tolerance 68
A.2.1.136 topological_representation_item 68
A.2.1.137 Transverse_position 68
A.2.1.138 Universal_resource_locator 68
A.2.1.139 vector 69
A.2.1.140 vertex 69
A.2.1.141 vertex_loop 69
A.2.1.142 vertex_point 69
A.2.1.143 Vertical_position 70
A.2.1.144 Void_compartment_design_definition 70
A.2.1.145 Void_compartment_functional_definition 70
A.2.1.146 Volume_Unit 70
A.2.1.147 Zone 71
A.2.1.148 Zone_design_definition 71
A.2.1.149 Zone_functional_definition 71
A.2.2 Application Assertions *72*
A.2.2.1 Capacity_properties to cartesian_point 72
A.2.2.2 Capacity_properties to cartesian_point 72
A.2.2.3 Cargo_bay_definition to Longitudinal_position 72
A.2.2.4 Cargo_bay_definition to Transverse_position 72
A.2.2.5 Cargo_bay_definition to Vertical_position 73
A.2.2.6 Cargo_compartment_design_definition to Cargo_bay_definition 73
A.2.2.7 Compartment_design_definition to Compartment 73
A.2.2.8 Compartment_design_definition to Compartment_extents 73
A.2.2.9 Compartment_design_definition to Non_structural_moulded_form_design_definition to Spacing_position 73
A.2.2.10 Compartment_design_to_functional_definition_relationship to Compartment_design_definition 73
A.2.2.11 Compartment_design_to_functional_definition_relationship to Compartment_functional_definition 73
A.2.2.12 Compartment_design_requirement 73
A.2.2.13 Compartment_extents to Longitudinal_position 73
A.2.2.14 Compartment_extents to Longitudinal_position 74
A.2.2.15 Compartment_extents to Transverse_position 74

A.2.2.16 Compartment_extents to Transverse_position 74
A.2.2.17 Compartment_extents to Vertical_position 74
A.2.2.18 Compartment_extents to Vertical_position 74
A.2.2.19 Compartment_functional_definition to Compartment 74
A.2.2.20 Compartment_property_to_design_definition_relationship to Compartment_property_set 74
A.2.2.21 Compartment_property_to_design_definition_relationship to Compartment_design_definition 74
A.2.2.22 Cargo_compartment_property_set to Capacity_properties 74
A.2.2.23 Compartment_property_set to Compartment 74
A.2.2.24 Compartment_volume to cartesian_point 75
A.2.2.25 Definable_object to Definition 75
A.2.2.26 Definition to Definable_object 75
A.2.2.27 Definition to Global_id 75
A.2.2.28 Definition to Simple_definition_version 75
A.2.2.29 Definition_relationship to Definition 75
A.2.2.30 Definition_relationship to Definition 75
A.2.2.31 Design_definition to representation 75
A.2.2.32 Design_requirement to Document_reference 75
A.2.2.33 General_characteristics_definition to Ship 75
A.2.2.34 General_compartment_property_set to Compartment_areas 75
A.2.2.35 General_compartment_property_set to Compartment_permeability 75
A.2.2.36 General_compartment_property_set to Compartment_volume 76
A.2.2.37 External_instance_reference to Global_id 76
A.2.2.38 Item to External_reference 76
A.2.2.39 Item to Ship 76
A.2.2.40 Spacing_position to Spacing_table 76
A.2.2.41 Spacing_table to Spacing_grid 76
A.2.2.42 Spacing_table to Spacing_position 76
A.2.2.43 Tank_compartment_property_set to Capacity_properties 76
A.2.2.44 Tank_compartment_property_set to Moments_of_inertia 76
A.2.2.45 Tank_compartment_property_set to Tank_geometric_parameters 76
A.2.2.46 Tank_compartment_property_set to Tank_piping_design_properties 77
A.2.2.47 Zone_design_definition to Compartment_design_definition 77
A.2.2.48 Zone_design_definition to Compartment_design_requirement 77
A.2.2.49 Zone_design_definition to Zone 77
A.2.2.50 Zone_functional_definition to Zone 77
A.2.2.51 Date_and_time to date 77
A.2.2.52 Date_and_time to local_time 77
A.2.2.53 Local_time to coordinated_universal_time_offset 77
A.2.2.54 Person_and_organization to person 77
A.2.2.55 Person_and_organization to organization 77
A.2.2.56 Simple_definition_version to date_and_time 78
A.2.2.57 Derived_Unit to Derived_Unit_Element 78
A.2.2.58 Derived_Unit_Element to Named_Unit 78
A.2.2.59 Extended_conversion_based_unit to Extended_measure_with_unit 78
A.2.2.60 Named_Unit to dimensional_exponents 78
A.2.2.61 Si_Unit to dimensional_exponents 78
A.2.2.62 axis2_placement_2d to direction 78
A.2.2.63 axis2_placement_2d to direction 78
A.2.2.64 axis2_placement_3d to direction 78
A.2.2.65 axis2_placement_3d to direction 78
A.2.2.66 axis2_placement_3d to direction 79
A.2.2.67 b_spline_curve to cartesian_point 79
A.2.2.68 composite_curve to composite_curve_segment 79
A.2.2.69 composite_curve_segment to curve 79
A.2.2.70 connected_face_set to face 79
A.2.2.71 edge to vertex 79
A.2.2.72 edge to vertex 79
A.2.2.73 face_based_surface_model to connected_face_set 79
A.2.2.74 face to face_bound 79
A.2.2.75 oriented_edge to edge 79
A.2.2.76 oriented_edge to vertex 79
A.2.2.77 oriented_edge to vertex 79

A.2.2.78 oriented_face to face 80
A.2.2.79 oriented_face to face_bound 80
A.2.2.80 composite_curve_on_surface to surface 80
A.2.2.81 line to cartesian_point 80
A.2.2.82 line to vector 80
A.2.2.83 pcurve to surface 80
A.2.2.84 pcurve to definitional_representation 80
A.2.2.85 polyline to cartesian_point 80
A.2.2.86 surface_curve to curve 80
A.2.2.87 surface_curve to surface 80
A.2.2.88 vector to direction 80
A.2.2.89 edge_curve to curve 80
A.2.2.90 elementary_surface to axis2_placement_3d 81
A.2.2.91 face_bound to loop 81
A.2.2.92 face_surface to surface 81
A.2.2.93 oriented_open_shell to face 81
A.2.2.94 oriented_closed_shell to closed_shell 81
A.2.2.95 oriented_open_shell to open_shell 81
A.2.2.96 oriented_closed_shell to face 81
A.2.2.97 oriented_path to path 81
A.2.2.98 placement to cartesian_point 81
A.2.2.99 representation to representation_item 81
A.2.2.100 representation to representation_context 81
A.2.2.101 representation_relationship to representation 81
A.2.2.102 representation_relationship to representation 82
A.2.2.103 representation_context to representation 82
A.2.2.104 vertex_loop to vertex 82
A.2.2.105 vertex_point to point 82
A.3 APPLICATION REFERENCE MODEL (ARM) 83
A.4 EXAMPLE TESTCASE 128
A.4.1 Testcase Definition 128
A.4.2 Entity Attribute Inheritance 131
A.4.3 Express-I Diagram 168
A.4.4 Preprocessor and Postprocessor Worksheets 177

Table of Figures

FIGURE 1: EXAMPLE OF SHIP COMPARTMENT 9
FIGURE 2: EXAMPLE OF DDG-51 DESIGN ZONES 10
FIGURE 3: EXAMPLE OF CG-60 FIRE ZONE BOUNDARIES 12
FIGURE A-1 (ABS) ITEM 87
FIGURE A-2 SPACING_TABLE 88
FIGURE A-3 COMPARTMENT 89
FIGURE A-4 ZONE 90
FIGURE A-5 NON STRUCTURAL MOLDED FORM DESIGN DEFINITION 91
FIGURE A-6 (ABS) DEFINITION 92
FIGURE A-7 (ABS) FUNCTIONAL_DEFINITION 93
FIGURE A-8 (ABS) DESIGN_DEFINITION 94
FIGURE A-9 DEFINITION_RELATIONSHIP 95
FIGURE A-10 (ABS) COMPARTMENT_PROPERTY_SET 96
FIGURE A-11 (ABS) DESIGN_DEFINITION 97
FIGURE A-12 REPRESENTATION 98
FIGURE A-13 REPRESENTATION_RELATIONSHIP 99
FIGURE A-14 REPRESENTATION_ITEM 100
FIGURE A-15 EDGE 101
FIGURE A-16 FACE 102
FIGURE A-17 PATH 103
FIGURE A-18 POINT 104
FIGURE A-19 MISCLLANEOUS GEOMETRY 105
FIGURE A-20 CURVE 106
FIGURE A-21 COMPOSITE_CURVE 107
FIGURE A-22 BOUNDED_CURVE 108
FIGURE A-23 SURFACE 109
FIGURE A-24 DIRECTION 110
FIGURE A-25 PLACEMENT 111
FIGURE A-26 (ABS) ITEM 112
FIGURE A-27 DESIGN_REQUIREMENT 113
FIGURE A-28 GENERAL_COMPARTMENT_PROPERTY_SET 114
FIGURE A-29 TANK_COMPARTMENT_PROPERTY_SET 115
FIGURE A-30 TANK_PIPING_DESIGN_PROPERTIES 116
FIGURE A-31 CAPACITY_PROPERTIES 117
FIGURE A-32 COMPARTMENT_EXTENTS 118
FIGURE A-33 DIMENSIONAL_EXPONENTS 119
FIGURE A-34 UNITS 120
FIGURE A-35 DATE 121
FIGURE A-36 ORGANIZATION 122
FIGURE A-37 UNIVERAL_RESOURCE_LOCATOR 123
FIGURE A-38 ADDRESS 124
FIGURE A-39 EXTERNAL_INSTANCE_REFERENCE 125
FIGURE A-40 DATE, UNITS, TOLERANCE 126
FIGURE A-41 REPRESENTATION_CONTEXT 127
FIGURE A-42 - COMPARTMENT 168
FIGURE A-47 - GENERAL_COMPARTMENT_PROPERTY_SET (1/2) 173
FIGURE A-48 - GENERAL_COMPARTMENT_PROPERTY_SET (2/2) 174
FIGURE A-50 - EXTERNAL_INSTANCE_REFERENCE 176

A. Ship Arrangements

This section contains information specific to the NSRP Ship Structures AP 215 schema including:

- Technical Discussion
- Information Content
- Application Reference Model
- Example Test case

A.1 Technical Discussion

A.1.1 Arrangement (Internal Subdivision)

The hull form of a ship is internally subdivided early in the design lifecycle by the introduction of many additional surfaces. These surfaces are associated with the molded hullform elements such as bulkheads and decks. Structural entities such as plate parts and stiffeners will be defined on these surfaces as the design progresses. A region of the ship --whether it be interior to the hull such as a tank or enclosing one if its exposed decks such as a helicopter landing platform-- is designated a space. Two types of spaces are addressed by this AP --compartments and zones. Compartments that represent physical bounded spaces and zones, which represent regions surrounded by some abstract boundary.

The most common type of spatial partitioning is the subdivision of a ship into compartments. A compartment is very similar to the idea of a room in a building. The compartment is bounded by the surfaces representing structural decks and bulkheads and also by non-structural (or non load bearing) surfaces that form "joiner bulkheads". Compartments may be classified according to the function they perform with regard to the operation of the ship. The types of spaces supported by this AP are cargo/stowage (both liquid and dry cargo), void, habitable, and machinery/equipment. Collections of attributes have been defined for the various compartments depending on its designated use. Compartments serve a vital function in configuration managing engineering part occurrences throughout the lifecycle of the ship.

In some cases, the same surfaces that subdivide a ship into compartments may also be used to subdivide the ship into zones. In other cases, additional hullform geometry elements and/or geometric surfaces may be required to define zone boundaries. On naval ships, multiple zone subdivisions --such as, pressure (Collective Protection System), subsafe, damage control, and arrangement zones-- will be defined and each subdivides the hull into an independent set of spaces. Sometimes, two zones may have the same boundary; however, each zone is still independently represented.

In addition to identifying the various spaces on the ship, it is important to represent the connectivity between these spaces. This model supports several types of relationships between spaces, specifically adjacency, functional, positional, and enclosing. Adjacency relationships are established via a connectivity network based on the connection/joint model presented earlier in this document. Characteristics such as accessibility, access time, and common surface area between adjacent spaces are provided such that analyses to determine transit times between areas of the ship and HVAC load calculations can be supported. Functional relationships can be used to record the fact that one space's design parameters are dependent on some functional characteristic of another spaces --such as a pair of port and starboard ballast tanks used for anti-roll stabilization. Positional relationships capture design intent expressing the fact that certain spaces must maintain geometric characteristics similar to another spaces' --such as two spaces that should maintain the same transverse width dimension. Finally, enclosing relationships allow the product model to record the fact that one space may be completely surrounded by another space --such as a free-standing Lube Oil Settling Tank in the

Machinery Space.
From a functional standpoint, the model has been developed to associate properties with the various compartments appropriate to their function. These properties include volumetric capacities, length measures, and cross-sectional areas. The ability to specify constraints on these properties is provided for where appropriate so as to assist engineers in the early stages of design. For example, it is possible to specify a minimum length for a compartment, as well as a maximum length for the compartment. Likewise, it is possible to record an estimated compartment volume, as well as a calculated and a measured. Longitudinal and transverse grids can be defined for an arrangement zone that can be used to restrict the free placement of bulkheads such that they align with the grid points.

A.1.2 Compartments

A ship is divided horizontally by decks, platforms, flats, levels, and the bottom shell. These divisions apply to the entire ship, both in the main hull and in the superstructure (or deckhouse). Deck gratings, false decks, or similar flats are not considered as division boundaries. Between horizontal division boundaries, the ship is divided vertically by tight or nontight bulkheads. Except for spaces designated as voids, cofferdams, or tanks, only tight boundaries are considered.
Every volume enclosed by horizontal and vertical boundaries (except for minor utility areas such as peacoat lockers, linen lockers, cleaning gear lockers, and other similar areas) is considered a compartment. Some compartments, by this definition, may or may not have access closures. Compartments are assigned a compartment name and a compartment number. Compartments that extend vertically through more than one horizontal division boundary, such as machinery spaces and deep tanks, are considered to be located on the lowest horizontal boundary.
Figure 1 illustrates the subdivision of one deck of a typical naval vessel.

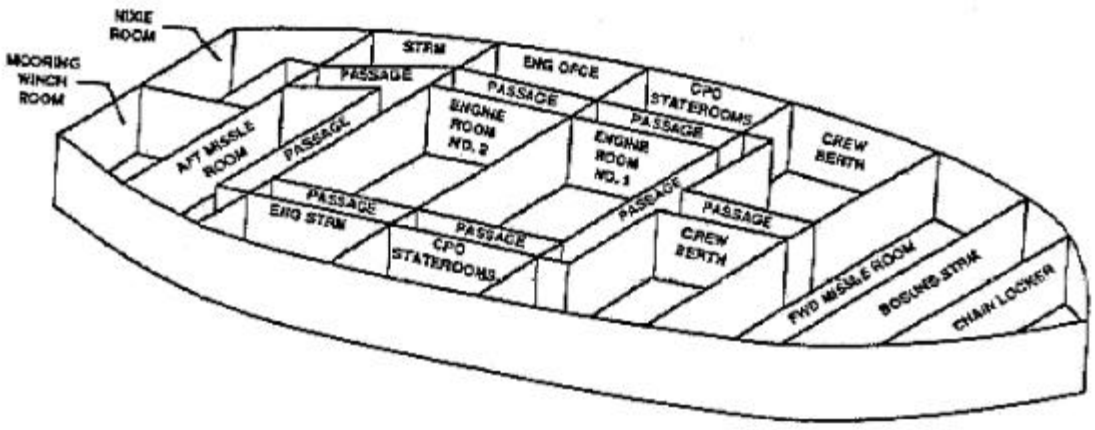


Figure 1: Example of Ship Compartment

The following table presents a listing of common compartment attributes --such as compartment name, compartment number, and applicable design zone-- for a similar vessel

COMPARTMENT NUMBER	COMPARTMENT	DESIGN ZONE
1 - 46-0 -M	5"/54 CALIBER LOADER DRUM & FAN ROOM	1350
3 - 52-1 -M	5"/54 CALIBER POWDER MAGAZINE NO.1	1330
3 - 42-01-M	5"/54 CALIBER POWDER PROJECTILE MAGAZINE	1330
1 -364-1 -T	ACCESS TRUNK	2450
1 -140-2 -Q	ADMINISTRATION OFFICE	4210
1 -196-1 -T	ACCESS TRUNK	2450
2 -310-1 -L	AFT BATTLE DRESSING STATION	3440

A.1.3 Zones

A.1.3.1 Design Zones

One common type of internal subdivision is the "design" zone. Whereas a compartment is a subdivision involving the functional aspects of a completed, or in-service, ship, a design zone is associated with the manufacturing of a ship by the design agent and/or shipbuilder. Design zones are used to break up the ship into blocks for facilitating design and construction.

Design zones, like compartments, are bounded by surfaces representing decks, bulkheads, the hull, and so on. They may also be bounded by other surfaces. A common use of the design zone is to configuration manage aspects of the ship design process within its bounds. For this reason, the subdivision model has been developed to allow product structuring by zone. Structural parts, structural assemblies, and engineering parts can all be associated with a particular zone of the ship. Figure 2 illustrates the design zones of the DDG-51, the lead ship of the Arleigh Burke class destroyer.

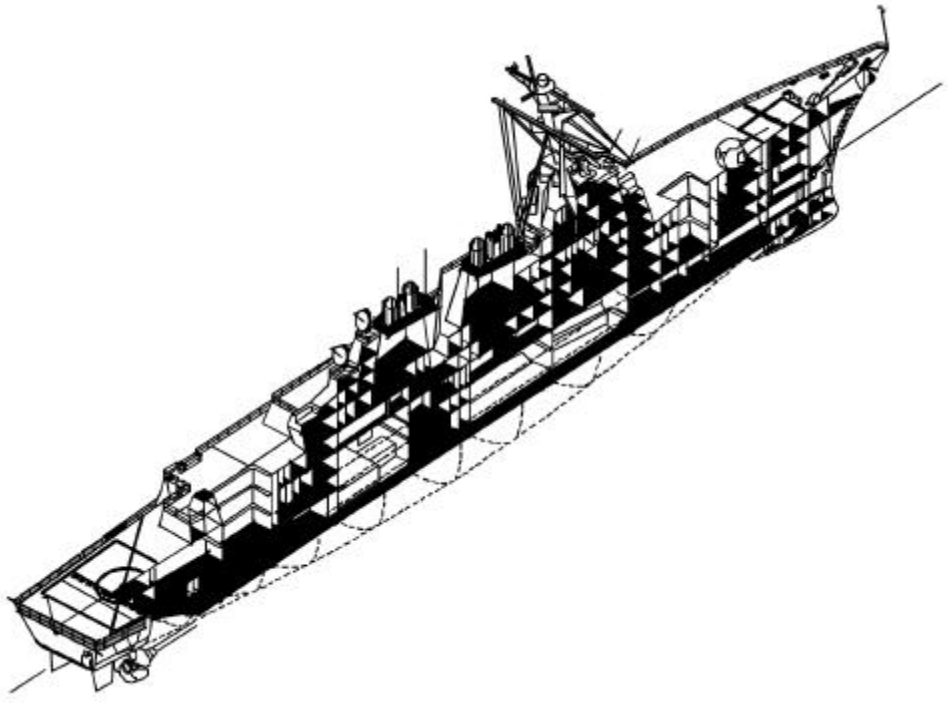


Figure 2: Example of DDG-51 Design Zones

A.1.3.2 Fire Zones

The design of a naval ship is likely to provide fire containment capabilities. A Damage Control Console provides remote control fire containment at a central site. The ship design process entails subdividing the vessel into a number of fire zones. A fire zone boundary is a physical boundary designed to retard the passage of flame and smoke from one area of the ship to the next. All fire zone boundaries are watertight or fumetight bulkheads. Fire zone boundaries in the hull are constructed of steel. Bulkheads in the superstructure, if aluminum, are covered with non-combustible thermal insulation. Each fire zone boundary on the Damage Control Deck is provided with spring loaded, joiner type, fire doors, each held open by an electromagnetic catch. An adjacent quick acting watertight door is also provided. Each fire zone has fire, smoke, and heat sensors which activate a central display on the hazard detection panel of the Damage Control Console and enable the ship's crew to safely and effectively monitor and control onboard fires. Figure 3 depicts the Fire Zone Boundaries and Fire Doors of a (Aegis) Ticonderoga class curiser.

A.1.3.3 Collective Protection System (CPS) Zones

Another type of zone common on naval ships is a pressure zone. These pressure zones are used to define regions of the ship that have been designed to maintain a pressure slightly higher than that of the outside atmosphere. These zones, commonly referred to as collective protection system (CPS) zones, are necessary to combat biological and nuclear warfare. The air pumped into these zones is specially filtered to remove harmful contaminants. As with fire zones discussed above, the boundaries of these zones are fitted with special types of automatic closures to secure the zones in an emergency.

A.1.3.4 Arrangement Zones

The last type of zone to be discussed here is the arrangement zone. This zone is used early in the design to control and manage the arrangement of compartments on the ship. An individual or workgroup may be assigned a collection of compartments that are to be arranged within a given domain (i.e. the arrangement zone). Working within the bounds of this zone, the designers can define the compartment boundaries according to the requirements for the spaces --such as number of crew, amount of cargo, etc. To facilitate standardization, longitudinal and transverse grids may be established that constrains the placement of bulkheads.

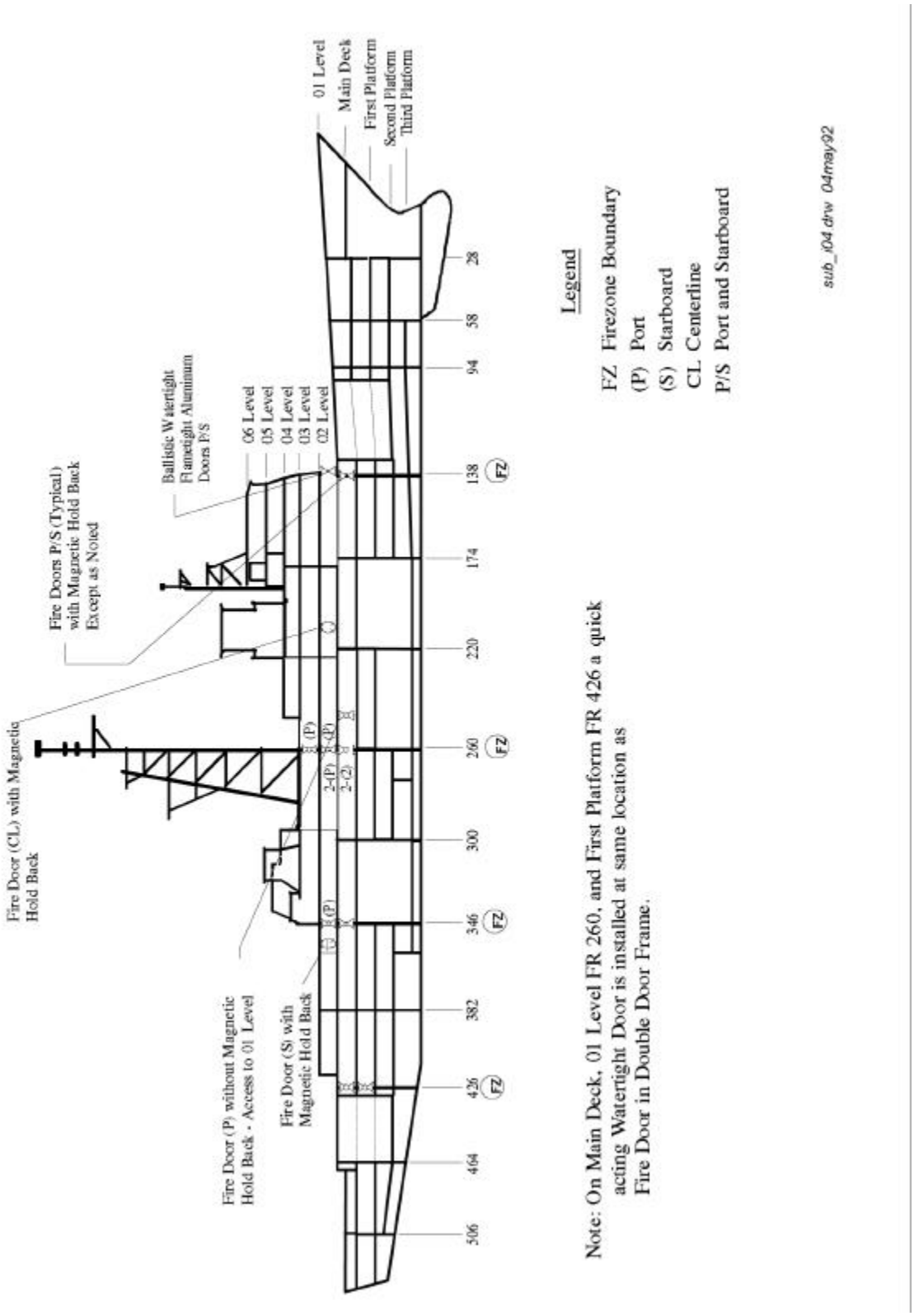


Figure 3: Example of CG-60 Fire Zone Boundaries

A.2 Information Content

The information content of the NSRP Ship Structures schema is defined by the following: 1) the Application Objects of the Application Reference Model; and 2) Application Assertions.

A.2.1 Application Objects

This section specifies the application objects for the Ship Structures application protocol. Each ap-plication object is an atomic element that embodies a unique application concept and contains attributes specifying the data elements of the object. The application objects and their definitions are given below.

A.2.1.1 address

The Address is the place where people and organizations are located. The Address entity is referenced by Any_address. The data associated with an address are the following:

— internal_location;
— street_number;
— street;
— postal_box;
— town;
— region;
— postal_code;
— country;
— facsimile_number;
— telephone_number;
— electronic_mail_address;
— telex_number.

A.2.1.1.1 internal_location

The Internal_location is the unique internal address used by a company to route mail or equipment, such as a mail stop number, a room number, or a department number.

A.2.1.1.2 street_number

The Street_number is the unique number assigned to a business or private residence based on its location on a street.

A.2.1.1.3 street

Street is a public road in a town or city.

A.2.1.1.4 postal_box

The Postal_box is the post office box number used to receive mail for addresses without street numbers.

A.2.1.1.5 town

Town is the name of a town or city.

A.2.1.1.6 region

Region is a subdivision of country, such as a state, province, or department.

A.2.1.1.7 postal_code

The Postal_code is a numbering system used by a post office to uniquely identify a region and town.

A.2.1.1.8 country

Country is the name of a country.

A.2.1.1.9 facsimile_number
Facsimile_number is the phone number of a fax machine and includes the area code followed by seven digits.

A.2.1.1.10 telephone_number
Telephone_number is the phone number of a business or private individual and includes the area code followed by seven digits.

A.2.1.1.11 electronic_mail_address
The Electronic_mail_address is the unique address used by the Internet to send and receive electronic mail.

A.2.1.1.12 telex_number
The Telex_number is the phone number of a telex machine and includes the area code followed by seven digits.

A.2.1.2 advanced_face
The Advanced_face is a special type of face_surface that has additional constraints to ensure that the geometry is directly and completely defined.

A.2.1.3 area_unit
The Area_unit entity is a subtype of Named_unit.

A.2.1.4 axis2_placement_2d
The Axis2_placement_2d is the location and orientation in two-dimensional space of two mutually perpendicular axes. The Axis2_placement_2d is defined in terms of a point and an axis.
The data associated with an axis2_placement_2d are the following:
— ref_direction;
— p

A.2.1.4.1 ref_direction
The Ref_direction is the direction used to determine the direction of the local X-axis.

A.2.1.4.2 p
The P is the axis set for the placement coordinate system.

A.2.1.5 axis2_placement_3d
The Axis2_placement_3d is the location and orientation in three-dimensional space of two mutually perpendicular axes. Defined in terms of a point and two axes.
The data associated with an axis2_placement_3d are the following:
— axis;
— ref_direction;
— p

A.2.1.5.1 axis
The Axis is the exact direction of the local Z-axis.

A.2.1.5.2 ref_direction
The Ref_direction is the direction used to determine the direction of the local X-axis.

A.2.1.5.3 p
The P is the axis for the placement coordinate system.

A.2.1.6 b_spline_curve
The B_spline_curve is a piecewise parametric polynomial or rational curve described in terms of control points and basis functions.
The data associated with a b_spline_curve are the following:
— degree;
— control_points_list;
— curve_form;
— closed_curve;
— self_intersect;
— upper_index_on_control_points;
— control_points

A.2.1.6.1 degree
The Degree is the algebraic degree of the basis functions.

A.2.1.6.2 control_points_list
The Control_points_list is the list of control points for the curve.

A.2.1.6.3 curve_form
The Curve_form is used to identify particular types of curves; it is for information only.
The curve_form shall be one of the following:
— polyline_form;
— circular_arc;
— elliptic_arc;
— parabolic_arc;
— hyperbolic_arc;
— unspecified

A.2.1.6.3.1 polyline_form
A Polyline_form is a connected sequence of line segments represented by a degree 1 B-spline basis function.

A.2.1.6.3.2 circular_arc
A Circular_arc is an arc of a circle, or a composite circle represented by a B-spline curve.

A.2.1.6.3.3 elliptic_arc
An Elliptic_arc is an arc of an ellipse, or a complete ellipse, represented by a B-spline curve.

A.2.1.6.3.4 parabolic_arc
A Parabolic_arc is an arc of finite length of a parabola represented by a B-spline curve.

A.2.1.6.3.5 hyperbolic_arc
A Hyperbolic_arc is an arc of finite length of one branch of hyperbola represented by a B-spline curve.

A.2.1.6.3.6 unspecified
An Unspecified is a B-spline curve for which no form is specified.

A.2.1.6.4 closed_curve
The Closed_curve is an indication of whether the curve is closed.

A.2.1.6.5 self_intersect
The Self_intersect is a flag to indicate whether the curve self intersects or not.

A.2.1.6.6 upper_index_on_control_points
The Upper_index_on_control_points is the upper index on the array of control points; the lower index is 0. Derived from the list of control points.

A.2.1.6.7 control_points
The Control_points are the array of control points used to define the geometry of the curve.

A.2.1.7 b_spline_curve_with_knots
The B_spline_curve_with_knots is the subtype of b_spline_curve for which the knot values are explicitly given. Used to represent non-uniform B-spline curves.
The data associated with a b_spline_curve_with_knots are the following:
— knot_multiplicities;
— knots;
— knot_spec;
— upper_index_on_knots.

A.2.1.7.1 knot_multiplicities
The Knot_multiplicities determines the number of times each knot in the knots list is to be repeated in constructing the knot array.

A.2.1.7.2 knots
The Knots are the list of distinct knots used to define the B-spline basis functions.

A.2.1.7.3 knot_spec
The Knot_spec is the description of the knot type.
The knot_spec shall be one of the following:
— uniform_knots;
— unspecified;
— quasi_uniform_knots;
— piecewise_bezier_knots.

A.2.1.7.3.1 uniform_knots
The Uniform_knots is the form of knots appropriate for a uniform B-spline curve.

A.2.1.7.3.2 unspecified
The Unspecified is the unspecified type of knots. This included the case of non-uniform knots.

A.2.1.7.3.3 quasi_uniform_knots
The Quasi_uniform_knots is the form of knots appropriate for a quasi-uniform B-spline curve.

A.2.1.7.3.4 piecewise_bezier_knots
The Piecewise_bezier_knots are the form of knots appropriate for a piecewise Bezier curve.

A.2.1.7.4 upper_index_on_knots
The Upper_index_on_knots is the upper index on the knot arrays; the lower index is 1.

A.2.1.8 b_spline_surface

The B_spline_surface is a general form of rational or polynomial parametric surface that is represented by control points, basis functions, and possibly, weights.
The data associated with a b_spline_surface are the following:

— u_degree;
— v_degree;
— control_points_list;
— surface_form;
— u_closed;
— v_closed;
— self_intersect;
— u_upper;
— v_upper;
— control_points

A.2.1.8.1 u_degree

The U_degree is the algebraic degree of basis functions on u.

A.2.1.8.2 v_degree

The V_degree is the algebraic degree of basis functions on v.

A.2.1.8.3 control_points_list

The Control_points_list is the list of lists of control points.

A.2.1.8.4 surface_form

The Surface_form is the indicator of special surface types.
The surface_form shall be one of the following:

— plane_surf;
— cylindrical_surf;
— conical_surf;
— spherical_surf;
— toroidal_surf;
— surf_of_revolution;
— ruled_surf;
— generalised_cone;
— quadric_surf;
— surf_of_linear_extrusion;
— unspecified

A.2.1.8.4.1 plane_surf

A Plane_surf is a bounded portion of a plane represented by a B-spline surface of degree 1 in each parameter.

A.2.1.8.4.2 cylindrical_surf

A Cylindrical_surf is a bounded portion of a cylindrical surface.

A.2.1.8.4.3 conical_surf

A Conical_surf is a bounded portion of the surface of a right circular cone.

A.2.1.8.4.4 spherical_surf
A Spherical_surf is a bounded portion of a sphere, or a complete sphere, represented by a B-spline surface.

A.2.1.8.4.5 toroidal_surf
A Toroidal_surf is a torus, or portion of a torus, represented by a B-spline surface.

A.2.1.8.4.6 surf_of_revolution
A Surf_of_revolution is a bounded portion of a surface of revolution.

A.2.1.8.4.7 ruled_surf
A Ruled_surf is a surface constructed from two parametric curves by joining with straight lines corresponding points with the same parameter value on each of the curves.

A.2.1.8.4.8 generalised_cone
A Generalised_cone is a special case of a ruled surface in which the second curve degenerates to a single point; when represented by a B-spline surface all the control points along one edge will be coincident.

A.2.1.8.4.9 quadric_surf
A Quadric_surf is a bounded portion of one of the class of surfaces of degree 2 in the variables X, Y, and Z.

A.2.1.8.4.10 surf_of_linear_extrusion
A Surf_of_linear_extrusion is a bounded portion of a surface of linear extrusion represented by a B-spline surface of degree 1 in one of the parameters.

A.2.1.8.4.11 unspecified
An Unspecified is a surface for which no particular form is specified.

A.2.1.8.5 u_closed
The U_closed is the indication of whether the surface is closed in the u direction.

A.2.1.8.6 v_closed
The V_closed is the indication of whether the surface is closed in the v direction.

A.2.1.8.7 self_intersect
The Self_intersect is the flag to indicate whether, or not, the surface is self-intersecting.

A.2.1.8.8 u_upper
The U_upper is the upper index on control points in the u direction.

A.2.1.8.9 v_upper
The V_upper in the upper index on control points in the v direction.

A.2.1.8.10 control_points
The Control_points is the array (two-dimensional) of control points defining surface geometry. This array is constructed from the control points list.

A.2.1.9 b_spline_surface_with_knots
The B_spline_surface_with_knots is a B-spline surface in which the knot values are explicitly given, and is used to represent non-uniform B-spline surfaces, and may also be used for other knot types.
The data associated with a b_spline_surface_with_knots are the following:

— u_multiplicities;
— v_multiplicities;
— u_knots;
— v_knots;
— knot_spec;
— knot_u_upper;
— knot_v_upper

A.2.1.9.1 u_multiplicities
The U_multiplicities are the multiplicities of the knots in the u parameter direction.

A.2.1.9.2 v_multiplicities
The V_multiplicities are the multiplicities of the knots in the v parameter direction.

A.2.1.9.3 u_knots
The U_knots is the list of the distinct knots in the u parameter direction.

A.2.1.9.4 v_knots
The V_knots is the list of the distinct knots in the v parameter direction.

A.2.1.9.5 knot_spec
The Knot_spec is the description of the knot type.
The knot_spec shall be one of the following:
— uniform_knots;
— unspecified;
— quasi_uniform_knots;
— piecewise_bezier_knots

A.2.1.9.5.1 uniform_knots
The Uniform_knots is the form of knots appropriate for a uniform B-spline curve.

A.2.1.9.5.2 Unspecified
The Unspecified is the unspecified type of knots. This included the case of non-uniform knots.

A.2.1.9.5.3 quasi_uniform_knots
The Quasi_uniform_knots is the form of knots appropriate for a quasi-uniform B-spline curve.

A.2.1.9.5.4 piecewise_bezier_knots
The Piecewise_bezier_knots are the form of knots appropriate for a piecewise Bezier curve.

A.2.1.9.6 knot_u_upper
The Knot_u_upper is the number of distinct knots in the u parameter direction.

A.2.1.9.7 knot_v_upper
The Knot_v_upper is the number of distinct knots in the v parameter direction.

A.2.1.10 bounded_curve
The Bounded_curve is a curve of finite arc length with identifiable end points.

A.2.1.11 bounded_pcurve
The Bounded_pcurve is a special subtype of pcurve that also has the properties of a bounded_curve.

A.2.1.12 bounded_surface
The Bounded_surface is a surface of finite area with identifiable boundaries.

A.2.1.13 bounded_surface_curve
The Bounded_surface_curve is a specialized subtype of surface_curve that also has the properties of a bounded_curve.

A.2.1.14 calendar_date
The Calendar_date is a date that is identified by a day in a month of a year. The Calendar_date entity is a subtype of Date. The data associated with a calendar_date are the following:
— day_component;
— month_component

A.2.1.14.1 day_component
The Day_component is the day element of the date.

A.2.1.14.2 month_component
The Month_component is the month element of the date.

A.2.1.15 Capacity_properties
Measures of volumetric characteristic of a Tank_compartment or Cargo_compartment computed at some specific combination of level, trim, and heel angle. The level represents the imaginary planar surface at the cargo non-cargo interface and is relative to a capacity level origin established for the compartment. The attitude of the plane is adjusted to coincide with a vector having a magnitude equal to the level and a direction reflecting the vessels heel and trim. A compartment may have any number of combinations of capacity values.
The data associated with a Capacity_properties are the following:
— capacity_level_origin;
— capacity_center;
— capacity_level;
— capacity_trim_angle;
— capacity_heel_angle;
— capacity_volume;
— capacity_context.

A.2.1.15.1 capacity_level_origin
The capacity_level_origin specifies point associated with a Tank_compartment or Cargo_compartment that represents the vertical reference for measuring the capacity depth levels corresponding to a set of compartment capacities. It may be chosen to represent the bottom of the compartment, the bottom of the sounding tube, or any other convenient location.

A.2.1.15.2 capacity_center
The capacity_center specifies the position of the volumetric center of the interior region of space formed by the compartment boundaries and the imaginary plane representing the cargo non-cargo interface.

A.2.1.15.3 capacity_level
The capacity_level specifies distance between the bottom of the compartment (expressed as the capacity_level_origin) and the top of an imaginary plane representing the cargo non-cargo

interface. It is measured along a vector offset from the vertical to reflect the capacity heel angle and the capacity trim angle.

A.2.1.15.4 capacity_trim_angle

The capacity_trim_angle specifies amount of rotation about the ships transverse centerline axis that has been factored into the capacity calculation for the plane representing the interface between the cargo and non-cargo regions of the compartment.

A.2.1.15.5 capacity_heel_angle

The capacity_heel_angle specifies amount of rotation about the ships longitudinal axis that has been factored into the capacity calculation for the plane representing the interface between the cargo and non-cargo regions of the compartment.

A.2.1.15.6 capacity_volume

The capacity_volume specifies the enclosed volumetric measurement of the interior region of space formed by the compartment and the imaginary plane representing the cargo non-cargo interface.

A.2.1.15.7 capacity_context

The capacity_context specifies values representing the two significant capacity states, The context must be unique with respect to a compartment, capacity level, capacity heel angle, capacity trim angle, and a compartment capacity type.
The capacity_context shall be one of the following:

— percent_100_capacity;
— percent_98_capacity

A.2.1.15.7.1 percent_100_capacity

The Percent_100_capacity is 100.0 percent full tank capacity state

A.2.1.15.7.2 percent_98_capacity

The Percent_98_capacity is 98.0 percent full tank capacity state

A.2.1.16 Cargo_bay_definition

The Cargo_bay_definition is a grid of positions within a cargo compartment that are used to specify the location of placement of cargo within that compartment.
The data associated with a Cargo_bay_definition are the following:

— longitudinal_cargo_positions;
— transverse_cargo_positions;
— vertical_cargo_positions.

A.2.1.16.1 longitudinal_cargo_positions

The longitudinal_cargo_positions specifies longitudinal positions that locate the placement of cargo.

A.2.1.16.2 transverse_cargo_positions

The transverse_cargo_positions specifies transverse positions that locate the placement of cargo.

A.2.1.16.3 vertical_cargo_positions

The vertical_cargo_positions specifies vertical positions that locate the placement of cargo.

A.2.1.17 Cargo_compartment_design_definition

A Cargo_compartment_design_definition is a type of Compartment_design_definition that provides the abstract definition of a version of a Cargo compartment from a design perspective.
The data associated with a Cargo_compartment_design_definition are the following:

— cargo_placement_locations_within_compartment.

A.2.1.17.1 cargo_placement_locations_within_compartment:

The Cargo_placement_locations_within_compartment is a Cargo_bay_definition and defines the location of cargo within a compartment in relation to the ship coordinate system.

A.2.1.18 Cargo_compartment_functional_definition

The Cargo_compartment_functional_definition defines the functional role of a Cargo_compartment; the role may be a pre-defined one or may be user-defined.
The data associated with a Cargo_compartment_functional_definition are the following:

— func.

The func specifies the name of a function that a specific Cargo_compartment may have in a ship.
The value of func shall be one of the following:

— dry_bulk_cargo;
— dry_unit_cargo;
— liquid_cargo;
— gaseous_cargo;
— user_defined

A.2.1.18.1.1 dry_bulk_cargo

The Dry_bulk_cargo function is a predefined dry bulk cargo compartment

A.2.1.18.1.2 dry_unit_cargo

The Dry_unit_cargo function is a predefined dry unit cargo compartment

A.2.1.18.1.3 liquid_cargo

The Liquid_cargo function is a predefined liquid cargo compartment

A.2.1.18.1.4 gaseous_cargo

The Gaseous_cargo function is a predefined gaseous cargo compartment

A.2.1.18.1.5 user_defined

The User_defined function is a predefined user defined cargo compartment

A.2.1.19 Cargo_compartment_property_set

Cargo_compartment_property_set describes properties for cargo capacities and cargo densities for which the Cargo_compartment is designed.
The data associated with a Cargo_compartment_property_set are the following:

— bulk_cargo_capacity;
— design_stowage_density

A.2.1.19.1 bulk_cargo_capacity

The bulk_cargo_capacity is optional and, if present, specifies the maximum capacity for bulk cargo.

A.2.1.19.2 design_stowage_density

The design_stowage_density specifies the measure of the quantity per unit volume of the dry bulk cargo for which the Cargo_compartment is designed.

A.2.1.20 cartesian_point

The Cartesian_point is a point defined by its coordinates in a rectangular Cartesian coordinate system, or in a parameter space. The entity is defined in a one, two or three-dimensional space as determined by the number of coordinates in the list.
The data associated with a cartesian_point are the following:
— coordinates
The first, second, or third coordinates of the point location, depending upon the dimensionality of the point.

A.2.1.21 circle

The Circle is defined by a radius and the location and orientation of the circle.
The data associated with a circle are the following:
— radius
The Radius of the circle, which shall be greater that zero.

A.2.1.22 closed_shell

The Closed_shell is a shell of dimensionality 2 that typically serves as a bound for a region in R3. A closed shell has no boundary, and has non-zero finite extent. If the shell has a domain with coordinate space R3, it divides that space into two connected regions, one finite and the other infinite. In this case, the topological normal of the shell is defined as being directed from the finite to the infinite region.

A.2.1.23 Compartment

A Compartment is a physical subdivision of space on a ship, designed to hold dry or liquid cargo, fuel, water, passengers, crew, machinery, equipment, etc. Each Compartment may have one or more functional definitions to specify the intended function (s) of the Compartment as well as one or more design definitions, which specify the compartment boundaries as well as its physical and geometric properties. Each Compartment may also have one or more Compartment_product structure_definitions, which reference the items bounding a compartment, such as moulded form regions or structural panels, or enclosed by a compartment, such as structural parts, outfitting objects, and equipment objects.

A.2.1.24 Compartment_areas

Compartment_areas specify required 2.0- dimensional cross-sectional areas for a Compartment, required to accommodate design requirements on the Compartment, such as placement of equipment.
The data associated with a Compartment_areas are the following:
— vertical_longitudinal_cross_sectional_area;
— vertical_transverse_cross_sectional_area;
— horizontal_cross_sectional_area;
— unstiffened_surface_area;
— stiffened_surface_area.

A.2.1.24.1 vertical_longitudinal_cross_sectional_area

The vertical_longitudinal_cross_sectional_area specifies an area measurement corresponding to a plane defined by the vertical and longitudinal axes.

NOTE - Typically, this area is used to reserve space early in the design process, such as the area needed for placement of a large piece of equipment.

A.2.1.24.2 vertical_transverse_cross_sectional_area

The vertical_transverse_cross_sectional_area specifies an area measurement corresponding to a plane defined by the vertical and transverse axes.

NOTE - Typically, this area is used to reserve space early in the design process, such as the area needed for placement of a large piece of equipment.

A.2.1.24.3 horizontal_cross_sectional_area

The horizontal_cross_sectional_area specifies an area measurement on a plane parallel to the baseline plane.

NOTE - Typically, the area is used to reserve space early in the design process, such as the area needed for placement of a propulsion system footprint (i.e. main engine, reduction gear, etc.).

A.2.1.24.4 unstiffened_surface_area

The unstiffened_surface_area specifies a measure of the amount of surface area for the compartment, excluding the surface area of any interior stiffeners on the bulkheads, decks, hull shell, etc.

A.2.1.24.5 stiffened_surface_area

The stiffened_surface_area specifies a measure of the amount of surface area for the compartment, including the surface area of any interior stiffeners on the bulkheads, decks, hull shell, etc.

A.2.1.25 Compartment_design_definition

The Compartment_design_definition is a definition of a version of a Compartment from a design perspective.

The data associated with a Compartment_design_definition are the following:

— design_requirements;
— boundaries;
— extents;
— representations;
— defined_for

A.2.1.25.1 design_requirements

The Design_requirements specifies reference to a description or formal specification that represents a constraint placed on the design.

A.2.1.25.2 boundaries

The Boundaries specifies topological reference to the Moulded_form design definitions that bound the Compartment.

A.2.1.25.3 extents

The Extents is optional and if present, specifies locations referencing the ships spacing grid of the boundaries of the Compartment.

A.2.1.25.4 representations
The representations specifies redefinition of the representation attribute a Compartment_design_definition shall only have Compartment_shape_representations.

A.2.1.25.5 defined_for
The defined_for specifies redefinition of the defined_for attribute a Compartment_design_definition is only valid for a Compartment.

A.2.1.26 Compartment_design_requirement
A type of specification that represents a constraint placed on a design. Requirements could be in the form of a reference to a set of rules or formula, such as NAVSEA Design Specifications, American Bureau of Shipping Rules, American Welding Society Rules, etc. or it may be an explicit requirement, such as an electrical requirement for 110.0 volt power, an HVAC requirement for operating temperatures in the range of 50.0 o F to 90.0 o F, etc.
The data associated with a Compartment_design_requirement are the following:
— requirement_type;
— requirement_description.

A.2.1.26.1 requirement_type
The requirement_type specifies an indicator used to denote the source placing the design requirement on the space.
The value of the requirement_type shall be one of the following:
— naval_architecture;
— structural;
— piping;
— hvac;
— electrical;
— electronic;
— combat_system;
— outfit_furnishing;
— painting_coating;
— user_defined

A.2.1.26.1.1 naval_architecture
The Naval_architecture Compartment design requirement originates from the naval_architecture discipline.

A.2.1.26.1.2 structural
The Structural Compartment_design_requirement originates from the structural discipline.

A.2.1.26.1.3 piping
The Piping Compartment_design_requirement originates from the piping discipline.

A.2.1.26.1.4 hvac
The Hvac Compartment_design_requirement originates from the hvac discipline.

A.2.1.26.1.5 electrical
The Electrical Compartment_design_requirement originates from the electrical discipline.

A.2.1.26.1.6 electronic
The Electronic Compartment_design_requirement originates from the electronic discipline

A.2.1.26.1.7 combat_system
The Combat_system Compartment_design_requirement originates from the combat_system discipline.

A.2.1.26.1.8 outfit_furnishing
The Outfit_furnishing Compartment_design_requirement originates from the outfit_furnishing discipline.

A.2.1.26.1.9 painting_coating
The Painting_coating Compartment_design_requirement originates from the painting_coating discipline.

A.2.1.26.1.10 user_defined
The User_defined Compartment_design_requirement is defined by the user.

A.2.1.26.2 requirement_description
The requirement_description specifies a textual description of the requirement that is to be met.

A.2.1.27 Compartment_design_to_functional_definition_relationship
The Compartment_design_to_functional_definition_relationship relates a Compartment_design_definition or any of its specialization's to a Compartment_functional_definition or any of its specialization's.
The data associated with a Compartment_design_to_functional_definition_relationship are the following:
— definition_1;
— definition_2

A.2.1.27.1 definition_1
The definition_1 specifies definition_1 shall be of type Compartment_design_definition.

A.2.1.27.2 definition_2
The definition_2 specifies definition_2 shall be of type Compartment_functional_definition.

A.2.1.28 Compartment_extents
The Compartment_extents are the locations in the global coordinate system of the boundaries of the Compartment through reference to positions in a Spacing_table.
The data associated with a Compartment_extents are the following:
— forwardmost_extent;
— aftmost_extent;
— portmost_extent;
— starboardmost_extent;
— topmost_extent;
— bottommost_extent

A.2.1.28.1 forwardmost_extent
The forwardmost_extent specifies forward end of the compartment, referring to a Spacing_position and offset in one of the longitudinal spacing tables.

A.2.1.28.2 aftmost_extent
The aftmost_extent specifies forward end of the compartment, referring to a Spacing_position and offset in one of the longitudinal spacing tables.

A.2.1.28.3 portmost_extent
The portmost_extent specifies furthest distance to port of the compartment, referring to a Spacing_position and offset in one of the transverse spacing tables.

A.2.1.28.4 starboardmost_extent
The starboardmost_extent specifies furthest distance to port of the compartment, referring to a Spacing_position and offset in one of the transverse spacing tables.

A.2.1.28.5 topmost_extent
The topmost_extent specifies highest vertical extent of the compartment, referring to a Spacing_position and offset in one of the vertical spacing tables.

A.2.1.28.6 bottommost_extent
The bottommost_extent specifies lowest vertical extent of the compartment, referring to a Spacing_position and offset in one of the vertical spacing tables.

A.2.1.29 Compartment_functional_definition
The Compartment_functional_definition defines the functional role of a Compartment; the role may be a pre-defined one which correspond to the Compartment subtypes or may be user-defined for combinations of the Compartment subtypes or identification of new types.
The data associated with a Compartment_functional_definition are the following:
— defined_for;
— used_for

A.2.1.29.1 defined_for
The defined_for specifies redefinition of the defined_for attribute a Compartment_functional_definition is only valid for a Compartment.

A.2.1.29.2 used_for
The used_for specifies the name of a function that a specific Compartment may have in a ship.
EXAMPLE 1 - Engine room and bow thruster room are types of Machinery compartments.
The value of the used_for shall be one of the following:
— cargo;
— habitable;
— machinery;
— tank;
— void;
— user_defined

A.2.1.29.2.1 cargo
Cargo: Predefined cargo functions corresponding to the specialization's of Compartment. The Compartment is designed to carry liquid, bulk, or containerized goods. These goods may be consumed during the voyage, as in the case of food or fuel, or they may be temporarily stored for transport between ports. A Tank compartment may also be used for the storage or transportation of liquid cargo.

A.2.1.29.2.2 habitable
Habitable: Predefined habitable functions corresponding to the specialization's of Compartment. The habitable the Compartment is designed as a habitable space, which is primarily designated as suitable for occupancy by humans. Passenger safety and comfort are subject to international, national, class_society or other regulations usually covered by product specifications and applicable class and register notations.

A.2.1.29.2.3 machinery

Machinery: Predefined machinery functions corresponding to the specialization's of Compartment. The Cargo_compartment is designed to contain machinery for the operation of the ship or in support of the vessels mission.

A.2.1.29.2.4 tank

Tank: Predefined tank functions corresponding to the specialization's of Compartment. The Compartment is designed to carry liquids used in the mission of the ship, or for the storage of liquid cargo's transported by the ship. Fuels for propulsion of the ship, potable water for the passengers and crew, waste products, petroleum product cargo, and fuel for aircraft supported by the ship are carried in Tank compartments.

A.2.1.29.2.5 void

Void: Predefined void functions corresponding to the specialization's of Compartment. The Compartment is designed as inaccessible, closed space that is never used to carry cargo or to be occupied by humans or to install any machinery or equipment in it. The main tasks of a Void compartment are segregating the cargo and or fluids, which are necessary to operate the ship and to create space for emergency access to other spaces.

A.2.1.29.2.6 user_defined

User_defined: Predefined user functions corresponding to the specialization's of Compartment. The type of Compartment is defined by the user_def_function attribute. This value may indicate the combination of two or more of the Pre_defined_compartment_function values, or any other user-defined classification for the Compartment.

A.2.1.30 Compartment_naval_administrative_property_set

A collection of identification and compartment design parameters that are applicable only to the design of naval vessels.

The data associated with a Compartment_naval_administrative_property_set are the following:

— compartment_abbreviated_name;
— department_ziplist_number;
— division_ziplist_number;
— maximum_accleration_g_force;
— nuclear_classification;
— noise_category;
— insulation_category;
— authorization_classification;
— safety_category;
— security_classification

A.2.1.30.1 compartment_abbreviated_name

The compartment_abbreviated_name specifies a short, compact, efficient means of referring to a particular compartment on a ship. The abbreviated name may or may not have an embedded meaning.

NOTE - In U.S. Naval vessels, abbreviated names are commonly used and are encoded such that they indicate the type of cargo, and the vertical, transverse, and longitudinal position, such as the abbreviation of 6.0 -55.0 -1.0 F for the freshwater tank above the 6.0th deck, beginning at frame 55.0, and located on the starboard side.

A.2.1.30.2 department_ziplist_number
The department_ziplist_number specifies an organization-specific identifier used for departmental control over the compartment during an overhaul or repair availability.

A.2.1.30.3 division_ziplist_number
The division_ziplist_number specifies an organization-specific identifier used for division control over the compartment during an overhaul or repair availability.

A.2.1.30.4 maximum_accleration_g_force
The maximum_accleration_g_force specifies a measure of the maximum allowable acceleration force, expressed as a ratio compared to the acceleration of gravity, allowed for a compartment. This force is represented as a single value and governs the accelerations in all three principal directions (e.g. vertical, longitudinal, and transverse).

EXAMPLE 2 - A value of 1.5 would represent one-and-a-half times the force of gravity (32.2 ft sec2), or 48.3 ft sec2.

A.2.1.30.5 nuclear_classification
The nuclear_classification specifies an indicator used to denote whether the compartment is designated a nuclear or non-nuclear space. This designation applies to spaces specifically designed to contain such things as nuclear reactors as well as spaces used for the storage or repair of nuclear weapons.
The value of the nuclear_classification shall be one of the following nuclear non_nuclear nuclear the Compartment contains nuclear propulsion systems or is used for storage or repair of nuclear weapons. Non_nuclear the Compartment does not contain nuclear propulsion systems nor is it used for storage o r repair of nuclear weapons.
The nuclear_classification shall be one of the following:
— nuclear;
— non_nuclear

A.2.1.30.5.1 nuclear
Nuclear: designated a nuclear space

A.2.1.30.5.2 non_nuclear
Non_nuclear: designated a non-nuclear space

A.2.1.30.6 noise_category
The noise_category specifies a single alphabetical character key used to denote whether special consideration is to be given to the compartment with respect to the internal level of sound.
The noise_category shall be one of the following:
— A;
— B;
— C;
— D;
— E;
— F

A.2.1.30.6.1 A
The Compartment shall be designed for intelligible speech (low noise).

A.2.1.30.6.2 B
The Compartment shall be designed for comfort.

A.2.1.30.6.3 C
The Compartment shall be designed for quiet.

A.2.1.30.6.4 D
The Compartment shall be designed for deafness avoidance.

A.2.1.30.6.5 E
The Compartment shall be designed for intelligible speech- high noise.

A.2.1.30.6.6 F
The Compartment shall be designed for intelligible speech- topside.

A.2.1.30.7 insulation_category
The insulation_category specifies an indicator used to denote what type of consideration is to be given to the compartment with regard to its thermal conductivity.
The value of the insulation_category shall be one of the following:
— A;
— B;
— C;
— D;
— E;
— F;
— G;
— H;
— I;
— J

A.2.1.30.7.1 A
The Compartment shall be designed for a specific temperature range and or compartment type.

A.2.1.30.7.2 B
The Compartment shall be designed for a specific temperature range and or compartment type.

A.2.1.30.7.3 C
The Compartment shall be designed for a specific temperature range and or compartment type.

A.2.1.30.7.4 D
The Compartment shall be designed for a specific temperature range and or compartment type.

A.2.1.30.7.5 E
The Compartment shall be designed for a specific temperature range and or compartment type.

A.2.1.30.7.6 F
The Compartment shall be designed for a specific temperature range and or compartment type.

A.2.1.30.7.7 G
The Compartment shall be designed for a specific temperature range and or compartment type.

A.2.1.30.7.8 H
The Compartment shall be designed for a specific temperature range and or compartment type.

A.2.1.30.7.9 I
The Compartment shall be designed for a specific temperature range and or compartment type.

A.2.1.30.7.10 J
The Compartment shall be designed for a specific temperature range and or compartment type.

A.2.1.30.8 authorization_classification
The authorization_classification specifies a type of crew restriction placed on the use of the compartment.
The value of the authorization_classification shall be one of the following:
— officers_only;
— crew_only;
— restricted;
— unrestricted

A.2.1.30.8.1 officers_only
The Compartment is designated for the use of officers only

A.2.1.30.8.2 crew_only
The Compartment is designated for the use of crew only

A.2.1.30.8.3 restricted
The Compartment is designated as a restricted area.

A.2.1.30.8.4 unrestricted
The Compartment access is not limited to any particular group.

A.2.1.30.9 safety_category
The safety_category specifies an indicator used to denote special consideration for the compartment with regard to a hazardous working environment for humans.
The value of the safety_category shall be one of the:
— A;
— B;
— C

A.2.1.30.9.1 A
Compartment is designated as safety class A.

A.2.1.30.9.2 B
Compartment is designated as safety class B.

A.2.1.30.9.3 C
Compartment is designated as safety class C.

A.2.1.30.10 security_classification
The security_classification specifies an indicator used to denote special considerations for the compartment with respect to accessibility and security clearances.
The value of the security_classification shall be one of the following:
— unclassified;
— classified;
— secret

A.2.1.30.10.1 unclassified
Unclassified: Unclassified accessibility, security clearances not required

A.2.1.30.10.2 classified
Classified: Classified accessibility and security clearances

A.2.1.30.10.3 secret
Secret: Secret accessibility and security clearances

A.2.1.31 Compartment_permeability
Compartment_permeability is a measure, expressed as a percentage of the volume or footprint area of the compartment representing open space (i.e. not occupied by equipment, structure, machinery, etc.) that would flood in the event the spaces watertight integrity was damaged. These are key parameters for the damage stability calculations for a ship.
The data associated with a Compartment_permeability are the following:
— volume_permeability;
— area_permeability

A.2.1.31.1 volume_permeability
The volume_permeability specifies the percentage of the occupied space of a Compartment.

A.2.1.31.2 area_permeability
The area_permeability specifies the percentage of the occupied surface of a Compartment.

A.2.1.32 Compartment_property_set
Compartment_property_set is a supertype of all entities defining properties for a compartment. A property is a measure of some significant characteristic of a compartment associated with a specific context. The contexts may be either maximum, minimum, estimated, calculated, or measured. A compartment may have multiple property sets; however, a property set must be unique with respect to the compartment property context, and compartment. A set of properties is related to the appropriate Compartment through the defined_for attribute and is assigned as applicable to a particular Compartment design definition or functional definition through a definition relationship.
The data associated with a Compartment_property_set are the following:
— contxt;
— defined_for.

A.2.1.32.1 contxt
The contxt specifies an indicator used to associate a design meaning with a compartment property.
The value of the context shall be one of the following:
— maximum;
— minimum;
— estimated;
— calculated;
— measured

A.2.1.32.1.1 maximum
Maximum: The maximum design limits for the property

A.2.1.32.1.2 minimum
Minimum: The minimum design limits for the property

A.2.1.32.1.3 estimated
Estimated: The estimated degree of accuracy for the property value

A.2.1.32.1.4 calculated
Calculated: The calculated degree of accuracy for the property value

A.2.1.32.1.5 measured
Measured: The measured degree of accuracy for the property value

A.2.1.32.2 defined_for
The defined_for specifies redefinition of the defined_for attribute; a Compartment_property_set is only valid for a Compartment.

A.2.1.33 Compartment_property_to_design_definition_relationship
The Compartment_property_definition_relationship relates a Compartment_property_set or any of its specialization's to a Compartment_design_definition or any of its specialization's.
The data associated with a Compartment_property_to_design_definition_relationship are the following:
— definition_1;
— definition_2

A.2.1.33.1 definition_1
The definition_1 specifies definition_1 shall be of type Compartment_property_set.

A.2.1.33.2 definition_2
The definition_2 specifies definition_2 shall be of type Compartment_design_definition.

A.2.1.34 Compartment_volume
Compartment_volume describes the volume properties of a Compartment.
The data associated with a Compartment_volume are the following:
— volume;
— centre_of_volume.

A.2.1.34.1 volume
The volume specifies the volume of a Compartment.

A.2.1.34.2 centre_of_volume
The centre_of_volume specifies the centre_of_volume of a Compartment in relation to the global coordinate system of the ship.

A.2.1.35 composite_curve
The Composite_curve is a collection of curves joined end to end. The individual segments of the curve are themselves defined as composite_curve_segments. The parameterization of the composite curve is an accumulation of the parametric ranges of the referenced bounded curves.
The data associated with a composite_curve are the following:
— segments;
— self_intersect;
— n_segments;
— closed_curve

A.2.1.35.1 segments
The Segments are the component bounded curves, their transitions and senses. The transition attribute for the last segment defines the transition between the end of the last segment and the start of the first; this transition attribute may take the value discontinuous, which indicates an open curve.

A.2.1.35.2 self_intersect
The Self_intersect is the indication of whether the curve intersects itself or not.

A.2.1.35.3 n_segments
The N_segments are the number of component curves.

A.2.1.35.4 closed_curve
The Closed_curve is the indication of whether the curve is closed or not; this is derived from the transition code on the last segment.

A.2.1.36 composite_curve_on_surface
The Composite_curve_on_surface is a collection of segments that are curves on a surface. Each segment shall lie on the basis surface, and may be a surface_curve, a pcurve, or a composite_curve_on_surface.
The data associated with a composite_curve_on_surface are the following:
— basis_surface.
The Basis_surface is the surface on which the composite curve is defined.

A.2.1.37 composite_curve_segment
The Composite_curve_segment is a bounded curve together with transition information that is used to construct a composite_curve.
The data associated with a composite_curve_segment are the following:
— transition;
— same_sense;
— parent_curve;
— using_curves

A.2.1.37.1 transition
The Transition is the state of transition (i.e., geometric continuity from the last point of this segment to the first point of the next segment) in a composite curve.
The transition shall be one of the following:
— discontinuous;
— continuous;
— cont_same_gradient;
— cont_same_gradient_same_curvature

A.2.1.37.1.1 discontinuous
In the Discontinuous enumeration, the segments, or patches, do not join. This is permitted only at the boundary of the curve or surface to indicate that it is not closed.

A.2.1.37.1.2 continuous
In the Continuous enumeration, the segments, or patches, join, but no condition on their tangents is implied.

A.2.1.37.1.3 cont_same_gradient

In the Cont_same_gradient enumeration, the segments, or patches, join, and their tangent vectors, or tangent planes, are parallel and have the same direction at the joint; equality of derivatives is not required.

A.2.1.37.1.4 cont_same_gradient_same_curvature

In the Cont_same_gradient_same_curvature enumeration, for a curve, the segments join, their tangent vectors are parallel and in the same direction, and their curvatures are equal at the joint; equality of derivatives is not required. For a surface this implies that the principal curvatures are the same and that the principal directions are coincident along the common boundary.

A.2.1.37.2 same_sense

The Same_sense is an indicator of whether or not the sense of the segment agrees with, or opposes, that of the parent curve. If Same_sense is false, the point with highest parameter value is taken as the first point of the segment.

A.2.1.37.3 parent_curve

The Parent_curve is the bounded curve that defines the geometry of the segment.

A.2.1.37.4 using_curves

The Using_curves is the set of composite_curves that use this composite_curve_segment as a segment. This set shall not be empty.

A.2.1.38 conic

The Conic is a planar curve that could be produced by intersecting a plane with a cone. A conic curve is defined in terms of its intrinsic geometric properties rather that being described in terms of other geometry. A conic entity always has a placement coordinate system defined by axis2_placement; the parametric representation is defined in terms of this placement coordinate system.

The data associated with a conic are the following:

— position

The Position is the location and orientation of the conic.

A.2.1.39 connected_face_set

The Connected_face_set is a set of faces such that the domain of the faces together with their bounding edges and vertices is connected.

The data associated with a connected_face_set are the following:

— cfs_faces

The Cfs_faces are a set of faces arcwise connected along common edges or vertexes.

A.2.1.40 coordinated_universal_time_offset

The Coordinated_universal_time_offset entity is used to relate a time to coordinated universal time by an offset (specified in hours and minutes) and a direction. It is referenced by Local_time and is composed of an hour_offset attribute, which is the hour in the day, a minute_offset attribute, which is the minute in the hour, and a sense attribute, which senses if the time is ahead or behind.

The data associated with a coordinated_universal_time_offset are the following:

— hour_offset;

— minute_offset;

— sense

A.2.1.40.1 hour_offset
The Hour_offset is the number of hours by which a time is offset from coordinated universal time.

A.2.1.40.2 minute_offset
The Minute_offset is the number of minutes by which a time is offset from coordinated universal time.

A.2.1.40.3 sense
The Sense indicates if the time is ahead or behind.
The sense shall be one of the following:
— ahead;
— behind

A.2.1.40.3.1 ahead
Ahead indicates that the time is ahead.

A.2.1.40.3.2 behind
Behind indicates that the time is behind.

A.2.1.41 curve
The Curve is a set of mathematical points which is the image, in two- or three-dimensional space, of a continuous function defined over a connected subset of the real line, and which is not a single point.

A.2.1.42 date
The Date entity is the identification of a day or week in a year, is a subtype of Calendar_date and is composed of a year_component attribute.
The data associated with a date are the following:
— year_component
TheYear_component is the year in which the date occurs.

A.2.1.43 date_and_time
The Date_and_time entity is a moment of time on a particular day, and is referenced by the Simple_definition_version and the Date_time_select entity. It is composed of a Date_component and a Time_component.
The data associated with a date_and_time are the following:
— date_component;
— time_component

A.2.1.43.1 date_component
The Date_component is the date element of the date time combination.

A.2.1.43.2 time_component
The Time_component is the time element of the date time combination.

A.2.1.44 Definable_object
The global unique identifier for SELF
The data associated with a Definable_object are the following:
— id;
— definitions

A.2.1.44.1 id
The id specifies the Definitions pointing to SELF

A.2.1.44.2 definitions
The Definitions describe the definable_object.

A.2.1.45 Definition
The global unique identifier for SELF
The data associated with a Definition are the following:
— id;
— defined_for;
— local_units;
— version

A.2.1.45.1 id
The id specifies the Definable_objects that are defined by Definition

A.2.1.45.2 defined_for
The defined_for specifies if the units that definition makes use of differ from the ones globally defined for the ship, they should be specified here.

A.2.1.45.3 local_units
The local_units specifies a simple version to support basic versioning as a conformance class. This replaces the full Configuration Management model in the ISO schemas.

A.2.1.45.4 version
The version_id provides simple version control. The version_id need not be specified for a particular Definition.

A.2.1.46 Definition_relationship
The Definition_relationship provides the possibility to relate two Definitions
The data associated with a Definition_relationship are the following:
— definition_1;
— definition_2;
— description

A.2.1.46.1 definition_1
The definition_1 specifies one of the related Definitions.

A.2.1.46.2 definition_2
The definition_2 specifies the related entities are only allowed to be of type Definition

A.2.1.46.3 description
The description is optional and if present, specifies the description, if required, for this relationship

A.2.1.47 Derived_Unit
The Derived_unit entity is an expression of units, and is composed of a set of Derived_unit_element.
The data associated with a Derived_Unit are the following:
— elements

The Elements is the group of units and their exponents that define the derived_unit.

A.2.1.48 Derived_Unit_Element

The Derived_unit_element entity is one of the unit quantities, which makes up a derived_unit, and is composed of a Unit and an Exponent.
The data associated with a Derived_Unit_Element are the following:
— unit;
— exponent

A.2.1.48.1 unit

The Unit is the fixed quantity that is used as the mathematical factor.

A.2.1.48.2 exponent

The Exponent is the power that is applied to the unit attribute.

A.2.1.49 Design_definition

The Design_definition is the SUPERTYPE for all kinds of design definitions
The data associated with a Design_definition are the following:
— representations
The representations specify the representations of the design definition; it is possible for a Design_definition to have multiple Representations.

A.2.1.50 Design_requirement

When a Design_requirement is created the related ship exists in the heads of the designers only. Therefore this information is not specific with respect to product item, because they are not yet defined, but it is very specific with respect to the expectations the future owner and the designers would like to express.
The data associated with a Design_requirement are the following:
— specification
The specification specifies Specification documenting the design requirement

A.2.1.51 dimensional_exponents

The Dimensional_exponents entity defines the powers of the dimensions of the base quantities. The dimensionality of any quantity can be expressed as a product of powers of the dimensions of base quantities. Dimensional_exponents are referenced by Named_unit and Si_unit and are composed of the following attributes:
— length_exponent;
— mass_exponent;
— time_exponent;
— electric_current_exponent;
— thermodynamic_temperature_exponent;
— amount_of_substance_exponent;
— luminous_intensity_exponent

A.2.1.51.1 length_exponent

The Length_exponent is the power of the length base quantity.

A.2.1.51.2 mass_exponent

The Mass_exponent is the power of the mass base quantity.

A.2.1.51.3 time_exponent
The Time_exponent is the power of the time base quantity.

A.2.1.51.4 electric_current_exponent
The Electric_current_exponent is the power of the electric current base quantity.

A.2.1.51.5 thermodynamic_temperature_exponent
The Thermodynamic_temperature_exponent is the power of the thermodynamic temperature base quantity.

A.2.1.51.6 amount_of_substance_exponent
The Amount_of_substance_exponent is the power of the amount of substance base quantity.

A.2.1.51.7 luminous_intensity_exponent
The Luminous_intensity_exponent is the power of the luminous intensity base quantity.

A.2.1.52 direction
The Direction defines a general direction vector in two- or three-dimensional space. The actual magnitudes of the components have no effect upon the direction being defined, only the rations z:y:z or x:y are significant. The data associated with a direction are the following:
— direction_ratios
The Direction_rations are the component in the X, Y, or Z-axis. Z will not be present in the case of a direction in two-dimensional coordinate space.

A.2.1.53 Document
A Document is an unambiguous identification of some human readable data item defined outside ISO 103030. Each document has an author and a version history.
The data associated with a Document are the following:
— has_author
The Has_author is the person and or organization responsible for the creation of the document.

A.2.1.54 Document_reference
A Document_reference is the qualification of a Document in terms of its source or location.

A.2.1.55 edge
The Edge is a topological construct corresponding to the connection between two vertices.
The data associated with an edge are the following:
— edge_start;
— edge_end

A.2.1.55.1 edge_start
The Edge_start is the start point (vertex) of the edge.

A.2.1.55.2 edge_end
The Edge_end is the end point (vertex) of the edge. The same vertex can be used for both edge_start and edge_end.

A.2.1.56 edge_curve
The Edge_curve is a special subtype of edge that has its geometry fully defined. The geometry is defined by associating the edge with a curve that may be unbounded. As the topological and geometric directions may be opposed, an indicator (same_sense) is used to identify whether the

edge and curve directions agree or are opposed. The logical value indicates whether the curve direction agrees with (TRUE) or is in the opposite direction (FALSE) to the edge direction. Any geometry associated with the vertices of the edge shall be consistent with the edge geometry. Multiple edges can reference the same curve.
The data associated with an edge_curve are the following:
— edge_geometry;
— same_sense

A.2.1.56.1 edge_geometry
The Edge_geometry is the curve that defines the shape and spatial location of the edge. This curve may be unbounded and is implicitly trimmed by the vertices of the edge; this defines the edge domain.

A.2.1.56.2 same_sense
The Same_sense is the logical flag indicating whether (TRUE), or not (FALSE) the senses of the edge and the curve defining the edge geometry are the same. The sense of an edge is from the edge start vertex to the edge end vertex; the sense of a curve is in the direction of increasing parameter.

A.2.1.57 edge_loop
The Edge_loop is a loop with nonzero extent. It is a path in which the start and end vertices are the same. Its domain, if present, is a closed curve. An edge_loop may overlap itself.
The data associated with an edge_loop are the following:
— ne
The Ne is the number of elements in the edge list of the path supertype.

A.2.1.58 elementary_surface
The Elementary_surface is a simple analytic surface with defined parametric representation.
The data associated with an elementary_surface are the following:
— position
The Position is the position and orientation of the surface. This attribute is used in the definition of the parameterization of the surface.

A.2.1.59 ellipse
The Ellipse is a conic section defined by the lengths of the semi-major and semi_minor diameters and the position (center or mid point of the line joining the foci) and orientation of the curve.
The data associated with an ellipse are the following:
— semi_axis_1;
— semi_axis_2

A.2.1.59.1 semi_axis_1
The Semi_axis_1 is the first radius of the ellipse that shall be positive.

A.2.1.59.2 semi_axis_2
The Semi_axis_2 is the second radius of the ellipse that shall be positive.

A.2.1.60 Extended_conversion_based_unit
An Extended_conversion_based_unit is a Unit that is defined based on an Extended_measure_with_unit. It is an extension of the P41 Entity Conversion_based_unit

EXAMPLE 3 - An inch is an Extended_conversion_based_unit. It is from the Imperial system, its name is inch and it can be related to the Si_unit, millimetre, through an Extended_measure_with_unit whose

value is 25.4 millimetre. A foot is also an Extended_conversion_based_unit. It is from the Imperial system, its name is foot and it can be related to the si unit, millimetre, either directly or through the unit called inch.

The data associated with an Extended_conversion_based_unit are the following:

— name;
— conversion_factor

A.2.1.60.1 name

The name specifies the word or group of words by which the Extended_conversion_based_unit is referred to.

A.2.1.60.2 conversion_factor

The conversion_factor specifies the physical quantity from which the Extended_conversion_based_unit is derived.

A.2.1.61 Extended_measure_with_unit

From P41 An Extended_measure_with_unit is the specification of a physical quantity as defined in ISO 31.0 (clause 2.0).

The data associated with an Extended_measure_with_unit are the following:

— value_component;
— unit_component

A.2.1.61.1 value_component

The value_component specifies the value of the physical quantity when expressed in the specified units.

A.2.1.61.2 unit_component

The unit_component specifies the unit of the physical quantity

A.2.1.62 External_instance_reference

Reference to an item or definition that is external to this transfer,

The data associated with an External_instance_reference are the following:

— schema_name;
— entity_type;
— GUID;
— version;
— Component

A.2.1.62.1 schema_name

The schema_name specifies the name of the schema that contains the definition instance.

A.2.1.62.2 entity_type

The entity_type specifies the type of the definition instance. This is a subtype of definition.

A.2.1.62.3 GUID

The GUID specifies A persistent, global identifier that uniquely identifies the definition.

A.2.1.62.4 version

The Version specifies the version of the external_instance_reference.

A.2.1.62.5 Component

The Component is optional and if present, specifies an optional field that can be used to specify a component of a definition, e.g. ports of a fitting or surfaces of a moulded_form_design_definition

A.2.1.63 External_reference

An External_reference is the abstract denotation of a data source external to the data set where an instance of this entity exists.

EXAMPLE 4 - a WWW uniform resource locator denotes such a data source

The data associated with an External_reference are the following:

— location;
— description

A.2.1.63.1 location

The location specifies the location of an external reference, in case of an Universal_resource_locator a computer accessible by a specified transmission protocol

A.2.1.63.2 description

The description specifies some additional information

A.2.1.64 face

The Face is a topological entity of dimensionality 2 corresponding to the intuitive notion of a piece of surface bounded by loops. Its domain, if present, is an oriented, connected, finite 2-manifold in Rm.

The data associated with a face are the following:

— bounds

The Bounds are the boundaries of the face; no more that one of these shall be a face_outer_bound.

A.2.1.65 face_based_surface_model

The Face_based_surface_model is described by a set of connected_face_sets of dimensionality 2. The connected_face_sets shall not intersect except at edges and vertices, except that a face in one connected face set may overlap a face in another connected face set, provided the face boundaries are identical. There shall be at least one connected_face_set. A connected_face_set may exist independently of a surface_model.

The data associated with a face_based_surface_model are the following:

— fbsm_faces

The Fbsm_faces are the set of connected_face_sets comprising the face_based_surface_model.

A.2.1.66 face_bound

The Face_bound is a loop that is intended to be used for bounding a face.

The data associated with a face_bound are the following:

— bound;
— orientation

A.2.1.66.1 bound

The Bound is the loop that will be used as a face boundary.

A.2.1.66.2 orientation

The Orientation indicates whether (TRUE), or not (FALSE) the loop has the same sense when used to bound the face as when first defined. If orientation is FALSE, the senses of all its component oriented edges are implicitly reversed when used in the face.

A.2.1.67 face_outer_bound

The Face_outer_bound is a special subtype of face_bound that carries the additional semantics of defining an outer boundary on the face. No more that one boundary of a face shall be of this type.

A.2.1.68 face_surface

The Face_surface is a subtype of face in which the geometry is defined by an associated surface. The portion of the surface used by the face shall be embeddable in the plane as an open disk, possibly with holes. However, the union of the face with the edges and vertices of its bounding loops need not be embeddable in the plane. It may, for example, cover an entire sphere or torus. As both a face and a geometric surface have defined normal directions, a Boolean flag (the orientation attribute) is used to indicate whether the surface normal agrees with (TRUE) or is opposed to (FALSE) the face normal direction. The geometry associated with any component of the loops of the face shall be consistent with the surface geometry, in the sense that the domains of all the vertex points and edge curves are contained in the face geometry surface. A surface may be referenced by more that one face_surface.
The data associated with a face_surface are the following:

— face_geometry;
— same_sense

A.2.1.68.1 face_geometry

The Face_geometry is the surface that defines the internal shape of the face. This surface may be unbounded. The domain of the face is defined by this surface and the bounding loops in the inherited attribute SELF/face.bounds.

A.2.1.68.2 same_sense

The Same_sense is the flag that indicates whether the sense of the surface normal agrees with (TRUE), or opposes (FALSE), the sense of the topological normal to the face.

A.2.1.69 Functional_definition

The Functional_definition is the SUPERTYPE for all kinds of functional definitions
The data associated with a Functional_definition are the following:

— user_def_function

The user_def_function is optional and, if present, specifies each non-abstract SUBTYPE of Functional_definition should declare (or inherit) an attribute the_function with a special ENUMERATION type containing the possible functions for the Definable_object subtype it points to with at least one entry called USER_DEFINED. In the case of the_function USER_DEFINED the user_def_function attribute has to be used to determine the user defined function.

A.2.1.70 General_characteristics_definition

The General_characteristics_definition provides a major part of the documentation of the vessel. It includes primary dimensions and capacities due to the contract of the product (ship).
The data associated with a General_characteristics_definition are the following:

— defined_for

The defined_for specifies A defined_for specifies a Ship or set of Ships for which the General_characteristics_definition applies.

A.2.1.71 General_compartment_property_set

General_compartment_property_set references all entities defining generic properties that are applicable to all types of compartments. A set of such property entities is attached to the Compartment through the redeclared defined_for attribute.

The data associated with a General_compartment_property_set are the following:

— volume;
— areas;
— permeability;
— required_bulkhead_tightness

A.2.1.71.1 volume

The volume specifies reference to the collection of volume properties for a Compartment.

A.2.1.71.2 areas

The areas specify reference to the collection of area properties for a Compartment.

A.2.1.71.3 permeability

The permeability specifies reference to the collection of key parameters for the stability calculations of a ship.

A.2.1.71.4 required_bulkhead_tightness

The required_bulkhead_tightness specifies reference to an indicator for the ability to prevent the passage of air and or liquid for all bulkheads forming the compartment boundary.

The value of the required_bulkhead_tightness shall be one of the following:

— air_tight;
— fume_tight;
— water_tight;
— oil_tight;
— non_tight;
— expanded_metal;
— user_defined

A.2.1.71.4.1 air_tight

Air_tight: Required ability to prevent the passage of air for all bulkheads

A.2.1.71.4.2 fume_tight

Fume_tight: Required ability to prevent the passage of funes for all bulkheads

A.2.1.71.4.3 water_tight

Water_tight: Required ability to prevent the passage of water for all bulkheads

A.2.1.71.4.4 oil_tight

Oil_tight: Required ability to prevent the passage of oil for all bulkheads

A.2.1.71.4.5 non_tight

Non_tight: Not required to prevent the passage of air, oil, water, or fumes for all bulkheads

A.2.1.71.4.6 expanded_metal
Expanded_metal: the Compartment boundaries may consist of expanded metal mesh and therefore will not be closed to prevent the passage of air, oil, water, or fumes.

A.2.1.71.4.7 user_defined
User_defined: Required ability to have user defined tightness for all bulkheads

A.2.1.72 geometric_representation_context
The Geometric_representation_context is a representation_context in which geometric_representation_items are geometrically founded. A geometric_representation_context is a distinct coordinate space, spatially unrelated to other coordinate spaces except as those coordinate spaces are specifically related by an appropriate transformation.
The data associated with a geometric_representation_context are the following:
— coordinate_space_dimension
The Coordinate_space_dimension is the integer dimension_count of the coordinate space that is the geometric_representation_context.

A.2.1.73 geometric_representation_item
The Geometric_representation_item is a representation_item that has the additional meaning of having geometric position or orientation or both. This meaning is present by virtue of:
-- being a cartesian_point or a direction;
-- referencing directly a cartesian_point or a direction;
-- referencing indirectly a ceartesian_point or a direction.
The data associated with a geometric_representation_item are the following:
— dim
The Dim is the coordinate dimension_count of the geometric_representation_item.

A.2.1.74 global_axis_placement
A global_axis_placement is a type of general_characteristics_definition that defines a fixed system of right handed orthogonal axes to which geometric data are referred. A global_axis_placement has a positive z-axis in an upwards direction starting from the base of the ship, positive x-axis running along the ship on the intersection of the centerline with the base and is in one case directed from the after part of the ship to the forward part of the ship or in the other case is directed from the forward part of the ship to the aft part of the ship. Origin of the global axis placement can be any point on the x-axis. The distance of the after perpendicular from the origin and the orientation of the x-axis shall be specified. If any other system of axes is used, local or global, then the transformation relations between it and the global_axis_placement shall be specified.
The data associated with a global_axis_placement are the following:
— orientation;
— after_perpendicular_offset;
— forward_perpendicular_offset;
— length_between_perpendiculars

A.2.1.74.1 orientation
The orientation specifies the direction of the x-axis.
The value of orientation shall be one of the following:
— aft_pointing;
— forward_pointing

A.2.1.74.1.1 aft_pointing

Aft_pointing: an orientation of a right handed ship coordinate system that has the positive x axis from the forward part of the ship directed to the aft part of the ship.

A.2.1.74.1.2 forward_pointing

Forward_pointing: an orientation of a right handed ship coordinate system that has the positive x axis from the aft part of the ship directed to the forward part of the ship.

A.2.1.74.2 after_perpendicular_offset

The after_perpendicular_offset specifies the distance from the origin of the global_axis_placement to the after perpendicular.

A.2.1.74.3 forward_perpendicular_offset

The Forward_perpendicular_offset specifies the distance from the origin of the global_axis_placement to the forward perpendicular.

A.2.1.74.4 length_between_perpendiculars

The Length_between_perpendicular defines the distance between perpendiculars.

A.2.1.75 Global_id

The Global_id is a persistent, global identifier that uniquely identifies the definition.
The data associated with a Global_id are the following:

— company_id;
— local_id

A.2.1.75.1 company_id

The company_id specifies A unique identifier for the company that created the data. The string is left justified and blank filled

A.2.1.75.2 local_id

The local_id specifies a persistent identifier that uniquely identifies this definition throughout the company. Assigned at the time a definition is created. The string is left justified and blank filled

A.2.1.76 Habitable_compartment_design_definition

A Habitable_compartment_design_definition is a type of Compartment_design_definition that provides the abstract definition of a version of a Habitable compartment from a design perspective.

A.2.1.77 Habitable_compartment_functional_definition

Defines the functional role of a Habitable_compartment; the role may be a pre-defined one or may be user-defined.
The data associated with a Habitable_compartment_functional_definition are the following:

— func.

The func specifies the name of a function that a specific Habitable_compartment may have in a ship.
The value of the func shall be one of the following:

— berthing;
— cabin;
— control;
— passageway;
— medical;

— lounge;
— access_trunk;
— user_defined

A.2.1.77.1.1 berthing
The Habitable_compartment is designed to be used as berthing space.

A.2.1.77.1.2 cabin
The Habitable_compartment is designed to be used as cabin space.

A.2.1.77.1.3 control
The Habitable_compartment is designed to be used for ship command and control functions.

A.2.1.77.1.4 passageway
The Habitable_compartment is designed to be used as a passageway.

A.2.1.77.1.5 medical
The Habitable_compartment is designed to be used as a medical space.

A.2.1.77.1.6 lounge
The Habitable_compartment is designed to be used as a lounge space.

A.2.1.77.1.7 access_trunk
The Habitable_compartment is designed to be used as an access trunk.

A.2.1.77.1.8 user_defined
The use of the Habitable_compartment is defined by the user_def_function attribute.

A.2.1.78 Habitable_compartment_property_set
Habitable_compartment_property describes properties specific to compartments designed for human habitation.
The data associated with a Habitable_compartment_property_set are the following:
— max_occupancy;
— air_circulation_rate;
— illumination_value

A.2.1.78.1 max_occupancy
The max_occupancy is optional and if present, specifies defines the maximum number of humans, which are allowed to occupy a Habitable_compartment.

A.2.1.78.2 air_circulation_rate
The air_circulation_rate specifies measure of the volume of air changes for the compartment per unit of time. This entity is used by applications performing HVAC load analyses.
NOTE - There are other attributes required to support HVAC activities and they will be defined in the future STEP HVAC AP.

A.2.1.78.3 illumination_value
The illumination_value specifies amount of lighting required for a compartment. Electrical applications will make use of this property in performing lighting analysis.

A.2.1.79 intersection_curve

The Intersection_curve is a curve that results from the intersection of two surfaces. It is represented as a special subtype of the surface_curve entity having two distinct surface associations defined via the associated geometry list.

A.2.1.80 Item

An Item is a discrete, identifiable thing used in one or more production activities. An Item is something (to be) created by a physical or mental activity or (automatically) derived from one or more other Items. An Item needs not to represent a physically realizable thing. It may also represent some abstract concept like activity, task etc. Item provides the functionality to have relationships to other Items and to be member in an Item_structure. Each entity that is intended to have such functionality should inherit from Item.
The data associated with an Item are the following:

— name;
— documentation;
— description;
— ship_context

A.2.1.80.1 name

The name specifies the (human readable) name of the concept that is represented by SELF

A.2.1.80.2 documentation

The documentation specifies documentation available (if it exists) for SELF

A.2.1.80.3 description

The description specifies the description for SELF

A.2.1.80.4 ship_context

The ship_context is optional and if present, specifies the context of SELF in terms of its applicability to a ship

A.2.1.81 Length_Unit

The Length_unit entity is a subtype of Named_unit.

A.2.1.82 line

The Line is an unbounded curve with constant tangent direction. A point and a direction define a line. The positive direction of the line is in the direction of the dir vector.
The data associated with a line are the following:

— pnt;
— dir

A.2.1.82.1 pnt

The Pnt is the location of the line.

A.2.1.82.2 dir

The Dir is the direction of the line; the magnitude and units of dir affect the parameterization of the line.

A.2.1.83 local_time

The Local_time entity is referenced by the Date_and_time entity and is composed of a hour_component, a minute_component, a second_component, and a zone attribute, which is a coordinated_universal_time_offset.

The data associated with a local_time are the following:

— hour_component;
— minute_component;
— second_component;
— zone

A.2.1.83.1 hour_component

The Hour_component is the hour portion of the local_time.

A.2.1.83.2 minute_component

The Minute_component is the minute portion of the local_time.

A.2.1.83.3 second_component

The Second_component is the second portion of the local_time.

A.2.1.83.4 zone

The Zone is the local time zone.

A.2.1.84 Longitudinal_position

A Longitudinal_position is a type of Spacing_position that specifies a location on the x-axis of the ship coordinate system. A Longitudinal_position may be defined by an offset distance from a given Spacing_position.

The data associated with a Longitudinal_position are the following:

— offset

The offset is optional and, if present, specifies the distance from the location specified in Spacing_position

A.2.1.85 loop

The Loop is a topological entity constructed from a single vertex, or by stringing together connected (oriented) edges, or linear segments beginning and ending at the same vertex. A loop is represented by a single vertex, or by an ordered collection of oriented_edges, or by an ordered collection of points.

A.2.1.86 Machinery_compartment_design_definition

A Machinery_compartment_design_definition is a type of Compartment_design_definition that provides the abstract definition of a version of a Machinery compartment from a design perspective.

A.2.1.87 Machinery_compartment_functional_definition

Defines the functional role of a Machinery_compartment; the role may be a pre-defined one or may be user-defined.

The data associated with a Machinery_compartment_functional_definition are the following:

— func.

The func specifies the name of a function that a specific Machinery_compartment may have in a ship.

The value of func shall be one of the following:

— main_engine_room;
— auxiliary_engine_room;
— bow_thruster_room;
— equipment_room;
— user_defined

A.2.1.87.1.1 main_engine_room
The Machinery_compartment is designed to be used as the main engine room.

A.2.1.87.1.2 auxiliary_engine_room
The Machinery_compartment is designed to be used as an auxiliary engine room space.

A.2.1.87.1.3 bow_thruster_room
The Machinery_compartment is designed to be used as the bow thruster room.

A.2.1.87.1.4 equipment_room
The Machinery_compartment is designed to be used as an equipment_room.

A.2.1.87.1.5 user_defined
The use of the Machinery_compartment is defined by the user_def_function attribute.

A.2.1.88 Moments_of_inertia
The Moments_of_inertia specify the moments_of_inertia represented by the value of 2.0nd moment of the boundary formed by the intersection of the compartment and a plane representing the cargo interface.
The data associated with a Moments_of_inertia are the following:
— long_moment_of_inertia;
— trans_moment_of_inertia;
— vert_moment_of_inertia

A.2.1.88.1 long_moment_of_inertia
The long_moment_of_inertia specifies the long_moment_of_inertia represents the value of 2.0nd moment of the boundary formed by the intersection of the compartment and a plane representing the cargo interface (i.e. liquid cargo static waterline). The lever of the moment is parallel to the longitudinal axis of the ship.

A.2.1.88.2 trans_moment_of_inertia
The trans_moment_of_inertia specifies the trans_moment_of_inertia represents the value of 2.0nd moment of the boundary formed by the intersection of the compartment and a plane representing the cargo interface (i.e. liquid cargo static waterline). The lever of the moment is parallel to the transverse axis of the ship.

A.2.1.88.3 vert_moment_of_inertia
The vert_moment_of_inertia specifies the vert_moment_of_inertia represents the value of 2.0nd moment of the boundary formed by the intersection of the compartment and a plane representing the cargo interface (i.e. liquid cargo static waterline). The lever of the moment is parallel to the vertical axis of the ship.

A.2.1.89 Named_Unit
The Named_unit is a unit quantity associated with the work, or group of words, by which the unit is identified.
The data associated with a Named_Unit are the following:

— dimensions
The Dimensions are the exponents of the base properties by which the named_unit is defined.

A.2.1.90 non_manifold_surface_shape_representation

The Non_manifold_surface_shape_representation is a type of moulded_form_representation that is based on surface shapes and that has a topology suitable for describing inter-connected surface patches.

A.2.1.91 Non_structural_moulded_form_design_definition

A Moulded_form_design_definition is a type of Design_definition that details the moulded form information for an Item. Which Items are defined and the nature of their associated shape are inherited from Design_defintion.
The data associated with a Non_structural_moulded_form_design_definition are the following:
— usage;
— intended_location;
— boundary_definitions;
— representations

A.2.1.91.1 usage

The usage specifies the area of the ship to which the Moulded_form_design_definition applies and thus the purpose the Moulded_form has in relation to the ship.
The usage shall be one of the following:
— non_structural_bulkhead;
— user_defined;
— grating

A.2.1.91.1.1 non_structural_bulkhead

Non_structural_bulkhead: a nonprimary bulkhead used for compartmentation and not structure

A.2.1.91.1.2 user_defined

User_defined: a region on the ship nonstructural moulded form whose function is unknown or not recorded

A.2.1.91.1.3 grating

Grating: decking within a compartment which is not a part of the main structural systems but which allows access by the ships crew.

A.2.1.91.2 intended_location

The intended_location is optional and if present, specifies an intended_location specifies the location of a Moulded_form_design_definition in relation to a Spacing_position in one of the Spacing_tables (e.g., a Frame position), rather than geometrically.

A.2.1.91.3 boundary_definitions

The boundary_definitions is optional and if present, specifies a boundary_definitions specifies the Moulded_form_design_definitions that form the topological boundary of the current Moulded_form_design_definition. This may be an External instance reference to a Moulded_form_design_definition outside of the current exchange file.

A.2.1.91.4 representations

The representations specifies redefinition of the representation attribute a Non_structural_moulded_form_design_definition shall only have Compartment_shape_representations.

A.2.1.92 open_shell
The Open_shell is a shell of dimensionality 2. Its domain, if present, is a finite, connected, oriented, 2-manifold with boundary, but is not a closed surface. It can be thought of as a closed_shell with one or more holes punched in it.

A.2.1.93 Organisation
The Organisation entity is a subtype of Organization, and is referenced by Any_author.

A.2.1.94 organization
The Organization is an administrative structure.
The data associated with an organization are the following:
— id;
— name;
— description

A.2.1.94.1 id
The Id is the means by which the organization's individuality may be deduced.

A.2.1.94.2 name
The Name is the work or group of words by which the organization is referred to.

A.2.1.94.3 description
The Description is the text that relates the nature of the organization.

A.2.1.95 oriented_closed_shell
The Oriented_closed_shell is a closed_shell constructed from another closed_shell and contains a Boolean orientation flag to indicate whether or not the orientation of the constructed closed_shell agrees with the orientation of the original closed_shell. The oriented_closed_shell is equivalent to the original closed_shell but may have the opposite orientation.
The data associated with the oriented_closed_shell are the following:
— closed_shell_element;
— orientation;
— cfs_faces

A.2.1.95.1 closed_shell_element
The Closed_shell_element is the closed shell that defines the faces of the oriented_closed_shell.

A.2.1.95.2 orientation
The Orientation is the relationship between the orientation of the oriented_closed_shell being defined and the closed_shell_element referenced.

A.2.1.95.3 cfs_faces
The Cfs_faces is the set of faces for the oriented_closed_shell, obtained from those of the closed_shell_element after possibly reversing their orientation.

A.2.1.96 oriented_edge
The Oriented_edge is an edge constructed from another edge and contains a Boolean orientation flag to indicate whether or not the orientation of the constructed edge agrees with the orientation of the original edge. Except for possible re-orientation, the oriented_edge is equivalent to the original edge.
The data associated with the oriented_edge are the following:

— edge_element;
— orientation;
— edge_start;
— edge_end

A.2.1.96.1 edge_element
The Edge_element is the edge entity used to construct this oriented_edge.

A.2.1.96.2 orientation
The Orientation is a boolean, and if TRUE, the topological orientation as used coincides with the orientation, from start vertex to end vertex, of the edge_element.

A.2.1.96.3 edge_start
The Edge_start is the start vertex of the oriented edge. This is derived from the vertices of the edge_element after taking account of the orientation.

A.2.1.96.4 edge_end
The Edge_end is the end vertex of the oriented edge. This is derived from the vertices of the edge_element after taking account of the orientation.

A.2.1.97 oriented_face
The Oriented_face is a subtype of face which contains an additional orientation Boolean flag to indicate whether, or not, the sense of the oriented face agrees with its sense as originally defined in the face element.
The data associated with the oriented_face are the following:
— face_element;
— orientation;
— bounds

A.2.1.97.1 face_element
The Face_element is the face entity used to construct this oriented_face.

A.2.1.97.2 orientation
The Orientation is the relationship of the topological orientation of this entity to that of the face_element. If TRUE, the topological orientation as used coincides with the orientation of the face_element.

A.2.1.97.3 bounds
The Bounds of the oriented_face are derived from those of the face_element after taking account of the orientation that may reverse the direction of these bounds.

A.2.1.98 oriented_open_shell
The Oriented_open_shell is an open_shell constructed from another open_shell and contains a Boolean direction flag to indicate whether or not the orientation of the constructed open_shell agrees with the orientation of the original open_shell. Except for perhaps orientation, the oriented_open_shell is equivalent to the original open_shell.
The data associated with the oriented_open_shell are the following:
— open_shell_element;
— orientation;
— cfs_faces

A.2.1.98.1 open_shell_element
The Open_shell_element is the open shell that defines the faces of the oriented_open_shell.

A.2.1.98.2 orientation
The Orientation is the relationship between the orientation of the oriented_open_shell being defined and the open_shell_element referenced.

A.2.1.98.3 cfs_faces
The Cfs_faces is the set of faces for the oriented_open_shell, obtained from those of the open_shell_element after possible reversing their orientation.

A.2.1.99 oriented_path
The Oriented_path is a path constructed from another path and contains a Boolean orientation flag to indicate whether or not the orientation of the constructed path agrees with the orientation of the original path. Except for perhaps orientation, the oriented_path is equivalent to the other path.
The data associated with the oriented_path are the following:
— path_element;
— orientation;
— edge_list

A.2.1.99.1 path_element
The Path_element is the path entity used to construct this oriented_path.

A.2.1.99.2 orientation
The Orientation is Boolean. If TRUE, the topological orientation as used coincides with the orientation of the path_element.

A.2.1.99.3 edge_list
The Edge_list is the list of oriented_edges that form the oriented_path. This list is derived from the path_element after taking account of the orientation attribute.

A.2.1.100 parametric_representation_context
The Parametric_representation_context is a representation_context in which the representation_items are defined in some parametric space. The length units in this space are dimensionless.

A.2.1.101 path
The Path is a topological entity consisting of an ordered collection of oriented_edges, such that the edge_start vertex of each edge coincides with the edge_end of its predecessor. The path is ordered from the edge_start of its first oriented_edge to the edge_end of its last oriented_edge. The Boolean value orientation in the oriented edge indicates whether the edge direction agrees with the direction of the path (TRUE) or is in the opposite direction (FALSE).
The data associated with a path are the following:
— edge_list
The Edge_list is a list of oriented_edge entities which are concatenated together to form this path.

A.2.1.102 pcurve
The Pcurve is a 3D curve defined by means of a 2D curve in the parameter space of a surface. A pcurve definition contains a reference to its basis_surface and an indirect reference to a 2D curve through a definitional_representation entity.

The data associated with a pcurve are the following:
— basis_surface;
— reference_to_curve

A.2.1.102.1 basis_surface
The Basis_surface is the surface in whose parameter space the curve is defined.

A.2.1.102.2 reference_to_curve
The Reference_to_curve is the reference to the parameter space curve that defines the pcurve.

A.2.1.103 person
The Person entity is an individual human being, and is referenced by Any_author, Person_and_organization, and Person_organization_select. The data associated with a person are the following:
— id;
— last_name;
— first_name;
— middle_names;
— prefix_titles;
— suffix_titles

A.2.1.103.1 id
The Id is a means by which the person may be identified.

A.2.1.103.2 last_name
The Last_name is the person's surname.

A.2.1.103.3 first_name
The First_name is the first element of the person's list of forenames.

A.2.1.103.4 middle_names
The Middle_names are the person's other forenames, if there are any.

A.2.1.103.5 prefix_titles
The Prefix_titles are the word, or group of words, which specify the person's social and/or professional standing and appear before his or her names.

A.2.1.103.6 suffix_titles
The Suffix_titles are the word, or group of words, which specify the person's social and/or professional standing and appear after his or her names.

A.2.1.104 Person_and_organisation
The Person_and_organisation is a specific subset of person_and_organization.

A.2.1.105 person_and_organization
The Person_and_organization is a person in an organization.
The data associated with a person_and_organization are the following:
— the_person;
— the_organization

A.2.1.105.1 the_person
The_person is the person who is related to the organization.

A.2.1.105.2 the_organization
The_organization is the organization to which the person is related.

A.2.1.106 placement
The Placement locates a geometric item with respect to the coordinate system of its geometric context. It locates the item to be defined and, in the case of the axis placement subtypes, gives its orientation.
The data associated with a placement are the following:
— location
The Location is the geometric position of a reference point, such as the center of a circle, of the item to be located.

A.2.1.107 plane
The Plane is an unbounded surface with a constant normal. A plane is defined by a point on the plane and the normal direction to the plane.

A.2.1.108 Plane_Angle_Unit
The Plane_angle_unit is the unit in which angles in planes are measured.

A.2.1.109 point
The Point is a location in some real Cartesian coordinate space.

A.2.1.110 polyline
The Polyline is a bounded_curve of n-1 linear segments, defined by a list of n points.
The data associated with a polyline are the following:
— points
The Points are the list of points that define the polyline.

A.2.1.111 rational_b_spline_curve
The Rational_b_spline_curve is a piecewise parametric rational curve described in terms of control points and basis functions. This subtype is instantiated with one of the other subtypes of b_spline_curve which explicitly or implicitly provide the knot values used to define the basis functions.
The data associated with a rational_b_spline_curve are the following:
— weights_data;
— weights

A.2.1.111.1 weights_data
The Weights_data are the supplied values of the weights.

A.2.1.111.2 weights
The Weights are the array of weights associated with the control points. This is derived from the weights_data.

A.2.1.112 rational_b_spline_surface
The Rational_b_spline_surface is a piecewise parametric rational surface described in terms of control points, associated weight values and basis functions. It is instantiated with any of the other subtypes of b_spline_surface, which provide explicit or implicit knot values from which the basis functions are defined.

The data associated with a rational_b_spline_surface are the following:
— weights_data;
— weights

A.2.1.112.1 weights_data
The Weights_data is the weights associated with the control points in the rational case.

A.2.1.112.2 weights
The Weights is the array (two_dimensional) of weight values constructed from the weights_data.

A.2.1.113 representation
The Representation is a collection of one or more representation_items that are related in a specified representation_context.
The data associated with a representation are the following:
— name;
— items;
— context_of_items

A.2.1.113.1 name
The Name is an identifier of the representation.

A.2.1.113.2 items
The Items are a set of representation_items that are related in the context_of_items.

A.2.1.113.3 context_of_items
The Context_of_items is a representation_context in which the items are related to form a representation of some concept.

A.2.1.114 representation_context
The Representation_context is a context in which a collection of representation_items is related.
The data associated with a representation_context are the following:
— context_identifier;
— context_type;
— representations_in_context

A.2.1.114.1 context_identifier
The Context_identifier is an identifier of the representation_context.

A.2.1.114.2 context_type
The Context_type is a description of the type of a representation_context.

A.2.1.114.3 representations_in_context
The Representations_in_context is at least one representation shall be associated with each representation_context.

A.2.1.115 representation_item
The Representation_item is an element of product data that participates in one or more representation or contributes to the definition of another representation_item.
The data associated with a representation_item are the following:
— name
The Name is an identifier of the representation_item.

A.2.1.116 representation_relationship

The Representation_relationship is the association of two representations.
The data associated with a representation_relationship are the following:

— name;
— description;
— rep_1;
— rep_2

A.2.1.116.1 name

The Name is an identifier of the representation_relationship.

A.2.1.116.2 description

The Description is a description of the representation_relationship.

A.2.1.116.3 rep_1

The Rep_1 is the first of two representations that are related.

A.2.1.116.4 rep_2

The Rep_2 is the second of two representations that are related.

A.2.1.117 seam_curve

The Seam_curve is a curve on a closed parametric surface that has two distinct representations as constant parameter curves at the two extremes of the parameter range for the surface.

A.2.1.118 shape_representation

The Shape_representation is a specific kind of representation that represents a shape.

A.2.1.119 Ship

A Ship is the naval architectural object in concern. All data defining the product shall be related to a ship, which might exist in any life cycle stage. A project, which represents a ship in the early design phase, for example before contract, is also regarded as a ship.
The data associated with a Ship are the following:

— units

The Units specifies a reference to a set of pre-defined units for all types of measures that may appear in the ship model.

NOTE - The name of the ship is specified as a Definition in Ship_designation, where it may be versioned.

A.2.1.120 Si_Unit

The Si_unit is the fixed quantity used as a standard in terms of which items are measured as defined by ISO 1000 (clause 2). The Si_unit entity is a subtype of Named_unit, and is composed of a prefix attribute, a name attribute, and a dimensional_exponents attribute. The data associated with a Si_Unit are the following:

— prefix;
— name;
— dimensions

A.2.1.120.1 prefix

The Prefix is the name of a prefix that may be associated with an si_unit.
The prefix shall be one of the following:

— exa;
— peta;
— tera;
— giga;
— mega;
— kilo;
— hecto;
— deca;
— deci;
— centi;
— milli;
— micro;
— nano;
— pico;
— femto;
— atto

A.2.1.120.1.1 exa
Exa: See definition in ISO 1000 (clause 3).

A.2.1.120.1.2 peta
Peta: See definition in ISO 1000 (clause 3).

A.2.1.120.1.3 tera
Tera: See definition in ISO 1000 (clause 3).

A.2.1.120.1.4 giga
Giga: See definition in ISO 1000 (clause 3).

A.2.1.120.1.5 mega
Mega: See definition in ISO 1000 (clause 3).

A.2.1.120.1.6 kilo
Kilo: See definition in ISO 1000 (clause 3).

A.2.1.120.1.7 hecto
Hecto: See definition in ISO 1000 (clause 3).

A.2.1.120.1.8 deca
Deca: See definition in ISO 1000 (clause 3).

A.2.1.120.1.9 deci
Deci: See definition in ISO 1000 (clause 3).

A.2.1.120.1.10 centi
Centi: See definition in ISO 1000 (clause 3).

A.2.1.120.1.11 milli
Milli: See definition in ISO 1000 (clause 3).

A.2.1.120.1.12 micro
Micro: See definition in ISO 1000 (clause 3).

A.2.1.120.1.13 nano
Nano: See definition in ISO 1000 (clause 3).

A.2.1.120.1.14 pico
Pico: See definition in ISO 1000 (clause 3).

A.2.1.120.1.15 femto
Femto: See definition in ISO 1000 (clause 3).

A.2.1.120.1.16 atto
Atto: See definition in ISO 1000 (clause 3).

A.2.1.120.2 name
The Name is the name of a SI unit.
The name shall be one of the following:
— metre;
— gram;
— second;
— ampere;
— kelvin;
— mole;
— candela;
— radian;
— steradian;
— hertz;
— newton;
— pascal;
— joule;
— watt;
— coulomb;
— volt;
— farad;
— ohm;
— siemens;
— weber;
— tesla;
— henry;
— degree_Celsius;
— lumen;
— lux;
— becquerel;
— gray;
— sievert

A.2.1.120.2.1 metre
Metre: See ISO 1000 (subclause 2.1).

A.2.1.120.2.2 gram
Gram: See ISO 1000 (subclause 2.1).

A.2.1.120.2.3 second
Second: See ISO 1000 (subclause 2.1).

A.2.1.120.2.4 ampere
Aampere: See ISO 1000 (subclause 2.1).

A.2.1.120.2.5 kelvin
Kelvin: See ISO 1000 (subclause 2.1).

A.2.1.120.2.6 mole
Mmole: See ISO 1000 (subclause 2.1).

A.2.1.120.2.7 candela
Candela: See ISO 1000 (subclause 2.1).

A.2.1.120.2.8 radian
Radian: See ISO 1000 (subclause 2.1).

A.2.1.120.2.9 steradian
Steradian: See ISO 1000 (subclause 2.1).

A.2.1.120.2.10 hertz
Hhertz: See ISO 1000 (subclause 2.1).

A.2.1.120.2.11 newton
Newton: See ISO 1000 (subclause 2.1).

A.2.1.120.2.12 pascal
Pascal: See ISO 1000 (subclause 2.1).

A.2.1.120.2.13 joule
Joule: See ISO 1000 (subclause 2.1).

A.2.1.120.2.14 watt
Watt: See ISO 1000 (subclause 2.1).

A.2.1.120.2.15 coulomb
Coulomb: See ISO 1000 (subclause 2.1).

A.2.1.120.2.16 volt
Volt: See ISO 1000 (subclause 2.1).

A.2.1.120.2.17 farad
Farad: See ISO 1000 (subclause 2.1).

A.2.1.120.2.18 ohm
Ohm: See ISO 1000 (subclause 2.1).

A.2.1.120.2.19 siemens
Siemens: See ISO 1000 (subclause 2.1).

A.2.1.120.2.20 weber
Weber: See ISO 1000 (subclause 2.1).

A.2.1.120.2.21 tesla
Tesla: See ISO 1000 (subclause 2.1).

A.2.1.120.2.22 henry
Henry: See ISO 1000 (subclause 2.1).

A.2.1.120.2.23 degree_Celsius
Degree_Celsius: See ISO 1000 (subclause 2.1).

A.2.1.120.2.24 lumen
Lumen: See ISO 1000 (subclause 2.1).

A.2.1.120.2.25 lux
Lux: See ISO 1000 (subclause 2.1).

A.2.1.120.2.26 becquerel
Becquerel: See ISO 1000 (subclause 2.1).

A.2.1.120.2.27 gray
Gray: See ISO 1000 (subclause 2.1).

A.2.1.120.2.28 sievert
Sievert: See ISO 1000 (subclause 2.1).

A.2.1.120.3 dimensions
The Dimensions are the exponents of the base properties by which the si_unit is defined.

A.2.1.121 Simple_definition_version
A simple version to support basic versioning as a conformance class. This replaces the full Configuration Management model in the ISO schemas.
The data associated with a Simple_definition_version are the following:
— version;
— date_time

A.2.1.121.1 version
The version specifies the version number of the definition. The string is left justified and blank filled

A.2.1.121.2 date_time
The date_time specifies the date and time this version was created or versioned.

A.2.1.122 Space
A Space is a defined volume on board a ship. A Space may be either a Compartment or a Zone. A Space is an item and as such may have functional definitions, design definitions, manufacturing definitions, and product_structure_definitions relating applicable properties to the Space. Geometry defining a Space consists of references to the deck and bulkhead moulded form surfaces that bound the Space.

A.2.1.123 Space_product_structure_definition
The Space_product_structure_definition entity is a subtype of Definition, and is composed of a Relating_space_definition attribute and a Related_design_definitions attribute.
The data associated with a Space_product_structure_definition are the following:

— relating_space_definition;
— related_design_definitions.

A.2.1.123.1 relating_space_definition
The compartment_design_definition or zone_design_definition for which the space_product_structure_definition applies.

A.2.1.123.2 related_design_definitions
The Relating_space_definition are the part of design_definitions that are contained in or form the boundary of the Compartment or Zone. These may be design_definitions for part items from any of the design disciplines such as structure, machinery, piping, electrical, etc.

A.2.1.124 Spacing_grid
A spacing_grid identifies all the spacing_tables used in a particular transfer. It does not have a GUID and is not configuration managed.

A.2.1.125 Spacing_position
The unique numerical identification given to the Spacing_position
The data associated with a Spacing_position are the following:
— position_number;
— name;
— location;
— defining_table

A.2.1.125.1 position_number
The position_number is optional and if present, specifies a shipping related term that is used to label the reference point. The name need not be specified for a particular Spacing_position

A.2.1.125.2 name
The name is optional and if present, specifies the distance of the Spacing_position from the origin location in the global coordinate system.

A.2.1.125.3 location
The location specifies the distance of the Spacing_position from the base of the After Perpendicular.

A.2.1.125.4 defining_table
The defining_table is the Spacing_table to which the position belongs, i.e. the Spacing_table that contains the Spacing_position in its table_positions.

A.2.1.126 Spacing_table
A Spacing_table is a collection of Spacing_positions that define a list of reference points along one of the coordinate axes of the Ship. The function of the Spacing_table is given by the table_usage.

EXAMPLE 5 - A frame spacing table is a type of Spacing_table. The frame numbers would be specified as table_positions and the fact that the table was a frame spacing table would be given by the value for table_usage

The data associated with a Spacing_table are the following:
— defined_for;
— table_positions;
— table_usage;

— description;
— name

A.2.1.126.1 defined_for
The defined_for is the Spacing_grid for which the Spacing_Table is defined.

A.2.1.126.2 table_positions
The table_positions specifies the positions on the coordinate axis that are of interest. There may be more than one table_positions in for a particular Spacing_table

A.2.1.126.3 table_usage
The table_usage specifies the purpose of the Spacing_table within the context of the design and manufacture of a ship
The table_usage shall be one of the following:
— longitudinal_table;
— transversal_table;
— vertical_table;
— frame_table;
— buttock_table;
— station_table;
— waterline_table;
— user_defined_table

A.2.1.126.3.1 longitudinal_table
Longitudinal_table: a table whose positions lie on the longitudinal axis of the ship

A.2.1.126.3.2 transversal_table
Transversal_table: a table whose positions lie on the transverse axis of the ship

A.2.1.126.3.3 vertical_table
Vertical_table: a table whose positions lie on the vertical axis of the ship

A.2.1.126.3.4 frame_table
Frame_table: a type of longitudinal table whose positions are a reference for the location of the ships frames

A.2.1.126.3.5 buttock_table
Buttock_table: a type of transversal table whose positions are a reference for buttock lines

A.2.1.126.3.6 station_table
Station_table: a type of longitudinal table whose positions are a reference for stations

A.2.1.126.3.7 waterline_table
Waterline_table: a type of vertical table whose positions are a reference for waterlines

A.2.1.126.3.8 user_defined_table
User_defined_table: a table of whose positions provide reference points of interest to the design and manufacture of the ship

A.2.1.126.4 description
The description is optional and if present, specifies the textual account of the reason why the Spacing_table was created and any additional text that is required to describe the purpose of the Spacing_table. The description need not be specified for a Spacing_table.

A.2.1.126.5 name
The name is optional and if present, specifies the context specific identification for the Spacing_table. The name need not be specified for a Spacing_table.

A.2.1.127 spherical_surface
The Spherical_surface is a surface that is at a constant distance (the radius) from a central point. A spherical_surface is defined by the radius and the location and orientation of the surface.
The data associated with a spherical_surface are the following:
— radius
The Radius is the radius of the sphere.

A.2.1.128 surface
The Surface is a set of mathematical points that are the image of a continuous function defined over a connected subset of the plane. A surface can be envisioned as a set of connected points in 3-dimensional space that are always locally 2-dimensional, but need not be a manifold. A surface shall not be a single point or in part, or entirely, a curve.

A.2.1.129 surface_curve
The Surface_curve is a curve on a surface. The curve is represented as a curve in three-dimensional space and possibly as a curve, corresponding to a pcurve, in the two-dimensional parametric space of a surface.
The data associated with a surface_curve are the following:
— curve_3d;
— associated_geometry;
— master_representation;
— basis_surface

A.2.1.129.1 curve_3d
The Curve_3d is the curve that is the three_dimensional representation of the surface_curve.

A.2.1.129.2 associated_geometry
The Associated_geometry is a list of one or two pcurves or surfaces that define the surface or surfaces associated with the surface curve. Two elements in this list indicate that the curve has two surface associations that need not be two distinct surfaces. When a pcurve is selected, it identifies a surface and also associates a basis curve in the parameter space of this surface.

A.2.1.129.3 master_representation
The Master_representation is an indication of representation “preferred”. The master representation defines the curve used to determine the unique parameterization of the surface curve.
The master_representation shall be one of the following:
— curve_3d;
— pcurve_s1;
— pcurve_s2

A.2.1.129.3.1 curve_3d
Curve_3d is the curve in 3-dimensional space.

A.2.1.129.3.2 pcurve_s1
Pcurve_s1 is the first pcurve.

A.2.1.129.3.3 pcurve_s2
Pcurve_s2 is the second pcurve.

A.2.1.129.4 basis_surface
The Basis_surface is the surface, or surface on which the surface_curve lies. This is determined from the associated_geometry list.

A.2.1.130 Tank_compartment_design_definition
A Tank_compartment_design_definition is a type of Compartment_design_definition that provides the abstract definition of a version of a Tank compartment from a design perspective.

A.2.1.131 Tank_compartment_functional_definition
Defines the functional role of a Tank_compartment; the role may be a pre-defined one or may be user-defined.
The data associated with a Tank_compartment_functional_definition are the following:
— func.
The func specifies the name of a function that a specific Tank_compartment may have in a ship. The value of func shall be one of the following:
— liquid_cargo;
— ballast_water;
— oil_fuel;
— potable_water;
— waste;
— jet_fuel;
— user_defined

A.2.1.131.1.1 liquid_cargo
The Tank_compartment is designed to carry liquid cargo.

A.2.1.131.1.2 ballast_water
The Tank_compartment is designed to carry ballast water.

A.2.1.131.1.3 oil_fuel
The Tank_compartment is designed to carry oil or fuel.

A.2.1.131.1.4 potable_water
The Tank_compartment is designed to carry potable water.

A.2.1.131.1.5 waste
The Tank_compartment is designed to carry waste.

A.2.1.131.1.6 jet_fuel
The Tank_compartment is designed to carry jet fuel.

A.2.1.131.1.7 user_defined
The use of the Tank_compartment is defined by the user_def_function attribute.

A.2.1.132 Tank_compartment_property_set
Tank_compartment_property_set describes properties for Compartments designated for carrying fluid cargo such as oil or fuel.
The data associated with a Tank_compartment_property_set are the following:

— moments_of_inertia;
— liquid_capacity;
— design_stowage_density;
— geometric_parameters;
— design_properties

A.2.1.132.1 moments_of_inertia
The moments_of_inertia specifies the design moments of inertia for the tank.

A.2.1.132.2 liquid_capacity
The liquid_capacity specifies the volume of the tank.

A.2.1.132.3 design_stowage_density
The design_stowage_density specifies the measure of the quantity per unit volume of the liquid cargo for which the Tank_compartment is designed.

A.2.1.132.4 geometric_parameters
The geometric_parameters specifies a set of geometric area, length, and location properties that provide information about the tank. A geometric_parameters need not be specified for a particular Tank_compartment_property_set.

A.2.1.132.5 design_properties
The design_properties is optional and, if present, specifies the

A.2.1.133 Tank_geometric_parameters
Tank_geometric_parameters describes geometric properties of the Tank_compartment used for analysis of fluid cargo sloshing. Other geometric parameters are included in the Compartment_design_definition.compartment_extents.
The data associated with a Tank_geometric_parameters are the following:
— length_wash;
— breadth_wash

A.2.1.133.1 length_wash
The length_wash is optional and if present, specifies length between effective wash bulkheads at the height of the load point.

A.2.1.133.2 breadth_wash
The breadth_wash is optional and if present, specifies breadth between effective wash bulkheads at the height of the load point.

A.2.1.134 Tank_piping_design_properties
The Tank_piping_design_properties is a collection of information about the piping in a tank that impact damaged stability calculations and the design of the tank structural. The data associated with a Tank_piping_design_properties are the following:
The data associated with a Tank_piping_design_properties are the following:
— sounding_pipe_height;
— airpipe_height;
— relief_valve_pressure_setting;
— filling_height.

A.2.1.134.1 sounding_pipe_height
The sounding_pipe_height specifies height of a sounding pipe.

A.2.1.134.2 airpipe_height
The airpipe_height specifies height from the base line to the top of the air pipe, if any.

A.2.1.134.3 relief_valve_pressure_setting
The relief_valve_pressure_setting is optional and if present, specifies pressure valve opening pressure.

A.2.1.134.4 filling_height
The filling_height is optional and if present, specifies height of maximum filling of the Tank_compartment.

A.2.1.135 Tolerance
Tolerance values used during an exchange
The data associated with a Tolerance are the following:
— minimum_point_spacing.
The minimum_point_spacing specifies the minimum distance two points are distinguishable.

A.2.1.136 topological_representation_item
The Topological_representation_item represents the topology, or connectivity, of entities that make up the representation of an object. The topological_representation_item is the supertype for all the representation items in the topology schema.

A.2.1.137 Transverse_position
A Transverse_position is a type of Spacing_position that specifies a location on the y-axis of the ship coordinate system. A Transverse_position may be defined by an offset distance from a given Spacing_position.
The data associated with a Transverse_position are the following:
— offset
The offset is optional and, if present, specifies the distance from the location specified in Spacing_position

A.2.1.138 Universal_resource_locator
The address of an electronic data source (i.e. an Internet address) This is an alternative to the common mail address as provided by ISO 10303.0 -41.0
The data associated with a Universal_resource_locator are the following:
— protocol;
— other_protocol_type;
— machine_adress;
— port;
— location

A.2.1.138.1 protocol
The protocol specifies the type of the transmission protocol (i.e. ftp, http...) Because there is a permanent development on this field it is not possible to cover all available protocol types with the enumeration. Because of this a protocol type USER_DEFINED is included and the optional other_protocol_type attribute is used to hold the protocol type in this case. A WHERE rule assures that in case of protocol USER_DEFINED the other_protocol_type attribute is in use.
The protocol shall be one of the following:
— HTTP;
— FTP;

— USER_DEFINED

A.2.1.138.1.1 HTTP
HTTP is the hypertext transfer protocol

A.2.1.138.1.2 FTP
FTP is the file transfer protocol

A.2.1.138.1.3 USER_DEFINED
USER_DEFINED is the user defined transfer protocol

A.2.1.138.2 other_protocol_type
The other_protocol_type is optional and if present, specifies this is the possibility to specify a transmission protocol apart from the ones of Protocol_type.

A.2.1.138.3 machine_adress
The machine_adress specifies the name of the target machine that provides the service (i.e. the ftp server name or an Internet address).

A.2.1.138.4 port
The port is optional and if present, specifies for some protocols (i.e. http) it is possible to use another portnumber then the standard one. If so it can be specified here.

A.2.1.138.5 location
The location specifies the path on the target machine where the document is located

A.2.1.139 vector
The Vector is defined in terms of direction and magnitude.
The data associated with a vector are the following:
— orientation;
— magnitude

A.2.1.139.1 orientation
The Orientation is the direction of the vector.

A.2.1.139.2 Magnitude
The Magnitude is the magnitude of the vector. All vectors of magnitude 0.0 are regarded as equal in value regardless of the orientation attribute. Magnitudes are 0 or greater.

A.2.1.140 vertex
The Vertex is the topological construct corresponding to a point. It has dimensionality 0 and extent 0.

A.2.1.141 vertex_loop
The Vertex_loop is a loop of zero genus consisting of a single vertex. A vertex can exist independently of a vertex_loop.
The data associated with a vertex_loop are the following:
— loop_vertex
The Loop_vertex is the vertex that defines the entire loop.

A.2.1.142 vertex_point
The Vertex_point is a vertex which has its geometry defined as a point.

The data associated with a vertex_point are the following:
— vertex_geometry
The Vertex_geometry is the geometric point that defines the position in geometric space of the vertex.

A.2.1.143 Vertical_position
A Vertical_position is a type of Spacing_position that specifies a location on the z-axis of the ship coordinate system. A Vertical_position may be defined by an offset distance from a given Spacing_position.
The data associated with a Vertical_position are the following:
— offset
The offset is optional and, if present, specifies the distance from the location specified in Spacing_position

A.2.1.144 Void_compartment_design_definition
A Void_compartment_design_definition is a type of Compartment_design_definition that provides the abstract definition of a version of a Void compartment from a design perspective.

A.2.1.145 Void_compartment_functional_definition
Defines the functional role of a Void_compartment; the role may be a pre-defined one or may be user-defined.
The data associated with a Void_compartment_functional_definition are the following:
— func.
The func specifies the name of a function that a specific Void_compartment may have in a ship.
The value of func shall be one of the following:
— void;
— cofferdam;
— trunk;
— shaft_alley;
— user_defined

A.2.1.145.1.1 void
The Void_compartment is designed to be used as a void space.

A.2.1.145.1.2 cofferdam
The Void_compartment is designed to be used as a cofferdam.

A.2.1.145.1.3 trunk
The Void_compartment is designed to be used as a trunk.

A.2.1.145.1.4 shaft_alley
The Void_compartment is designed to be used as a shaft alley.

A.2.1.145.1.5 user_defined
The use of the Void_compartment is defined by the user_def_function attribute.

A.2.1.146 Volume_Unit
The Volume_unit is a unit in which the solid content of a body is measured.

A.2.1.147 Zone

A type of Space that represents an abstract bounded volume identifying a region of a ship with unique requirements or characteristics which must be specially treated in the design and or manufacturing process.

EXAMPLE 6 - Zones carry designations as Design Zone, Fire Zone, CPS Zone, Subsafe Zone, and Ship Work Authorization Boundary Zone.

A.2.1.148 Zone_design_definition

The Zone_design_definition is the abstract definition of a version of a Zone from a design perspective.

The data associated with a Zone_design_definition are the following:

— governed_by_design_requirement;
— boundaries;
— constituent_compartments;
— representations;
— defined_for

A.2.1.148.1 governed_by_design_requirement

The governed_by_design_requirement specifies reference to a description or formal specification that represents a constraint placed on the design.

A.2.1.148.2 boundaries

The boundaries is optional and if present, specifies topological reference to the Moulded_form items that bound the Zone

A.2.1.148.3 constituent_compartments

The constituent_compartments specifies the design_definitions of the compartments that are contained within the design definition of a Zone.

A.2.1.148.4 representations

The representations specifies redefinition of the representation attribute a Zone_design_definition shall only have Compartment_shape_ representations.

A.2.1.148.5 defined_for

The defined_for specifies redefinition of the defined_for attribute a Zone_design_definition is only valid for a Zone.

A.2.1.149 Zone_functional_definition

Defines the functional role of a Zone; the role may be a pre-defined one or may be user-defined.

The data associated with a Zone_functional_definition are the following:

— defined_for;
— used_for

A.2.1.149.1 defined_for

The defined_for specifies redefinition of the defined_for attribute a Zone_functional_definition is only valid for a Zone.

A.2.1.149.2 used_for

The used_for specifies the name of a function that a specific Zone may have in a ship.

The value of the used_for shall be one of the following:

— subsafe_zone;

— pressure_zone;
— fire_zone;
— design_zone;
— damage_control_zone;
— arrangement_zone;
— user_defined

A.2.1.149.2.1 subsafe_zone
The Zone is defined to be a subsafe zone.

A.2.1.149.2.2 pressure_zone
The Zone is defined to be a pressure zone.

A.2.1.149.2.3 fire_zone
The Zone is defined to be a fire zone.

A.2.1.149.2.4 design_zone
The Zone is defined to be a design zone.

A.2.1.149.2.5 damage_control_zone
The Zone is defined to be a damage control zone.

A.2.1.149.2.6 arrangement_zone
The Zone is defined to be an arrangement zone.

A.2.1.149.2.7 user_defined
The Zone function is defined by the user_def_function attribute.

A.2.2 Application Assertions

This subclause specifies the application assertions for the Ship Structures application protocol. Applica-tion assertions specify the relationships between application objects, the cardinality of the relationships, and the rules required for the integrity and validity of the application objects and UoFs. The application assertions and their definitions are given below.

A.2.2.1 Capacity_properties to cartesian_point
Each Capacity_properties has capacity_level_origin defined by exactly one cartesian_point. A cartesian_point defines the capacity_level_origin for a Capacity_properties.

A.2.2.2 Capacity_properties to cartesian_point
Each Capacity_properties has capacity_center defined by exactly one cartesian_point. A cartesian_point defines the capacity_center for a Capacity_properties.

A.2.2.3 Cargo_bay_definition to Longitudinal_position
Each Cargo_bay_definition has longitudinal_cargo_positions defined by zero up to many Longitudinal_position. A set of Longitudinal_position objects defines the longitudinal_cargo_positions for a Cargo_bay_definition.

A.2.2.4 Cargo_bay_definition to Transverse_posi tion
Each Cargo_bay_definition has transverse_cargo_positions defined by zero up to many Transverse_position. A set of Transverse_position objects defines the transverse_cargo_positions for a Cargo_bay_definition.

A.2.2.5 Cargo_bay_definition to Vertical_position

Each Cargo_bay_definition has vertical_cargo_positions defined by zero up to many Vertical_position. A set of Vertical_position objects defines the vertical_cargo_positions for a Cargo_bay_definition.

A.2.2.6 Cargo_compartment_design_definition to Cargo_bay_definition

Each Cargo_compartment_design_definition has cargo_placement_locations_within_compartment defined by zero or one Cargo_bay_definition. A Cargo_bay_definition defines the cargo_placement_locations_within_compartment for a Cargo_compartment_design_definition.

A.2.2.7 Compartment_design_definition to Compartment

Each Compartment_design_definition has defined_for defined by one up to many Compartments. A set of Compartment objects defines the defined_for for a Compartment_design_definition.

A.2.2.8 Compartment_design_definition to Compartment_extents

Each Compartment_design_definition has extents defined by zero or one Compartment_extents. A Compartment_extents defines the extents for a Compartment_design_definition.

A.2.2.9 Compartment_design_definition to Non_structural_moulded_form_design_definition to Spacing_position

Each Non_structural_moulded_form_design_definition has intended_location defined by zero or one Spacing_position. A Spacing_position defines the intended_location for a Non_structural_moulded_form_design_definition.

A.2.2.10 Compartment_design_to_functional_definition_relationship to Compartment_design_definition

Each Compartment_design_to_functional_definition_relationship has definition_1 defined by exactly one Compartment_design_definition. A Compartment_design_definition defines the definition_1 for a Compartment_design_to_functional_definition_relationship.

A.2.2.11 Compartment_design_to_functional_definition_relationship to Compartment_functional_definition

Each Compartment_design_to_functional_definition_relationship has definition_2 defined by exactly one Compartment_functional_definition. A Compartment_functional_definition defines the definition_2 for a Compartment_design_to_functional_definition_relationship.

A.2.2.12 Compartment_design_requirement

Each Compartment_design_definition has design_requirements defined by zero up to many Compartment_design_requirement. A set of Compartment_design_requirement objects defines the design_requirements for a Compartment_design_definition.

A.2.2.13 Compartment_extents to Longitudinal_position

Each Compartment_extents has forwardmost_extent defined by exactly one Longitudinal_position. A Longitudinal_position defines the forwardmost_extent for a Compartment_extents.

A.2.2.14 Compartment_extents to Longitudinal_position
Each Compartment_extents has aftmost_extent defined by exactly one Longitudinal_position. A Longitudinal_position defines the aftmost_extent for a Compartment_extents.

A.2.2.15 Compartment_extents to Transverse_position
Each Compartment_extents has portmost_extent defined by exactly one Transverse_position. A Transverse_position defines the portmost_extent for a Compartment_extents.

A.2.2.16 Compartment_extents to Transverse_position
Each Compartment_extents has starboardmost_extent defined by exactly one Transverse_position. A Transverse_position defines the starboardmost_extent for a Compartment_extents.

A.2.2.17 Compartment_extents to Vertical_position
Each Compartment_extents has topmost_extent defined by exactly one Vertical_position. A Vertical_position defines the topmost_extent for a Compartment_extents.

A.2.2.18 Compartment_extents to Vertical_position
Each Compartment_extents has bottommost_extent defined by exactly one Vertical_position. A Vertical_position defines the bottommost_extent for a Compartment_extents.

A.2.2.19 Compartment_functional_definition to Compartment
Each Compartment_functional_definition has defined_for defined by one up to many Compartments. A set of Compartment objects defines the defined_for for a Compartment_functional_definition.

A.2.2.20 Compartment_property_to_design_definition_relationship to Compartment_property_set
Each Compartment_property_to_design_definition_relationship has definition_1 defined by exactly one Compartment_property_set. A Compartment_property_set defines the definition_1 for a Compartment_property_to_design_definition_relationship.

A.2.2.21 Compartment_property_to_design_definition_relationship to Compartment_design_definition
Each Compartment_property_to_design_definition_relationship has definition_2 defined by exactly one Compartment_design_definition. A Compartment_design_definition defines the definition_2 for a Compartment_property_to_design_definition_relationship.

A.2.2.22 Cargo_compartment_property_set to Capacity_properties
Each Cargo_compartment_property_set has bulk_cargo_capacity defined by zero or one Capacity_properties. A Capacity_properties defines the bulk_cargo_capacity for a Cargo_compartment_property_set.

A.2.2.23 Compartment_property_set to Compartment
Each Compartment_property_set has defined_for defined by one up to many Compartments. A set of Compartment objects defines the defined_for for a Compartment_property_set.

A.2.2.24 Compartment_volume to cartesian_point
Each Compartment_volume has centre_of_volume defined by exactly one cartesian_point. A cartesian_point defines the centre_of_volume for a Compartment_volume.

A.2.2.25 Definable_object to Definition
Each Definable_object has definitions defined by zero up to many Definition. A set of Definition objects defines the definitions for a Definable_object.

A.2.2.26 Definition to Definable_object
Each Definition has defined_for defined by one up to many Definable_object. A set of Definable_object objects defines the defined_for for a Definition.

A.2.2.27 Definition to Global_id
Each Definition has id defined by exactly one Global_id. A Global_id defines the id for a Definition.

A.2.2.28 Definition to Simple_definition_version
Each Definition has version defined by zero or one Simple_definition_version. A Simple_definition_version defines the version for a Definition.

A.2.2.29 Definition_relationship to Definition
Each Definition_relationship has definition_1 defined by exactly one Definition. A Definition defines the definition_1 for a Definition_relationship.

A.2.2.30 Definition_relationship to Definition
Each Definition_relationship has definition_2 defined by exactly one Definition. A Definition defines the definition_2 for a Definition_relationship.

A.2.2.31 Design_definition to representation
Each Design_definition has representations defined by zero up to many representations. A set of representation objects defines the representations for a Design_definition.

A.2.2.32 Design_requirement to Document_reference
Each Design_requirement has specification defined by zero up to many Document_reference. A set of Document_reference objects defines the specification for a Design_requirement.

A.2.2.33 General_characteristics_definition to Ship
Each General_characteristics_definition has defined_for defined by one up to many Ships. A set of Ship objects defines the defined_for for a General_characteristics_definition.

A.2.2.34 General_compartment_property_set to Compartment_areas
Each General_compartment_property_set has areas defined by exactly one Compartment_areas. A Compartment_areas defines the areas for a General_compartment_property_set.

A.2.2.35 General_compartment_property_set to Compartment_permeability
Each General_compartment_property_set has permeability defined by exactly one Compartment_permeability. A Compartment_permeability defines the permeability for a General_compartment_property_set.

A.2.2.36 General_compartment_property_set to Compartment_volume

Each General_compartment_property_set has volume defined by exactly one Compartment_volume. A Compartment_volume defines the volume for a General_compartment_property_set.

A.2.2.37 External_instance_reference to Global_id

Each External_instance_reference has GUID defined by exactly one Global_id. A Global_id defines the GUID for an External_instance_reference.

A.2.2.38 Item to External_reference

Each Item has documentation defined by zero up to many External_reference. A set of External_reference objects defines the documentation for an Item.

A.2.2.39 Item to Ship

Each Item has ship_context defined by zero or one Ship. A Ship defines the ship_context for an Item.

A.2.2.40 Spacing_position to Spacing_table

Each Spacing_position has defining_table defined by exactly one Spacing_table. A Spacing_table defines the defining_table for a Spacing_position.

A.2.2.41 Spacing_table to Spacing_grid

Each Spacing_table has defined_for defined by one up to many Spacing_grid. A set of Spacing_grid objects defines the defined_for for a Spacing_table.

A.2.2.42 Spacing_table to Spacing_position

Each Spacing_table has table_positions defined by one up to many Spacing_position. A list of Spacing_position objects defines the table_positions for a Spacing_table.

A.2.2.43 Tank_compartment_property_set to Capacity_properties

Each Tank_compartment_property_set has liquid_capacity defined by zero up to many Capacity_properties. A set of Capacity_properties objects defines the liquid_capacity for a Tank_compartment_property_set.

A.2.2.44 Tank_compartment_property_set to Moments_of_inertia

Each Tank_compartment_property_set has moments_of_inertia defined by exactly one Moments_of_inertia. A Moments_of_inertia defines the moments_of_inertia for a Tank_compartment_property_set.

A.2.2.45 Tank_compartment_property_set to Tank_geometric_parameters

Each Tank_compartment_property_set has geometric_parameters defined by zero or one Tank_geometric_parameters. A Tank_geometric_parameters defines the geometric_parameters for a Tank_compartment_property_set.

A.2.2.46 Tank_compartment_property_set to Tank_piping_design_properties

Each Tank_compartment_property_set has design_properties defined by zero or one Tank_piping_design_properties. A Tank_piping_design_properties defines the design_properties for a Tank_compartment_property_set.

A.2.2.47 Zone_design_definition to Compartment_design_definition

Each Zone_design_definition has constituent_compartments defined by zero up to many Compartment_design_definition. A set of Compartment_design_definition objects defines the constituent_compartments for a Zone_design_definition.

A.2.2.48 Zone_design_definition to Compartment_design_requirement

Each Zone_design_definition has governed_by_design_requirement defined by exactly one Compartment_design_requirement. A Compartment_design_requirement defines the governed_by_design_requirement for a Zone_design_definition.

A.2.2.49 Zone_design_definition to Zone

Each Zone_design_definition has defined_for defined by one up to many Zone. A set of Zone objects defines the defined_for for a Zone_design_definition.

A.2.2.50 Zone_functional_definition to Zone

Each Zone_functional_definition has defined_for defined by one up to many Zone. A set of Zone objects defines the defined_for for a Zone_functional_definition.

Supporting Entities

A.2.2.51 Date_and_time to date

Each date_and_time has date_component defined by exactly one date. A date defines the date_component for a date_and_time.

A.2.2.52 Date_and_time to local_time

Each date_and_time has time_component defined by exactly one local_time. A local_time defines the time_component for a date_and_time.

A.2.2.53 Local_time to coordinated_universal_time_offset

Each local_time has zone defined by exactly one coordinated_universal_time_offset. A coordinated_universal_time_offset defines the zone for a local_time.

A.2.2.54 Person_and_organization to person

Each person_and_organization has the_person defined by exactly one person. A person defines the the_person for a person_and_organization.

A.2.2.55 Person_and_organization to organization

Each person_and_organization has the_organization defined by exactly one organization. An organization defines the the_organization for a person_and_organization.

A.2.2.56 Simple_definition_version to date_and_time

Each Simple_definition_version has date_time defined by exactly one date_and_time. A date_and_time defines the date_time for a Simple_definition_version.

Units

A.2.2.57 Derived_Unit to Derived_Unit_Element

Each Derived_Unit has elements defined by one up to many Derived_Unit_Element. A set of Derived_Unit_Element objects defines the elements for a Derived_Unit.

A.2.2.58 Derived_Unit_Element to Named_Unit

Each Derived_Unit_Element has unit defined by exactly one Named_Unit. A Named_Unit defines the unit for a Derived_Unit_Element.

A.2.2.59 Extended_conversion_based_unit to Extended_measure_with_unit

Each Extended_conversion_based_unit has conversion_factor defined by exactly one Extended_measure_with_unit. An Extended_measure_with_unit defines the conversion_factor for an Extended_conversion_based_unit.

A.2.2.60 Named_Unit to dimensional_exponents

Each Named_Unit has dimensions defined by exactly one dimensional_exponents. A dimensional_exponents defines the dimensions for a Named_Unit.

A.2.2.61 Si_Unit to dimensional_exponents

Each Si_Unit has dimensions defined by exactly one dimensional_exponents. A dimensional_exponents defines the dimensions for a Si_Unit.

Geometry

A.2.2.62 axis2_placement_2d to direction

Each axis2_placement_2d has ref_direction defined by zero or one direction. A direction defines the ref_direction for an axis2_placement_2d.

A.2.2.63 axis2_placement_2d to direction

Each axis2_placement_2d has p defined by two up to many directions. A list of direction objects defines the p for an axis2_placement_2d.

A.2.2.64 axis2_placement_3d to direction

Each axis2_placement_3d has axis defined by zero or one direction. A direction defines the axis for an axis2_placement_3d.

A.2.2.65 axis2_placement_3d to direction

Each axis2_placement_3d has ref_direction defined by zero or one direction. A direction defines the ref_direction for an axis2_placement_3d.

A.2.2.66 axis2_placement_3d to direction
Each axis2_placement_3d has p defined by three up to many directions. A list of direction objects defines the p for an axis2_placement_3d.

A.2.2.67 b_spline_curve to cartesian_point
Each b_spline_curve has control_points_list defined by two up to many cartesian_point. A list of cartesian_point objects defines the control_points_list for a b_spline_curve.

A.2.2.68 composite_curve to composite_curve_segment
Each composite_curve has segments defined by one up to many composite_curve_segment. A list of composite_curve_segment objects defines the segments for a composite_curve.

A.2.2.69 composite_curve_segment to curve
Each composite_curve_segment has parent_curve defined by exactly one curve. A curve defines the parent_curve for a composite_curve_segment.

A.2.2.70 connected_face_set to face
Each connected_face_set has cfs_faces defined by one up to many face. A set of face objects defines the cfs_faces for a connected_face_set.

A.2.2.71 edge to vertex
Each edge has edge_start defined by exactly one vertex. A vertex defines the edge_start for an edge.

A.2.2.72 edge to vertex
Each edge has edge_end defined by exactly one vertex. A vertex defines the edge_end for an edge.

A.2.2.73 face_based_surface_model to connected_face_set
Each face_based_surface_model has fbsm_faces defined by one up to many connected_face_set. A set of connected_face_set objects defines the fbsm_faces for a face_based_surface_model.

A.2.2.74 face to face_bound
Each face has bounds defined by one up to many face_bound. A set of face_bound objects defines the bounds for a face.

A.2.2.75 oriented_edge to edge
Each oriented_edge has edge_element defined by exactly one edge. An edge defines the edge_element for an oriented_edge.

A.2.2.76 oriented_edge to vertex
Each oriented_edge has edge_start defined by exactly one vertex. A vertex defines the edge_start for an oriented_edge.

A.2.2.77 oriented_edge to vertex
Each oriented_edge has edge_end defined by exactly one vertex. A vertex defines the edge_end for an oriented_edge.

A.2.2.78 oriented_face to face
Each oriented_face has face_element defined by exactly one face. A face defines the face_element for an oriented_face.

A.2.2.79 oriented_face to face_bound
Each oriented_face has bounds defined by one up to many face_bound. A set of face_bound objects defines the bounds for an oriented_face.

A.2.2.80 composite_curve_on_surface to surface
Each composite_curve_on_surface has basis_surface defined by two up to many surface. A set of surface objects defines the basis_surface for a composite_curve_on_surface.

A.2.2.81 line to cartesian_point
Each line has pnt defined by exactly one cartesian_point. A cartesian_point defines the pnt for a line.

A.2.2.82 line to vector
Each line has dir defined by exactly one vector. A vector defines the dir for a line.

A.2.2.83 pcurve to surface
Each pcurve has basis_surface defined by exactly one surface. A surface defines the basis_surface for a pcurve.

A.2.2.84 pcurve to definitional_representation
Each pcurve has reference_to_curve defined by exactly one definitional_representation. A definitional_representation defines the reference_to_curve for a pcurve.

A.2.2.85 polyline to cartesian_point
Each polyline has points defined by two up to many cartesian_point. A list of cartesian_point objects defines the points for a polyline.

A.2.2.86 surface_curve to curve
Each surface_curve has curve_3d defined by exactly one curve. A curve defines the curve_3d for a surface_curve.

A.2.2.87 surface_curve to surface
Each surface_curve has basis_surface defined by two up to many surfaces. A set of surface objects defines the basis_surface for a surface_curve.

A.2.2.88 vector to direction
Each vector has orientation defined by exactly one direction. A direction defines the orientation for a vector.

A.2.2.89 edge_curve to curve
Each edge_curve has edge_geometry defined by exactly one curve. A curve defines the edge_geometry for an edge_curve.

A.2.2.90 elementary_surface to axis2_placement_3d
Each elementary_surface has position defined by exactly one axis2_placement_3d. An axis2_placement_3d defines the position for an elementary_surface.

A.2.2.91 face_bound to loop
Each face_bound has bound defined by exactly one loop. A loop defines the bound for a face_bound.

A.2.2.92 face_surface to surface
Each face_surface has face_geometry defined by exactly one surface. A surface defines the face_geometry for a face_surface.

A.2.2.93 oriented_open_shell to face
Each oriented_open_shell has cfs_faces defined by one up to many faces. A set of face objects defines the cfs_faces for an oriented_open_shell.

A.2.2.94 oriented_closed_shell to closed_shell
Each oriented_closed_shell has closed_shell_element defined by exactly one closed_shell. A closed_shell defines the closed_shell_element for an oriented_closed_shell.

A.2.2.95 oriented_open_shell to open_shell
Each oriented_open_shell has open_shell_element defined by exactly one open_shell. An open_shell defines the open_shell_element for an oriented_open_shell.

A.2.2.96 oriented_closed_shell to face
Each oriented_closed_shell has cfs_faces defined by one up to many faces. A set of face objects defines the cfs_faces for an oriented_closed_shell.

A.2.2.97 oriented_path to path
Each oriented_path has path_element defined by exactly one path. A path defines the path_element for an oriented_path.

A.2.2.98 placement to cartesian_point
Each placement has location defined by exactly one cartesian_point. A cartesian_point defines the location for a placement.

A.2.2.99 representation to representation_item
Each representation has items defined by one up to many representation_item. A set of representation_item objects defines the items for a representation.

A.2.2.100 representation to representation_context
Each representation has context_of_items defined by exactly one representation_context. A representation_context defines the context_of_items for a representation.

A.2.2.101 representation_relationship to representation
Each representation_relationship has rep_1 defined by exactly one representation. A representation defines the rep_1 for a representation_relationship.

A.2.2.102 representation_relationship to representa tion

Each representation_relationship has rep_2 defined by exactly one representation. A representation defines the rep_2 for a representation_relationship.

A.2.2.103 representation_context to representation

Each representation_context has representations_in_context defined by one up to many representations. A set of representation objects defines the representations_in_context for a representation_context.

A.2.2.104 vertex_loop to vertex

Each vertex_loop has loop_vertex defined by exactly one vertex. A vertex defines the loop_vertex for a vertex_loop.

A.2.2.105 vertex_point to point

Each vertex_point has vertex_geometry defined by exactly one point. A point defines the vertex_geometry for a vertex_point.

A.3 Application Reference Model (ARM)

This section provides the application reference model for the NSRP 0429 Ship Structure schema and is given in figures A-1 through A-41. The application reference model is a graphical representation of the structure and con-straints of the application objects specified in section A-2-1. The graphical form of the application reference model is presented in the EXPRESS-G format. The application reference model is independent from any implementation method.

NSRP Entities

STEP Entity	Figure Number
Shipbuilding Entities	
capacity_properties	A-10, A-18, A-31
cargo_bay_definition	A-2, A-6, A-8
cargo_compartment_design_definition	A-8, A-9
cargo_compartment_functional_definition	A-7, A-9
cargo_compartment_property_set	A-9, A-10, A-31
compartment	A-1, A-3, A-7, A-8, A-10
compartment_areas	A-10, A-28
compartment_design_definition	A-3, A-4, A-6, A-8, A-9, A-11, A-27, A-32
compartment_design_requirement	A-4, A-8, A-11
compartment_design_to_functional_definition_relationship	A-8, A-9
compartment_extents	A-2, A-8, A-32
compartment_functional_definition	A-3, A-7, A-9
compartment_naval_administrative_property_set	A-9, A-10
compartment_permeability	A-10
compartment_property_to_design_definition_relationship	A-8, A-9, A-10
compartment_volume	A-10, A-18
date_and_time	A-6, A-35, A-40
definition_relationship	A-6, A-9
external_instance_reference	A-39
general_compartment_property_set	A-9, A-10, A-28
global_axis_placement	A-1
global_id	A-6, A-39
habitable_compartment_design_definition	v8, A-9
habitable_compartment_functional_definition	A-7, A-9
habitable_compartment_property_set	A-9, A-10
longitudinal_position	
machinery_compartment_design_definition	A-2, A-32
machinery_compartment_functional_definition	A-7, A-9
moments_of_inertia	A-9, A-10
non_structural_moulded_form_design_definition	A-5, A-6, A-11
ship	A-1
simple_definition_version	A-6, A-35
space_product_structure_definition	A-6
spacing_grid	A-1, A-2
spacing_position	A-2, A-5
spacing_table	A-2, A-6
tank_compartment_design_definition	A-8, A-9
tank_compartment_functional_definition	A-7, A-9

tank_compartment_property_set	A-9, A-10, A-29, A-30, A-31
tank_geometric_parameters	A-10
tank_piping_design_properties	A-10, A-30
tolerance	A-40
transverse_position	A-2, A-32
vertical_position	A-2, A-32
void_compartment_design_definition	A-8, A-9
void_compartment_functional_definition	A-7, A-9
zone	A-1, A-4, A-7, A-27
zone_design_definition	A-4, A-6, A-11, A-27
zone_functional_definition	A-4, A-7
Unit Entities	
area_unit	A-34, A-40
derived_unit	A-40
derived_unit_element	v40
dimensional_exponents	A-33, A-34
extended_conversion_based_unit	A-34, A-40
extended_measure_with_unit	A-34
length_unit	A-34, A-40
plane_angle_unit	A-34, A-40
si_unit	A-33, A-34, A-40
Support Entities	
address	A-38
calendar_date	A-35
coordinated_universal_time_offset	A-35
date	A-35
document	A-26
document_reference	A-11, A-26, A-27
external_reference	A-1, A-26
local_time	A-35
organisation	A-36
person	A-38
person_and_organisation	A-36
universal_resource_locator	A-37
Geometry Entities	
advanced_face	A-19
axis2_placement_2d	A-24, A-25
axis2_placement_3d	A-23, A-24, A-25
bounded_curve	A-20, A-22
bounded_pcurve	A-20, A-22
bounded_surface_curve	A-20, A-22
b_spline_curve_with_knots	A-22
b_spline_surface_with_knots	A-23
cartesian_point	A-18, A-20, A-22, A-23, A-25, A-31
circle	A-20
closed_shell	A-16
composite_curve	A-20, A-21, A-22
composite_curve_on_surface	A-22
composite_curve_segment	A-20, A-21
definitional_representation	A-12

direction	A-14, A-24, A-25
edge_curve	A-14, A-15, A-19, A-20
edge_loop	A-16, A-17
ellipse	A-20
face_based_surface_model	A-14, A-19
face_bound	A-14, A-16
face_outer_bound	A-16
face_surface	A-14, A-16, A-19
geometric_representation_context	A-41
intersection_curve	A-20
line	A-18, A-20
loop	A-14, A-16, A-17
non_manifold_surface_shape_representation	A-12
open_shell	A-16
oriented_closed_shell	A-16
oriented_edge	A-15, A-17
oriented_face	A-16
oriented_open_shell	A-16
oriented_path	A-17
path	A-14, A-17
pcurve	A-20
plane	A-23, A-24
polyline	A-18, A-20, A-22
seam_curve	A-20
spherical_surface	A-23, A-2
surface_curve	A-20
vector	A-14, A-20, A-24
vertex	A-14, A-15, A-16
vertex_loop	A-16
vertex_point	A-14, A-16, A-18, A-19
compartment_property_set	A-3, A-6, A-9, A-10
definable_object	A-1
definition	A-1, A-6, A-9
design_definition	A-6, A-8, A-11
design_requirement	A-6, A-11, A-26, A-27
functional_definition	A-6, A-7
general_characteristics_definition	A-1, A-6
item	A-1, A-26
space	A-1
area_unit	
b_spline_curve	A-18, A-20, A-22
b_spline_surface	A-18, A-23
bounded_curve	A-20, A-22
bounded_surface	A-23
conic	A-20
connected_face_set	A-14, A-16, A-19
curve	A-14, A-19, A-20
edge	A-14, A-15
elementary_surface	A-23, A-24, A-25
face	A-14, A-16

geometric_representation_item	A-14
named_unit	A-33, A-34
organization	A-36
parametric_representation_context	A-41
person_and_organization	A-36
placement	A-14, A-18, A-25
point	A-14, A-16, A-18, A-19
rational_b_spline_curve	A-22
representation	A-6, A-11, A-12, A-13, A-41
representation_context	A-12, A-41
representation_item	A-12, A-14
representation_relationship	A-13
shape_representation	A-12
surface	A-14, A-19, A-20, A-23
topological_representation_item	A-14
volume_unit	A-34

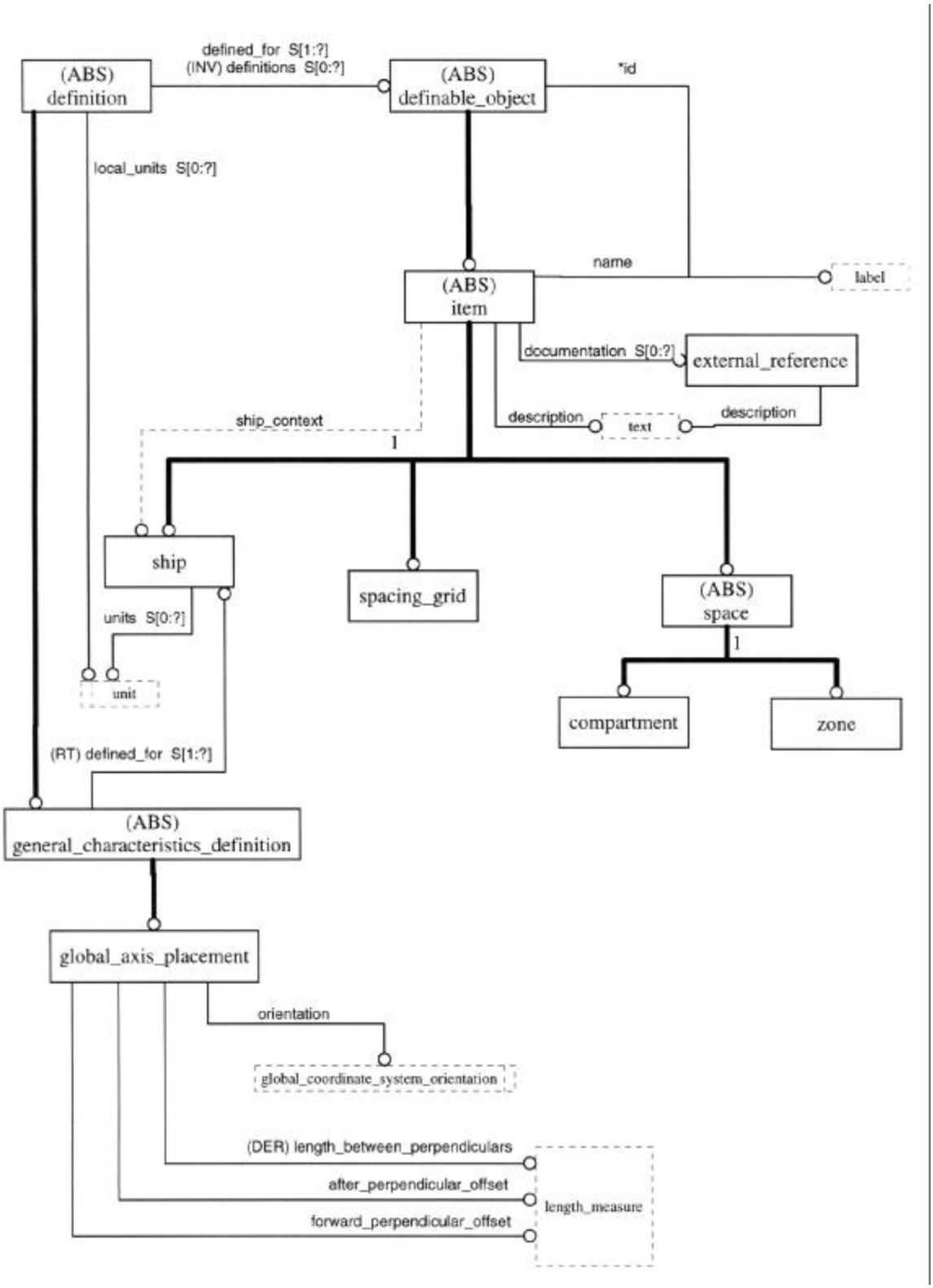


Figure A-1 (ABS) Item

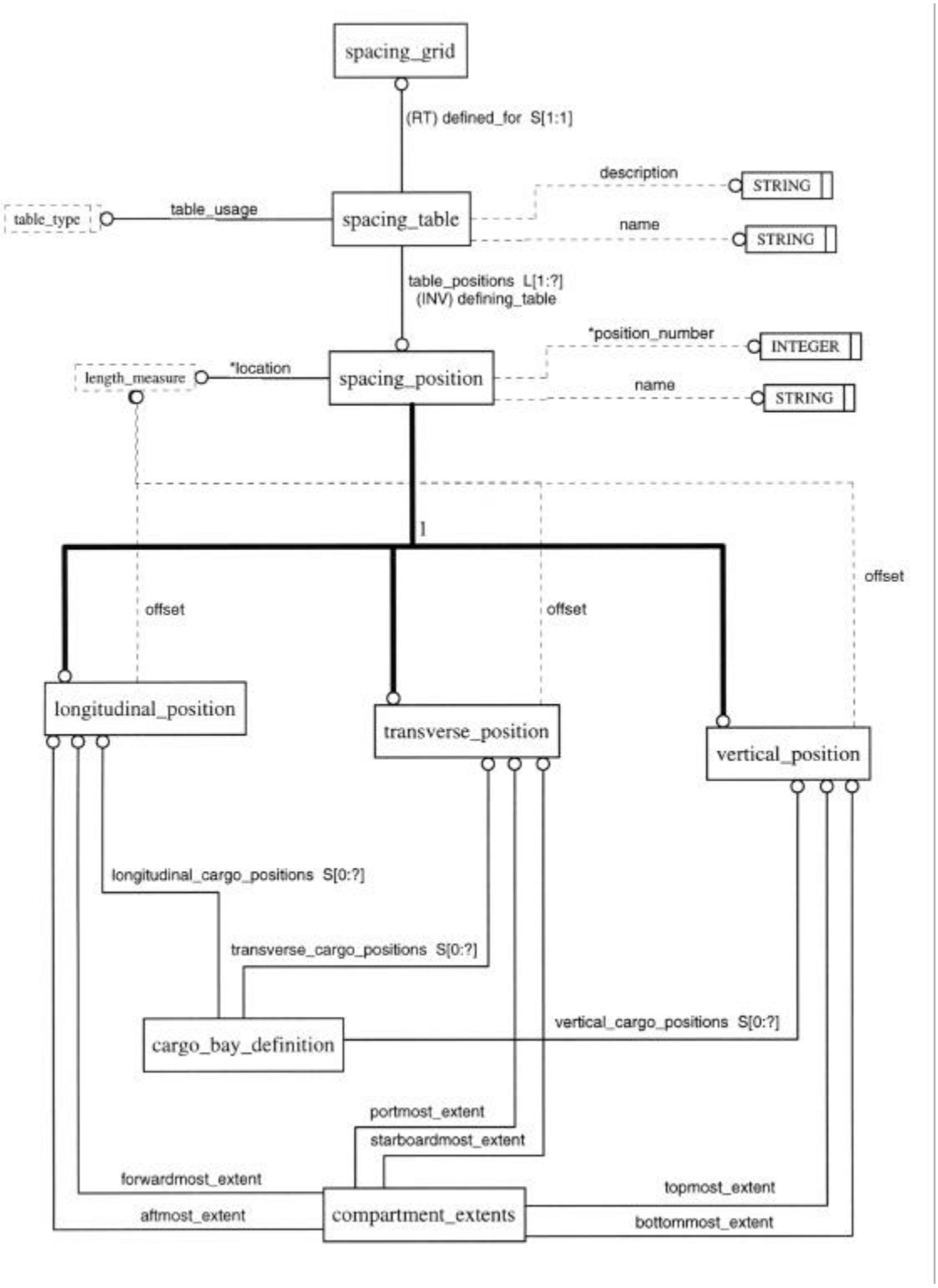


Figure A-2 Spacing_Table

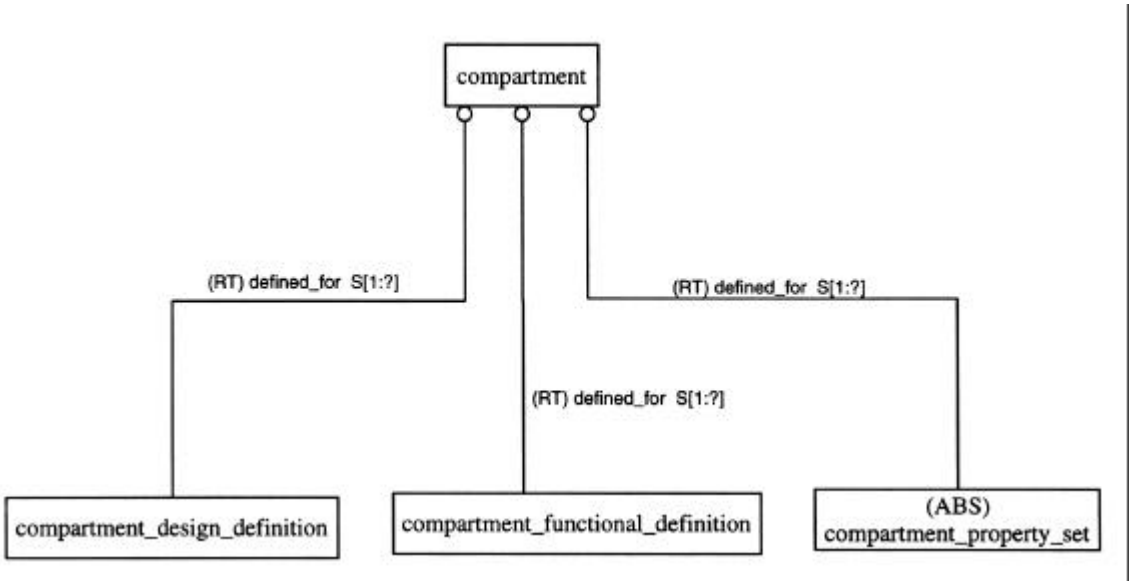


Figure A-3 Compartment

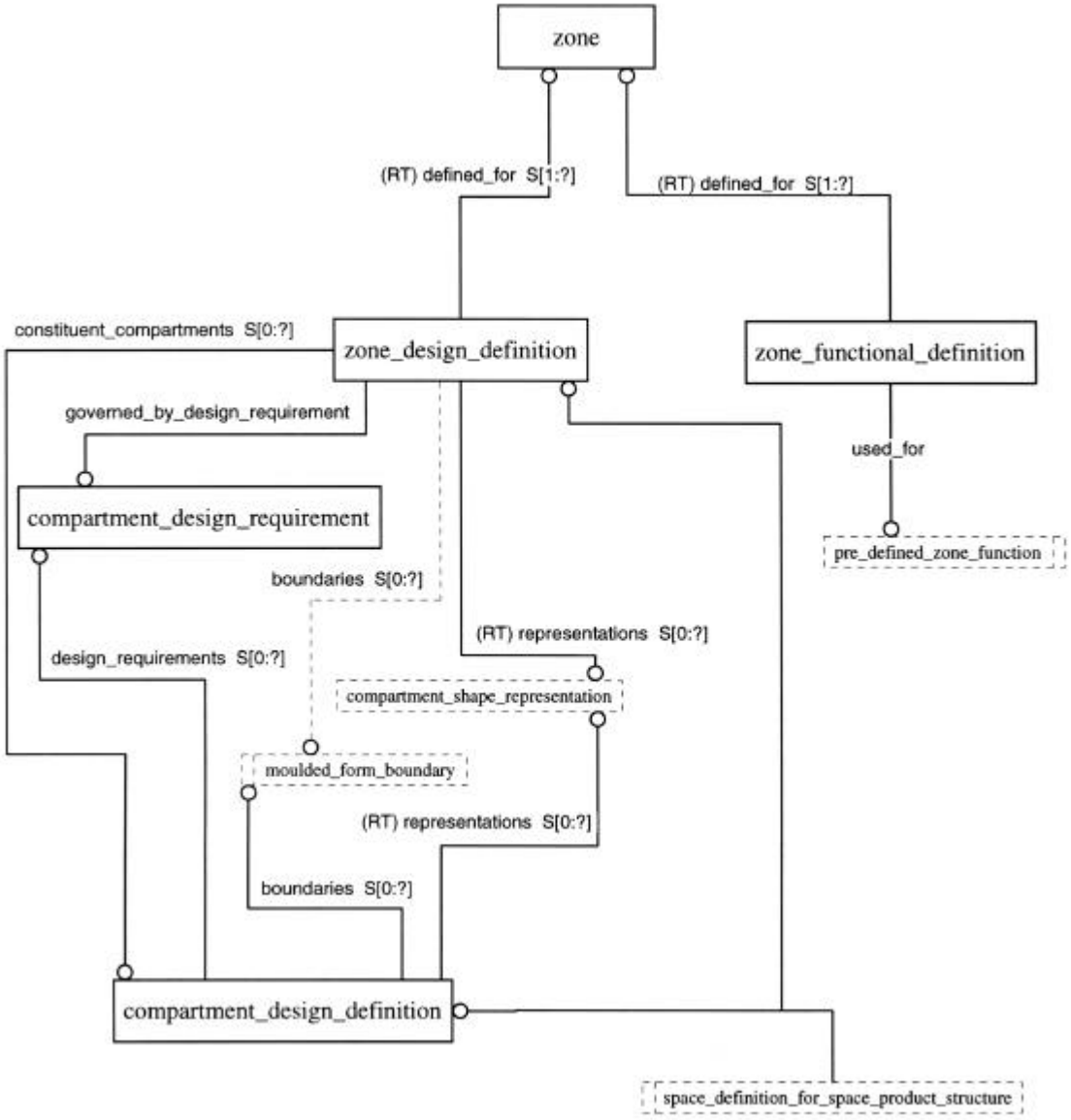


Figure A-4 Zone

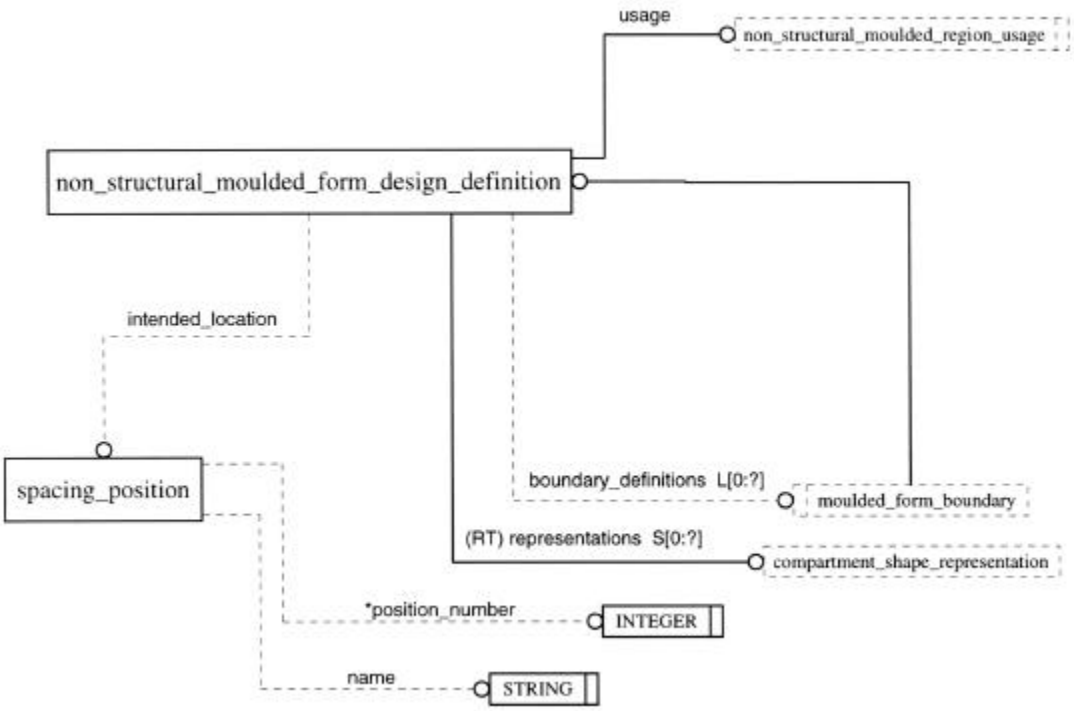


Figure A-5 Non Structural Molded Form Design Definition

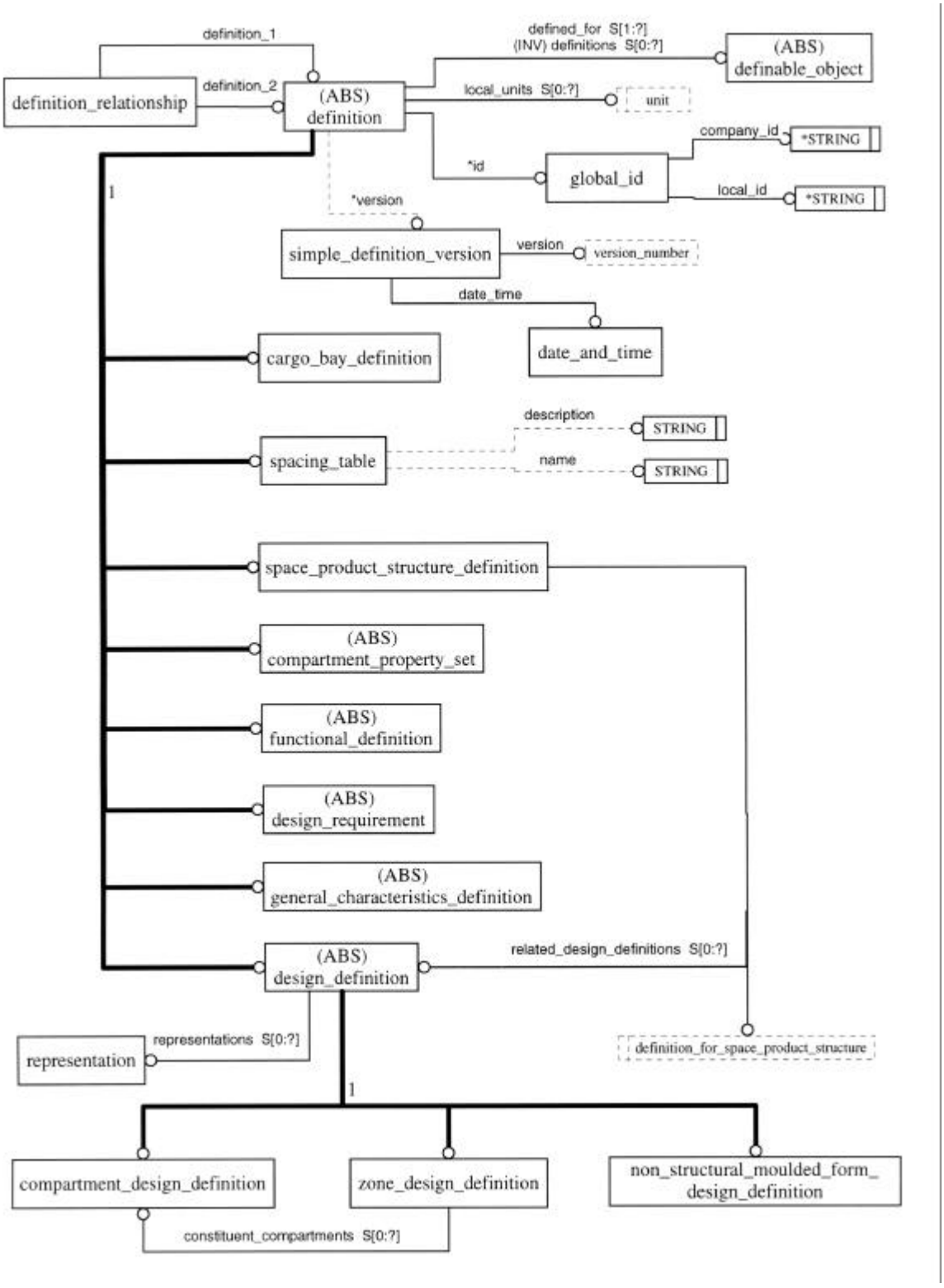


Figure A-6 (ABS) Definition

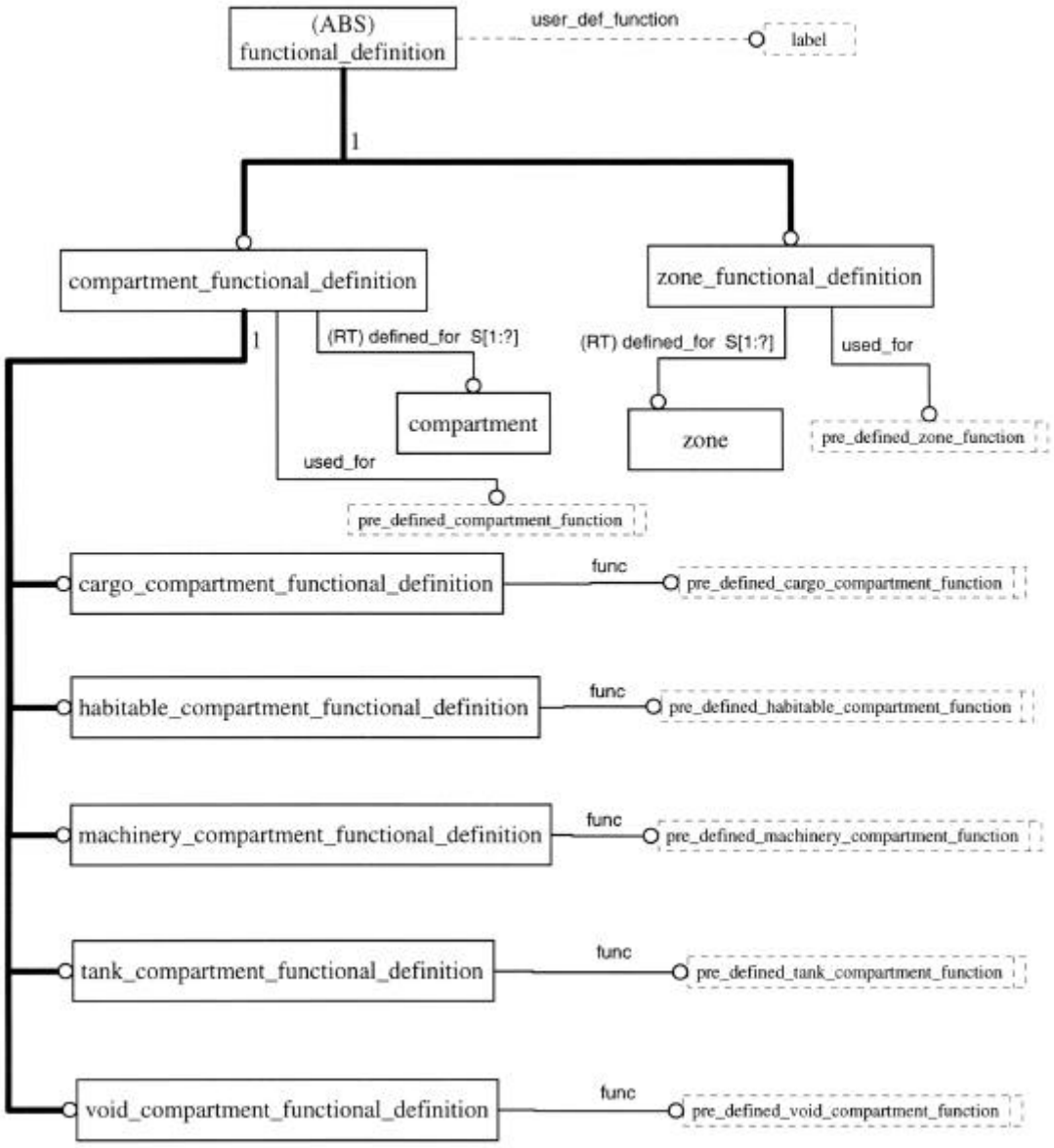


Figure A-7 (ABS) Functional_Definition

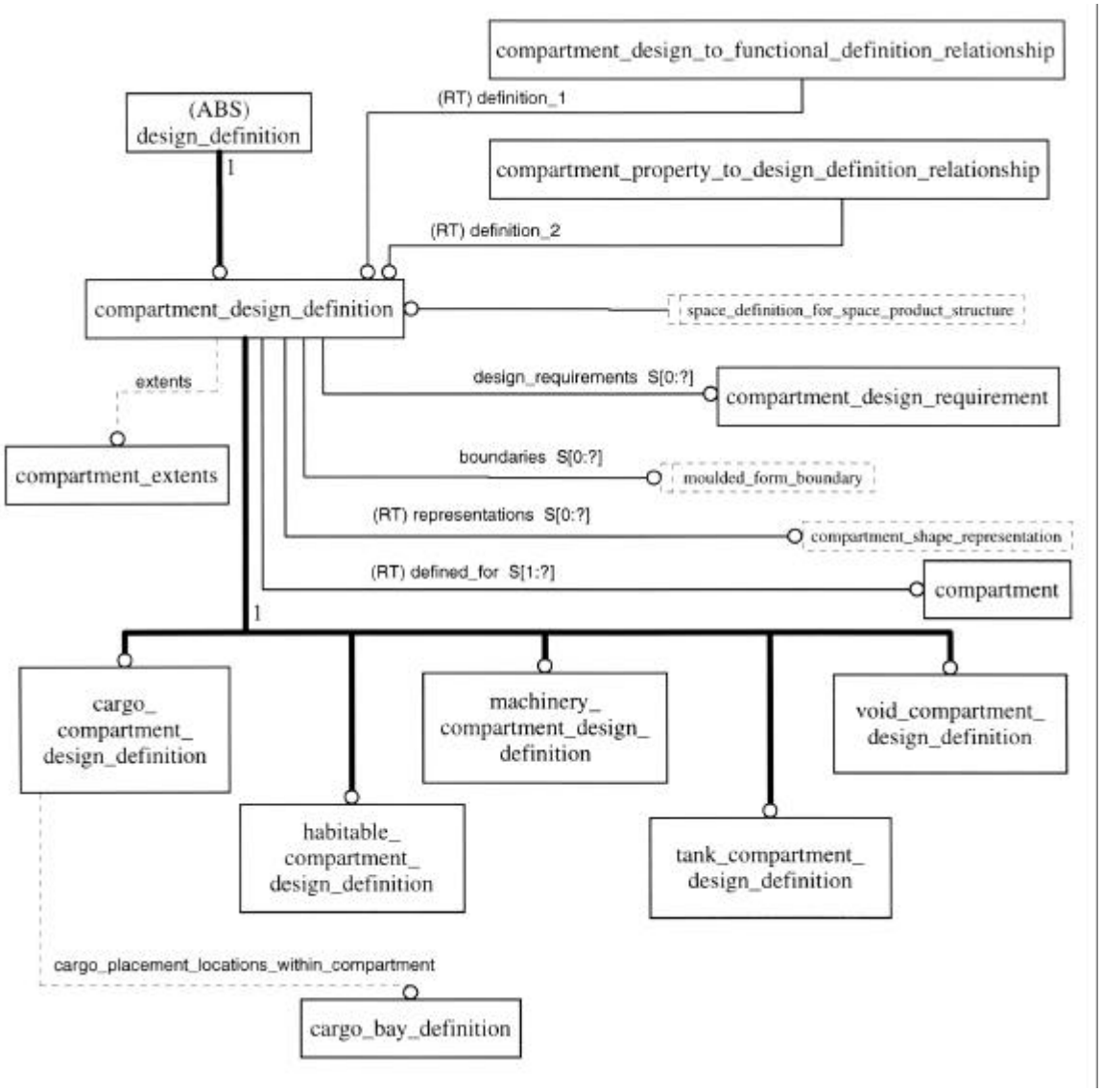


Figure A-8 (ABS) Design_Definition

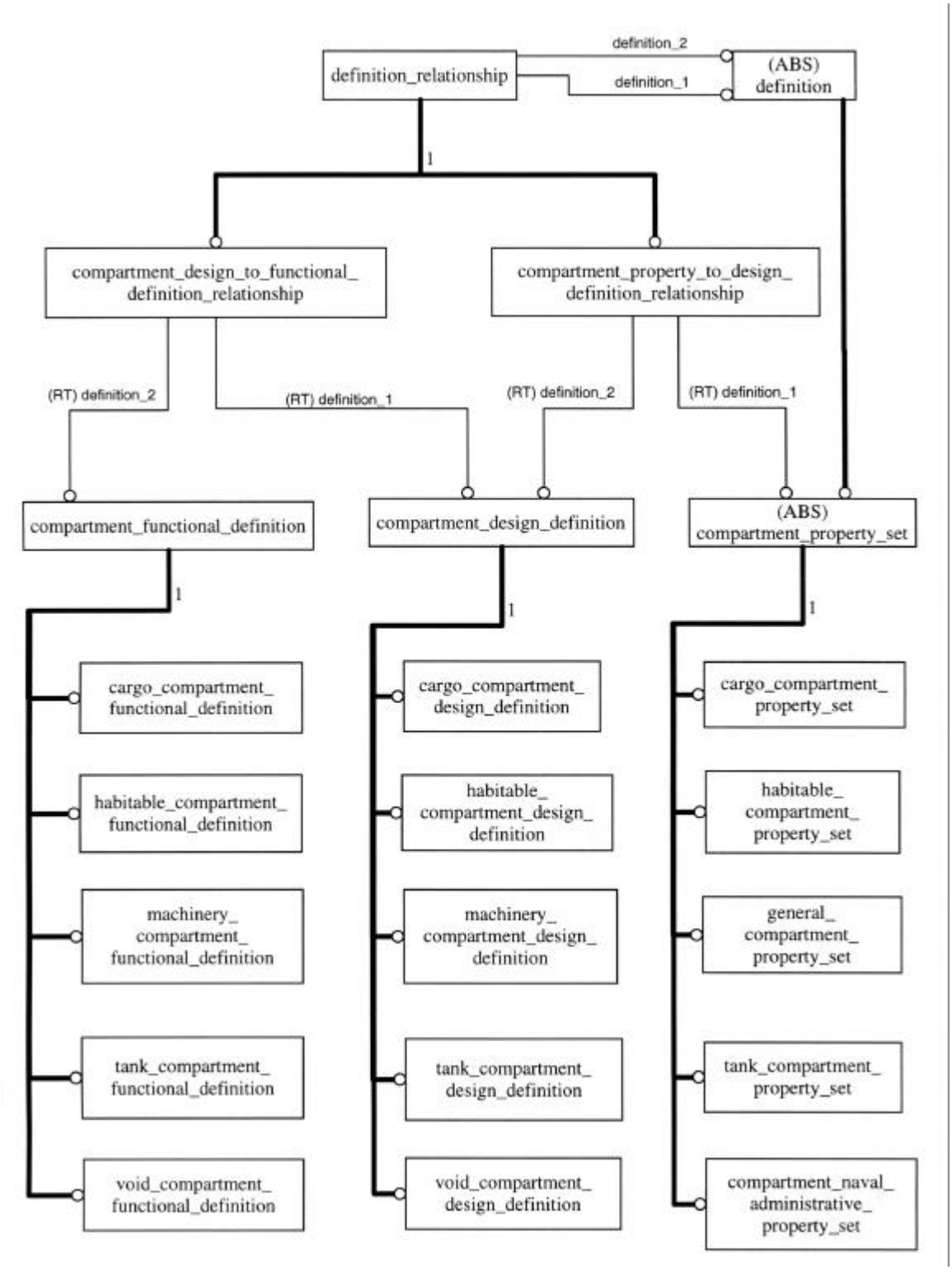


Figure A-9 Definition_Relationship

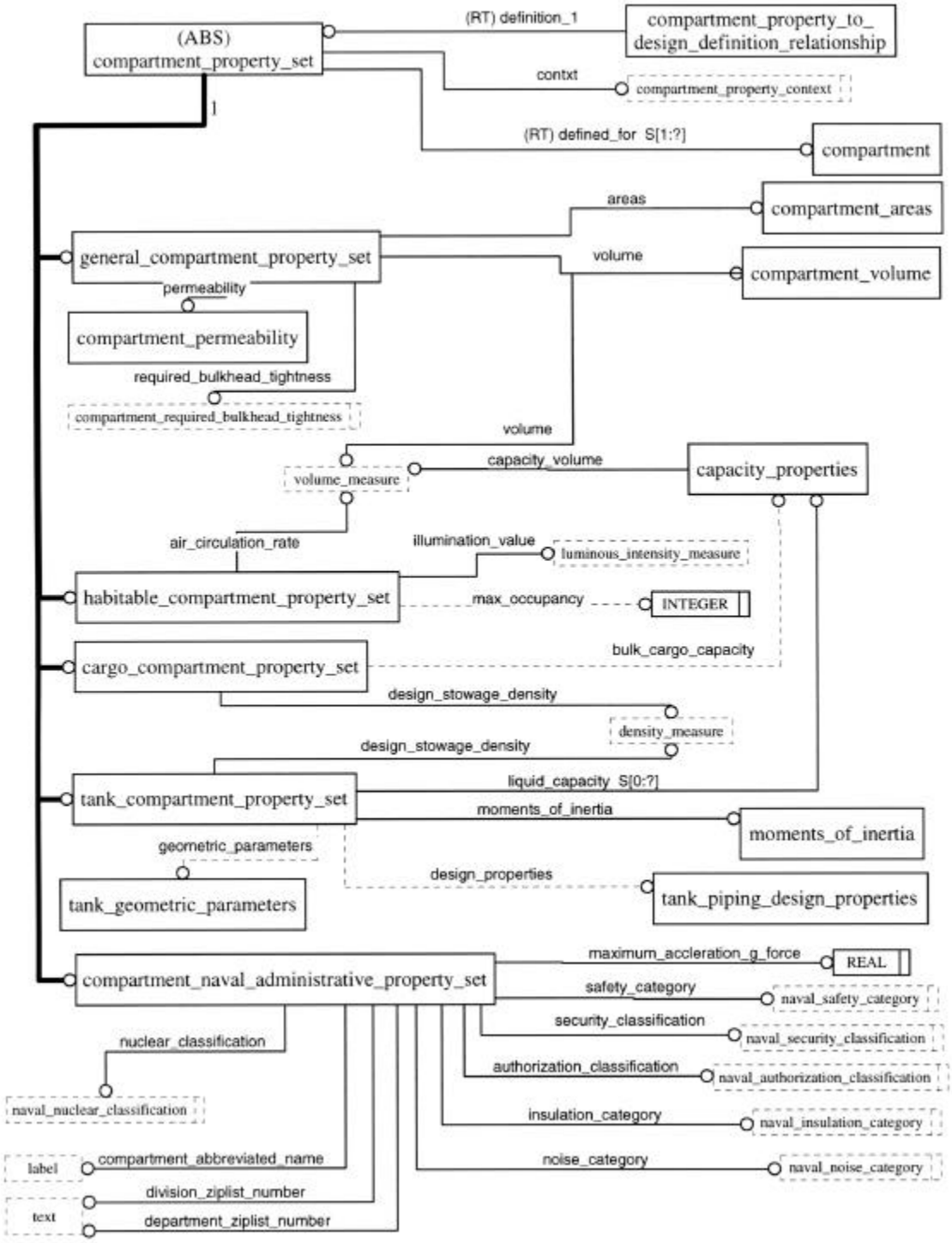


Figure A-10 (ABS) Compartment_Property_Set

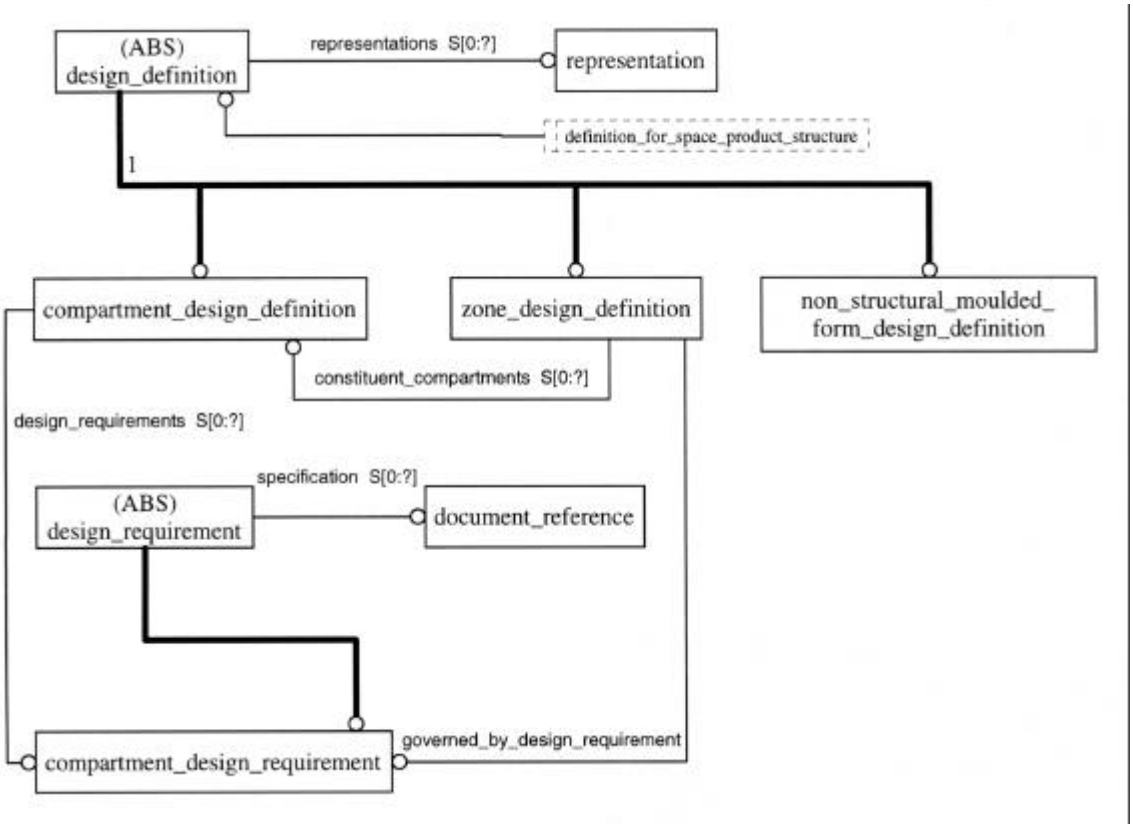


Figure A-11 (ABS) Design_Definition

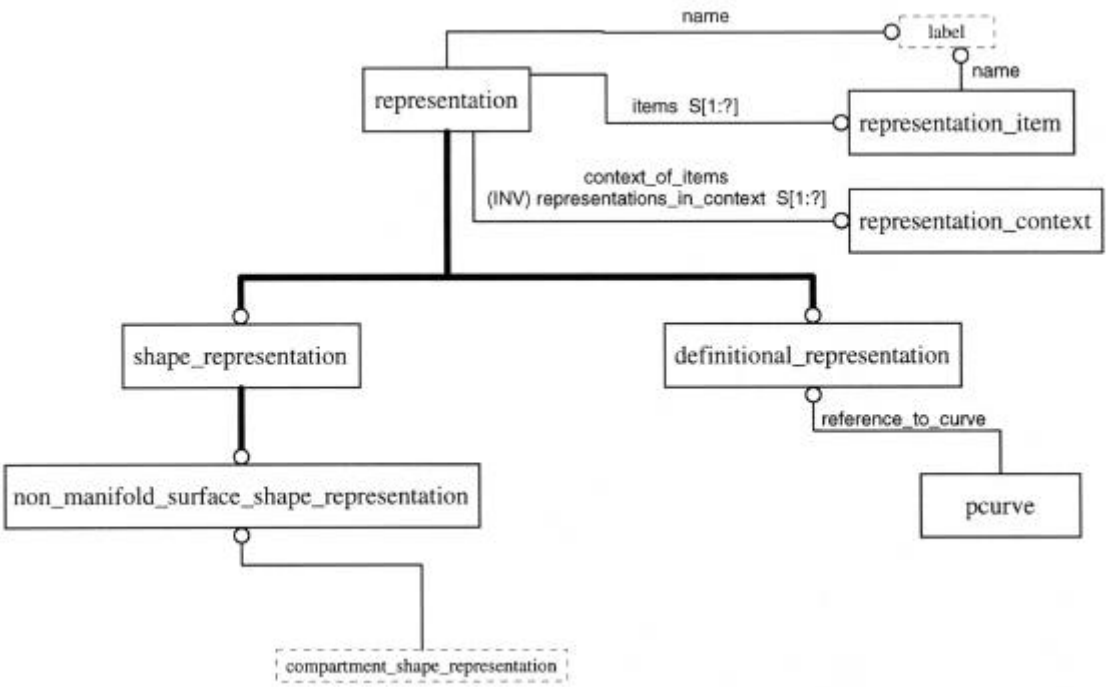


Figure A-12 Representation

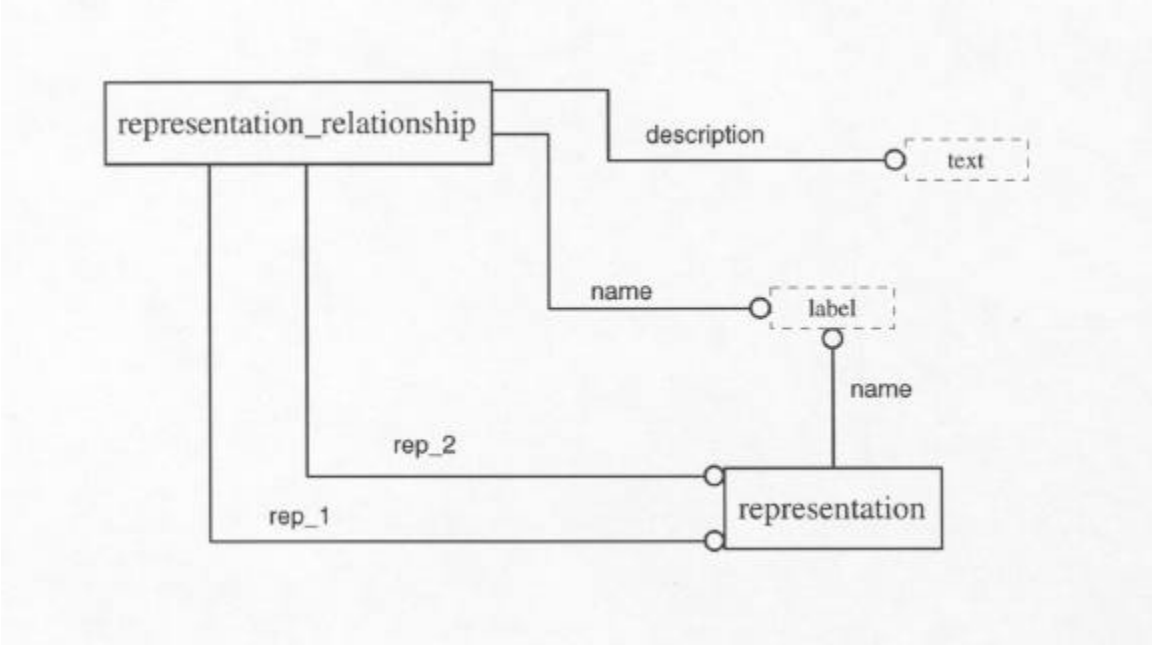


Figure A-13 Representation_Relationship

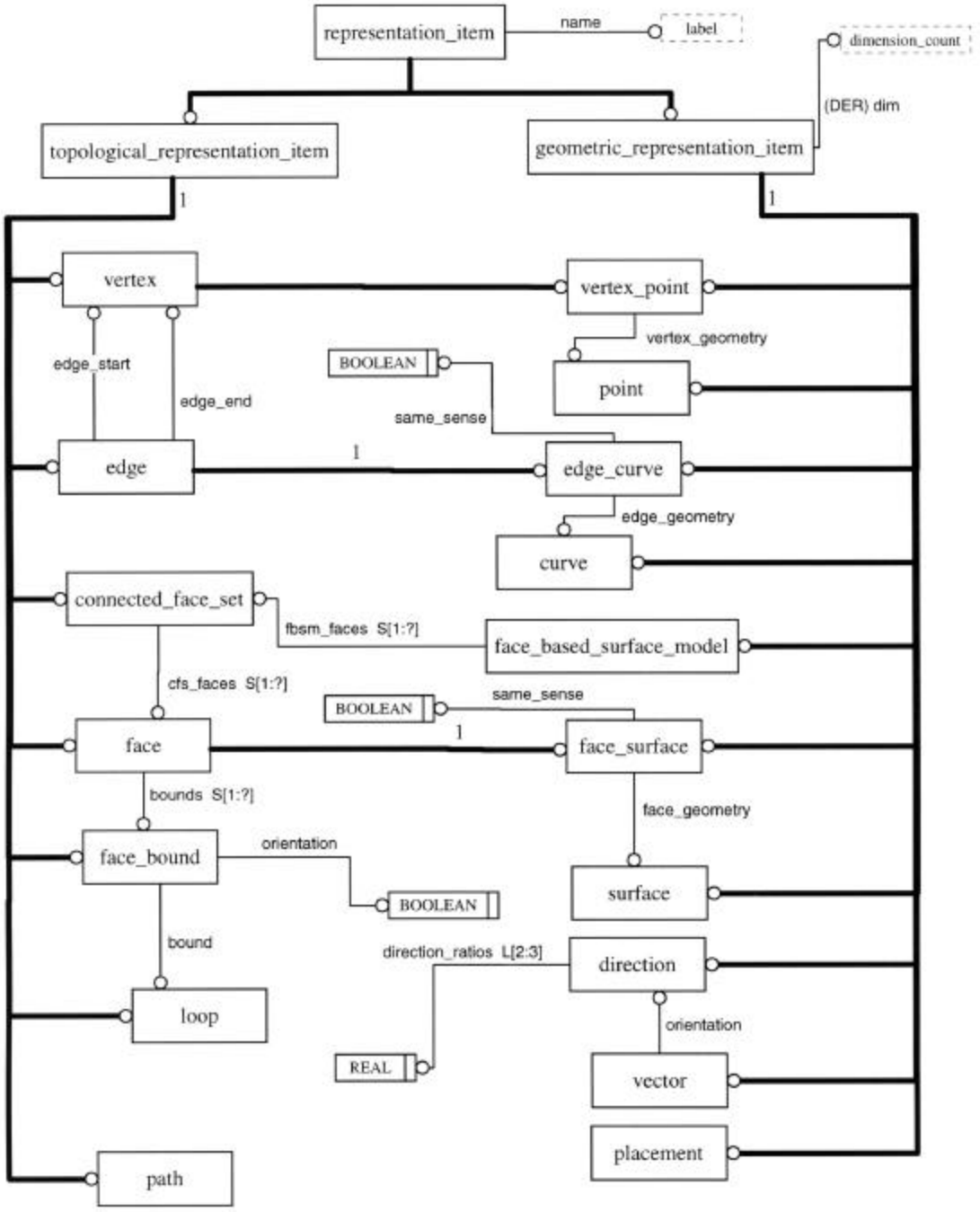


Figure A-14 Representation_Item

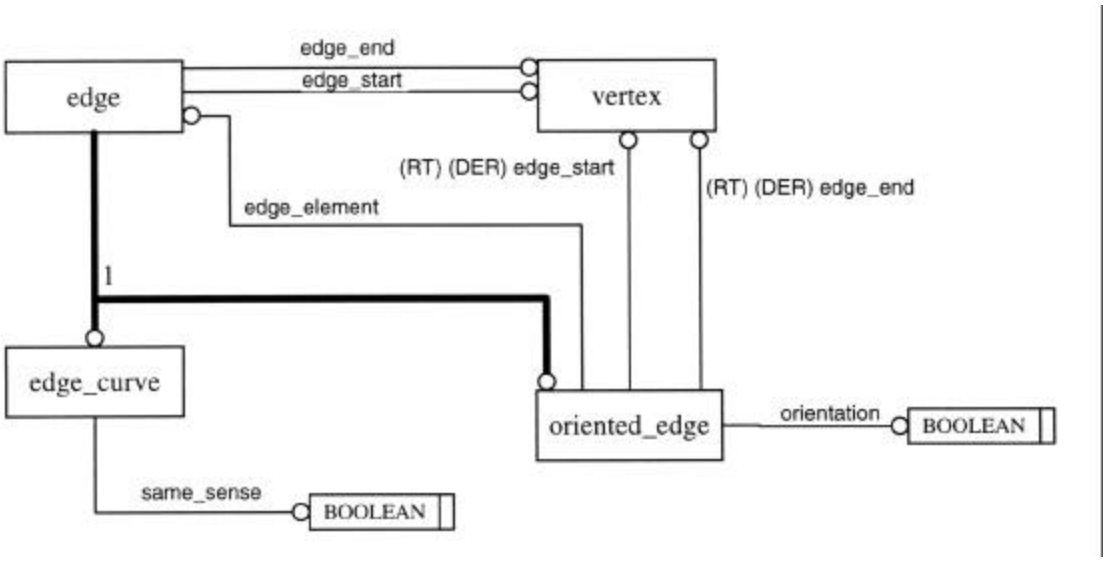


Figure A-15 Edge

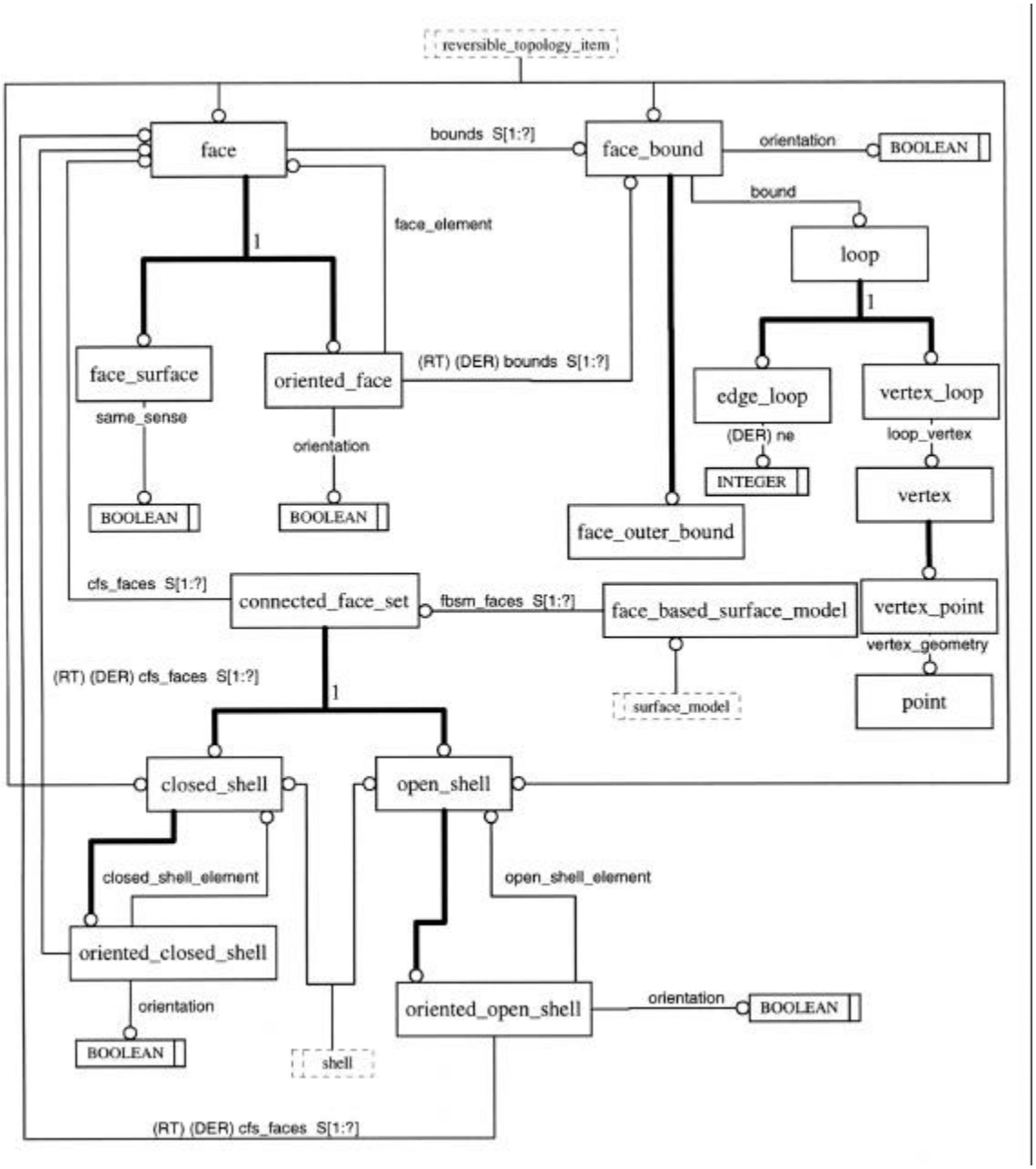


Figure A-16 Face

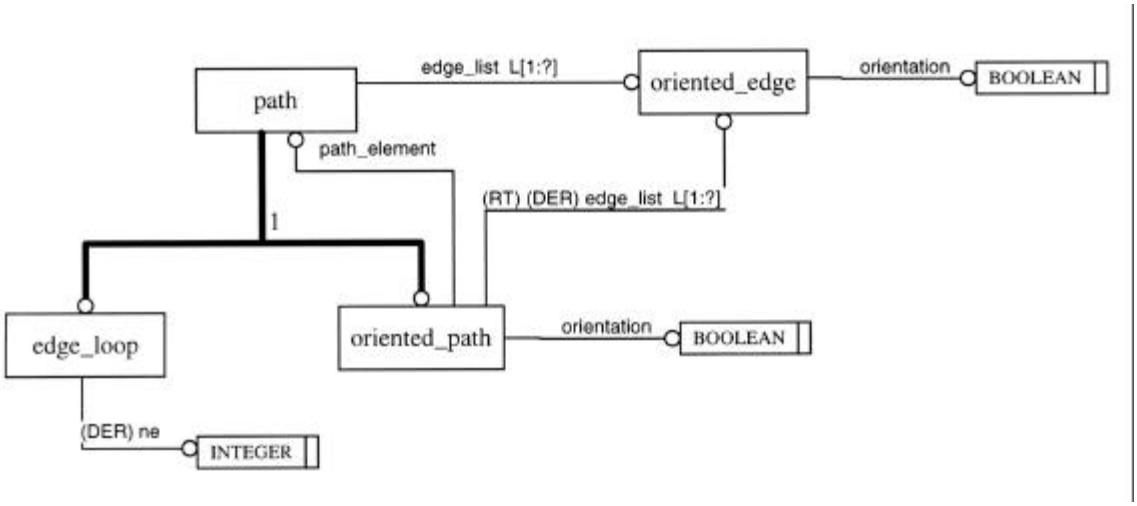


Figure A-17 Path

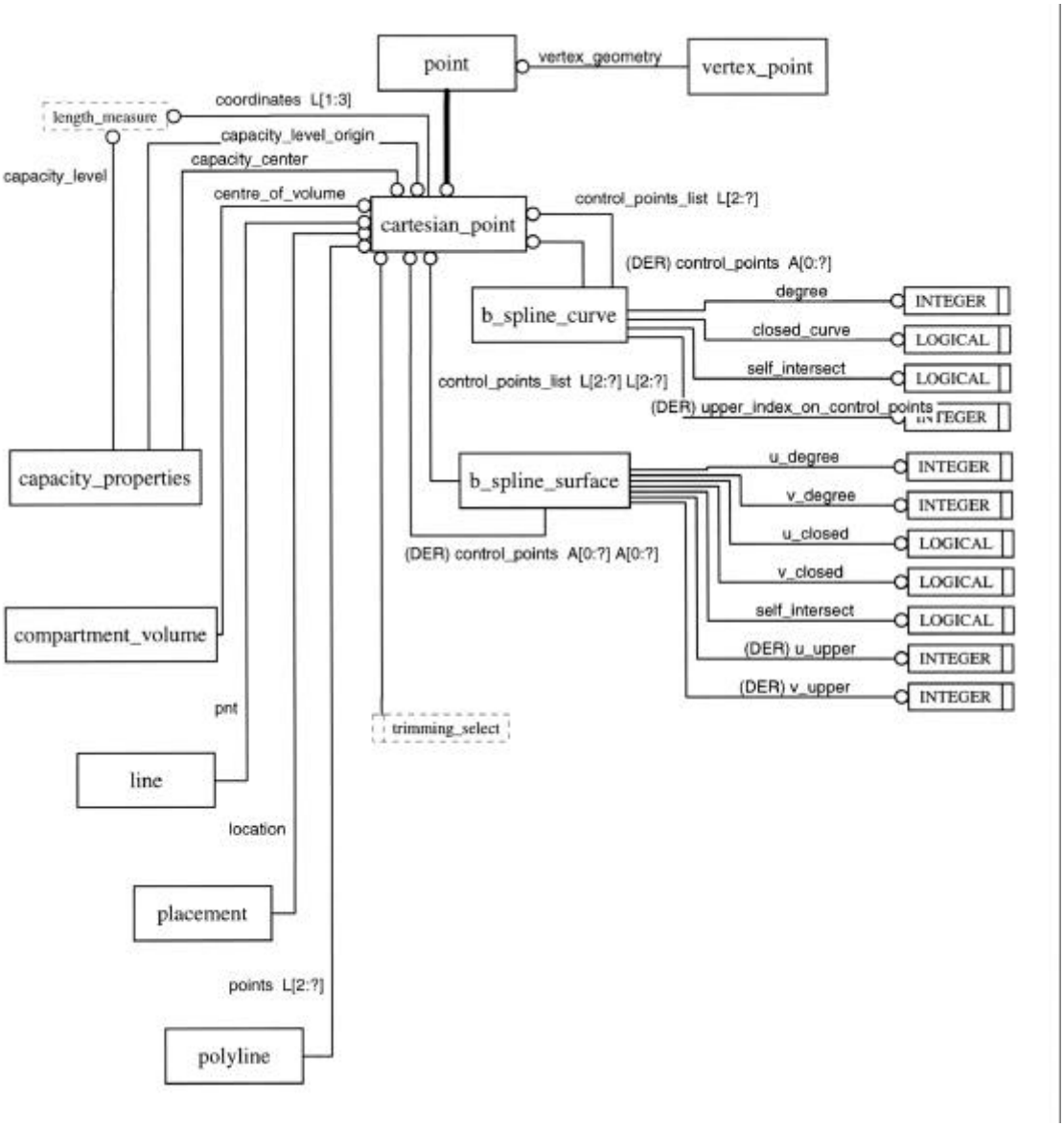


Figure A-18 Point

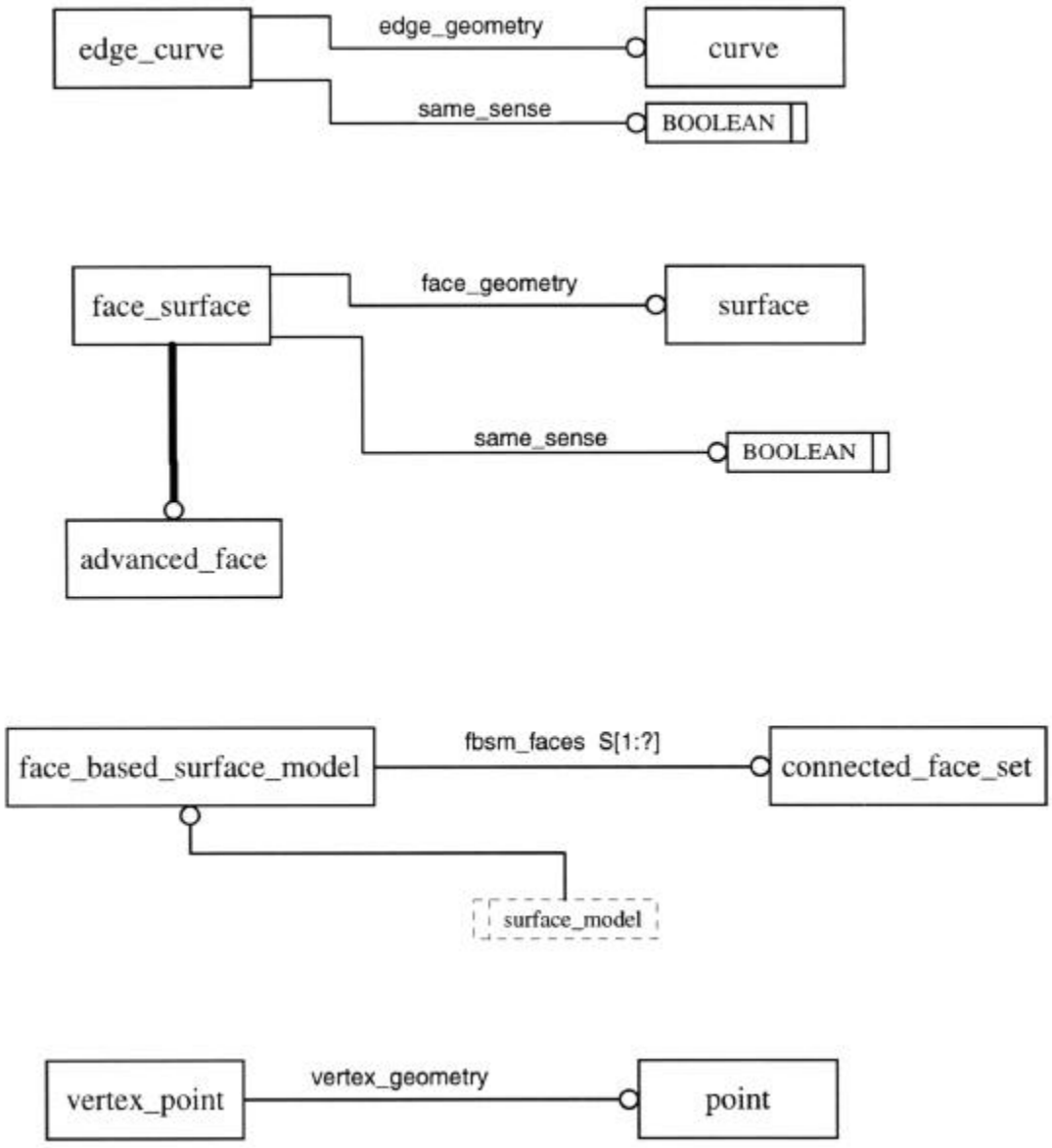


Figure A-19 Miscllaneous Geometry

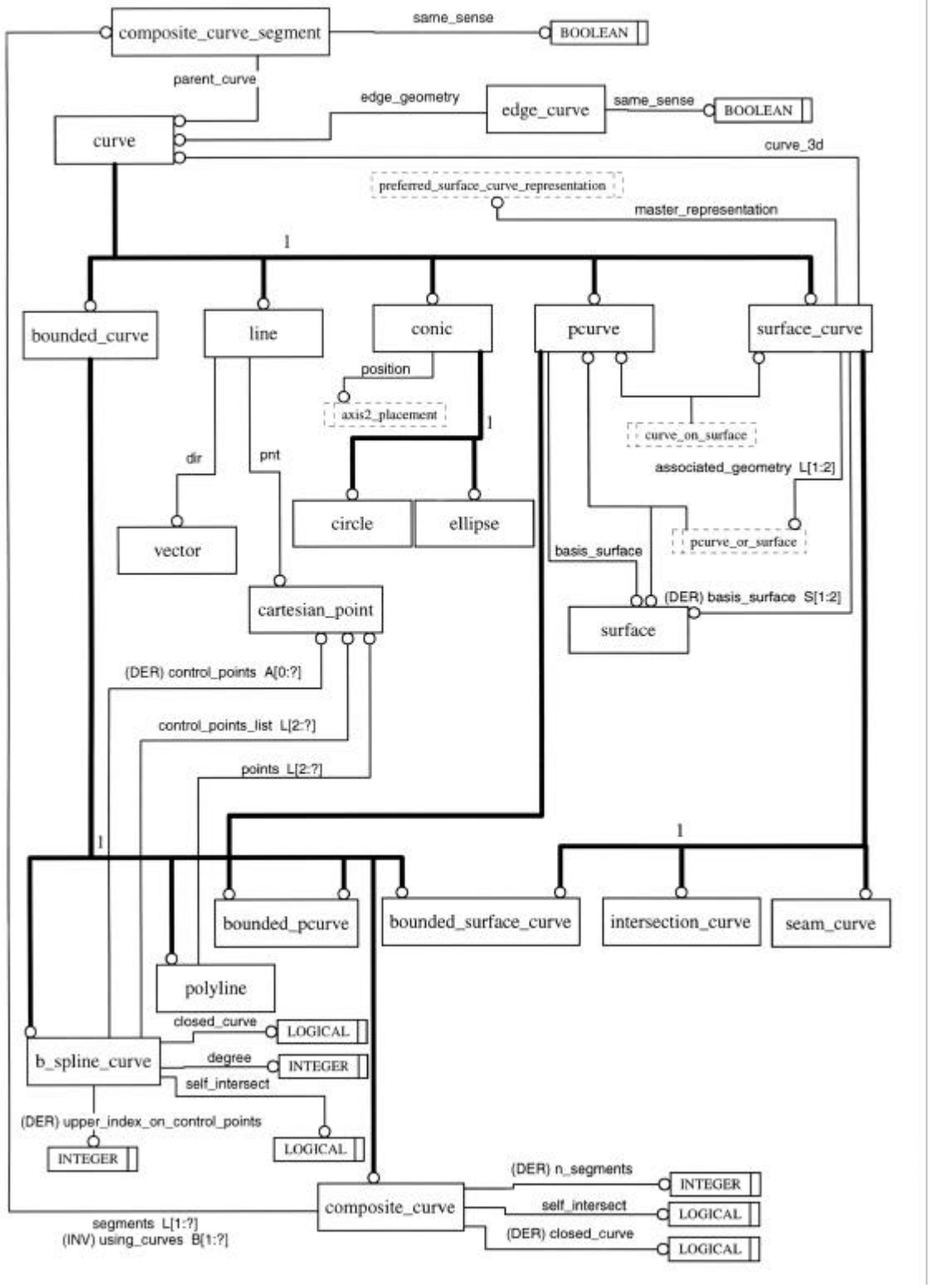


Figure A-20 Curve

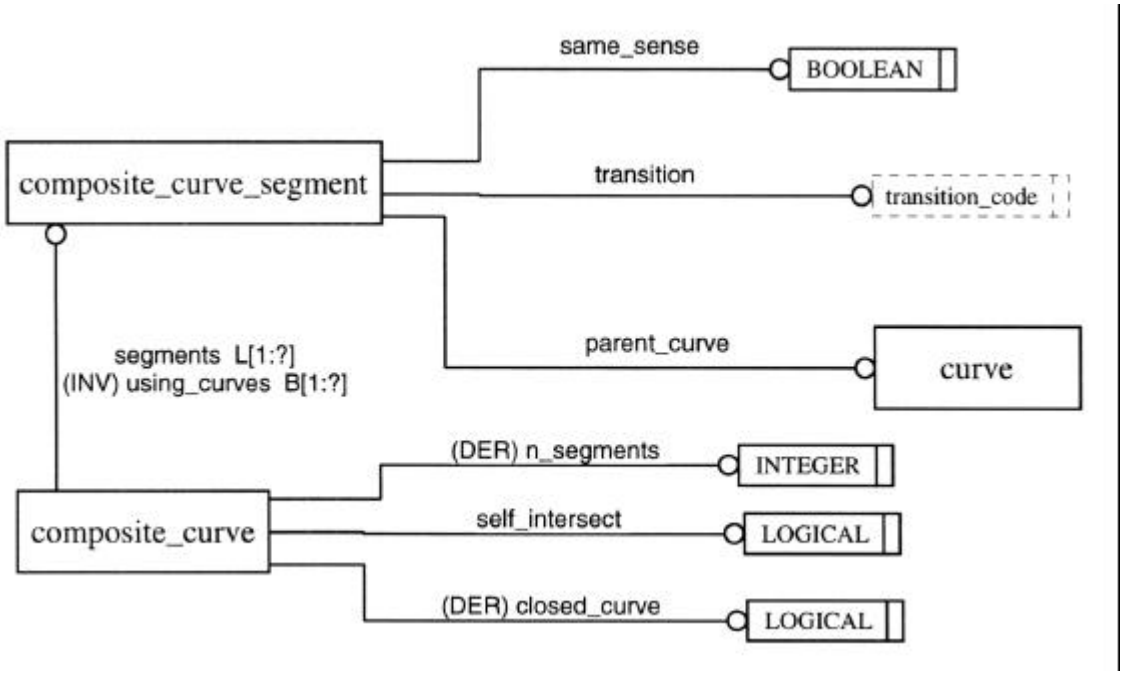


Figure A-21 Composite_Curve

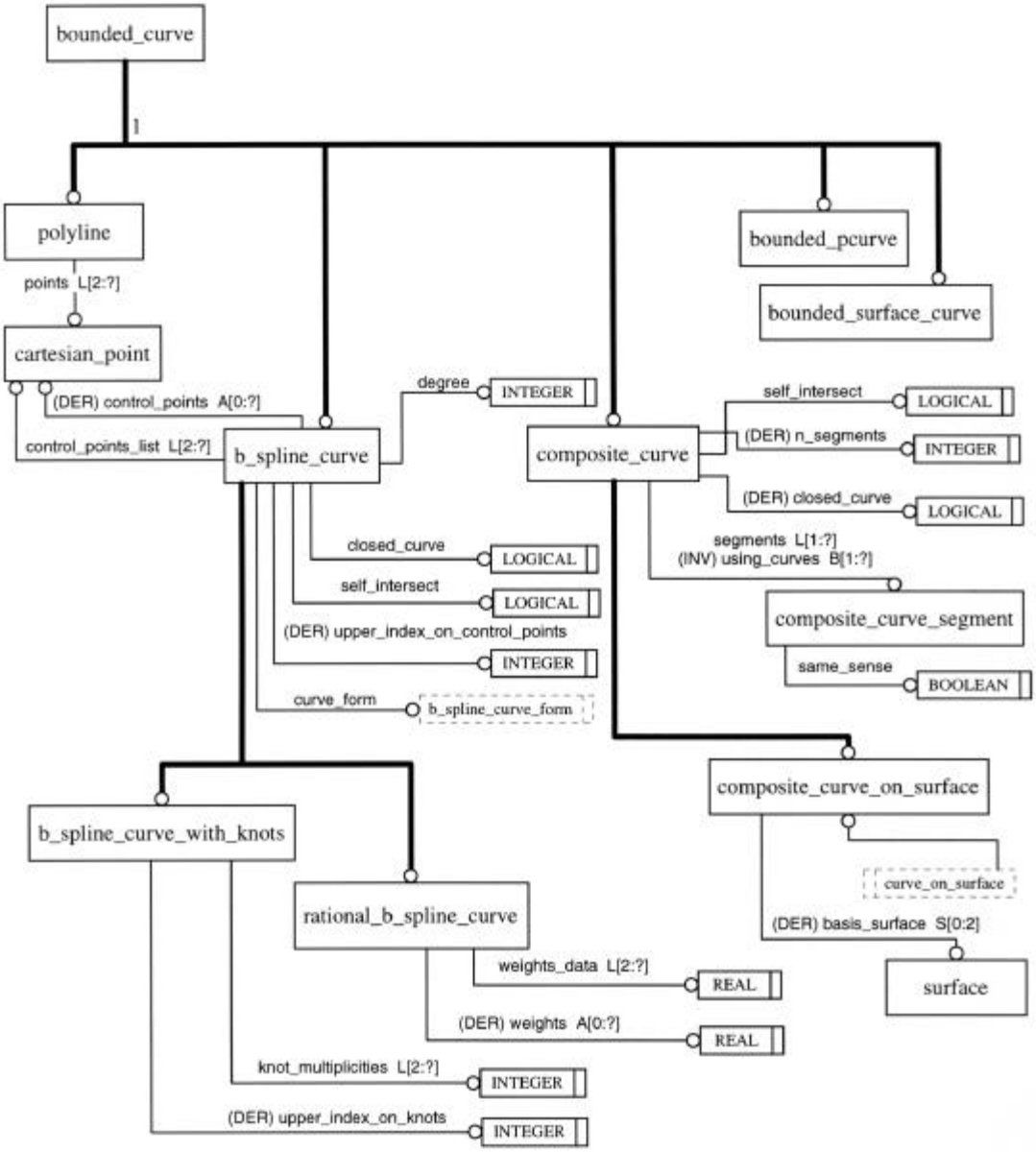


Figure A-22 Bounded_Curve

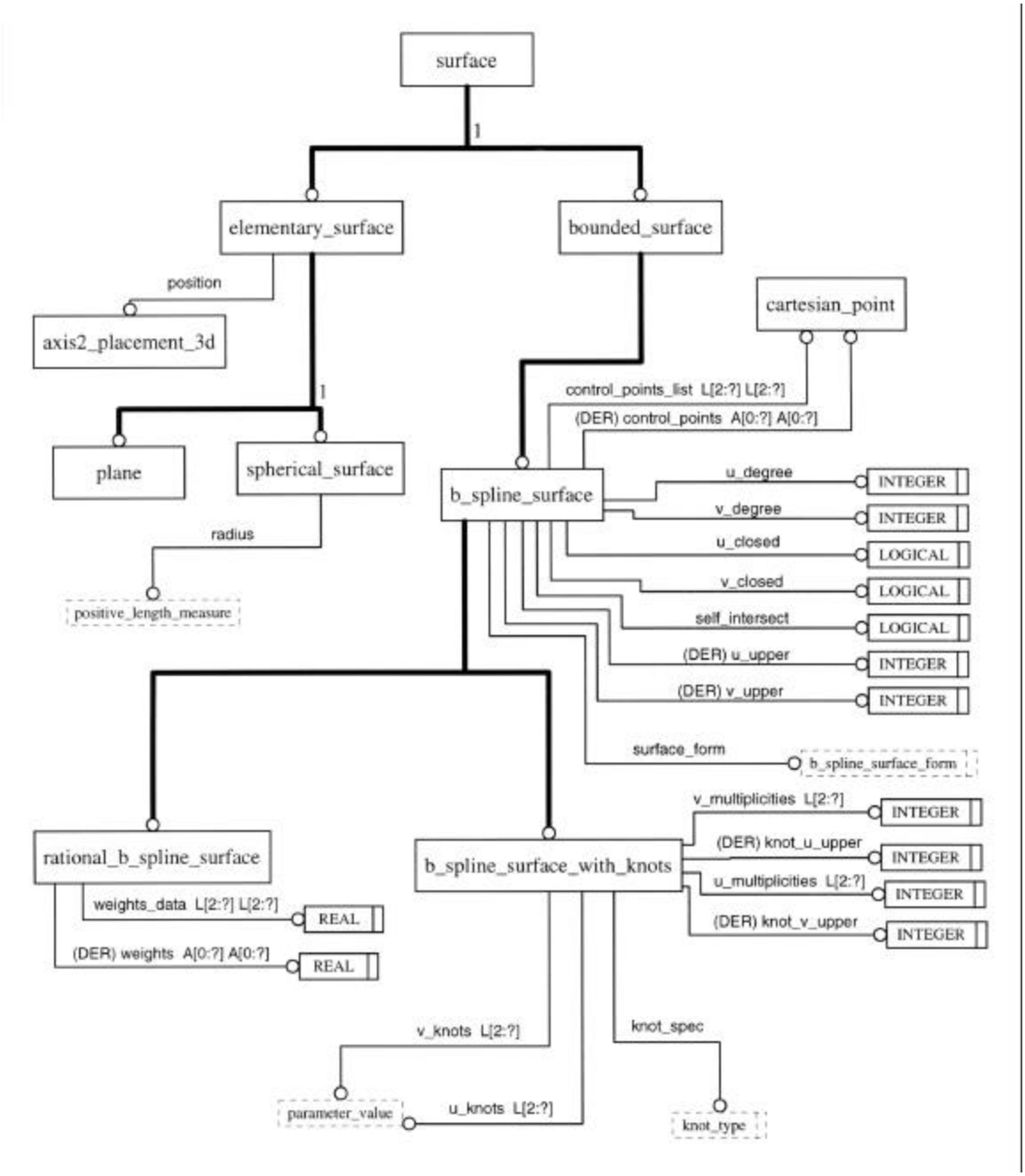


Figure A-23 Surface

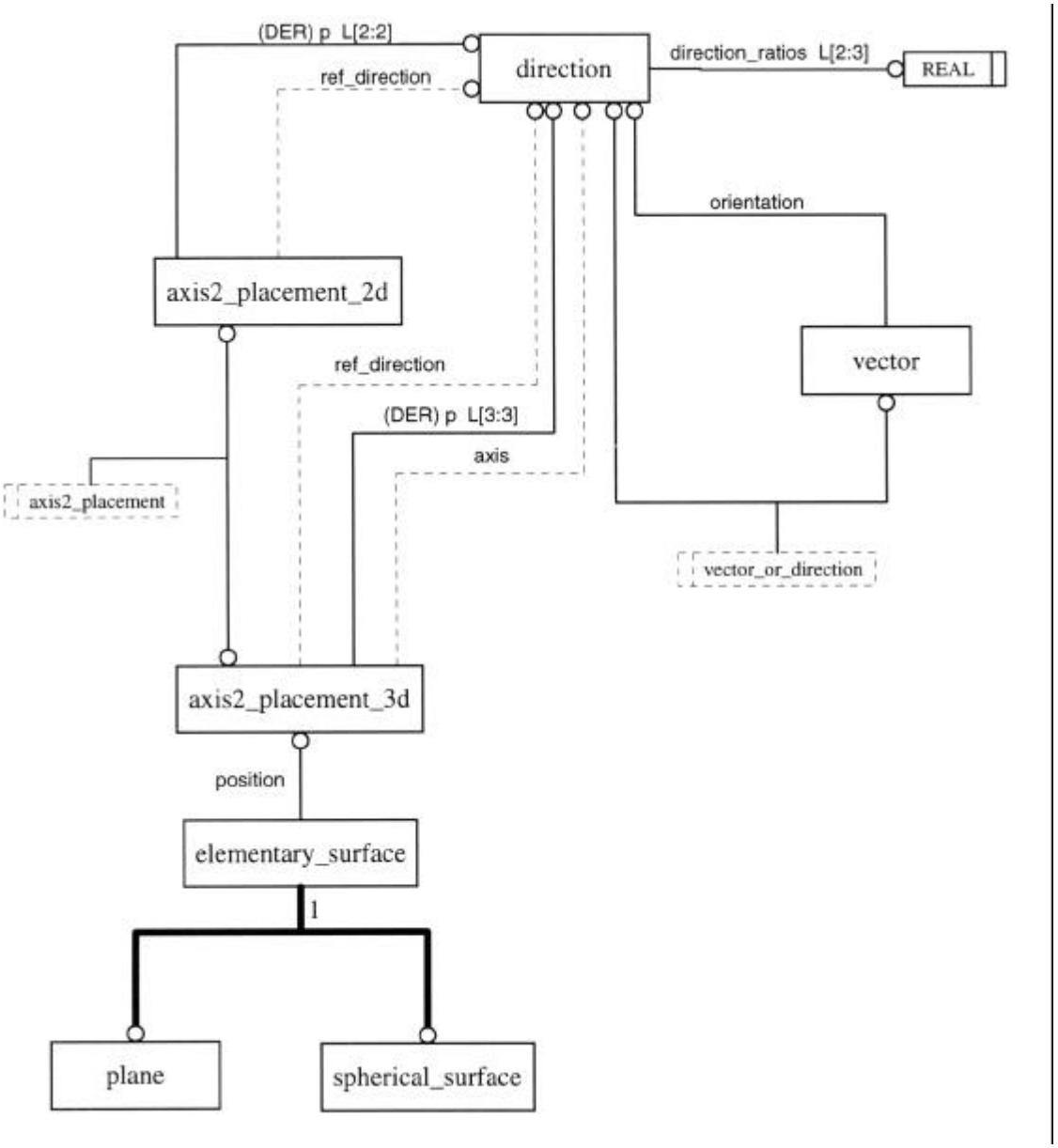


Figure A-24 Direction

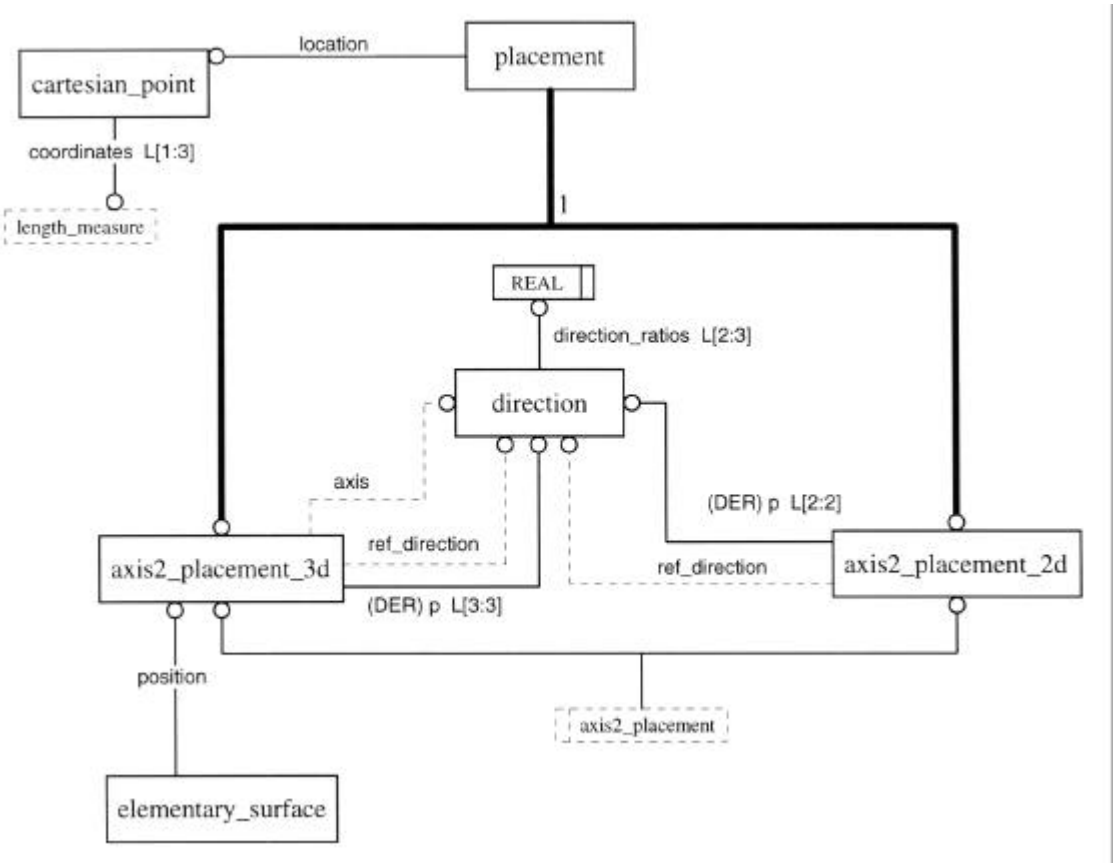


Figure A-25 Placement

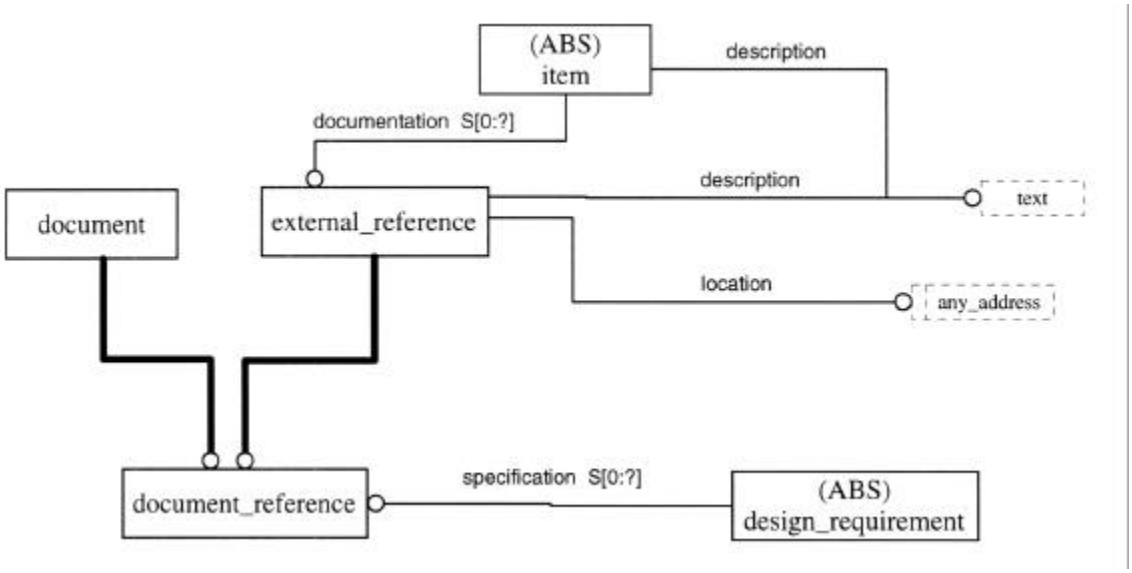


Figure A-26 (ABS) Item

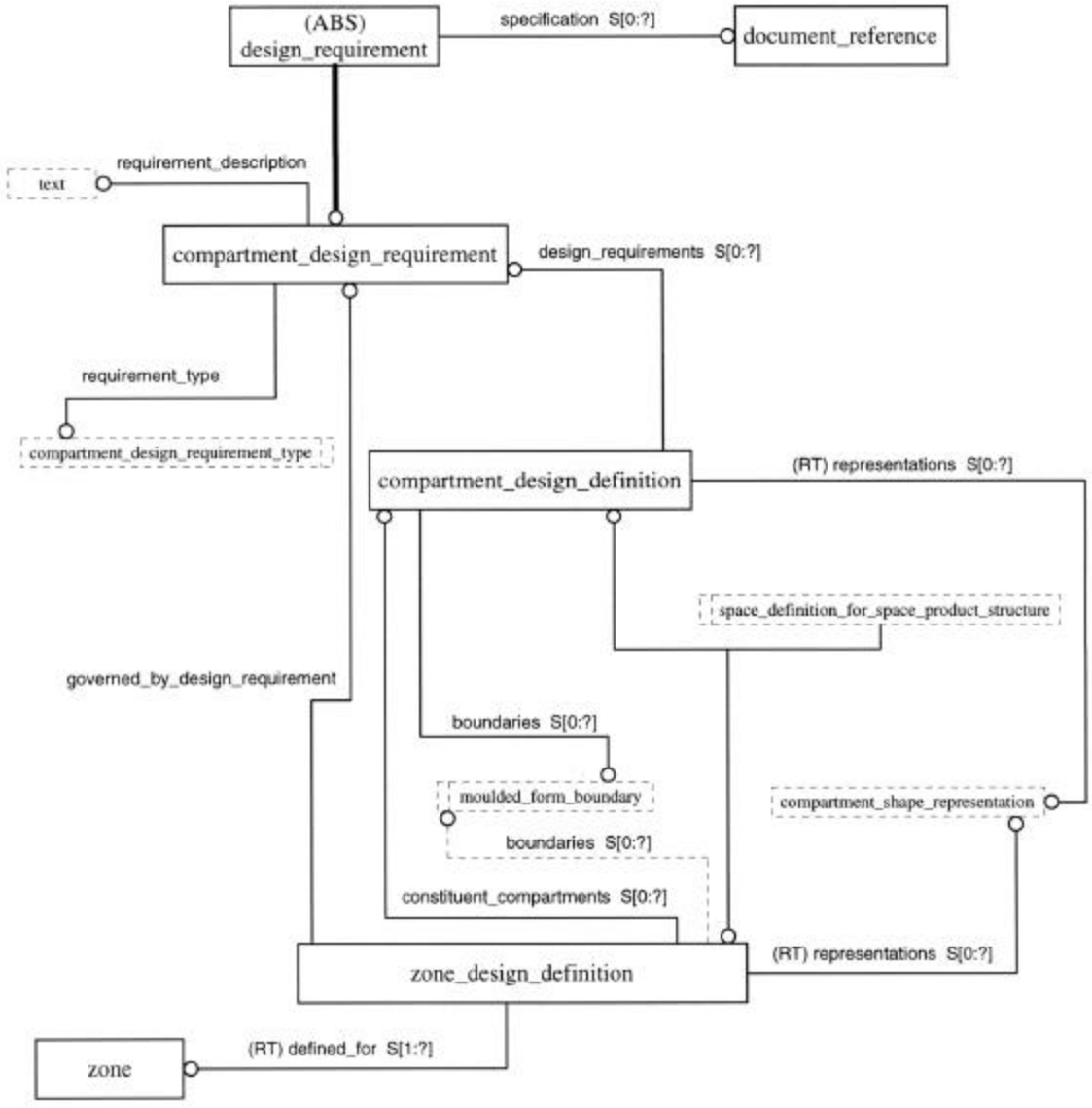


Figure A-27 Design_Requirement

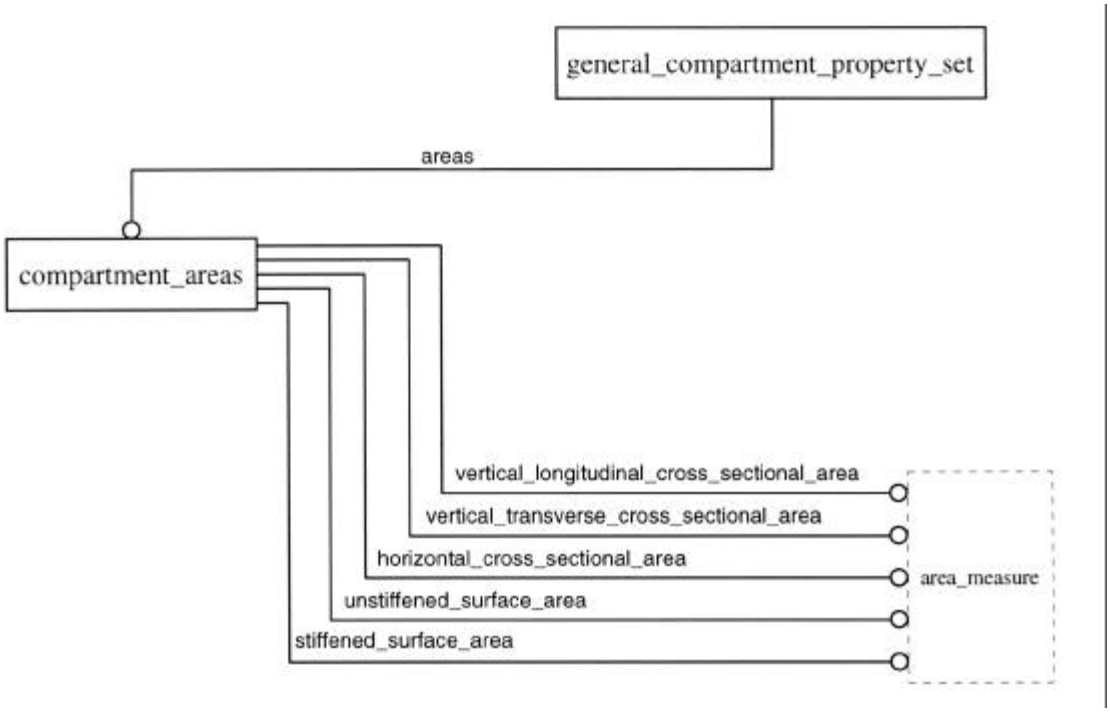


Figure A-28 General_Compartment_Property_Set

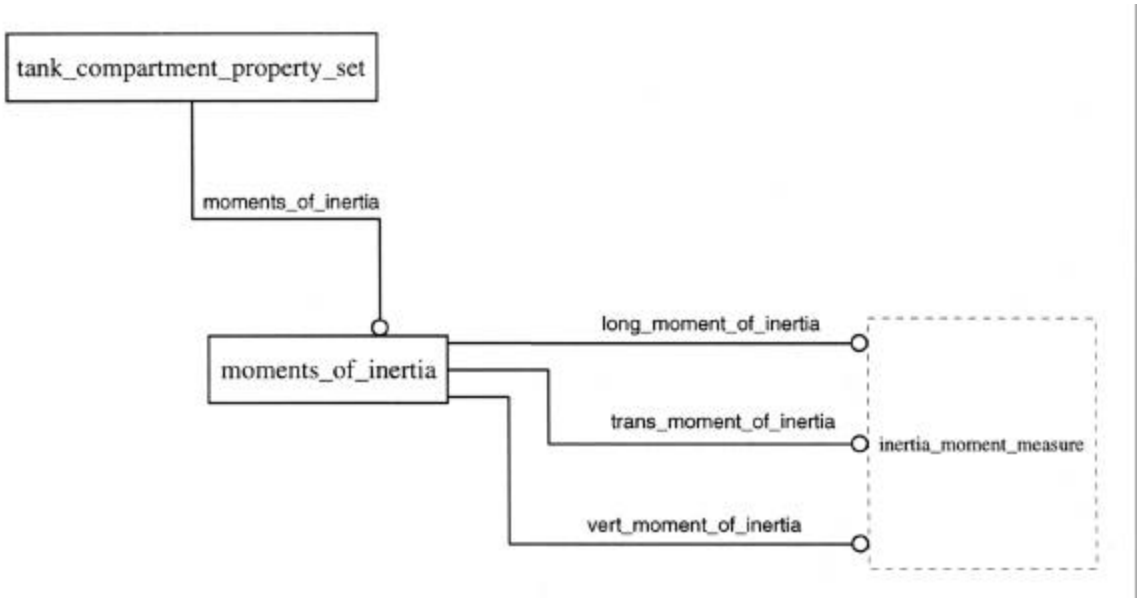


Figure A-29 Tank_Compartment_Property_Set

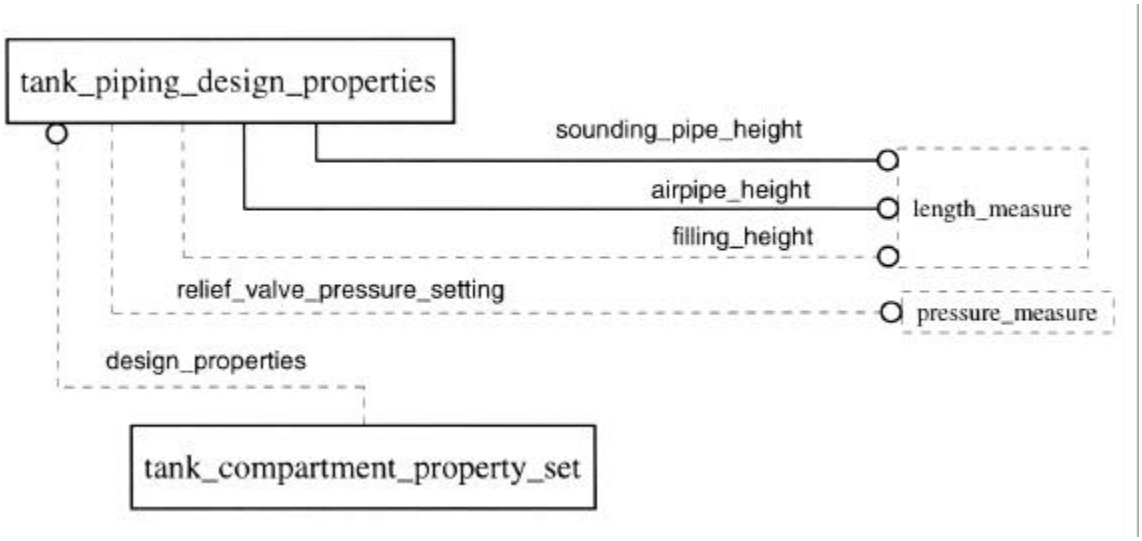


Figure A-30 Tank_Piping_Design_Properties

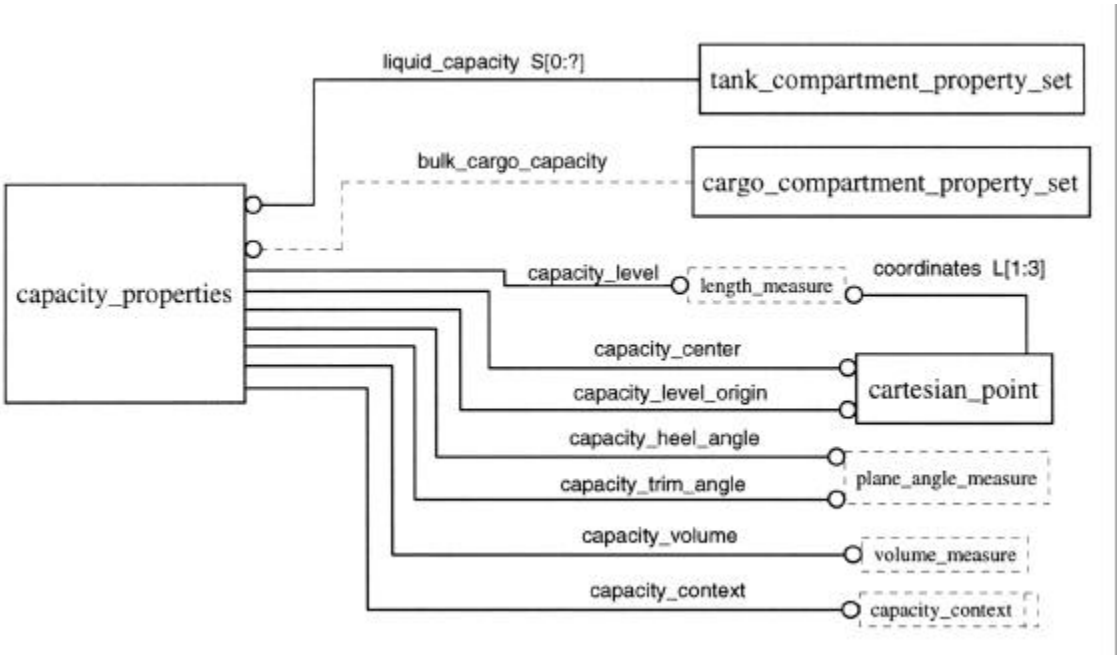


Figure A-31 Capacity_Properties

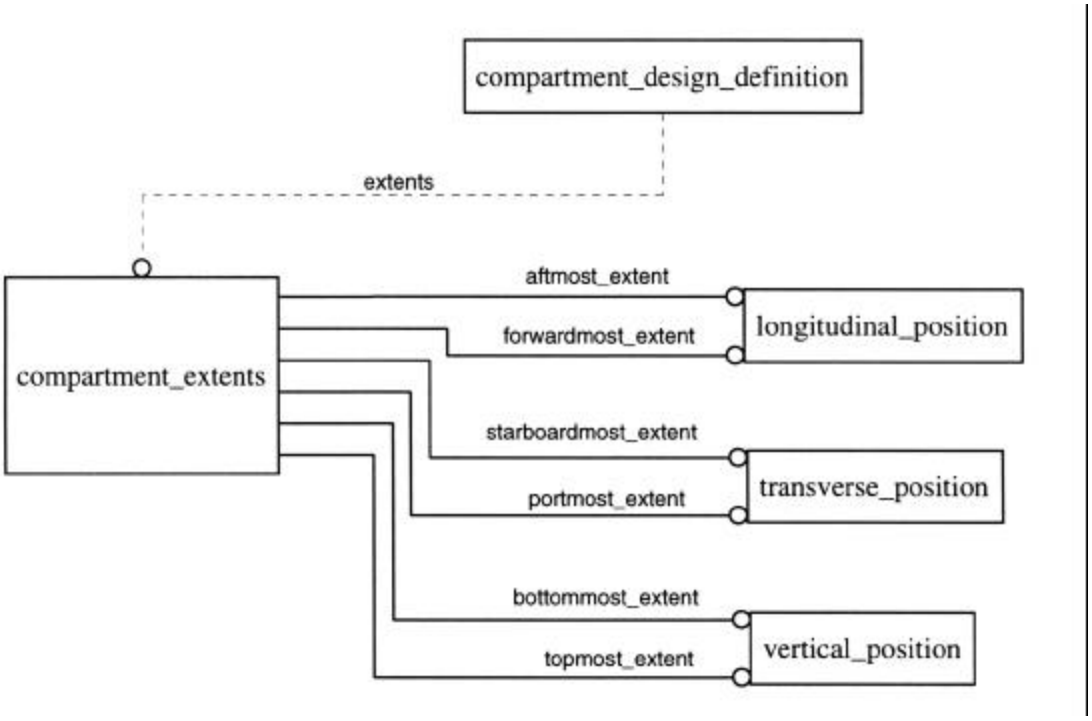


Figure A-32 Compartment_Extents

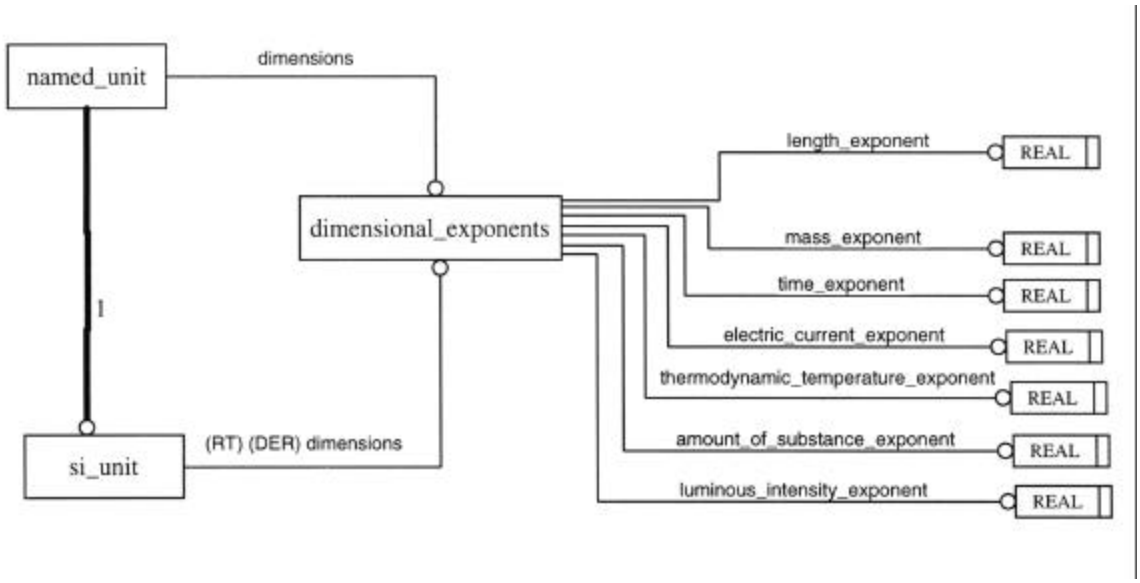


Figure A-33 Dimensional_Exponents

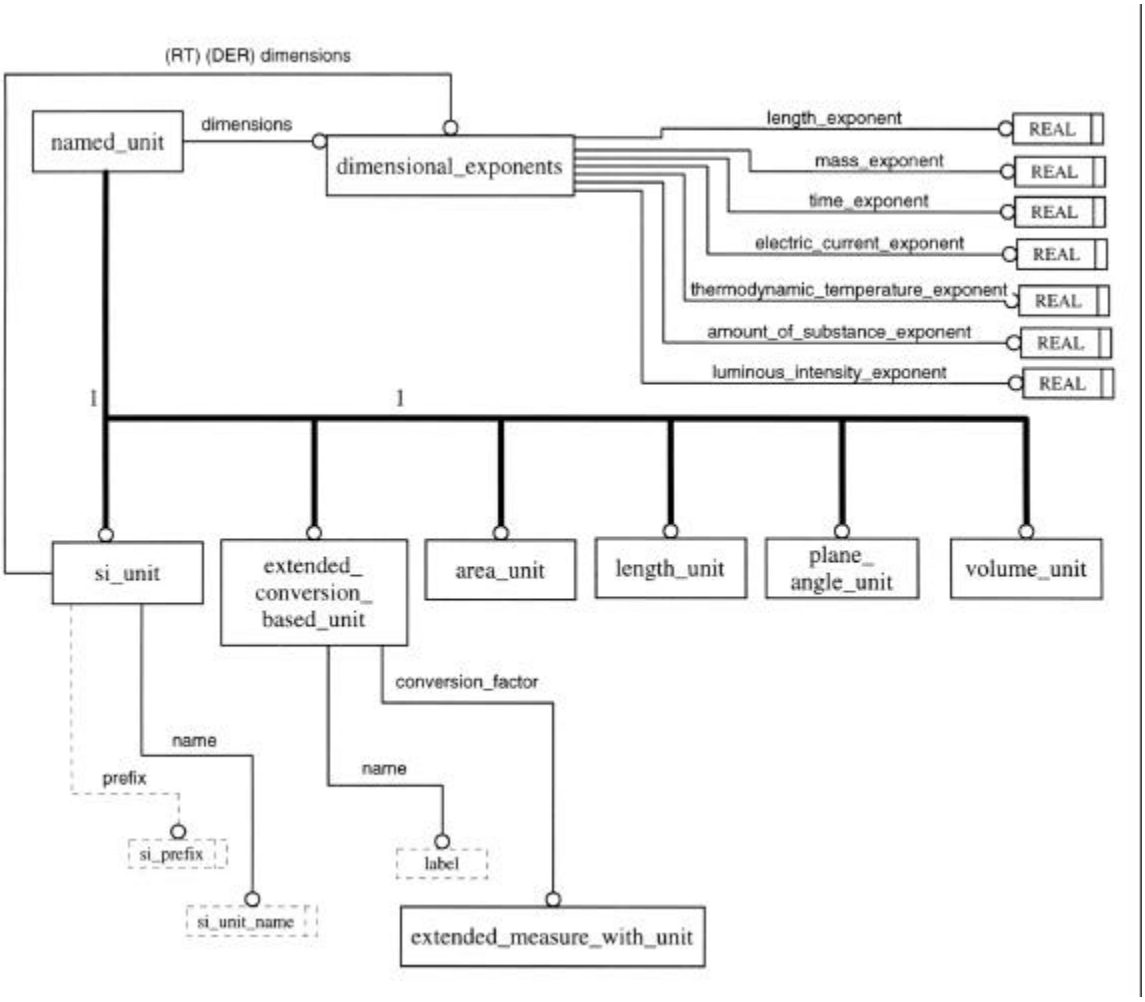


Figure A-34 Units

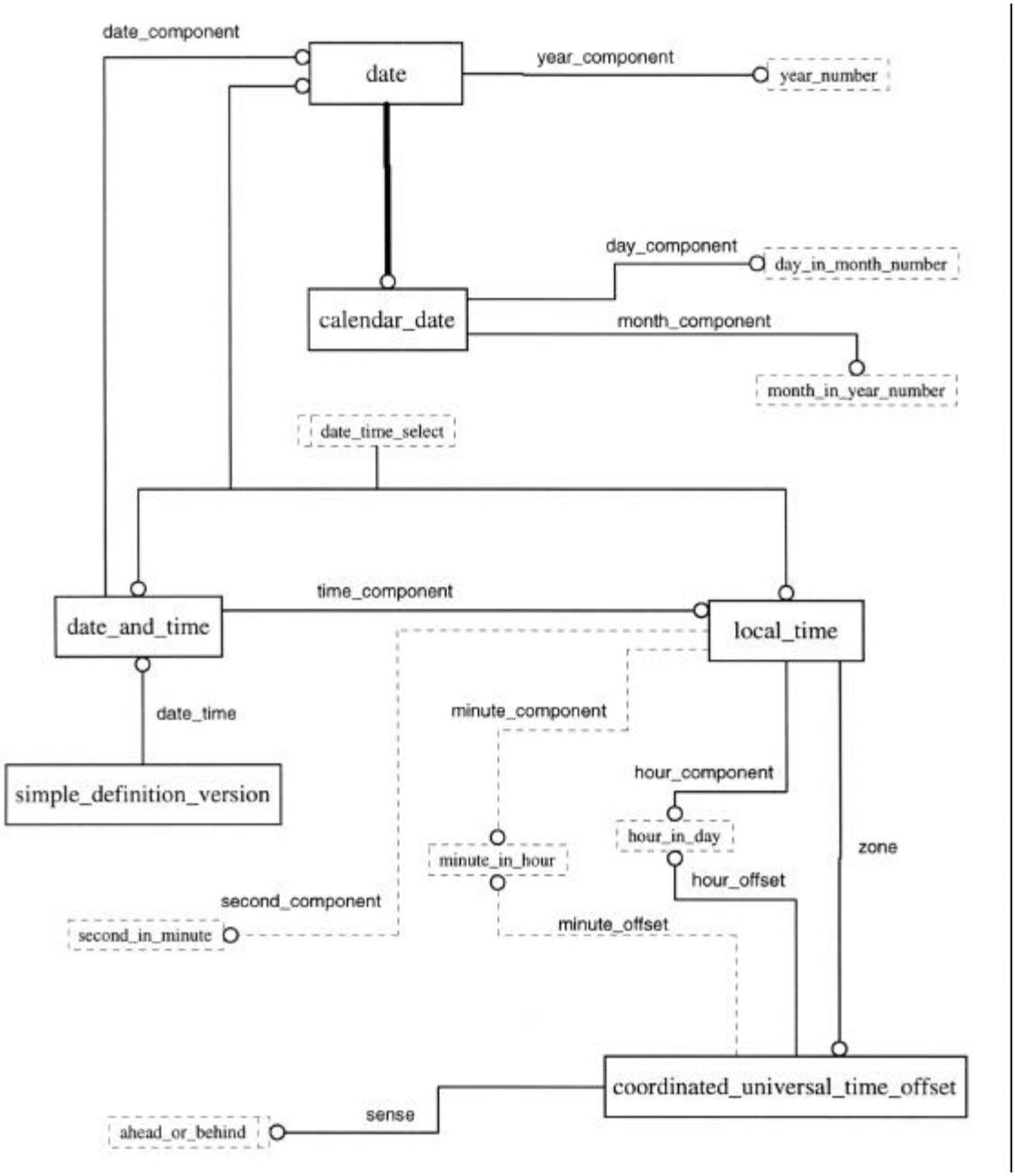


Figure A-35 Date

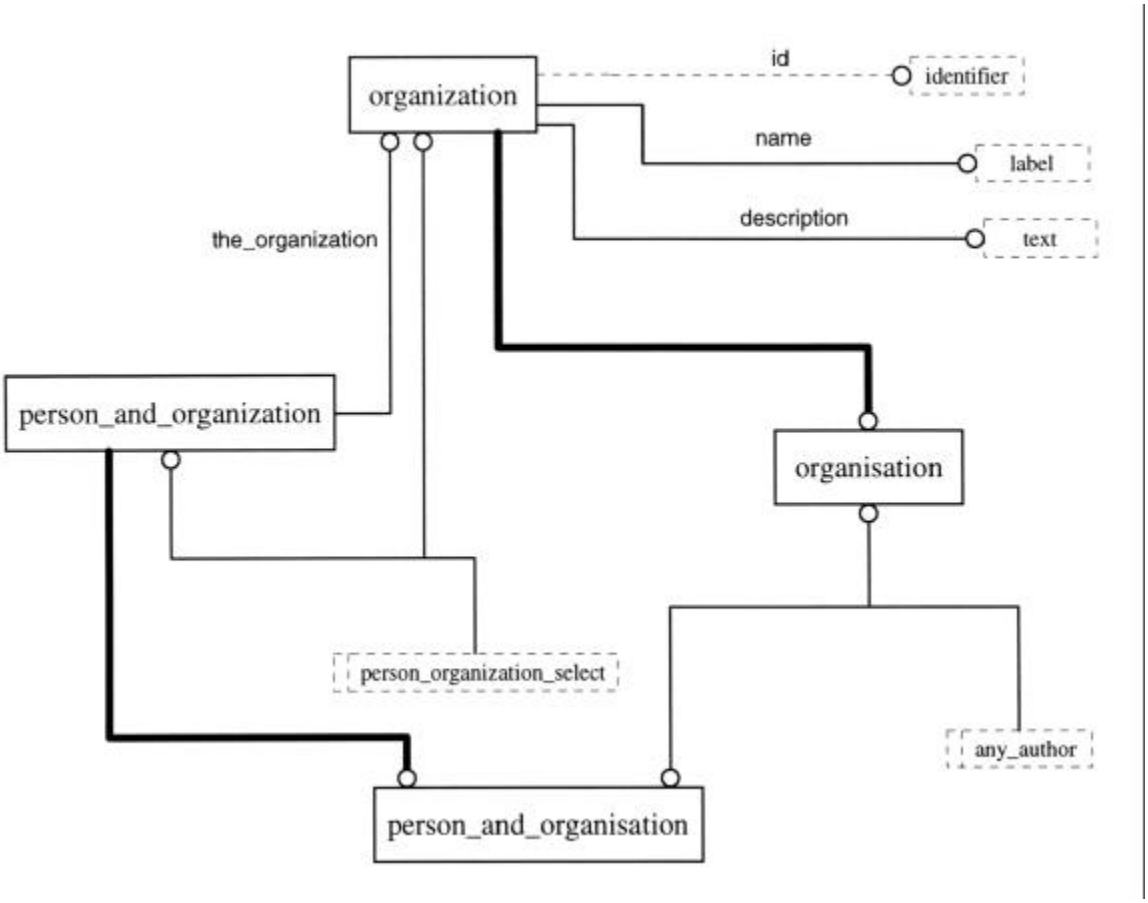


Figure A-36 Organization

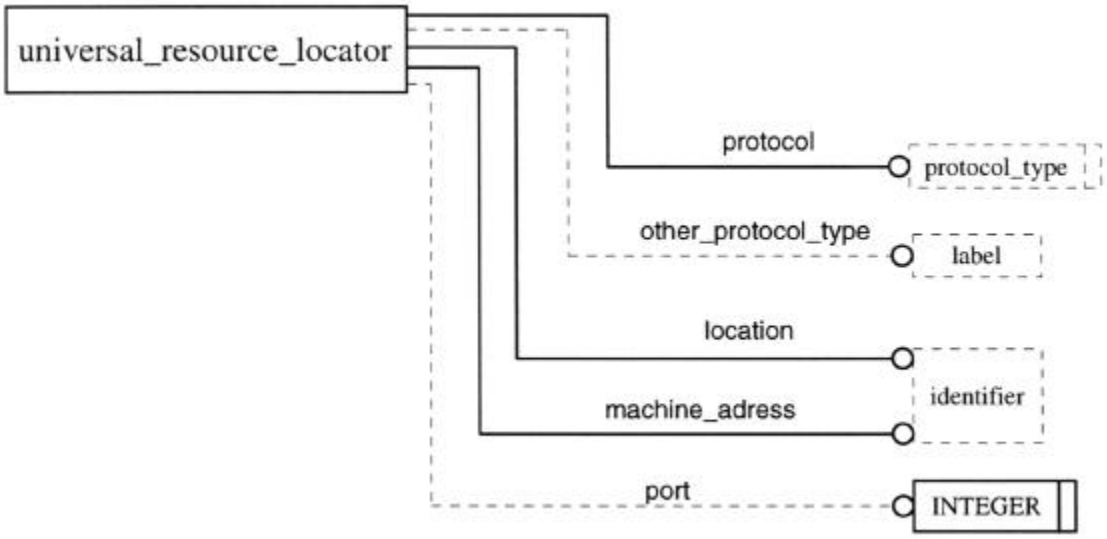


Figure A-37 Univeral_Resource_Locator

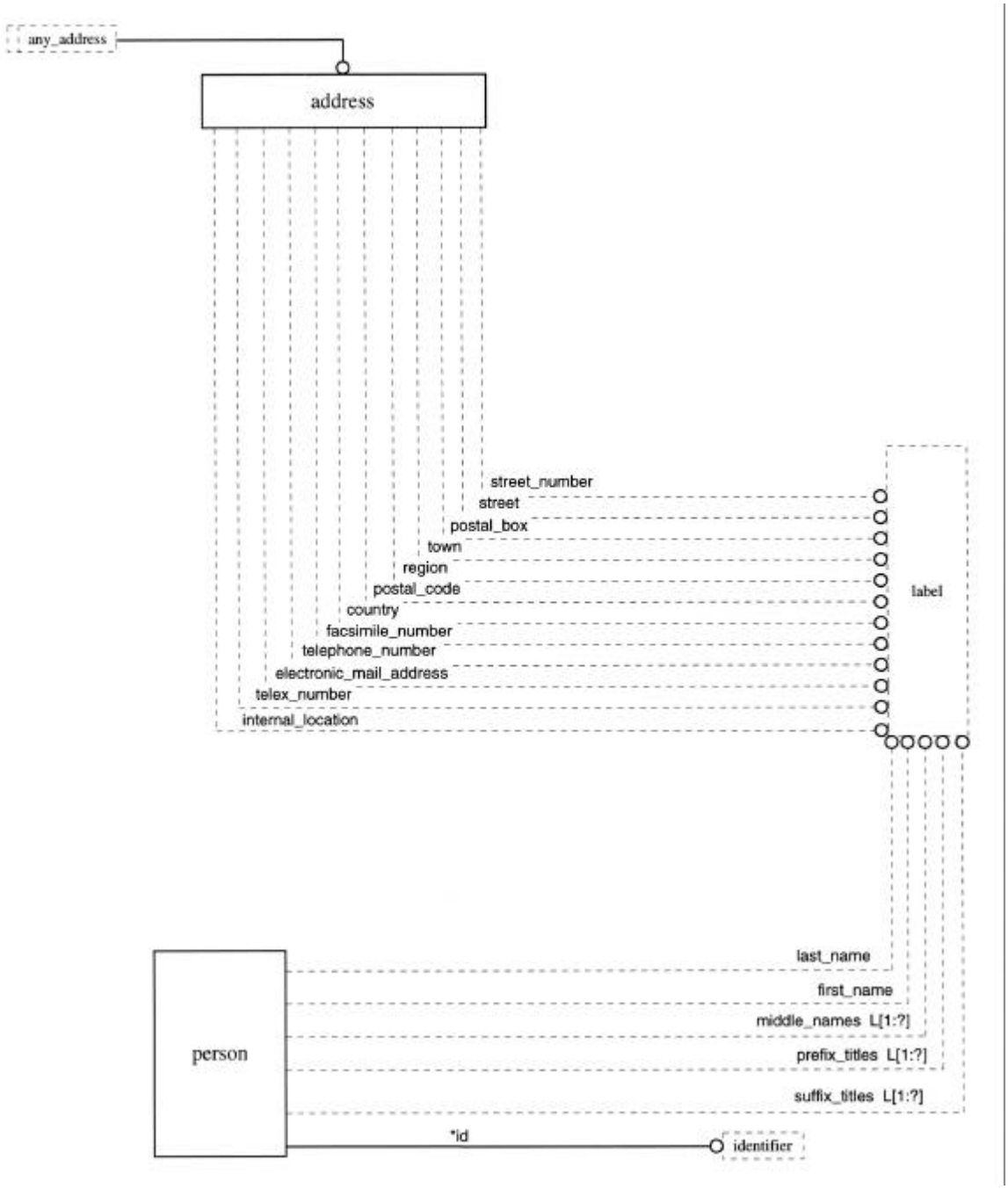


Figure A-38 Address

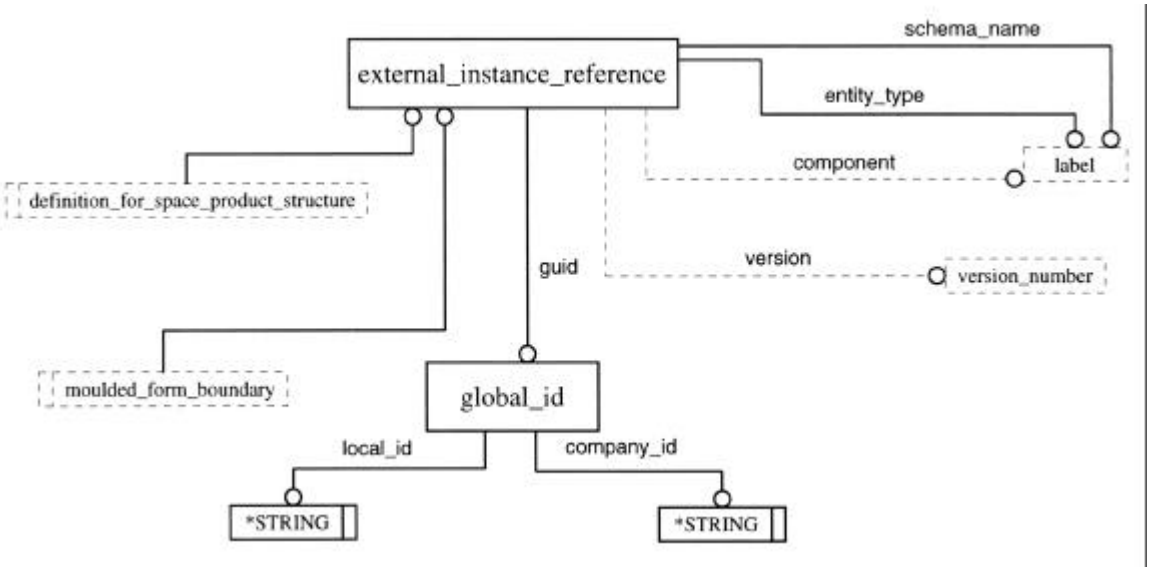


Figure A-39 External_Instance_Reference

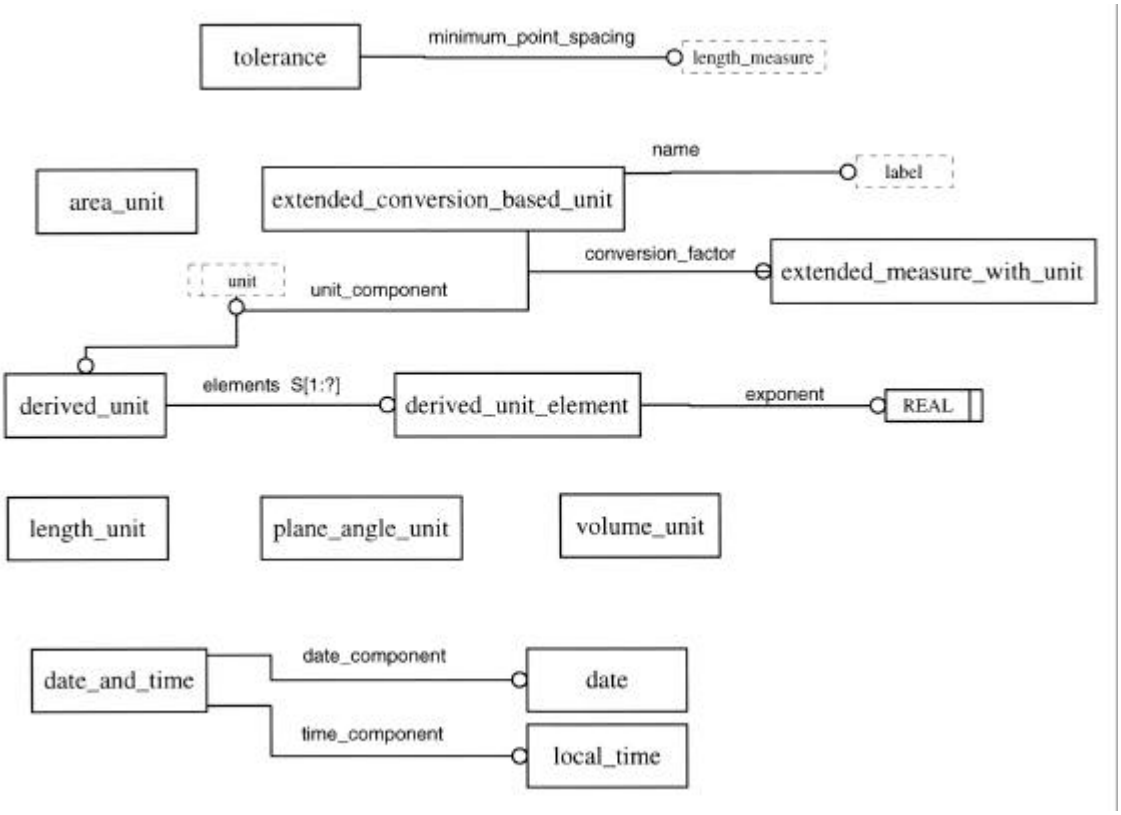


Figure A-40 Date, Units, Tolerance

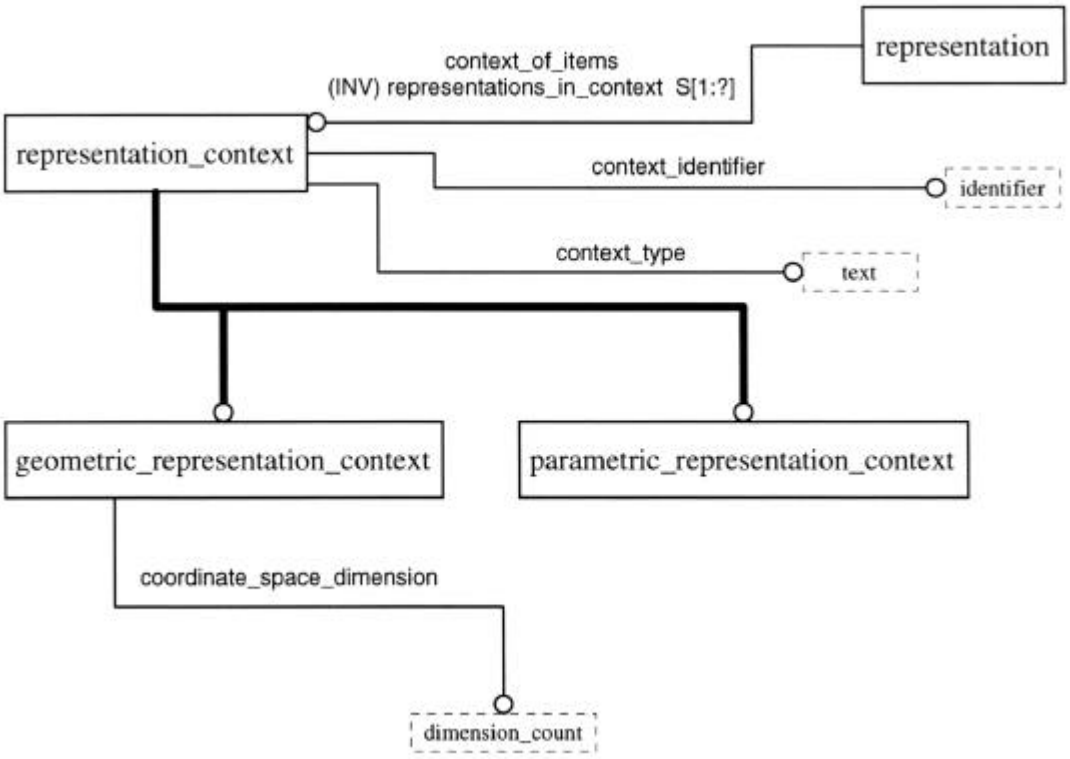


Figure A-41 Representation_Context

A.4 Example Testcase

This section contains an example of an AP 215 testcase from the suite of MariSTEP Ship Arrangements testcases as described by the following sections:

- Testcase Definition
- Entity Attribute Inheritance
- Express-I Diagrams of the Testcase
- Preprocessor and Postprocessor Test Worksheets

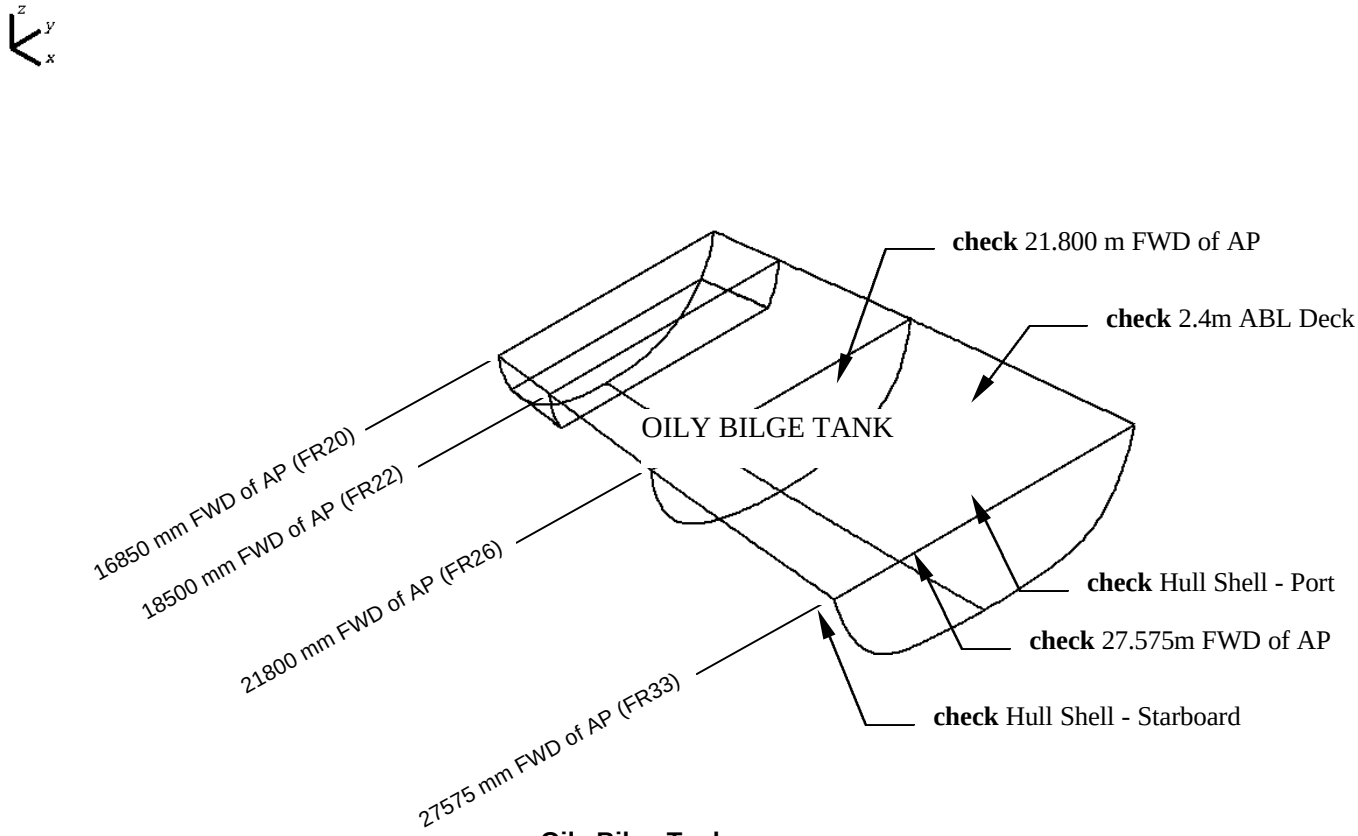
A.4.1 Testcase Definition

This testcase contains one tank compartment that is geometrically defined by structural molded forms exchanged using the NSRP Ship Moulded Forms schema.

Test Case No.: **AP215-2-1**
Name: **Oily Bilge Tank**

Description of Test Case:
Transfer tank compartment with external references to AP 216 - Moulded Forms including the tank compartment functional definition and property set.
The Oily Bilge Tank is located between Frame 26 - 33 at level 2.4M and below.
The test case will include:

- compartment
- documentation for the tank compartment
- tank compartment functional definition
- tank compartment design definition
- general compartment property set
- tank compartment property set
- external instance references to AP216 - Moulded Forms.



Oily Bilge Tank

Entities that are required to be checked (minus geometry):

Capacity_propertes
Compartment
Compartment_areas
Compartment_design_requirement
Compartment_design_to_functional_definition_relationship
Compartment_property_to_design_definition_relationship
Compartment_volume
External_instance_reference
General_compartment_property_set
Global_id
Tank_compartment_design_definition
Tank_compartment_functional_definition
Tank_compartment_property_set
Derived_unit
Derived_unit_element
Dimensional_exponents
Address
Document_reference
External_reference
Organisation
Person
Person_and_organisation
Compartment_permeability

A.4.2 Entity Attribute Inheritance

Each AP 215 entity has been "flattened" below to show the inheritance of attributes from its supertype entities. The "flattened entity" structure relates the order of the inheritance as well as what attributes is inherited from which supertype entity.

For example, the entity **cargo_compartment_design_definition** inherits attributes from **compartment_design_definition**, which inherits attributes from **design_definition**, which inherits attributes from **definition**. **Cargo_compartment_design_definition** has one attribute of its own: cargo_placement_locations_within_compartment. From **compartment_design_definition**, it inherits three attributes: design_requirements, boundaries, and extents. From **design_definition**, it inherits one attribute: representations. From **definition**, it inherits four attributes: id, defined_for, local_units, and version. Note that the following attributes are redefined:

- representations is redefined by **compartment_design_definition** to limit the entitiy set to be of type COMPARTMENT_SHAPE_REPRESENTATION
- defined_for is redefined by **compartment_design_definition** to limit the entity set to be of type **Compartment**

----------INSTANTIABLE "SHIPBUILDING" ENTITIES

ENTITY **capacity_properties;**
capacity_level_origin : cartesian_point;
capacity_center : cartesian_point;
capacity_level : LENGTH_MEASURE;
capacity_trim_angle : PLANE_ANGLE_MEASURE;
capacity_heel_angle : PLANE_ANGLE_MEASURE;
capacity_volume : VOLUME_MEASURE;
capacity_context : CAPACITY_CONTEXT;
END_ENTITY;

ENTITY **cargo_bay_definition;**
ENTITY definition;
id : global_id;
defined_for : SET [1:?] OF definable_object;
local_units : SET OF UNIT;
version : OPTIONAL simple_definition_version;
ENTITY cargo_bay_definition;
longitudinal_cargo_positions : SET OF longitudinal_position;
transverse_cargo_positions : SET OF transverse_position;
vertical_cargo_positions : SET OF vertical_position;
END_ENTITY;

```
ENTITY cargo_compartment_design_definition;
  ENTITY definition;
    id                                : global_id;
    [defined_for                      : SET [1:?] OF definable_object;]  Redeclared in
                                              compartment_design_definition
    local_units                       : SET OF UNIT;
    version                           : OPTIONAL simple_definition_version;
  ENTITY design_definition;
    [representations                  : SET OF representation;]  Redeclared in
                                              compartment_design_definition
  ENTITY compartment_design_definition;
    design_requirements               : SET OF compartment_design_requirement;
    boundaries                        : SET OF MOULDED_FORM_BOUNDARY;
    extents                           : OPTIONAL compartment_extents;
    design_definition.representations: SET OF COMPARTMENT_SHAPE_REPRESENTATION;
    definition.defined_for            : SET [1:?] OF compartment;
  ENTITY cargo_compartment_design_definition;
    cargo_placement_locations_within_compartment : OPTIONAL cargo_bay_definition;
END_ENTITY;

ENTITY cargo_compartment_functional_definition;
  ENTITY definition;
    id                          : global_id;
    [defined_for                : SET [1:?] OF definable_object;]  Redeclared in
                                        compartment_functional_definition
    local_units                 : SET OF UNIT;
    version                     : OPTIONAL simple_definition_version;
  ENTITY functional_definition;
    user_def_function           : OPTIONAL LABEL;
  ENTITY compartment_functional_definition;
    definition.defined_for      : SET [1:?] OF compartment;
    used_for                    : PRE_DEFINED_COMPARTMENT_FUNCTION;
  ENTITY cargo_compartment_functional_definition;
    func                        : PRE_DEFINED_CARGO_COMPARTMENT_FUNCTION;
END_ENTITY;

ENTITY cargo_compartment_property_set;
  ENTITY definition;
    id                          : global_id;
    [defined_for                : SET [1:?] OF definable_object;]  Redeclared in
                                        compartment_property_set
    local_units                 : SET OF UNIT;
    version                     : OPTIONAL simple_definition_version;
  ENTITY compartment_property_set;
    contxt                      : COMPARTMENT_PROPERTY_CONTEXT;
    definition.defined_for      : SET [1:?] OF compartment;
  ENTITY cargo_compartment_property_set;
    bulk_cargo_capacity         : OPTIONAL capacity_properties;
    design_stowage_density      : DENSITY_MEASURE;
END_ENTITY;
```

```
ENTITY compartment;
   ENTITY definable_object;
      id                    : LABEL;
    INVERSE
      definitions           : SET OF definition FOR defined_for
   ENTITY item;
      name                  : LABEL;
      documentation         : SET OF external_reference;
      description           : TEXT;
      ship_context          : OPTIONAL ship;
   ENTITY space;
   ENTITY compartment;
END_ENTITY;

ENTITY compartment_areas;
      vertical_longitudinal_cross_sectional_area      : AREA_MEASURE;
      vertical_transverse_cross_sectional_area : AREA_MEASURE;
      horizontal_cross_sectional_area         : AREA_MEASURE;
      unstiffened_surface_area                : AREA_MEASURE;
      stiffened_surface_area                  : AREA_MEASURE;
END_ENTITY;

ENTITY compartment_design_definition;
   ENTITY definition;
      id                                : global_id;
      [defined_for                      : SET [1:?] OF definable_object;]  Redeclared in
                                        compartment_design_definition
      local_units                       : SET OF UNIT;
      version                           : OPTIONAL simple_definition_version;
   ENTITY design_definition;
      [representations                  : SET OF representation;]  Redeclared in
                                        compartment_design_definition
   ENTITY compartment_design_definition;
      design_requirements               : SET OF compartment_design_requirement;
      boundaries                        : SET OF MOULDED_FORM_BOUNDARY;
      extents                           : OPTIONAL compartment_extents;
      design_definition.representations: SET OF COMPARTMENT_SHAPE_REPRESENTATION;
      definition.defined_for            : SET [1:?] OF compartment;
END_ENTITY;
```

```
ENTITY compartment_design_requirement;
   ENTITY definition;
      id                              : global_id;
      defined_for                     : SET [1:?] OF definable_object;
      local_units                     : SET OF UNIT;
      version                         : OPTIONAL simple_definition_version;
   ENTITY design_requirement;
      specification                   : SET OF document_reference;
   ENTITY compartment_design_requirement;
      requirement_type                        : COMPARTMENT_DESIGN_REQUIREMENT_TYPE;
      requirement_description         : TEXT;
END_ENTITY;

ENTITY compartment_design_to_functional_definition_relationship;
   ENTITY definition_relationship;
      [definition_1                   : definition;]  Redeclared in
                                      compartment_design_to_functional_definition_relationship
      [definition_2                   : definition;]  Redeclared in
                                      compartment_design_to_functional_definition_relationship
      description                     : OPTIONAL TEXT;
   ENTITY compartment_design_to_functional_definition_relationship;
      definition_relationship.definition_1      : compartment_design_definition;
      definition_relationship.definition_2      : compartment_functional_definition;
END_ENTITY;

ENTITY compartment_extents;
      forwardmost_extent      : longitudinal_position;
      aftmost_extent          : longitudinal_position;
      portmost_extent         : transverse_position;
      starboardmost_extent    : transverse_position;
      topmost_extent          : vertical_position;
      bottommost_extent               : vertical_position;
END_ENTITY;

ENTITY compartment_functional_definition;
   ENTITY definition;
      id                      : global_id;
      [defined_for            : SET [1:?] OF definable_object;]  Redeclared in
                                      compartment_functional_definition
      local_units             : SET OF UNIT;
      version                 : OPTIONAL simple_definition_version;
   ENTITY functional_definition;
      user_def_function               : OPTIONAL LABEL;
   ENTITY compartment_functional_definition;
      definition.defined_for  : SET [1:?] OF compartment;
      used_for                : PRE_DEFINED_COMPARTMENT_FUNCTION;
END_ENTITY;
```

```
ENTITY compartment_naval_administrative_property_set;
   ENTITY definition;
      id                              : global_id;
      [defined_for                    : SET [1:?] OF definable_object;]  Redeclared in
                                              compartment_property_set
      local_units                     : SET OF UNIT;
      version                         : OPTIONAL simple_definition_version;
   ENTITY compartment_property_set;
      contxt                          : COMPARTMENT_PROPERTY_CONTEXT;
      definition.defined_for          : SET [1:?] OF compartment;
   ENTITY compartment_naval_administrative_property_set;
      compartment_abbreviated_name : LABEL;
      department_ziplist_number               : TEXT;
      division_ziplist_number         : TEXT;
      maximum_accleration_g_force     : REAL;
      nuclear_classification          : NAVAL_NUCLEAR_CLASSIFICATION;
      noise_category                  : NAVAL_NOISE_CATEGORY;
      insulation_category             : NAVAL_INSULATION_CATEGORY;
      authorization_classification            : NAVAL_AUTHORIZATION_CLASSIFICATION;
      safety_category                 : NAVAL_SAFETY_CATEGORY;
      security_classification         : NAVAL_SECURITY_CLASSIFICATION;
END_ENTITY;

ENTITY compartment_permeability;
      volume_permeability     : PERCENTAGE;
      area_permeability               : PERCENTAGE;
END_ENTITY;

ENTITY compartment_property_to_design_definition_relationship;
   ENTITY definition_relationship;
      [definition_1                   : definition;]  Redeclared in
                                        compartment_property_to_design_definition_relationship
      [definition_2                   : definition;]  Redeclared in
                                        compartment_property_to_design_definition_relationship
      description                     : OPTIONAL TEXT;
   ENTITY compartment_property_to_design_definition_relationship;
      definition_relationship.definition_1       : compartment_property_set;
      definition_relationship.definition_2       : compartment_design_definition;
END_ENTITY;

ENTITY compartment_volume;
      volume                  : VOLUME_MEASURE;
      centre_of_volume                : cartesian_point;
END_ENTITY;

ENTITY date_and_time;
      date_component          : date;
      time_component          : local_time;
END_ENTITY;
```

```
ENTITY definition_relationship;
    definition_1          : definition;
    definition_2          : definition;
    description           : OPTIONAL TEXT;
END_ENTITY;

ENTITY external_instance_reference;
    schema_name           : LABEL;
    entity_type           : LABEL;
    guid                  : global_id;
    version               : OPTIONAL VERSION_NUMBER;
    component             : OPTIONAL LABEL;
END_ENTITY;

ENTITY general_compartment_property_set;
  ENTITY definition;
    id                    : global_id;
    [defined_for          : SET [1:?] OF definable_object;]  Redeclared in
                                compartment_property_set
    local_units           : SET OF UNIT;
    version               : OPTIONAL simple_definition_version;
  ENTITY compartment_property_set;
    contxt                : COMPARTMENT_PROPERTY_CONTEXT;
    definition.defined_for : SET [1:?] OF compartment;
  ENTITY general_compartment_property_set;
    volume                : compartment_volume;
    areas                 : compartment_areas;
    permeability          : compartment_permeability;
    required_bulkhead_tightness     : COMPARTMENT_REQUIRED_BULKHEAD_TIGHTNESS;
END_ENTITY;

ENTITY global_axis_placement;
  ENTITY definition;
    id                              : global_id;
    [defined_for                    : SET [1:?] OF definable_object;]  Redeclared in
                                          general_characteristics_definition
    local_units                     : SET OF UNIT;
    version                         : OPTIONAL simple_definition_version;
  ENTITY general_characteristics_definition;
    definition.defined_for          : SET [1:?] OF ship;
  ENTITY global_axis_placement;
    orientation                     : GLOBAL_COORDINATE_SYSTEM_ORIENTATION;
    after_perpendicular_offset            : LENGTH_MEASURE;
    forward_perpendicular_offset    : LENGTH_MEASURE;
   DERIVE
    length_between_perpendiculars   : LENGTH_MEASURE;
END_ENTITY;
```

```
ENTITY global_id;
    company_id              : STRING (4) FIXED;
    local_id                : STRING (64) FIXED;
END_ENTITY;

ENTITY habitable_compartment_design_definition;
  ENTITY definition;
    id                              : global_id;
    [defined_for                    : SET [1:?] OF definable_object;]  Redeclared in
                                          compartment_design_definition
    local_units                     : SET OF UNIT;
    version                         : OPTIONAL simple_definition_version;
  ENTITY design_definition;
    [representations                : SET OF representation;]  Redeclared in
                                          compartment_design_definition
  ENTITY compartment_design_definition;
    design_requirements             : SET OF compartment_design_requirement;
    boundaries                      : SET OF MOULDED_FORM_BOUNDARY;
    extents                         : OPTIONAL compartment_extents;
    design_definition.representations: SET OF COMPARTMENT_SHAPE_REPRESENTATION;
    definition.defined_for          : SET [1:?] OF compartment;
  ENTITY habitable_compartment_design_definition;
END_ENTITY;

ENTITY habitable_compartment_functional_definition;
  ENTITY definition;
    id                      : global_id;
    [defined_for            : SET [1:?] OF definable_object;]  Redeclared in
                                  compartment_functional_definition
    local_units             : SET OF UNIT;
    version                 : OPTIONAL simple_definition_version;
  ENTITY functional_definition;
    user_def_function               : OPTIONAL LABEL;
  ENTITY compartment_functional_definition;
    definition.defined_for  : SET [1:?] OF compartment;
    used_for                : PRE_DEFINED_COMPARTMENT_FUNCTION;
  ENTITY habitable_compartment_functional_definition;
    func                    : PRE_DEFINED_HABITABLE_COMPARTMENT_FUNCTION;
END_ENTITY;
```

```
ENTITY habitable_compartment_property_set;
   ENTITY definition;
      id                        : global_id;
      [defined_for              : SET [1:?] OF definable_object;]  Redeclared in
                                        compartment_property_set
      local_units               : SET OF UNIT;
      version                   : OPTIONAL simple_definition_version;
   ENTITY compartment_property_set;
      contxt                    : COMPARTMENT_PROPERTY_CONTEXT;
      definition.defined_for    : SET [1:?] OF compartment;
   ENTITY habitable_compartment_property_set;
      max_occupancy             : OPTIONAL INTEGER;
      air_circulation_rate      : VOLUME_MEASURE;
      illumination_value                : LUMINOUS_INTENSITY_MEASURE;
END_ENTITY;

ENTITY longitudinal_position;
   ENTITY spacing_position;
      position_number           : OPTIONAL INTEGER;
      name                      : OPTIONAL STRING;
      location                  : LENGTH_MEASURE;
    INVERSE
      defining_table            : spacing_table FOR table_positions
   ENTITY longitudinal_position;
      offset                    : OPTIONAL LENGTH_MEASURE;
END_ENTITY;

ENTITY machinery_compartment_design_definition;
   ENTITY definition;
      id                                : global_id;
      [defined_for                      : SET [1:?] OF definable_object;]  Redeclared in
                                              compartment_design_definition
      local_units                       : SET OF UNIT;
      version                           : OPTIONAL simple_definition_version;
   ENTITY design_definition;
      [representations                  : SET OF representation;]  Redeclared in
                                              compartment_design_definition
   ENTITY compartment_design_definition;
      design_requirements               : SET OF compartment_design_requirement;
      boundaries                        : SET OF MOULDED_FORM_BOUNDARY;
      extents                           : OPTIONAL compartment_extents;
      design_definition.representations: SET OF COMPARTMENT_SHAPE_REPRESENTATION;
      definition.defined_for            : SET [1:?] OF compartment;
   ENTITY machinery_compartment_design_definition;
END_ENTITY;
```

```
ENTITY machinery_compartment_functional_definition;
   ENTITY definition;
      id                        : global_id;
      [defined_for              : SET [1:?] OF definable_object;]  Redeclared in
                                          compartment_functional_definition
      local_units               : SET OF UNIT;
      version                   : OPTIONAL simple_definition_version;
   ENTITY functional_definition;
      user_def_function                 : OPTIONAL LABEL;
   ENTITY compartment_functional_definition;
      definition.defined_for    : SET [1:?] OF compartment;
      used_for                  : PRE_DEFINED_COMPARTMENT_FUNCTION;
   ENTITY machinery_compartment_functional_definition;
      func                      : PRE_DEFINED_MACHINERY_COMPARTMENT_FUNCTION;
END_ENTITY;

ENTITY moments_of_inertia;
      long_moment_of_inertia          : INERTIA_MOMENT_MEASURE;
      trans_moment_of_inertia         : INERTIA_MOMENT_MEASURE;
      vert_moment_of_inertia : INERTIA_MOMENT_MEASURE;
END_ENTITY;

ENTITY non_structural_moulded_form_design_definition;
   ENTITY definition;
      id                        : global_id;
      defined_for               : SET [1:?] OF definable_object;
      local_units               : SET OF UNIT;
      version                   : OPTIONAL simple_definition_version;
   ENTITY design_definition;
      [representations          : SET OF representation;]  Redeclared in
                        non_structural_moulded_form_design_definition
   ENTITY non_structural_moulded_form_design_definition;
      usage                     : NON_STRUCTURAL_MOULDED_REGION_USAGE;
      intended_location                 : OPTIONAL spacing_position;
      boundary_definitions      : OPTIONAL LIST OF MOULDED_FORM_BOUNDARY;
      design_definition.representations      : SET OF COMPARTMENT_SHAPE_REPRESENTATION;
END_ENTITY;
```

```
ENTITY ship;
   ENTITY definable_object;
      id                : LABEL;
    INVERSE
      definitions       : SET OF definition FOR defined_for
   ENTITY item;
      name              : LABEL;
      documentation     : SET OF external_reference;
      description       : TEXT;
      ship_context      : OPTIONAL ship;
   ENTITY ship;
      units             : SET OF UNIT;
END_ENTITY;

ENTITY simple_definition_version;
      version                   : VERSION_NUMBER;
      date_time                       date_and_time;
END_ENTITY;

ENTITY space_product_structure_definition;
   ENTITY definition;
      id                              : global_id;
      defined_for                     : SET [1:?] OF definable_object;
      local_units                     : SET OF UNIT;
      version                         : OPTIONAL simple_definition_version;
   ENTITY space_product_structure_definition;
      relating_space_definition             :
                              SPACE_DEFINITION_FOR_SPACE_PRODUCT_STRUCTURE;
      related_design_definitions            : SET OF
                              DEFINITION_FOR_SPACE_PRODUCT_STRUCTURE;
END_ENTITY;

ENTITY spacing_grid;
   ENTITY definable_object;
      id                    : LABEL;
    INVERSE
      definitions           : SET OF definition FOR defined_for
   ENTITY item;
      name                  : LABEL;
      documentation         : SET OF external_reference;
      description           : TEXT;
      ship_context          : OPTIONAL ship;
   ENTITY spacing_grid;
END_ENTITY;
```

```
ENTITY spacing_position;
      position_number          : OPTIONAL INTEGER;
      name                     : OPTIONAL STRING;
      location                 : LENGTH_MEASURE;
    INVERSE
      defining_table           : spacing_table FOR table_positions
END_ENTITY;

ENTITY spacing_table;
  ENTITY definition;
      id                       : global_id;
      [defined_for             : SET [1:?] OF definable_object;]  Redeclared in spacing_table
      local_units              : SET OF UNIT;
      version                  : OPTIONAL simple_definition_version;
  ENTITY spacing_table;
      definition.defined_for   : SET [1:1] OF spacing_grid;
      table_positions          : LIST [1:?] OF spacing_position;
      table_usage              : TABLE_TYPE;
      description              : OPTIONAL STRING;
      name                     : OPTIONAL STRING;
END_ENTITY;

ENTITY tank_compartment_design_definition;
  ENTITY definition;
      id                                : global_id;
      [defined_for                      : SET [1:?] OF definable_object;]  Redeclared in
                                            compartment_design_definition
      local_units                       : SET OF UNIT;
      version                           : OPTIONAL simple_definition_version;
  ENTITY design_definition;
      [representations                  : SET OF representation;]  Redeclared in
                                            compartment_design_definition
  ENTITY compartment_design_definition;
      design_requirements               : SET OF compartment_design_requirement;
      boundaries                        : SET OF MOULDED_FORM_BOUNDARY;
      extents                           : OPTIONAL compartment_extents;
      design_definition.representations : SET OF COMPARTMENT_SHAPE_REPRESENTATION;
      definition.defined_for            : SET [1:?] OF compartment;
  ENTITY tank_compartment_design_definition;
END_ENTITY;
```

```
ENTITY tank_compartment_functional_definition;
   ENTITY definition;
      id                        : global_id;
      [defined_for              : SET [1:?] OF definable_object;]  Redeclared in
                                        compartment_functional_definition
      local_units               : SET OF UNIT;
      version                   : OPTIONAL simple_definition_version;
   ENTITY functional_definition;
      user_def_function                 : OPTIONAL LABEL;
   ENTITY compartment_functional_definition;
      definition.defined_for    : SET [1:?] OF compartment;
      used_for                  : PRE_DEFINED_COMPARTMENT_FUNCTION;
   ENTITY tank_compartment_functional_definition;
      func                      : PRE_DEFINED_TANK_COMPARTMENT_FUNCTION;
END_ENTITY;

ENTITY tank_compartment_property_set;
   ENTITY definition;
      id                        : global_id;
      [defined_for              : SET [1:?] OF definable_object;]  Redeclared in
                                        compartment_property_set
      local_units               : SET OF UNIT;
      version                   : OPTIONAL simple_definition_version;
   ENTITY compartment_property_set;
      contxt                    : COMPARTMENT_PROPERTY_CONTEXT;
      definition.defined_for    : SET [1:?] OF compartment;
   ENTITY tank_compartment_property_set;
      moments_of_inertia        : moments_of_inertia;
      liquid_capacity           : SET OF capacity_properties;
      design_stowage_density    : DENSITY_MEASURE;
      geometric_parameters      : OPTIONAL tank_geometric_parameters;
      design_properties                 : OPTIONAL tank_piping_design_properties;
END_ENTITY;

ENTITY tank_geometric_parameters;
      length_wash               : OPTIONAL LENGTH_MEASURE;
      breadth_wash              : OPTIONAL LENGTH_MEASURE;
END_ENTITY;

ENTITY tank_piping_design_properties;
      sounding_pipe_height              : LENGTH_MEASURE;
      airpipe_height                    : LENGTH_MEASURE;
      relief_valve_pressure_setting     : OPTIONAL PRESSURE_MEASURE;
      filling_height                    : OPTIONAL LENGTH_MEASURE;
END_ENTITY;

ENTITY tolerance;
      minimum_point_spacing: LENGTH_MEASURE;
END_ENTITY;
```

```
ENTITY transverse_position;
   ENTITY spacing_position;
      position_number          : OPTIONAL INTEGER;
      name                     : OPTIONAL STRING;
      location                 : LENGTH_MEASURE;
    INVERSE
      defining_table           : spacing_table FOR table_positions
   ENTITY transverse_position;
      offset                   : OPTIONAL LENGTH_MEASURE;
END_ENTITY;

ENTITY vertical_position;
   ENTITY spacing_position;
      position_number          : OPTIONAL INTEGER;
      name                     : OPTIONAL STRING;
      location                 : LENGTH_MEASURE;
    INVERSE
      defining_table           : spacing_table FOR table_positions
   ENTITY vertical_position;
      offset                   : OPTIONAL LENGTH_MEASURE;
END_ENTITY;

ENTITY void_compartment_design_definition;
   ENTITY definition;
      id                                : global_id;
      [defined_for                      : SET [1:?] OF definable_object;]  Redeclared in
                                            compartment_design_definition
      local_units                       : SET OF UNIT;
      version                           : OPTIONAL simple_definition_version;
   ENTITY design_definition;
      [representations                  : SET OF representation;]  Redeclared in
                                            compartment_design_definition
   ENTITY compartment_design_definition;
      design_requirements               : SET OF compartment_design_requirement;
      boundaries                        : SET OF MOULDED_FORM_BOUNDARY;
      extents                           : OPTIONAL compartment_extents;
      design_definition.representations : SET OF COMPARTMENT_SHAPE_REPRESENTATION;
      definition.defined_for            : SET [1:?] OF compartment;
   ENTITY void_compartment_design_definition;
END_ENTITY;
```

```
ENTITY void_compartment_functional_definition;
   ENTITY definition;
      id                                : global_id;
      [defined_for                      : SET [1:?] OF definable_object;]  Redeclared in
                                              compartment_functional_definition
      local_units                       : SET OF UNIT;
      version                           : OPTIONAL simple_definition_version;
   ENTITY functional_definition;
      user_def_function                       : OPTIONAL LABEL;
   ENTITY compartment_functional_definition;
      definition.defined_for            : SET [1:?] OF compartment;
      used_for                          : PRE_DEFINED_COMPARTMENT_FUNCTION;
   ENTITY void_compartment_functional_definition;
      func                              : PRE_DEFINED_VOID_COMPARTMENT_FUNCTION;
END_ENTITY;

ENTITY zone;
   ENTITY definable_object;
      id                : LABEL;
    INVERSE
      definitions       : SET OF definition FOR defined_for
   ENTITY item;
      name              : LABEL;
      documentation     : SET OF external_reference;
      description       : TEXT;
      ship_context      : OPTIONAL ship;
   ENTITY space;
   ENTITY zone;
END_ENTITY;

ENTITY zone_design_definition;
   ENTITY definition;
      id                                : global_id;
      [defined_for                      : SET [1:?] OF definable_object;]  Redeclared in
                                              zone_design_definition
      local_units                       : SET OF UNIT;
      version                           : OPTIONAL simple_definition_version;
   ENTITY design_definition;
      [representations                  : SET OF representation;]  Redeclared in
                                              zone_design_definition
   ENTITY zone_design_definition;
      governed_by_design_requirement          : compartment_design_requirement;
      boundaries                        : OPTIONAL SET OF MOULDED_FORM_BOUNDARY;
      constituent_compartments                : SET OF compartment_design_definition;
      design_definition.representations: SET OF COMPARTMENT_SHAPE_REPRESENTATION;
      definition.defined_for            : SET [1:?] OF zone;
END_ENTITY;
```

```
ENTITY zone_functional_definition;
  ENTITY definition;
    id                      : global_id;
    [defined_for            : SET [1:?] OF definable_object;]  Redeclared in
                                  zone_functional_definition
    local_units             : SET OF UNIT;
    version                 : OPTIONAL simple_definition_version;
  ENTITY functional_definition;
    user_def_function               : OPTIONAL LABEL;
  ENTITY zone_functional_definition;
    definition.defined_for  : SET [1:?] OF zone;
    used_for                : PRE_DEFINED_ZONE_FUNCTION;
END_ENTITY;


----------INSTANTIABLE "UNITS" ENTITIES

ENTITY area_unit+extended_conversion_based_unit;
  ENTITY named_unit;
    dimensions                      : dimensional_exponents;
  ENTITY area_unit;
  ENTITY extended_conversion_based_unit;
    name                            : LABEL;
    conversion_factor                       : extended_measure_with_unit;
  ENTITY area_unit+extended_conversion_based_unit;
END_ENTITY;

ENTITY area_unit+si_unit;
  ENTITY named_unit;
    [dimensions                     : dimensional_exponents;]  Redeclared in si_unit
  ENTITY area_unit;
  ENTITY si_unit;
    prefix                          : OPTIONAL SI_PREFIX;
    name                            : SI_UNIT_NAME;
   DERIVE
    named_unit.dimensions           : dimensional_exponents;
  ENTITY area_unit+si_unit;
END_ENTITY;

ENTITY derived_unit;
    elements                        : SET [1:?] OF derived_unit_element;
END_ENTITY;

ENTITY derived_unit_element;
    unit                    : named_unit;
    exponent                        : REAL;
END_ENTITY;
```

```
ENTITY dimensional_exponents;
    length_exponent                     : REAL;
    mass_exponent                       : REAL;
    time_exponent                       : REAL;
    electric_current_exponent           : REAL;
    thermodynamic_temperature_exponent  : REAL;
    amount_of_substance_exponent        : REAL;
    luminous_intensity_exponent         : REAL;
END_ENTITY;

ENTITY extended_conversion_based_unit;
  ENTITY named_unit;
    dimensions              : dimensional_exponents;
  ENTITY extended_conversion_based_unit;
    name                    : LABEL;
    conversion_factor               : extended_measure_with_unit;
END_ENTITY;

ENTITY extended_conversion_based_unit+length_unit;
  ENTITY named_unit;
    dimensions                      : dimensional_exponents;
  ENTITY extended_conversion_based_unit;
    name                            : LABEL;
    conversion_factor                       : extended_measure_with_unit;
  ENTITY length_unit;
  ENTITY extended_conversion_based_unit+length_unit;
END_ENTITY;

ENTITY extended_conversion_based_unit+plane_angle_unit;
  ENTITY named_unit;
    dimensions                      : dimensional_exponents;
  ENTITY extended_conversion_based_unit;
    name                            : LABEL;
    conversion_factor                       : extended_measure_with_unit;
  ENTITY plane_angle_unit;
  ENTITY extended_conversion_based_unit+plane_angle_unit;
END_ENTITY;

ENTITY extended_conversion_based_unit+volume_unit;
  ENTITY named_unit;
    dimensions                      : dimensional_exponents;
  ENTITY extended_conversion_based_unit;
    name                            : LABEL;
    conversion_factor                       : extended_measure_with_unit;
  ENTITY volume_unit;
  ENTITY extended_conversion_based_unit+volume_unit;
END_ENTITY;
```

```
ENTITY extended_measure_with_unit;
    value_componen              : EXTENDED_MEASURE_VALUE;
    unit_component          : UNIT;
END_ENTITY;

ENTITY length_unit+si_unit;
  ENTITY named_unit;
    [dimensions                 : dimensional_exponents;]  Redeclared in si_unit
  ENTITY length_unit;
  ENTITY si_unit;
    prefix                      : OPTIONAL SI_PREFIX;
    name                        : SI_UNIT_NAME;
   DERIVE
    named_unit.dimensions       : dimensional_exponents;
  ENTITY length_unit+si_unit;
END_ENTITY;

ENTITY plane_angle_unit+si_unit;
  ENTITY named_unit;
    [dimensions                 : dimensional_exponents;]  Redeclared in si_unit
  ENTITY plane_angle_unit;
  ENTITY si_unit;
    prefix                      : OPTIONAL SI_PREFIX;
    name                        : SI_UNIT_NAME;
   DERIVE
    named_unit.dimensions       : dimensional_exponents;
  ENTITY plane_angle_unit+si_unit;
END_ENTITY;

ENTITY si_unit;
  ENTITY named_unit;
    [dimensions             : dimensional_exponents;]  Redeclared in si_unit
  ENTITY si_unit;
    prefix                  : OPTIONAL SI_PREFIX;
    name                    : SI_UNIT_NAME;
   DERIVE
    named_unit.dimensions   : dimensional_exponents;
END_ENTITY;

ENTITY si_unit+volume_unit;
  ENTITY named_unit;
    [dimensions                 : dimensional_exponents;]  Redeclared in si_unit
  ENTITY si_unit;
    prefix                      : OPTIONAL SI_PREFIX;
    name                        : SI_UNIT_NAME;
   DERIVE
    named_unit.dimensions       : dimensional_exponents;
  ENTITY volume_unit;
  ENTITY si_unit+volume_unit;
END_ENTITY;
```

----------INSTANTIABLE “SUPPORT” ENTITIES

```
ENTITY address;
    internal_location       : OPTIONAL LABEL;
    street_number           : OPTIONAL LABEL;
    street                  : OPTIONAL LABEL;
    postal_box              : OPTIONAL LABEL;
    town                    : OPTIONAL LABEL;
    region                  : OPTIONAL LABEL;
    postal_code             : OPTIONAL LABEL;
    country                 : OPTIONAL LABEL;
    facsimile_number            : OPTIONAL LABEL;
    telephone_number            : OPTIONAL LABEL;
    electronic_mail_address : OPTIONAL LABEL;
    telex_number            : OPTIONAL LABEL;
END_ENTITY;

ENTITY calendar_date;
  ENTITY date;
    year_component          : YEAR_NUMBER;
  ENTITY calendar_date;
    day_component           : DAY_IN_MONTH_NUMBER;
    month_component             : MONTH_IN_YEAR_NUMBER;
END_ENTITY;

ENTITY coordinated_universal_time_offset;
    hour_offset             : HOUR_IN_DAY;
    minute_offset           : OPTIONAL MINUTE_IN_HOUR;
    sense                   : AHEAD_OR_BEHIND;
END_ENTITY;

ENTITY date;
    year_component          : YEAR_NUMBER;
END_ENTITY;

ENTITY date_and_time;
    date_component          : date;
    time_component          : local_time;
END_ENTITY;

ENTITY document;
    has_author              : ANY_AUTHOR;
END_ENTITY;
```

```
ENTITY document_reference;
   ENTITY external_reference;
      location                : ANY_ADDRESS;
      description             : TEXT;
   ENTITY document;
      has_author              : ANY_AUTHOR;
   ENTITY document_reference;
END_ENTITY;

ENTITY external_reference;
      location                : ANY_ADDRESS;
      description             : TEXT;
END_ENTITY;

ENTITY local_time;
      hour_component          : HOUR_IN_DAY;
      minute_component              : OPTIONAL MINUTE_IN_HOUR;
      second_component              : OPTIONAL SECOND_IN_MINUTE;
      zone                    : coordinated_universal_time_offset;
END_ENTITY;

ENTITY organisation;
   ENTITY organization;
      id                      : OPTIONAL IDENTIFIER;
      name                    : LABEL;
      description             : TEXT;
   ENTITY organisation;
END_ENTITY;

ENTITY person;
      id                      : IDENTIFIER;
      last_name                     : OPTIONAL LABEL;
      first_name              : OPTIONAL LABEL;
      middle_names            : OPTIONAL LIST [1:?] OF LABEL;
      prefix_titles           : OPTIONAL LIST [1:?] OF LABEL;
      suffix_titles           : OPTIONAL LIST [1:?] OF LABEL;
END_ENTITY;

ENTITY person_and_organisation;
   ENTITY person_and_organization;
      the_person              : person;
      the_organization        : organization;
   ENTITY person_and_organisation;
END_ENTITY;
```

```
ENTITY universal_resource_locator;
      protocol                : PROTOCOL_TYPE;
      other_protocol_type     : OPTIONAL LABEL;
      machine_adress          : IDENTIFIER;
      port                    : OPTIONAL INTEGER;
      location                : IDENTIFIER;
END_ENTITY;

------------------INSTANTIABLE “GEOMETRY” ENTITIES

ENTITY advanced_face;
  ENTITY representation_item;
    name                      : LABEL;
  ENTITY topological_representation_item;
  ENTITY face;
    bounds                    : SET [1:?] OF face_bound;
  ENTITY geometric_representation_item;
   DERIVE
    dim                       : DIMENSION_COUNT;
  ENTITY face_surface;
    face_geometry             : surface;
    same_sense                : BOOLEAN;
  ENTITY advanced_face;
END_ENTITY;

ENTITY axis2_placement_2d;
  ENTITY representation_item;
    name                      : LABEL;
  ENTITY geometric_representation_item;
   DERIVE
    dim                       : DIMENSION_COUNT;
  ENTITY placement;
    location                  : cartesian_point;
  ENTITY axis2_placement_2d;
    ref_direction             : OPTIONAL direction;
   DERIVE
    p                                 : LIST [2:2] OF direction;
END_ENTITY;
```

```
ENTITY axis2_placement_3d;
   ENTITY representation_item;
      name                    : LABEL;
   ENTITY geometric_representation_item;
    DERIVE
      dim                     : DIMENSION_COUNT;
   ENTITY placement;
      location                : cartesian_point;
   ENTITY axis2_placement_3d;
      axis                    : OPTIONAL direction;
      ref_direction           : OPTIONAL direction;
    DERIVE
      p                              : LIST [3:3] OF direction;
END_ENTITY;

ENTITY bounded_curve+conic;
   ENTITY representation_item;
      name                     : LABEL;
   ENTITY geometric_representation_item;
    DERIVE
      dim                     : DIMENSION_COUNT;
   ENTITY curve;
   ENTITY bounded_curve;
   ENTITY conic;
      position                 : AXIS2_PLACEMENT;
   ENTITY bounded_curve+conic;
END_ENTITY;


ENTITY bounded_curve+line;
   ENTITY representation_item;
      name                    : LABEL;
   ENTITY geometric_representation_item;
    DERIVE
      dim                     : DIMENSION_COUNT;
   ENTITY curve;
   ENTITY bounded_curve;
   ENTITY line;
      pnt                     : cartesian_point;
      dir                     : vector;
   ENTITY bounded_curve+line;
END_ENTITY;
```

```
ENTITY bounded_curve+pcurve;
   ENTITY representation_item;
      name                    : LABEL;
   ENTITY geometric_representation_item;
    DERIVE
      dim                     : DIMENSION_COUNT;
   ENTITY curve;
   ENTITY bounded_curve;
   ENTITY pcurve;
      basis_surface           : surface;
      reference_to_curve              : definitional_representation;
   ENTITY bounded_curve+pcurve;
END_ENTITY;

ENTITY bounded_curve+surface_curve;
   ENTITY representation_item;
      name                    : LABEL;
   ENTITY geometric_representation_item;
    DERIVE
      dim                     : DIMENSION_COUNT;
   ENTITY curve;
   ENTITY bounded_curve;
   ENTITY surface_curve;
      curve_3d                        : curve;
      associated_geometry     : LIST [1 : 2] OF PCURVE_OR_SURFACE;
      master_representation   : PREFERRED_SURFACE_CURVE_REPRESENTATION;
    DERIVE
      basis_surface           : SET [1 : 2] OF surface;
   ENTITY bounded_curve+surface_curve;
END_ENTITY;

ENTITY bounded_pcurve;
   ENTITY representation_item;
      name                    : LABEL;
   ENTITY geometric_representation_item;
    DERIVE
      dim                     : DIMENSION_COUNT;
   ENTITY curve;
   ENTITY pcurve;
      basis_surface           : surface;
      reference_to_curve              : definitional_representation;
   ENTITY bounded_curve;
   ENTITY bounded_pcurve;
END_ENTITY;
```

```
ENTITY bounded_surface_curve;
   ENTITY representation_item;
      name                    : LABEL;
   ENTITY geometric_representation_item;
    DERIVE
      dim                     : DIMENSION_COUNT;
   ENTITY curve;
   ENTITY surface_curve;
      curve_3d                        : curve;
      associated_geometry     : LIST [1:2] OF PCURVE_OR_SURFACE;
      master_representation   : PREFERRED_SURFACE_CURVE_REPRESENTATION;
    DERIVE
      basis_surface           : SET [1:2] OF surface;
   ENTITY bounded_curve;
   ENTITY bounded_surface_curve;
END_ENTITY;

ENTITY bounded_surface_curve+intersection_curve;
   ENTITY representation_item;
      name                    : LABEL;
   ENTITY geometric_representation_item;
    DERIVE
      dim                     : DIMENSION_COUNT;
   ENTITY curve;
   ENTITY surface_curve;
      curve_3d                        : curve;
      associated_geometry     : LIST [1 : 2] OF PCURVE_OR_SURFACE;
      master_representation   : PREFERRED_SURFACE_CURVE_REPRESENTATION;
    DERIVE
      basis_surface           : SET [1 : 2] OF surface;
   ENTITY bounded_curve;
   ENTITY bounded_surface_curve;
   ENTITY intersection_curve;
   ENTITY bounded_surface_curve+intersection_curve;
END_ENTITY;
```

```
ENTITY bounded_surface_curve+seam_curve;
   ENTITY representation_item;
      name                    : LABEL;
   ENTITY geometric_representation_item;
    DERIVE
      dim                     : DIMENSION_COUNT;
   ENTITY curve;
   ENTITY surface_curve;
      curve_3d                      : curve;
      associated_geometry     : LIST [1 : 2] OF PCURVE_OR_SURFACE;
      master_representation   : PREFERRED_SURFACE_CURVE_REPRESENTATION;
    DERIVE
      basis_surface           : SET [1 : 2] OF surface;
   ENTITY bounded_curve;
   ENTITY bounded_surface_curve;
   ENTITY seam_curve;
   ENTITY bounded_surface_curve+seam_curve;
END_ENTITY;

ENTITY b_spline_curve_with_knots;
   ENTITY representation_item;
      name                    : LABEL;
   ENTITY geometric_representation_item;
    DERIVE
      dim                     : DIMENSION_COUNT;
   ENTITY curve;
   ENTITY bounded_curve;
   ENTITY b_spline_curve;
      degree                  : INTEGER;
      control_points_list           : LIST [2:?] OF cartesian_point;
      curve_form              : B_SPLINE_CURVE_FORM;
      closed_curve            : LOGICAL;
      self_intersect          : LOGICAL;
    DERIVE
      upper_index_on_control_points  : INTEGER;
      control_points                 : ARRAY OF cartesian_point;
   ENTITY b_spline_curve_with_knots;
      knot_multiplicities                   : LIST [2:?] OF INTEGER;
      knots                          : LIST [2:?] OF PARAMETER_VALUE;
      knot_spec                             : KNOT_TYPE;
    DERIVE
      upper_index_on_knots           : INTEGER;
END_ENTITY;
```

```
ENTITY b_spline_curve_with_knots+rational_b_spline_curve;
  ENTITY representation_item;
    name                          : LABEL;
  ENTITY geometric_representation_item;
   DERIVE
    dim                           : DIMENSION_COUNT;
  ENTITY curve;
  ENTITY bounded_curve;
  ENTITY b_spline_curve;
    degree                        : INTEGER;
    control_points_list                   : LIST [2 : ?] OF cartesian_point;
    curve_form                    : B_SPLINE_CURVE_FORM;
    closed_curve                  : LOGICAL;
    self_intersect                : LOGICAL;
   DERIVE
    upper_index_on_control_points : INTEGER;
    control_points                : ARRAY OF cartesian_point;
  ENTITY b_spline_curve_with_knots;
    knot_multiplicities                   : LIST [2 : ?] OF INTEGER;
    knots                         : LIST [2 : ?] OF PARAMETER_VALUE;
    knot_spec                             : KNOT_TYPE;
   DERIVE
    upper_index_on_knots          : INTEGER;
  ENTITY rational_b_spline_curve;
    weights_data                  : LIST [2 : ?] OF REAL;
   DERIVE
    weights                       : ARRAY OF REAL;
  ENTITY b_spline_curve_with_knots+rational_b_spline_curve;
END_ENTITY;
```

```
ENTITY b_spline_surface_with_knots;
   ENTITY representation_item;
      name                    : LABEL;
   ENTITY geometric_representation_item;
    DERIVE
      dim                     : DIMENSION_COUNT;
   ENTITY surface;
   ENTITY bounded_surface;
   ENTITY b_spline_surface;
      u_degree                      : INTEGER;
      v_degree                      : INTEGER;
      control_points_list           : LIST [2:?] OF LIST [2:?] OF cartesian_point;
      surface_form            : B_SPLINE_SURFACE_FORM;
      u_closed                      : LOGICAL;
      v_closed                      : LOGICAL;
      self_intersect          : LOGICAL;
    DERIVE
      u_upper                 : INTEGER;
      v_upper                 : INTEGER;
      control_points          : ARRAY OF ARRAY OF cartesian_point;
   ENTITY b_spline_surface_with_knots;
      u_multiplicities        : LIST [2:?] OF INTEGER;
      v_multiplicities        : LIST [2:?] OF INTEGER;
      u_knots                 : LIST [2:?] OF PARAMETER_VALUE;
      v_knots                 : LIST [2:?] OF PARAMETER_VALUE;
      knot_spec                     : KNOT_TYPE;
    DERIVE
      knot_u_upper            : INTEGER;
      knot_v_upper            : INTEGER;
END_ENTITY;
```

```
ENTITY b_spline_surface_with_knots+rational_b_spline_surface;
  ENTITY representation_item;
    name                    : LABEL;
  ENTITY geometric_representation_item;
   DERIVE
    dim                     : DIMENSION_COUNT;
  ENTITY surface;
  ENTITY bounded_surface;
  ENTITY b_spline_surface;
    u_degree                        : INTEGER;
    v_degree                        : INTEGER;
    control_points_list             : LIST [2 : ?] OF LIST [2 : ?] OF cartesian_point;
    surface_form            : B_SPLINE_SURFACE_FORM;
    u_closed                        : LOGICAL;
    v_closed                        : LOGICAL;
    self_intersect          : LOGICAL;
   DERIVE
    u_upper                 : INTEGER;
    v_upper                 : INTEGER;
    control_points          : ARRAY OF ARRAY OF cartesian_point;
  ENTITY b_spline_surface_with_knots;
    u_multiplicities        : LIST [2 : ?] OF INTEGER;
    v_multiplicities        : LIST [2 : ?] OF INTEGER;
    u_knots                 : LIST [2 : ?] OF PARAMETER_VALUE;
    v_knots                 : LIST [2 : ?] OF PARAMETER_VALUE;
    knot_spec                       : KNOT_TYPE;
   DERIVE
    knot_u_upper            : INTEGER;
    knot_v_upper            : INTEGER;
  ENTITY rational_b_spline_surface;
    weights_data            : LIST [2 : ?] OF LIST [2 : ?] OF REAL;
   DERIVE
    weights                 : ARRAY OF ARRAY OF REAL;
  ENTITY b_spline_surface_with_knots+rational_b_spline_surface;
END_ENTITY;

ENTITY cartesian_point;
  ENTITY representation_item;
    name                    : LABEL;
  ENTITY geometric_representation_item;
   DERIVE
    dim                     : DIMENSION_COUNT;
  ENTITY point;
  ENTITY cartesian_point;
    coordinates             : LIST [1:3] OF LENGTH_MEASURE;
END_ENTITY;
```

```
ENTITY circle;
   ENTITY representation_item;
      name                    : LABEL;
   ENTITY geometric_representation_item;
    DERIVE
      dim                     : DIMENSION_COUNT;
   ENTITY curve;
   ENTITY conic;
      position                : AXIS2_PLACEMENT;
   ENTITY circle;
      radius                  : POSITIVE_LENGTH_MEASURE;
END_ENTITY;

ENTITY closed_shell;
   ENTITY representation_item;
      name                    : LABEL;
   ENTITY topological_representation_item;
   ENTITY connected_face_set;
      cfs_faces                       : SET [1:?] OF face;
   ENTITY closed_shell;
END_ENTITY;

ENTITY composite_curve;
   ENTITY representation_item;
      name                    : LABEL;
   ENTITY geometric_representation_item;
    DERIVE
      dim                     : DIMENSION_COUNT;
   ENTITY curve;
   ENTITY bounded_curve;
   ENTITY composite_curve;
      segments                        : LIST [1:?] OF composite_curve_segment;
      self_intersect          : LOGICAL;
    DERIVE
      n_segments              : INTEGER;
      closed_curve            : LOGICAL;
END_ENTITY;
```

```
ENTITY composite_curve_on_surface;
   ENTITY representation_item;
      name                    : LABEL;
   ENTITY geometric_representation_item;
    DERIVE
      dim                     : DIMENSION_COUNT;
   ENTITY curve;
   ENTITY bounded_curve;
   ENTITY composite_curve;
      segments                        : LIST [1:?] OF composite_curve_segment;
      self_intersect          : LOGICAL;
    DERIVE
      n_segments              : INTEGER;
      closed_curve            : LOGICAL;
   ENTITY composite_curve_on_surface;
    DERIVE
      basis_surface           : SET [0:2] OF surface;
END_ENTITY;

ENTITY composite_curve_segment;
      transition                      : TRANSITION_CODE;
      same_sense              : BOOLEAN;
      parent_curve            : curve;
    INVERSE
      using_curves            : BAG [1:?] OF composite_curve FOR segments
END_ENTITY;

ENTITY definitional_representation;
   ENTITY representation;
      name                    : LABEL;
      items                   : SET [1:?] OF representation_item;
      context_of_items                : representation_context;
   ENTITY definitional_representation;
END_ENTITY;

ENTITY definitional_representation+shape_representation;
   ENTITY representation;
      name                    : LABEL;
      items                   : SET [1 : ?] OF representation_item;
      context_of_items                : representation_context;
   ENTITY definitional_representation;
   ENTITY shape_representation;
   ENTITY definitional_representation+shape_representation;
END_ENTITY;
```

```
ENTITY direction;
   ENTITY representation_item;
      name                    : LABEL;
   ENTITY geometric_representation_item;
    DERIVE
      dim                     : DIMENSION_COUNT;
   ENTITY direction;
      direction_ratios        : LIST [2:3] OF REAL;
END_ENTITY;

ENTITY edge_curve;
   ENTITY representation_item;
      name                    : LABEL;
   ENTITY topological_representation_item;
   ENTITY edge;
      edge_start                    : vertex;
      edge_end                      : vertex;
   ENTITY geometric_representation_item;
    DERIVE
      dim                     : DIMENSION_COUNT;
   ENTITY edge_curve;
      edge_geometry           : curve;
      same_sense              : BOOLEAN;
END_ENTITY;

ENTITY edge_loop;
   ENTITY representation_item;
      name                    : LABEL;
   ENTITY topological_representation_item;
   ENTITY loop;
   ENTITY path;
      edge_list                     : LIST [1:?] OF UNIQUE oriented_edge;
   ENTITY edge_loop;
    DERIVE
      ne                      : INTEGER;
END_ENTITY;

ENTITY ellipse;
   ENTITY representation_item;
      name                    : LABEL;
   ENTITY geometric_representation_item;
    DERIVE
      dim                     : DIMENSION_COUNT;
   ENTITY curve;
   ENTITY conic;
      position                : AXIS2_PLACEMENT;
   ENTITY ellipse;
      semi_axis_1             : POSITIVE_LENGTH_MEASURE;
      semi_axis_2             : POSITIVE_LENGTH_MEASURE;
END_ENTITY;
```

```
ENTITY face_based_surface_model;
   ENTITY representation_item;
      name                  : LABEL;
   ENTITY geometric_representation_item;
    DERIVE
      dim                   : DIMENSION_COUNT;
   ENTITY face_based_surface_model;
      fbsm_faces            : SET [1:?] OF connected_face_set;
END_ENTITY;

ENTITY face_bound;
   ENTITY representation_item;
      name                  : LABEL;
   ENTITY topological_representation_item;
   ENTITY face_bound;
      bound                 : loop;
      orientation           : BOOLEAN;
END_ENTITY;

ENTITY face_outer_bound;
   ENTITY representation_item;
      name                  : LABEL;
   ENTITY topological_representation_item;
   ENTITY face_bound;
      bound                 : loop;
      orientation           : BOOLEAN;
   ENTITY face_outer_bound;
END_ENTITY;

ENTITY face_surface;
   ENTITY representation_item;
      name                  : LABEL;
   ENTITY topological_representation_item;
   ENTITY face;
      bounds                : SET [1:?] OF face_bound;
   ENTITY geometric_representation_item;
    DERIVE
      dim                   : DIMENSION_COUNT;
   ENTITY face_surface;
      face_geometry         : surface;
      same_sense            : BOOLEAN;
END_ENTITY;
```

```
ENTITY geometric_representation_context;
   ENTITY representation_context;
      context_identifier                : IDENTIFIER;
      context_type                      : TEXT;
    INVERSE
      representations_in_context        : SET [1:?] OF representation FOR context_of_items
   ENTITY geometric_representation_context;
      coordinate_space_dimension        : DIMENSION_COUNT;
END_ENTITY;

ENTITY intersection_curve;
   ENTITY representation_item;
      name                              : LABEL;
   ENTITY geometric_representation_item;
    DERIVE
      dim                               : DIMENSION_COUNT;
   ENTITY curve;
   ENTITY surface_curve;
      curve_3d                          : curve;
      associated_geometry               : LIST [1:2] OF PCURVE_OR_SURFACE;
      master_representation             : PREFERRED_SURFACE_CURVE_REPRESENTATION;
    DERIVE
      basis_surface                     : SET [1:2] OF surface;
   ENTITY intersection_curve;
END_ENTITY;

ENTITY line;
   ENTITY representation_item;
      name                              : LABEL;
   ENTITY geometric_representation_item;
    DERIVE
      dim                               : DIMENSION_COUNT;
   ENTITY curve;
   ENTITY line;
      pnt                               : cartesian_point;
      dir                               : vector;
END_ENTITY;

ENTITY loop+path;
   ENTITY representation_item;
      name                              : LABEL;
   ENTITY topological_representation_item;
   ENTITY loop;
   ENTITY path;
      edge_list                         : LIST [1 : ?] OF UNIQUE oriented_edge;
   ENTITY loop+path;
END_ENTITY;
```

```
ENTITY non_manifold_surface_shape_representation;
   ENTITY representation;
      name                          : LABEL;
      items                         : SET [1:?] OF representation_item;
      context_of_items                    : representation_context;
   ENTITY shape_representation;
   ENTITY non_manifold_surface_shape_representation;
END_ENTITY;

ENTITY open_shell;
   ENTITY representation_item;
      name                    : LABEL;
   ENTITY topological_representation_item;
   ENTITY connected_face_set;
      cfs_faces                     : SET [1:?] OF face;
   ENTITY open_shell;
END_ENTITY;

ENTITY oriented_closed_shell;
   ENTITY representation_item;
      name                    : LABEL;
   ENTITY topological_representation_item;
   ENTITY connected_face_set;
      [cfs_faces                    : SET [1:?] OF face;]  Redeclared in oriented_closed_shell
   ENTITY closed_shell;
   ENTITY oriented_closed_shell;
      closed_shell_element    : closed_shell;
      orientation             : BOOLEAN;
    DERIVE
      connected_face_set.cfs_faces    : SET [1:?] OF face;
END_ENTITY;

ENTITY oriented_edge;
   ENTITY representation_item;
      name                    : LABEL;
   ENTITY topological_representation_item;
   ENTITY edge;
      [edge_start             : vertex;]  Redeclared in oriented_edge
      [edge_end                     : vertex;]  Redeclared in oriented_edge
   ENTITY oriented_edge;
      edge_element            : edge;
      orientation             : BOOLEAN;
    DERIVE
      edge.edge_start         : vertex;
      edge.edge_end           : vertex;
END_ENTITY;
```

```
ENTITY oriented_face;
  ENTITY representation_item;
    name                    : LABEL;
  ENTITY topological_representation_item;
  ENTITY face;
    [bounds                 : SET [1:?] OF face_bound;]  Redeclared in oriented_face
  ENTITY oriented_face;
    face_element            : face;
    orientation             : BOOLEAN;
   DERIVE
    face.bounds             : SET [1:?] OF face_bound;
END_ENTITY;

ENTITY oriented_open_shell;
  ENTITY representation_item;
    name                    : LABEL;
  ENTITY topological_representation_item;
  ENTITY connected_face_set;
    [cfs_faces                      : SET [1:?] OF face;]  Redeclared in oriented_open_shell
  ENTITY open_shell;
  ENTITY oriented_open_shell;
    open_shell_element      : open_shell;
    orientation             : BOOLEAN;
   DERIVE
    connected_face_set.cfs_faces    : SET [1:?] OF face;
END_ENTITY;

ENTITY oriented_path;
  ENTITY representation_item;
    name                    : LABEL;
  ENTITY topological_representation_item;
  ENTITY path;
    [edge_list                      : LIST [1:?] OF UNIQUE oriented_edge;]  Redeclared in
oriented_path
  ENTITY oriented_path;
    path_element            : path;
    orientation             : BOOLEAN;
   DERIVE
    path.edge_list          : LIST [1:?] OF UNIQUE oriented_edge;
END_ENTITY;

ENTITY path;
  ENTITY representation_item;
    name                    : LABEL;
  ENTITY topological_representation_item;
  ENTITY path;
    edge_list                       : LIST [1:?] OF UNIQUE oriented_edge;
END_ENTITY;
```

```
ENTITY pcurve;
   ENTITY representation_item;
      name                    : LABEL;
   ENTITY geometric_representation_item;
    DERIVE
      dim                     : DIMENSION_COUNT;
   ENTITY curve;
   ENTITY pcurve;
      basis_surface           : surface;
      reference_to_curve              : definitional_representation;
END_ENTITY;

ENTITY plane;
   ENTITY representation_item;
      name                    : LABEL;
   ENTITY geometric_representation_item;
    DERIVE
      dim                     : DIMENSION_COUNT;
   ENTITY surface;
   ENTITY elementary_surface;
      position                : axis2_placement_3d;
   ENTITY plane;
END_ENTITY;

ENTITY polyline;
   ENTITY representation_item;
      name                    : LABEL;
   ENTITY geometric_representation_item;
    DERIVE
      dim                     : DIMENSION_COUNT;
   ENTITY curve;
   ENTITY bounded_curve;
   ENTITY polyline;
      points                  : LIST [2:?] OF cartesian_point;
END_ENTITY;
```

```
ENTITY seam_curve;
   ENTITY representation_item;
      name                    : LABEL;
   ENTITY geometric_representation_item;
    DERIVE
      dim                     : DIMENSION_COUNT;
   ENTITY curve;
   ENTITY surface_curve;
      curve_3d                        : curve;
      associated_geometry     : LIST [1:2] OF PCURVE_OR_SURFACE;
      master_representation   : PREFERRED_SURFACE_CURVE_REPRESENTATION;
    DERIVE
      basis_surface           : SET [1:2] OF surface;
   ENTITY seam_curve;
END_ENTITY;

ENTITY spherical_surface;
   ENTITY representation_item;
      name                    : LABEL;
   ENTITY geometric_representation_item;
    DERIVE
      dim                     : DIMENSION_COUNT;
   ENTITY surface;
   ENTITY elementary_surface;
      position                : axis2_placement_3d;
   ENTITY spherical_surface;
      radius                  : POSITIVE_LENGTH_MEASURE;
END_ENTITY;

ENTITY surface_curve;
   ENTITY representation_item;
      name                    : LABEL;
   ENTITY geometric_representation_item;
    DERIVE
      dim                     : DIMENSION_COUNT;
   ENTITY curve;
   ENTITY surface_curve;
      curve_3d                        : curve;
      associated_geometry     : LIST [1:2] OF PCURVE_OR_SURFACE;
      master_representation   : PREFERRED_SURFACE_CURVE_REPRESENTATION;
    DERIVE
      basis_surface           : SET [1:2] OF surface;
END_ENTITY;
```

```
ENTITY vector;
   ENTITY representation_item;
      name                    : LABEL;
   ENTITY geometric_representation_item;
    DERIVE
      dim                     : DIMENSION_COUNT;
   ENTITY vector;
      orientation             : direction;
      magnitude               : LENGTH_MEASURE;
END_ENTITY;

ENTITY vertex;
   ENTITY representation_item;
      name                    : LABEL;
   ENTITY topological_representation_item;
   ENTITY vertex;
END_ENTITY;

 ENTITY vertex_loop;
   ENTITY representation_item;
      name                    : LABEL;
   ENTITY topological_representation_item;
   ENTITY loop;
   ENTITY vertex_loop;
      loop_vertex             : vertex;
END_ENTITY;

ENTITY vertex_point;
   ENTITY representation_item;
      name                    : LABEL;
   ENTITY topological_representation_item;
   ENTITY vertex;
   ENTITY geometric_representation_item;
    DERIVE
      dim                     : DIMENSION_COUNT;
   ENTITY vertex_point;
      vertex_geometry         : point;
END_ENTITY;
```

A.4.3 Express-I Diagram

This Express-I diagram is a graphical representation of the AP215-2-1-INGR.stp product model. It shows the relationships of the model entity attributes to other entities in the model.

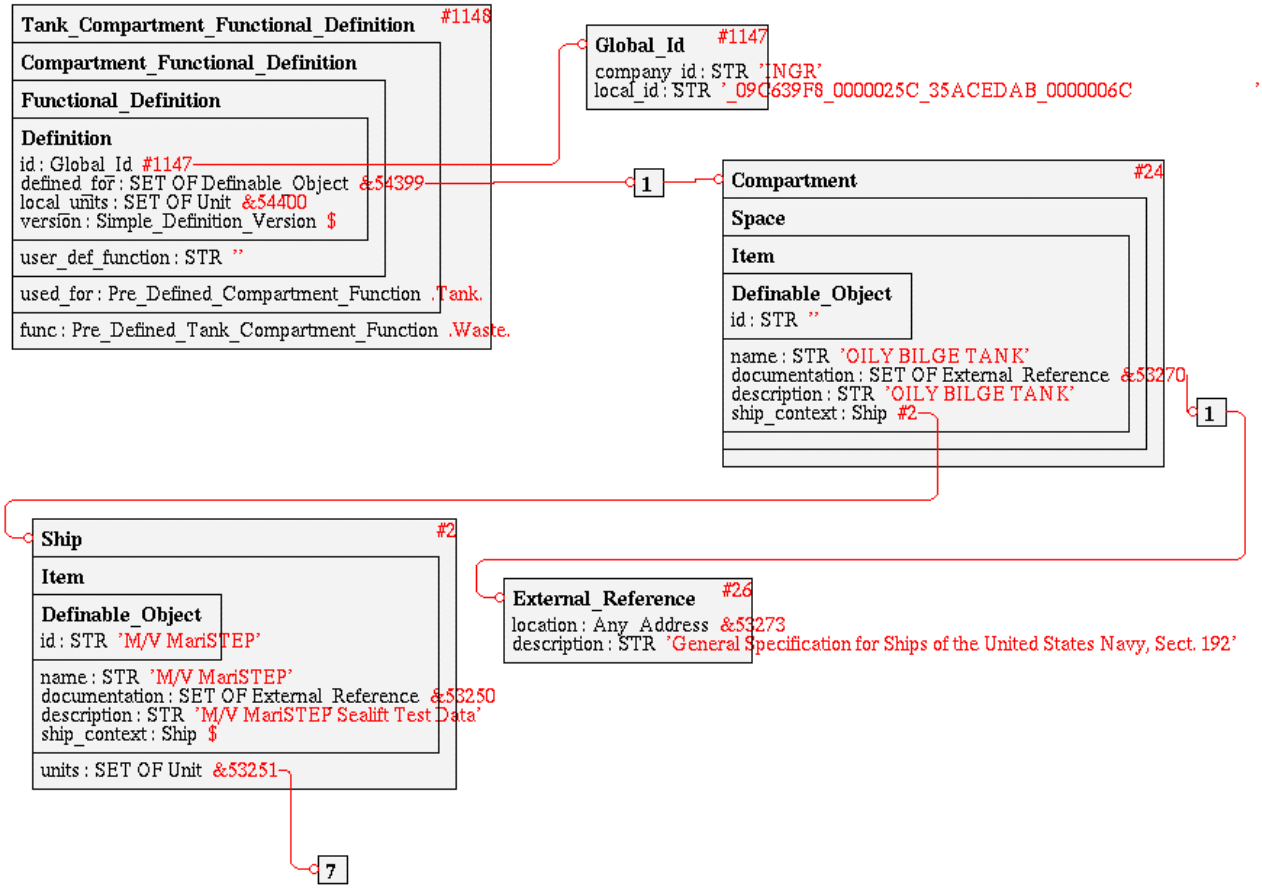


Figure A-42 - Compartment

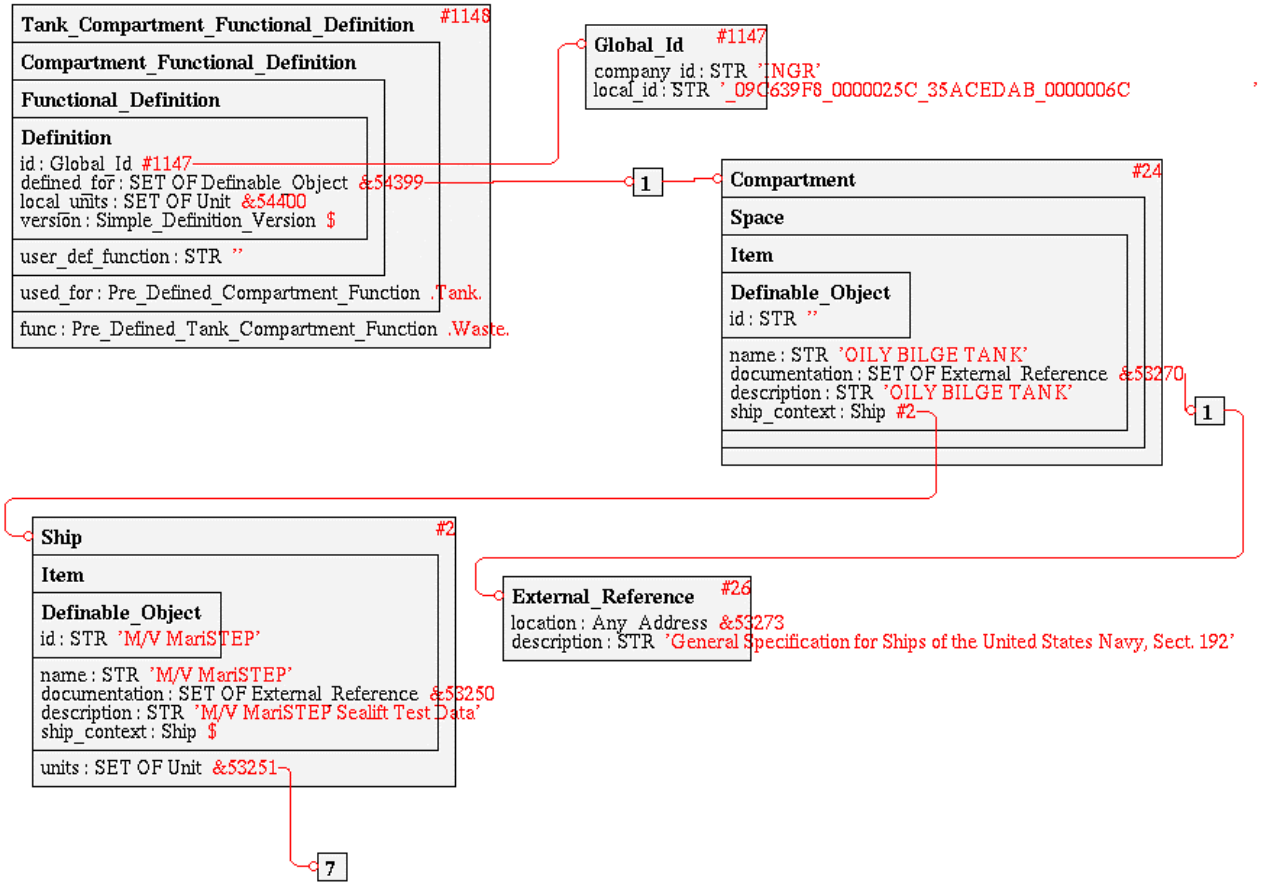


Figure A-43 - Tank_Compartment_Functional_Definition

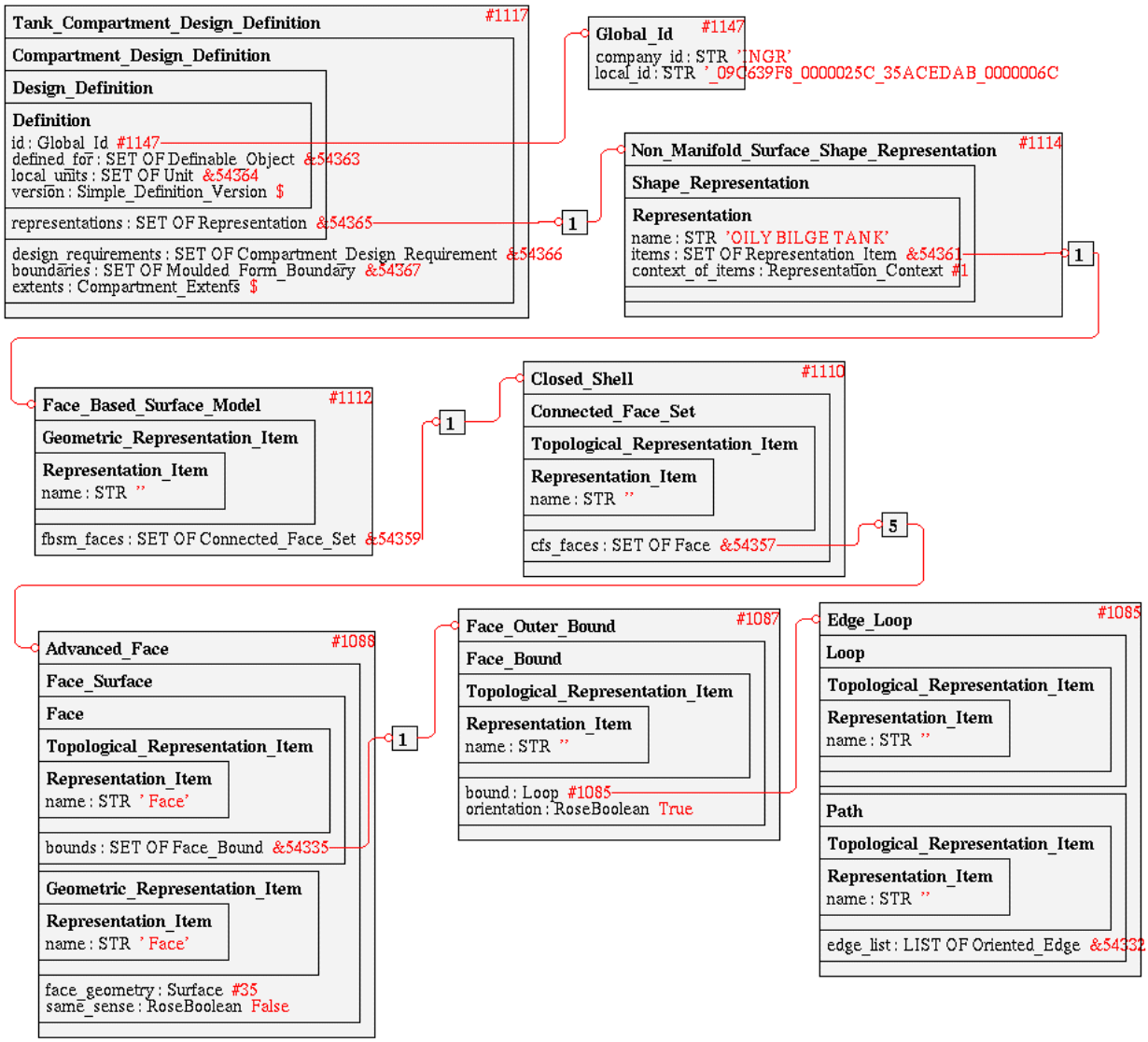


Figure A-44 - Tank_Compartment_Design_Definition

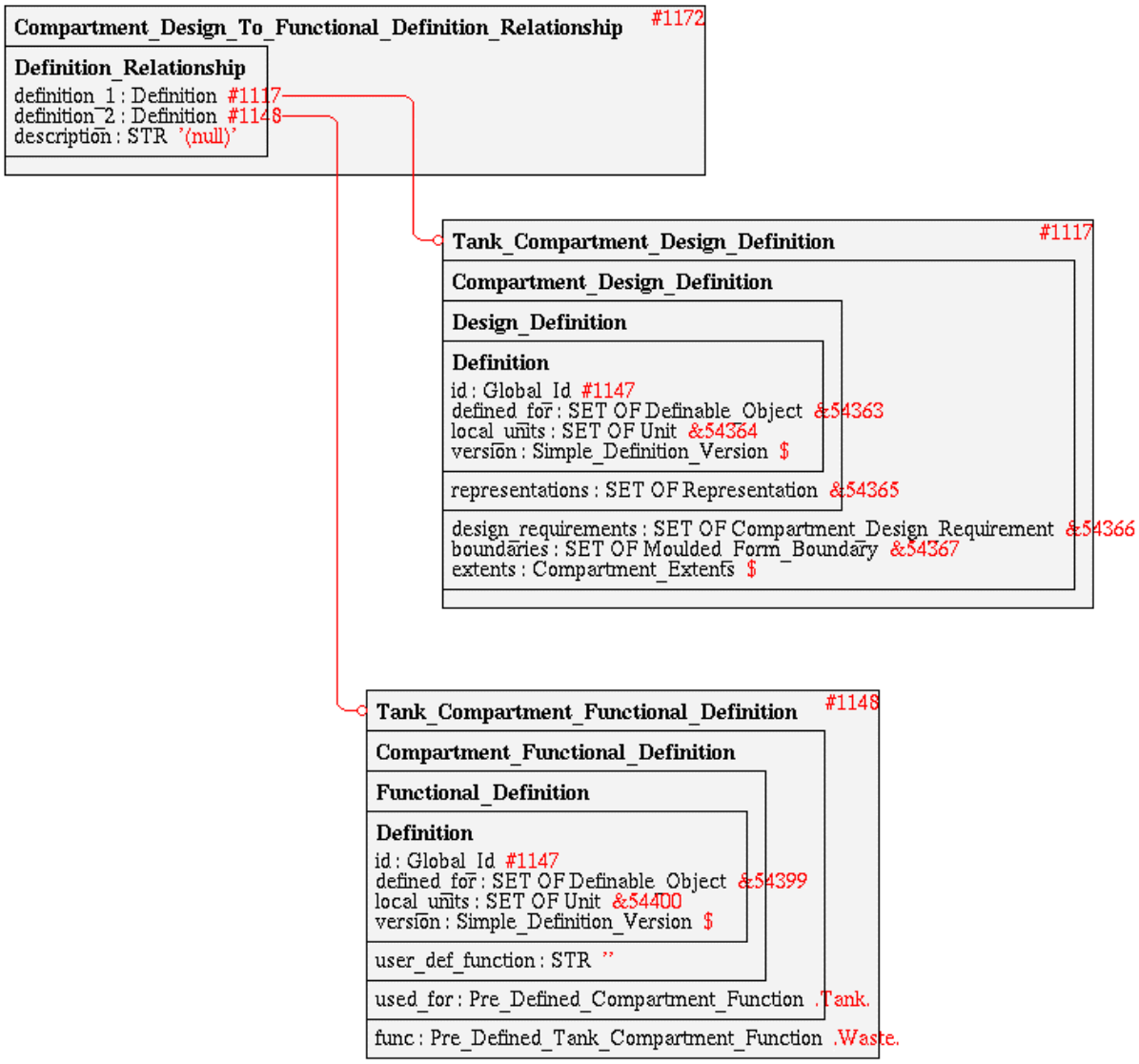


Figure A-45 - Compartment_Design_To_Functional_Definition_Relationship

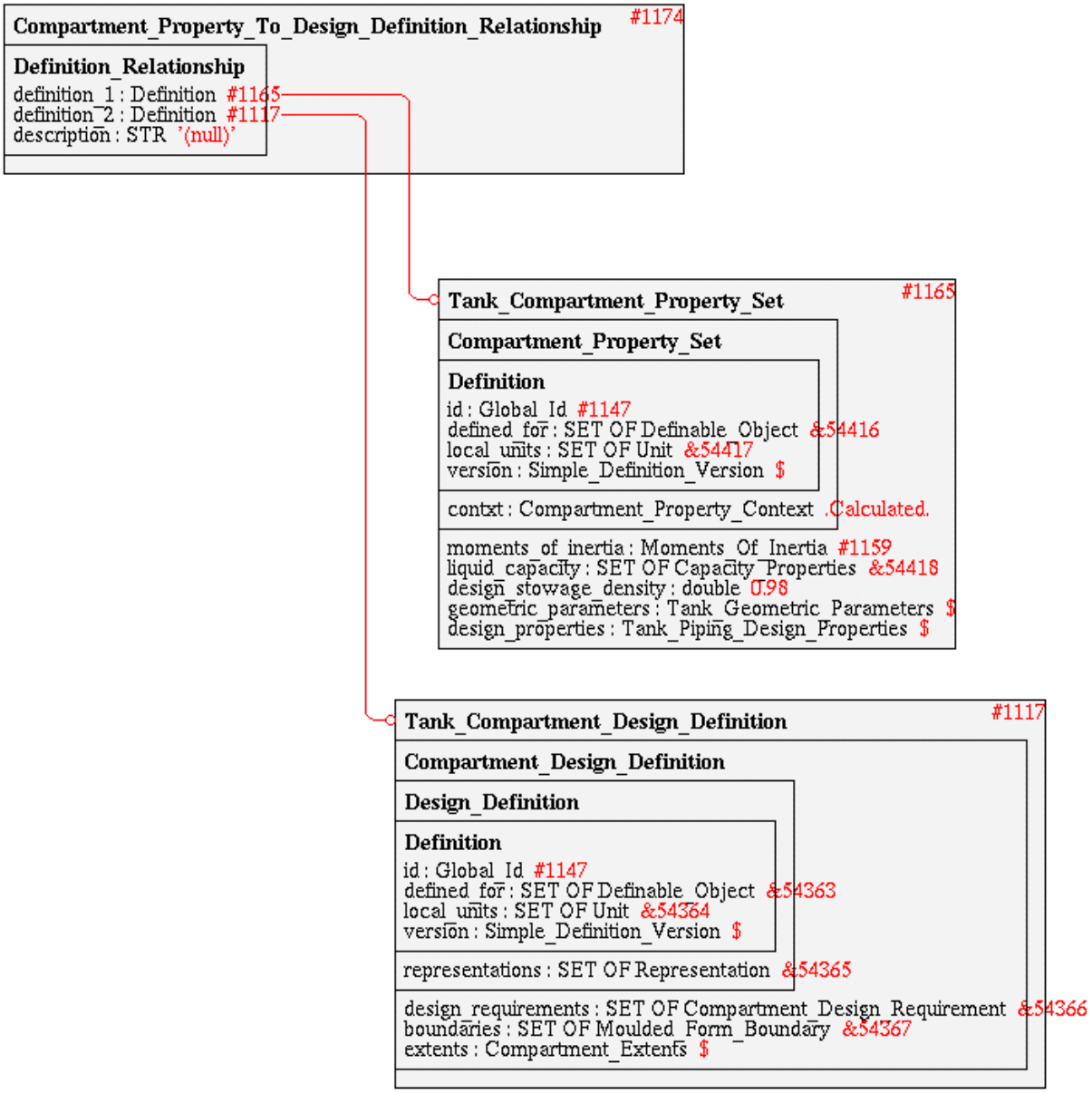


Figure A-46 - Tank_Compartment_Property_Set

Figure A-47 - General_Compartment_Property_Set (1/2)

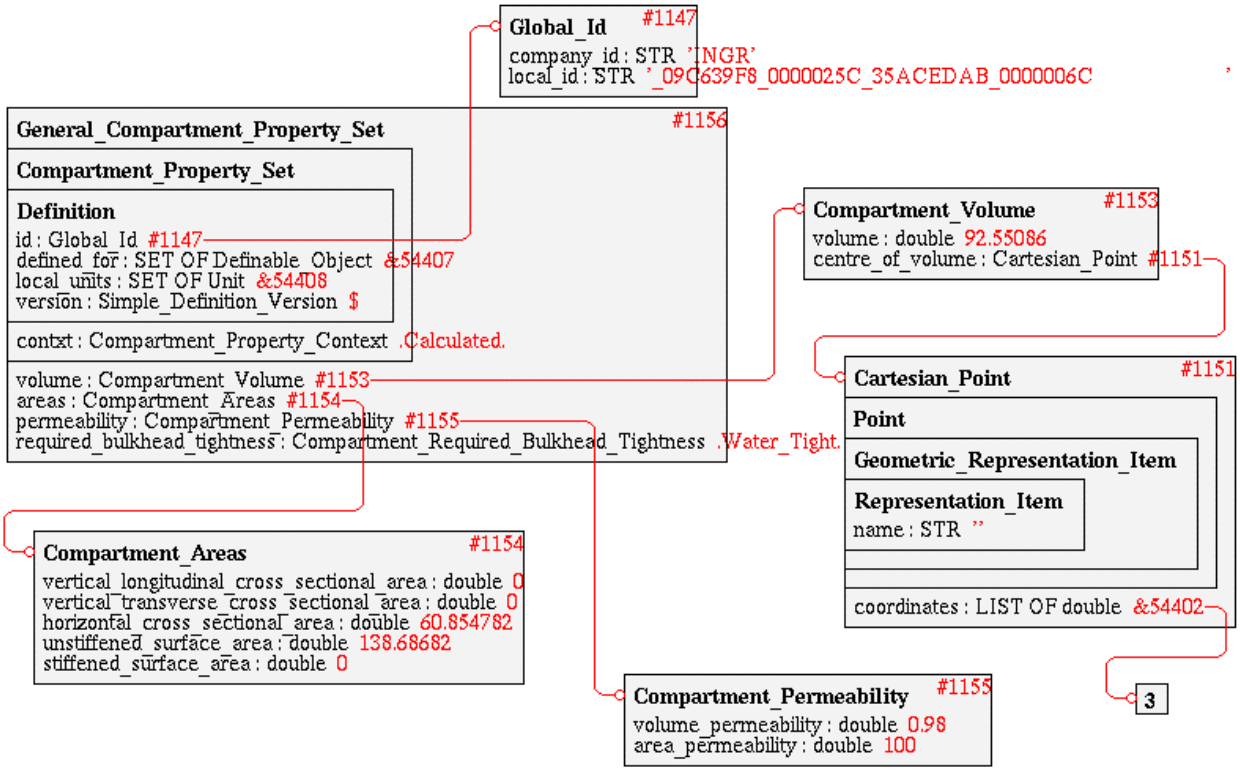


Figure A-48 - General_Compartment_Property_Set (2/2)

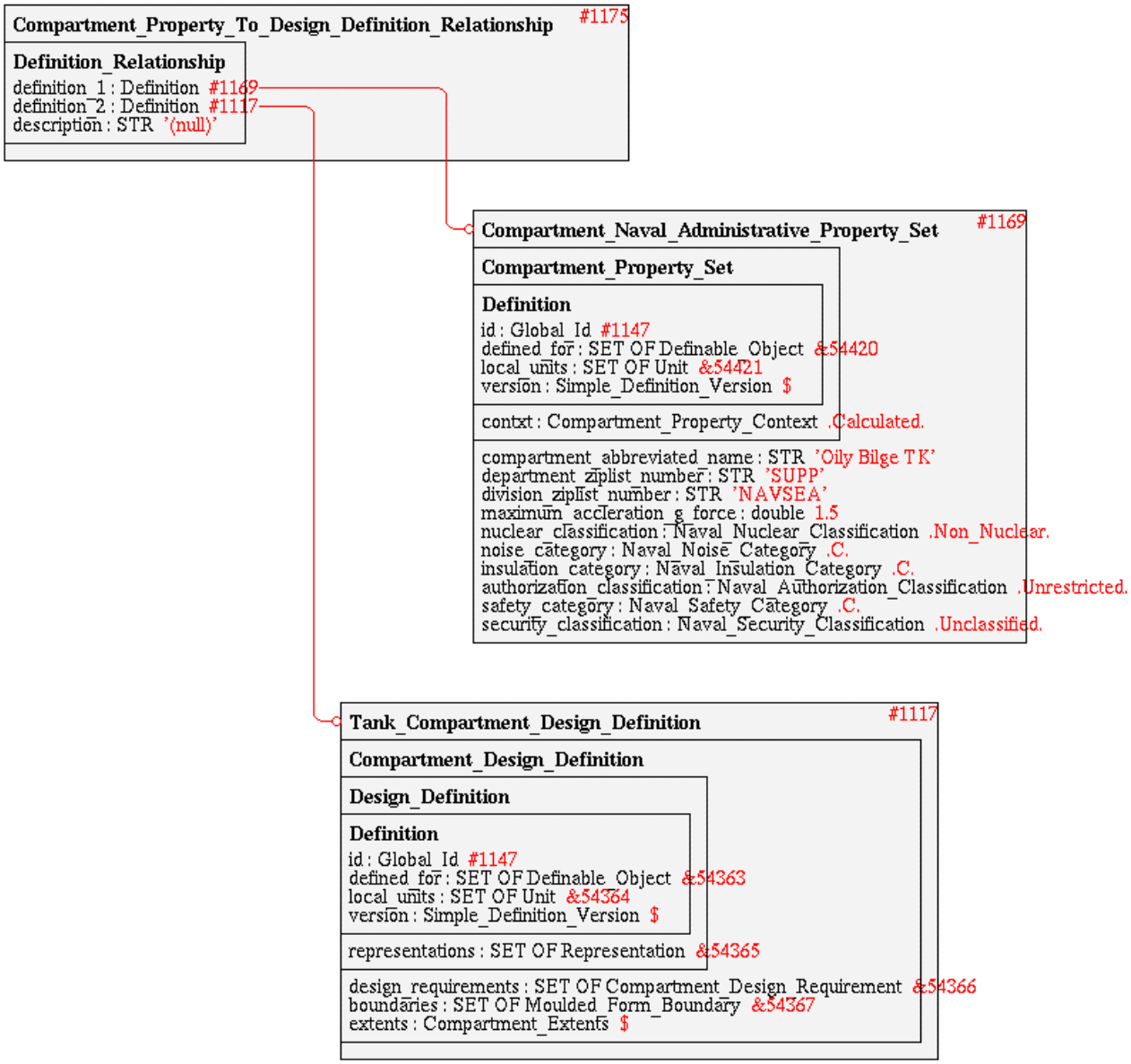


Figure A-49 - Compartment_Naval_Administrative_Property_Set

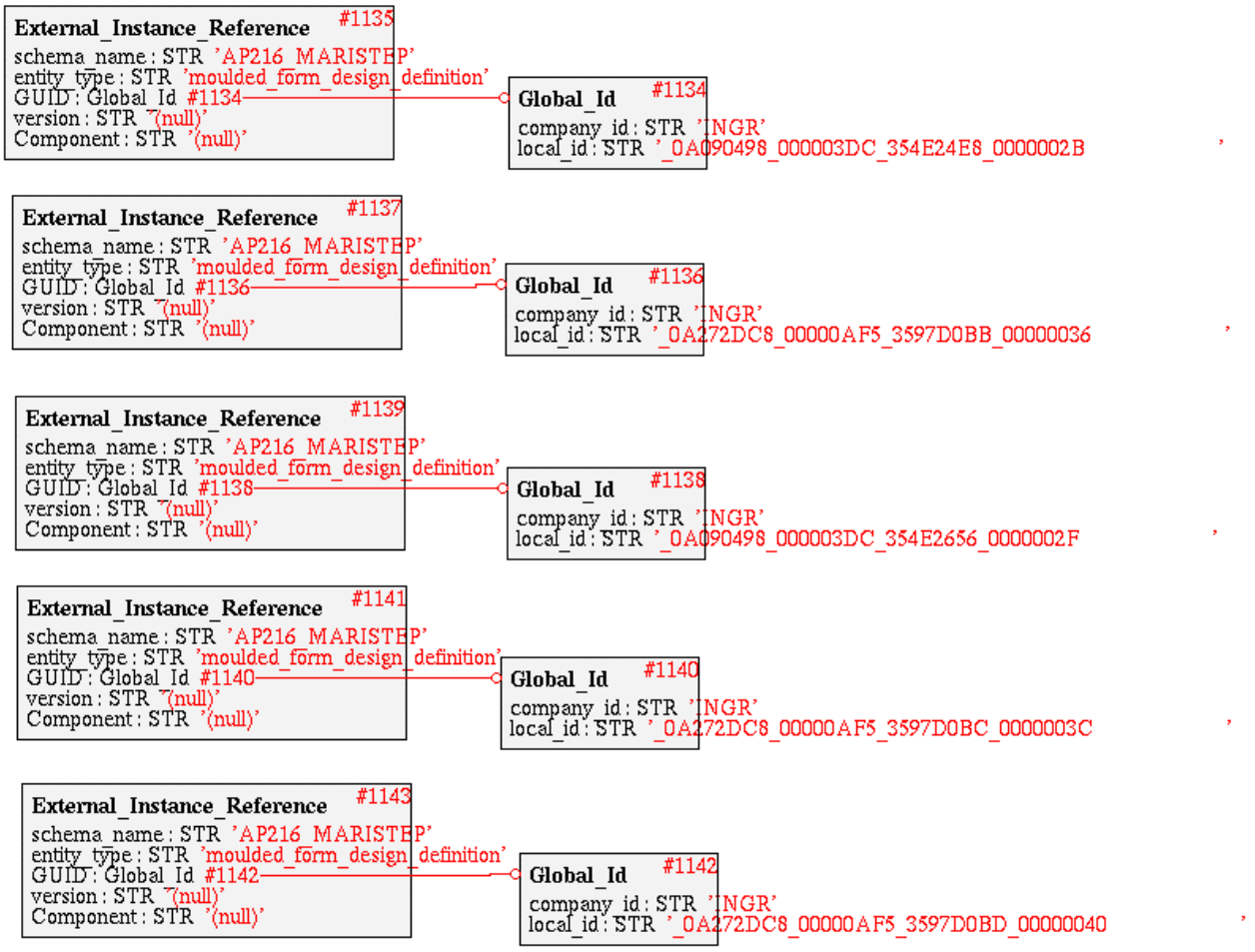


Figure A-50 - External_Instance_Reference

A.4.4 Preprocessor and Postprocessor Worksheets

The following are the preprocessor and postprocessor worksheets that were used by the MariSTEP project to capture test results for AP 215 testing efforts.

Preprocessor Worksheet

MariSTEP Test # : AP215-2-1 **Round #:** **Date Run:**

Team Member Name: **Stage / Track:** **Translator Version:**

	√	Procedure	Comments
Start - State 1			
1		Graphically review the native model to be translated	
Translate - State 2			
2		Preprocess native model	
3		Record results: Did it complete with errors & warnings identified? List and describe the fatal errors.	
Analyze Results - State 3			
Graphical			
4		Run STEP Part 21 Viewer	
5		Compare graphical output with view of native model just translated Was it graphically equivalent? If not, how was it different?	
Entities / Attributes (Part 21 File)			
6		**COMPARTMENT**	
7		name = Oily Bilge Tank	
8		documentation → external_reference	
9		description = <company defined>	
10		**EXTERNAL_REFERENCE**	
11		location → address	
12		description = General Specification for Ships of the United States Navy, Sect. 192	
13		**ADDRESS** (for external_reference)	
14		internal_location = ( )	
15		street_number = ( )	
16		street = ( )	
17		town = Washington, D.C.	
18		postal_code = 20362-5101	
19		country = USA	
20		telephone_number= ( )	
21		**TANK_COMPARTMENT_FUNCTIONAL_DEFINITION**	
22		defined_for → compartment (Oily Bilge Tank)	
23		used_for = Tank	
24		func = waste	
25		**TANK_COMPARTMENT_DESIGN_DEFINITION**	
26		id → Global_ID	
27		defined_for → compartment (Oily Bilge Tank)	
28		representations → non_manifold_surface_shape_representation	

MariSTEP Test # : AP215-2-1 **Round #:** **Date Run:**

Team Member Name: **Stage / Track:** **Translator Version:**

	√	Procedure	Comments
29		design_requirements → set of 1 compartment_design_ requirement	
30		boundaries → set of 5 external_instance_reference	
31		**GLOBAL_ID** (for Tank_Compartment_Design_Defintion)	
32		company_id = <company specific>	
33		local_id = <company specific>	
34		**NON_MANIFOLD_SURFACE_SHAPE_REPRESENTATION**	
35		items → set of 1 face_based_surface_model	
36		context_of_items → geometric_representation_context	
37		**FACE_BASED_SURFACE_MODEL**	
38		fbsm_faces -> set of one or more closed_shells	
39		CLOSED_SHELL (1 or more instances)	
40		**COMPARTMENT_DESIGN_REQUIREMENT**	
41		defined_for	
42		specification →document_reference	
43		requirement_type = piping	
44		requirement_description = Section 505, General Requirements for Piping System	
45		**DOCUMENT_REFERENCE**	
46		location → address	
47		description = NAVSEA S9AAD-AA-SPN-010/GEN-SPEC (NSN-0910-LP-432-5900)	
48		has_author → person_and_organisation	
49		**ADDRESS** (for document_reference)	
50		internal_location = <company specific>	
51		street_number = <company specific>	
52		street = <company specific>	
53		town = <company specific>	
54		postal_code = <company specific>	
55		country = <company specific>	
56		telephone_number= <company specific>	
57		**PERSON_AND_ORGANISATION**	
58		the_person → person	
59		the_organisation → organisation	
60		**PERSON**	
61		last_name = ( )	
62		first_name = ( )	
63		**ORGANISATION**	
64		name = Department of the Navy, Naval Sea System Command	
65		description = NAVSEA	
66		1 of 5 external_instance_reference	
67		**EXTERNAL_INSTANCE_REFERENCE** (FR33 transverse bulkhead)	
68		schema_name → AP216_MARISTEP	
69		entity_type → Moulded_form_design_definition	
70		guid → global_id	
71		**GLOBAL_ID**	
72		company_id = <company specific>	
73		local_id = <company specific>	
74		2 of 5 external_instance_reference	
75		**EXTERNAL_INSTANCE_REFERENCE** (FR26 transverse bulkhead)	
76		schema_name → AP216_MARISTEP	
77		entity_type → Moulded_form_design_definition	

MariSTEP Test # : AP215-2-1 **Round #:** **Date Run:**

Team Member Name: **Stage / Track:** **Translator Version:**

	√	Procedure	Comments
78		guid =global_id	
79		**GLOBAL_ID**	
80		company_id = <company specific>	
81		local_id = <company specific>	
82		**3 of 5 external_instance_reference**	
83		**EXTERNAL_INSTANCE_REFERENCE** (stbd hull shell section)	
84		schema_name → AP216_MARISTEP	
85		entity_type → Moulded_form_design_definition	
86		guid = global_id	
87		**GLOBAL_ID**	
88		company_id = <company specific>	
89		local_id = <company specific>	
90		**4 of 5 external_instance_reference**	
91		**EXTERNAL_INSTANCE_REFERENCE** (port hull shell section)	
92		schema_name → AP216_MARISTEP	
93		entity_type → Moulded_form_design_definition	
94		guid = global_id	
95		**GLOBAL_ID**	
96		company_id = <company specific>	
97		local_id = <company specific>	
98		**5 of 5 external_instance_reference**	
100		**EXTERNAL_INSTANCE_REFERENCE** (2.4 ABL Deck)	
101		schema_name = AP216_MARISTEP	
102		entity_type = Moulded_form_design_definition	
103		guid → global_id	
104		**GLOBAL_ID**	
105		company_id = <company specific>	
106		local_id = <company specific>	
107		**GENERAL_COMPARTMENT_PROPERTY_SET**	
108		defined_for → Compartment	
109		contxt = calculated	
110		volume → compartment_volume	
111		areas → compartment_areas	
112		permeability → compartment_permeability	
113		required_bulkhead_tightness = water_tight	
114		**COMPARTMENT_VOLUME**	
115		volume = <company calculated or indeterminate>	
116		centre_of_volume = <company calculated or indeterminate>	
117		**COMPARTMENT_AREAS**	
118		vertical_longitudinal_cross_sectional_area = <company calculated or indeterminate>	
119		horizontal_cross_sectional_area = <company calculated or indeterminate>	
120		unstiffened_surface_area = <company calculated or indeterminate>	
121		stiffened_surface_area = <company calculated or indeterminate>	
122		**COMPARTMENT_PERMEABILITY**	
123		volume_permeability = <company calculated or indeterminate>	
124		area_permeability = <company calculated or indeterminate>	
125		**TANK_COMPARTMENT_PROPERTY_SET**	
126		moments_of_inertia → moments_of_inertia	
127		liquid_capacity → set of 1 capacity_properties	
128		design_stowage_density = <company calculated>	
129		**MOMENTS_OF_INERTIA**	

MariSTEP Test # : AP215-2-1 **Round #:** **Date Run:**

Team Member Name: **Stage / Track:** **Translator Version:**

	√	Procedure	Comments
130		long_moment_of_inertia → <company calculated or indeterminate>	
131		trans_moment_of_inertia → <company calculated or indeterminate>	
132		**CAPACITY_PROPERTIES**	
133		capacity_level_origin = <company calculated or indeterminate>	
134		capacity_center = <company calculated or indeterminate>	
135		capacity_level = <company calculated or indeterminate>	
136		capacity_trim_angle = <company calculated or indeterminate>	
137		capacity_heel_angle = <company calculated or indeterminate>	
138		capacity_volume = <company calculated or indeterminate>	
139		capacity_context = <company calculated or indeterminate>	
140		**COMPARTMENT_DESIGN_TO_FUNCTIONAL_DEFINITION_ RELATIONSHIP**	
141		definition_1 → tank_compartment_design_definition for Oily Bilge Tank	
142		definition_2 → tank_compartment_functional_defintion for Oily Bilge Tank	
143		**COMPARTMENT_PROPERTY_TO_DESIGN_DEFINITION_RE LATIONSHIP**	
144		definition_1 → general_compartment_property_set for Oily Bilge Tank	
145		definition_2 → tank_compartment_design_definition for Oily Bilge Tank	
146		**COMPARTMENT_PROPERTY_TO_DESIGN_DEFINITION_RE LATIONSHIP**	
147		definition_1 → tank_compartment_property_set for the Oily Bilge Tank	
148		definition_2 → tank_compartment_design_definition for Oily Bilge Tank	
149		**SHIP**	
150		name = M/V MariSTEP	
151		units → Set of units (length, plane_angle, area, volume, density, area_moment_of_inertia, luminous_intensity)	
152		Length	
153		**LENGTH_UNIT + SI_UNIT**	
154		name = metre	
155		Plane Angle	
156		**PLANE_ANGLE_UNIT + SI_UNIT**	
157		name = radian	
158		Area	
159		**AREA_UNIT + SI_UNIT**	
160		name = metre	
161		Volume	
162		**SI_UNIT+VOLUME_UNIT** (volume)	
163		name = metre	
164		Density	
165		**DERIVED_UNIT**	
166		elements = SET of derived_unit_element (mass, volume)	
167		**DERIVED_UNIT_ELEMENT** (mass)	
168		unit → SI_unit	
169		exponent = 1	
170		**SI_UNIT** (mass)	
171		prefix = kilo	

MariSTEP Test # : AP215-2-1 **Round #:** **Date Run:**

Team Member Name: **Stage / Track:** **Translator Version:**

	√	Procedure	Comments
172		name = gram	
173		**DERIVED_UNIT_ELEMENT** (volume)	
174		unit → volume_unit + SI_Unit	
175		exponent → -1	
176		Area Moment of Inertia	
177		**DERIVED_UNIT** (area_moment_of_inertia)	
178		elements = SET of derived_unit_element	
179		**DERIVED_UNIT_ELEMENT** (area_moment_of_inertia)	
180		unit → length_unit+SI_unit	
181		exponent = 4	
182		Luminous Intensity	
183		**SI_UNIT**	
184		name = lux	
STEP Model Validation			
185		Validate STEP model generated using your STEP toolkit.	
186		Record results, describing errors that occurred. Errors? Y/N Global Rule Where Rule Required Attribute Inverse Attribute Attribute Datatype Attribute Value Uniqueness Aggregate Size Aggregate Uniqueness Array Sparseness	

Preprocessor Test Worksheet

MariSTEP Test # : AP215-2-1 **Round #:** **Date Run:**

Team Member Name: **Stage / Track:** **Translator Version:**

Postprocessor Worksheet

MariSTEP Test # : AP215-2-1 **Round #:** **Date Run:**

Team Member Name: **Stage / Track:** **Translator Version:**

PMDB File Input: AP215-2-1- **PMB File Created by:**

	√	Procedure	Comments
Start - State 1			
1		Run STEP Part 21 Viewer to graphically review the PMDB model to be translated	
Translate - State 2			
2		Postprocess PMDB model	
3		Record results: Did it complete with errors & warnings identified? List and describe the fatal errors.	
Analyze Results - State 3			
		Graphical	
4		Graphically compare the generated native model with the view of the PMDB model just translated Was it graphically equivalent? If not, how was it different?	
Entities / Attributes			
5		Compartment name is **Oily Bilge Tank**	
6		Is the tank function waste.	
7		5 external instance references to AP216.	
8		Does the Oily Bilge Tank Compartment include the following External references: • **document reference -** • **compartment design requirement** • **Does an address for Document Reference and compartment design requirement exist?**	
9		Is the product model in metric units?	
10		Do the following property sets exist: **general_compartment_property_set** • required bulkhead tightness = **water_tight** • tank_compartment_property_set	
Geometry			
11		Forward boundary of compartment is **27.575 m FWD of AP**	
12		Aft boundary of compartment is **21.80 m FWD of AP**	
13		Bounded on top by 2**.4 m ABL deck**	
14		Bounded by **hull shell (port and starboard) between frame 26 and 33**	

Postprocessor Worksheet

MariSTEP Test # : AP215-2-1 **Round #:** **Date Run:**

Team Member Name: **Stage / Track:** **Translator Version:**

PMDB File Input: AP215-2-1- **PMB File Created by:**

REVISION HISTORY

Rev	Rev Date	Revised By	Description	Reason
0	9/30/99		Initial Release	NSRP-0429 Version 2.0