ФЕДЕРАЛЬНОЕ АГЕНТСТВО

ПО ТЕХНИЧЕСКОМУ РЕГУЛИРОВАНИЮ И МЕТРОЛОГИИ

НАЦИОНАЛЬНЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

ГОСТ Р ИСО 10303-502— 2006

Системы автоматизации производства и их интеграция

ПРЕДСТАВЛЕНИЕ ДАННЫХ ОБ ИЗДЕЛИИ И ОБМЕН ЭТИМИ ДАННЫМИ

Часть 502

Прикладные интерпретированные конструкции. Каркасное представление формы на основе оболочек

ISO 10303-502:2000
Industrial automation systems and integration — Product data representation and exchange — Part 502: Application interpreted construct: Shell-based wireframe
(IDT)

Издание официальное

БЗ 5—2006/93

Москва
Стандартинформ
2007

gost.e-ecolog.ru

Предисловие

Цели и принципы стандартизации в Российской Федерации установлены Федеральным законом от 27 декабря 2002 г. № 184-ФЗ «О техническом регулировании», а правила применения национальных стандартов Российской Федерации — ГОСТ Р 1.0—2004 «Стандартизация в Российской Федерации. Основные положения»

Сведения о стандарте

1 ПОДГОТОВЛЕН Государственным научным учреждением «Центральный научно-исследовательский и опытно-конструкторский институт робототехники и технической кибернетики» на основе собственного аутентичного перевода стандарта, указанного в пункте 4

2 ВНЕСЕН Техническим комитетом по стандартизации ТК 459 «Информационная поддержка жизненного цикла изделий»

3 УТВЕРЖДЕН И ВВЕДЕН В ДЕЙСТВИЕ Приказом Федерального агентства по техническому регулированию и метрологии от 27 декабря 2006 г. № 487-ст

4 Настоящий стандарт идентичен международному стандарту ИСО 10303-502:2000 «Системы автоматизации производства и их интеграция. Представление данных об изделии и обмен этими данными. Часть 502. Прикладные интерпретированные конструкции. Каркасное представление формы на основе оболочек» (ISO 10303-502:2000 «Industrial automation systems and integration — Product data representation and exchange — Part 502: Application interpreted construct: Shell-based wireframe»). При применении настоящего стандарта рекомендуется использовать вместо ссылочных международных стандартов соответствующие им национальные стандарты Российской Федерации, сведения о которых приведены в дополнительном приложении Е

5 ВВЕДЕН ВПЕРВЫЕ

Информация об изменениях к настоящему стандарту публикуется в ежегодно издаваемом информационном указателе «Национальные стандарты», а текст изменений и поправок — в ежемесячно издаваемых информационных указателях «Национальные стандарты». В случае пересмотра (замены) или отмены настоящего стандарта соответствующее уведомление будет опубликовано в ежемесячно издаваемом информационном указателе «Национальные стандарты». Соответствующая информация, уведомление и тексты размещаются также в информационной системе общего пользования — на официальном сайте Федерального агентства по техническому регулированию и метрологии в сети Интернет

© Стандартинформ, 2007

Настоящий стандарт не может быть полностью или частично воспроизведен, тиражирован и распространен в качестве официального издания без разрешения Федерального агентства по техническому регулированию и метрологии

Содержание

1 Область применения 1
2 Нормативные ссылки 1
3 Термины и определения 2
4 Сокращенный листинг на языке EXPRESS 2
4.1 Введение 3
4.2 Основные понятия и допущения 3
4.3 Определение объекта shell_based_wireframe_shape_representation схемы aic_shell_based_wireframe 3
4.4 Определения функций схемы aic_shell_based_wireframe 7
Приложение А (обязательное) Сокращенное наименование объекта 9
Приложение В (обязательное) Регистрация информационного объекта 10
Приложение С (справочное) EXPRESS-G диаграммы 11
Приложение D (справочное) Машинно-интерпретируемые листинги 18
Приложение Е (справочное) Сведения о соответствии национальных стандартов Российской Федерации ссылочным международным стандартам 19

Введение

Стандарты комплекса ИСО 10303 распространяются на компьютерное представление информации об изделиях и обмен данными об изделиях. Их целью является обеспечение нейтрального механизма, способного описывать изделия на всем протяжении их жизненного цикла. Этот механизм применим не только для нейтрального обмена файлами, но является также основой для реализации и совместного доступа к базам данных об изделиях и организации архивирования.

Стандарты комплекса ИСО 10303 представляют собой набор отдельно издаваемых стандартов (частей). Стандарты данного комплекса относятся к одной из следующих тематических групп: методы описания, интегрированные ресурсы, прикладные интерпретированные конструкции, прикладные протоколы, комплекты абстрактных тестов, методы реализации и аттестационное тестирование. Группы стандартов данного комплекса описаны в ИСО 10303-1. Настоящий стандарт входит в группу прикладных интерпретированных конструкций.

Прикладная интерпретированная конструкция (ПИК) обеспечивает логическую группировку интерпретированных конструкций, поддерживающих конкретную функциональность для использования данных об изделии в разнообразных прикладных контекстах. Интерпретированная конструкция представляет собой обычную интерпретацию интегрированных ресурсов, поддерживающую требования совместного использования информации прикладными протоколами.

Настоящий стандарт определяет прикладную интерпретированную конструкцию для описания формы изделия посредством трехмерных моделей каркасного представления формы, ограниченных связанными совокупностями оболочек.

НАЦИОНАЛЬНЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

Системы автоматизации производства и их интеграция

ПРЕДСТАВЛЕНИЕ ДАННЫХ ОБ ИЗДЕЛИИ И ОБМЕН ЭТИМИ ДАННЫМИ

Часть 502

Прикладные интерпретированные конструкции.
Каркасное представление формы на основе оболочек

Industrial automation systems and integration. Product data representation and exchange.
Part 502. Application interpreted constructions. Shell-based wireframe

Дата введения — 2007—07—01

1 Область применения

Настоящий стандарт определяет интерпретацию интегрированных ресурсов, обеспечивающую соответствие требованиям к представлению формы изделия посредством использования трехмерных каркасных моделей, ограниченных совокупностью оболочек.

Область применения настоящего стандарта распространяется на:

- представление каркасных моделей, описываемых графом, состоящим из ребер и вершин, в котором ребра пересекаются только в их вершинах;
- представление каркасной модели одной или несколькими оболочками, которые должны перекрываться или пересекаться только в их вершинах или ребрах;
- точки, описываемые в трехмерном координатном пространстве;
- кривые, включая b-сплайны, описываемые в трехмерном координатном пространстве;
- представление отдельной каркасной модели или сборки каркасных моделей.

Область применения настоящего стандарта не распространяется на:

- геометрию, определенную в двумерном координатном пространстве;
- геометрию поверхности;
- геометрию твердого тела;
- ссылки на исключенные конструкции.

2 Нормативные ссылки

В настоящем стандарте использованы ссылки на следующие международные стандарты:

ИСО/МЭК 8824-1:1995 Информационные технологии. Взаимосвязь открытых систем. Абстрактная синтаксическая нотация версии один (АСН.1). Часть 1. Спецификация основной нотации

ИСО 10303-1:1994 Системы автоматизации производства и их интеграция. Представление данных об изделии и обмен этими данными. Часть 1. Общие представления и основополагающие принципы

ИСО 10303-11:1994 Системы автоматизации производства и их интеграция. Представление данных об изделии и обмен этими данными. Часть 11. Методы описания: Справочное руководство по языку EXPRESS

ИСО 10303-41:1994 Системы автоматизации производства и их интеграция. Представление данных об изделии и обмен этими данными. Часть 41. Интегрированные обобщенные ресурсы. Основы описания и поддержки изделий

ИСО 10303-42:1994 Системы автоматизации производства и их интеграция. Представление данных об изделии и обмен этими данными. Часть 42. Интегрированные обобщенные ресурсы. Геометрическое и топологическое представление

Издание официальное

ИСО 10303-43:1994 Системы автоматизации производства и их интеграция. Представление данных об изделии и обмен этими данными. Часть 43. Интегрированные обобщенные ресурсы. Структуры представлений

ИСО 10303-202:1996 Системы автоматизации производства и их интеграция. Представление данных об изделии и обмен этими данными. Часть 202. Прикладные протоколы. Ассоциативные чертежи

3 Термины и определения

3.1 Термины, определенные в ИСО 10303-1

В настоящем стандарте применены следующие термины:

- **приложение** (application);
- **прикладной контекст** (application context);
- **прикладной протокол**; ПП (application protocol; AP);
- **метод реализации** (implementation method);
- **интегрированный ресурс** (integrated resource);
- **интерпретация** (interpretation);
- **модель** (model);
- **изделие** (product);
- **данные об изделии** (product data).

3.2 Термин, определенный в ИСО 10303-202

В настоящем стандарте применен следующий термин:

- **прикладная интерпретированная конструкция**; ПИК (application interpreted construct; AIC).

4 Сокращенный листинг на языке EXPRESS

В настоящем разделе определена EXPRESS-схема, в которой используются элементы интегрированных ресурсов и содержатся типы, конкретизации объектов и функции, относящиеся к настоящему стандарту.

Примечание — Допускается существование подтипов и элементов списков выбора, которые появляются в интегрированных ресурсах, не импортированных в ПИК. Такие конструкции исключаются из дерева подтипов или списка выбора посредством правил неявного интерфейса, определенных в ИСО 10303-11.

EXPRESS-спецификация

```
*)
SCHEMA aic_shell_based_wireframe ;

USE FROM geometric_model_schema                    -- ISO 10303-42
    (shell_based_wireframe_model) ;

USE FROM geometry_schema                           -- ISO 10303-42
    (axis2_placement_3d,
    b_spline_curve_with_knots,
    bezier_curve,
    cartesian_transformation_operator_3d,
    circle,
    conic,
    curve,
    curve_replica,
    ellipse,
    geometric_representation_context,
    hyperbola,
    line,
    offset_curve_3d,
    parabola,
    point,
```

```
    point_replica,
    polyline,
    quasi_uniform_curve,
    rational_b_spline_curve,
    uniform_curve);

USE FROM product_property_representation_schema        -- ISO 10303-41
    (shape_representation);

USE FROM representation_schema                         -- ISO 10303-43
    (mapped_item);

USE FROM topology_schema                               -- ISO 10303-42
    (edge_curve,
     edge_loop,
     path,
     vertex_loop,
     vertex_point
     vertex_shell,
     wire_shell);
(*
```

Примечание — Схемы, ссылки на которые даны выше, можно найти в следующих стандартах комплекса ИСО 10303:

geometric_model_schema	— ИСО 10303-42;
geometry_schema	— ИСО 10303-42;
product_property_representation_schema	— ИСО 10303-41;
topology_schema	— ИСО 10303-42;
representation_schema	— ИСО 10303-43.

4.1 Введение

В настоящем стандарте определяются геометрические и топологические структуры для представления трехмерных форм с использованием оболочек размерности 0 (объект **vertex_shell**) или 1 (объект **wire_shell**). Данная ПИК представлена объектом **shell_based_wireframe_shape_representation,** который относится к типу **shape_representation**.

4.2 Основные понятия и допущения

Данное каркасное представление формы основывается на модели **shell_based_wireframe_model** и необходимых геометрических и топологических объектах. Определения форм, представленных объектом **shell_based_wireframe_shape_representation**, могут быть обеспечены совокупностями объектов **vertex_shell** или **wire_shell**. Каркасные представления могут быть компоновками других каркасных представлений, отображаемых на то же пространство координат.

4.3 Определение объекта shell_based_wireframe_shape_representation схемы aic_shell_based_wireframe

Объект **shell_based_wireframe_shape_representation** является трехмерным подтипом объекта **shape_representation**, представляющим форму или часть формы изделия каркасными конструкциями, определяющими не выраженный явно объем. Это включает в себя все трехмерные кривые и топологические объекты, определяющие граф, состоящий из вершин, ребер и замкнутых цепей.

Примечание — Прикладной протокол, в котором используется данная ПИК, может обеспечить реализацию объекта **shape_representation** как объекта **shell_based_wireframe_shape_representation**.

EXPRESS-спецификация

```
*)
ENTITY shell_based_wireframe_shape_representation
    SUBTYPE OF (shape_representation);
  WHERE
    WR1: SIZEOF (QUERY (it <* SELF.items |
           NOT
           (SIZEOF( ['AIC_SHELL_BASED_WIREFRAME. SHELL_BASED_WIREFRAME_MODEL',
```

```
                    'AIC_SHELL_BASED_WIREFRAME.MAPPED_ITEM',
                    'AIC_SHELL_BASED_WIREFRAME.AXIS2_PLACEMENT_3D'] *
                TYPEOF (it)) = 1
        ))) = 0 ;
    WR2: SIZEOF (QUERY (it <* SELF.items |
        SIZEOF( ['AIC_SHELL_BASED_WIREFRAME.SHELL_BASED_WIREFRAME_MODEL',
        'AIC_SHELL_BASED_WIREFRAME.MAPPED_ITEM'] * TYPEOF (it)) = 1
        )) >= 1 ;
    WR3: SIZEOF (QUERY (sbwm <* QUERY (it <* SELF.items |
        ('AIC_SHELL_BASED_WIREFRAME.SHELL_BASED_WIREFRAME_MODEL'
            IN TYPEOF (it))) |
        NOT (SIZEOF (QUERY (ws <* QUERY (sb <*
                    sbwm \ shell_based_wireframe_model.sbwm_boundary |
                ('AIC_SHELL_BASED_WIREFRAME.WIRE_SHELL' IN TYPEOF (sb))) |
            NOT (SIZEOF (QUERY (eloop <* QUERY (wsb <*
                    ws \ wire_shell.wire_shell_extent |
                ('AIC_SHELL_BASED_WIREFRAME.EDGE_LOOP' IN TYPEOF (wsb))) |
                NOT (SIZEOF (QUERY (el <* eloop\path.edge_list |
                    NOT ('AIC_SHELL_BASED_WIREFRAME.EDGE_CURVE' IN
                        TYPEOF (el.edge_element)))) = 0)
                )) = 0)
            )) = 0)
        )) = 0 ;
    WR4: SIZEOF (QUERY (sbwm <* QUERY (it <* SELF.items |
        ('AIC_SHELL_BASED_WIREFRAME.SHELL_BASED_WIREFRAME_MODEL'
            IN TYPEOF (it))) |
        NOT (SIZEOF (QUERY (ws <* QUERY (sb <*
                    sbwm \ shell_based_wireframe_model.sbwm_boundary |
                ('AIC_SHELL_BASED_WIREFRAME.WIRE_SHELL' IN TYPEOF (sb))) |
        NOT (SIZEOF (QUERY (eloop <* QUERY (wsb <*
                    ws \ wire_shell.wire_shell_extent |
            ('AIC_SHELL_BASED_WIREFRAME.EDGE_LOOP' IN TYPEOF (wsb))) |
            NOT (SIZEOF (QUERY (pline_el <*
                    QUERY (el <* eloop \ path.edge_list |
                ('AIC_SHELL_BASED_WIREFRAME.POLYLINE' IN
                    TYPEOF (el.edge_element \ edge_curve.edge_geometry))) |
            NOT (SIZEOF (pline_el.edge_element \ edge_curve.
                    edge_geometry \ polyline.points) >2)
        )) = 0)
        )) = 0)
        )) = 0)
    )) = 0 ;
    WR5: SIZEOF (QUERY (sbwm <* QUERY (it <* SELF.items |
        ('AIC_SHELL_BASED_WIREFRAME.SHELL_BASED_WIREFRAME_MODEL'
            IN TYPEOF (it))) |
        NOT (SIZEOF (QUERY (ws <* QUERY (sb <*
                    sbwm \ shell_based_wireframe_model.sbw_boundary |
                ('AIC_SHELL_BASED_WIREFRAME.WIRE_SHELL' IN TYPEOF (sb))) |
            NOT (SIZEOF (QUERY (eloop <* QUERY (wsb <*
                    ws \ wire_shell.wire_shell_extent |
                ('AIC_SHELL_BASED_WIREFRAME.EDGE_LOOP' IN TYPEOF (wsb))) |
                NOT (SIZEOF (QUERY (el <* eloop \ path.edge_list |
                    NOT (valid_wireframe_edge_curve
                        (el.edge_element \ edge_curve.edge_geometry ))) = 0)
        )) = 0)
        )) = 0)
    )) = 0 ;
WR6: SIZEOF (QUERY (sbwm <* QUERY (it <* SELF.items |
```

```
        ('AIC_SHELL_BASED_WIREFRAME.SHELL_BASED_WIREFRAME_MODEL'
            IN TYPEOF(it))) |
        NOT (SIZEOF (QUERY (ws <* QUERY (sb <*
                                sbwm \ shell_based_wireframe_model.sbwm_boundary |
                        ('AIC_SHELL_BASED_WIREFRAME.WIRE_SHELL' IN TYPEOF (sb))) |
            NOT (SIZEOF (QUERY (eloop <* QUERY (wsb <*
                                ws \ wire_shell.wire_shell_extent |
                        ('AIC_SHELL_BASED_WIREFRAME.EDGE_LOOP' IN TYPEOF (wsb))) |
                NOT (SIZEOF (QUERY (el <* eloop \ path.edge_list |
                    NOT (('AIC_SHELL_BASED_WIREFRAME.VERTEX_POINT'IN
                                TYPEOF (el.edge_element.edge_start))
                        AND
                                ('AIC_SHELL_BASED_WIREFRAME.VERTEX_POINT' IN
                                    TYPEOF (el.edge_element.edge_end))) )) = 0)
            )) = 0)
        )) = 0)
    )) = 0 ;
WR7: SIZEOF (QUERY (sbwm <* QUERY (it <* SELF.items |
        ('AIC_SHELL_BASED_WIREFRAME.SHELL_BASED_WIREFRAME_MODEL'
            IN TYPEOF (it))) |
        NOT (SIZEOF (QUERY (ws <* QUERY (sb <*
                                sbwm \ shell_based_wireframe_model.sbwm_boundary |
                        ('AIC_SHELL_BASED_WIREFRAME.WIRE_SHELL' IN TYPEOF (sb))) |
            NOT (SIZEOF (QUERY (eloop <* QUERY (wsb <*
                                ws \ wire_shell.wire_shell_extent |
                    ('AIC_SHELL_BASED_WIREFRAME.EDGE_LOOP' IN TYPEOF (wsb))) |
                NOT (SIZEOF (QUERY (el <* eloop \ path.edge_list |
                        NOT ((valid_wireframe_vertex_point
                                (el.edge_element.
                                 edge_start \ vertex_point.vertex_geometry))
                    AND
                        (valid_wireframe_vertex_point
                     (el.edge_element.edge_end \ vertex_point.vertex_geometry)))
            )) = 0)
        )) = 0)
    )) = 0)
)) = 0 ;
WR8: SIZEOF (QUERY (sbwm <* QUERY (it <* SELF.items |
        ('AIC_SHELL_BASED_WIREFRAME.SHELL_BASED_WIREFRAME_MODEL'
            IN TYPEOF (it))) |
        NOT (SIZEOF (QUERY (ws <* QUERY (sb <*
                                sbwm \ shell_based_wireframe_model.sbwm_boundary |
                        ('AIC_SHELL_BASED_WIREFRAME.WIRE_SHELL' IN TYPEOF (sb))) |
            NOT (SIZEOF (QUERY (vloop <* QUERY (wsb <*
                                ws \ wire_shell.wire_shell_extent |
                    ('AIC_SHELL_BASED_WIREFRAME.VERTEX_LOOP' IN TYPEOF (wsb))) |
                NOT ('AIC_SHELL_BASED_WIREFRAME.VERTEX_POINT' IN
                        TYPEOF (vloop \ vertex_loop.loop_vertex))
        )) = 0)
    )) = 0)
)) = 0;
WR9: SIZEOF (QUERY (sbwm <* QUERY (it <* SELF.items |
        ('AIC_SHELL_BASED_WIREFRAME.SHELL_BASED_WIREFRAME_MODEL'
            IN TYPEOF (it))) |
        NOT (SIZEOF (QUERY (ws <* QUERY (SB <*
                                sbwm \ shell_based_wireframe_model.sbwm_boundary |
                    ('AIC_SHELL_BASED_WIREFRAME.WIRE_SHELL' IN TYPEOF (sb))) |
```

```
      NOT (SIZEOF (QUERY (vloop <* QUERY (wsb <*
                              ws \ wire_shell.wire_shell_extent |
            ('AIC_SHELL_BASED_WIREFRAME.VERTEX_LOOP' IN TYPEOF (wsb))) |
          NOT (valid_wireframe_vertex_point (vloop \ vertex_loop.
               loop_vertex \ vertex_point.vertex_geometry))
          )) = 0)
        )) = 0)
    )) = 0 ;
  WR10: SIZEOF (QUERY (sbwm <* QUERY (it <* SELF.items |
        ('AIC_SHELL_BASED_WIREFRAME.SHELL_BASED_WIREFRAME_MODEL'
            IN TYPEOF (it))) |
        NOT (SIZEOF (QUERY (vs <* QUERY (sb <*
                                sbwm \ shell_based_wireframe_model.sbwm_boundary |
                    ('AIC_SHELL_BASED_WIREFRAME.VERTEX_SHELL' IN TYPEOF (sb))) |
          NOT ('AIC_SHELL_BASED_WIREFRAME.VERTEX_POINT' IN
                    TYPEOF (vs \ vertex_shell.vertex_shell_extent.loop_vertex))
        )) = 0)
    )) = 0 ;
  WR11: SIZEOF (QUERY (sbwm <* QUERY (it <* SELF.items |
        ('AIC_SHELL_BASED_WIREFRAME.SHELL_BASED_WIREFRAME_MODEL'
            IN TYPEOF (it))) |
        NOT (SIZEOF (QUERY (vs <* QUERY (sb <*
                                sbwm \ shell_based_wireframe_model.sbwm_boundary |
                    ('AIC_SHELL_BASED_WIREFRAME.VERTEX_SHELL' IN TYPEOF (sb))) |
          NOT (valid_wireframe_vertex_point (vs \ vertex_shell.
                vertex_shell_extent.loop_vertex\vertex_point.
                vertex_geometry))
        )) = 0)
    )) = 0 ;
  WR12: SIZEOF (QUERY (mi <* QUERY (it <* SELF.items |
          ('AIC_SHELL_BASED_WIREFRAME.MAPPED_ITEM' IN TYPEOF (it))) |
          NOT ('AIC_SHELL_BASED_WIREFRAME.' +
                'SHELL_BASED_WIREFRAME_SHAPE_REPRESENTATION' IN
            TYPEOF (mi \ mapped_item.mapping_source.mapped_representation)
      ))) = 0;
  WR13: SELF.context_of_items \ geometric_representation_context.
            coordinate_space_dimension = 3 ;
END_ENTITY ;
(*
```

Формальные утверждения

WR1 — элементами **items** в объекте **shell_based_wireframe_shape_representation** должны быть **shell_based_wireframe_model, mapped_item** или **axis2_placement_3d**.

WR2 — по крайней мере один из элементов **items** в объекте **shell_based_wireframe_-shape_representation** должен быть либо **shell_based_wireframe_model**, либо **mapped_item**.

WR3 — все ребра (объекты **edge**), образующие замкнутую цепь (объект **edge_loop**) в **shell_based_wireframe_model**, должны иметь тип **edge_curve**.

WR4 — каждый объект **polyline**, который лежит в основе объекта **edge** в **shell_based_-wireframe_model**, должен быть определен более чем двумя различными точками (объектами **point**).

WR5 — объект **edge_geometry**, который лежит в основе объекта **edge** для **edge_based_-wireframe_model**, должен иметь типы **line**, **conic**, **b_spline_curve**, **offset_curve_3d**, **polyline** или **curve_replica**, а кривые, имеющие базу, заданную другими кривыми, представляются таким же образом. Для объектов **offset_curve_3d** или **curve_replica** кривая, указанная как базовая (объект **basis_curve**), должна быть одного из вышеупомянутых типов.

WR6 — каждая вершина (объект **vertex**), определенная как начальная или конечная вершина для ребра (объект **edge**) в **shell_based_wireframe_model**, должна иметь тип **vertex_point**.

WR7 — объект **vertex_geometry**, который лежит в основе вершин, определяющих границы ребер (объектов **edge**) в замкнутой цепи (объект **edge_loop**) для **shell_based_wireframe_model**, должен быть

либо **cartesian_point**, либо **point_replica**, а объект **point_replica** должен воспроизводить либо другой объект **point_replica**, либо **cartesian_point**.

WR8 — вершина, которая определяет объект **vertex_loop**, используемый в качестве границы в **shell_based_wireframe_model**, должна иметь тип **vertex_point**.

WR9 — в основе вершины, которая определяет объект **vertex_loop**, используемый в качестве границы в **shell_based_wireframe_model**, должен лежать объект **cartesian_point** или **point_replica**, а объект **point_replica** должен воспроизводить либо другой объект **point_replica**, либо **cartesian_point**.

WR10 — вершина, которая определяет объект **vertex_loop**, используемый как **vertex_shell_extent** для **vertex_shell** в **shell_based_wireframe_model**, должна быть **vertex_point**.

WR11 — в основе вершины, которая определяет объект **vertex_loop**, используемый как **vertex_shell_extent** для объекта **vertex_shell** в **shell_based_wireframe_model**, должен лежать объект **cartesian_point** или **point_replica**, а объект **point_replica** должен воспроизводить либо другой объект **point_replica**, либо **cartesian_point**.

WR12 — если в объекте **shell_based_wireframe_shape_representation** имеется объект **mapped_item**, то источником объекта **mapped_item** должен быть объект **shell_based_wireframe_shape_representation**.

WR13 — значение параметра **coordinate_space_dimension** объекта **shell_based_wireframe_shape_representation** должно быть равно трем.

4.4 Определения функций схемы aic_shell_based_wireframe

4.4.1 Функция **valid_wireframe_edge_curve** проверяет, действительно ли заданная аргументом кривая является допустимой для использования в представлении формы, определенной топологически ограниченным каркасом.

EXPRESS-спецификация

```
*)
FUNCTION valid_wireframe_edge_curve (crv : curve ) : BOOLEAN;
      -- проверка на допустимость основных типов кривых

      IF SIZEOF ( ['AIC_SHELL_BASED_WIREFRAME.LINE',
                   'AIC_SHELL_BASED_WIREFRAME.CONIC',
                   'AIC_SHELL_BASED_WIREFRAME.B_SPLINE_CURVE',
                   'AIC_SHELL_BASED_WIREFRAME.POLYLINE' ] * TYPEOF (crv)) = 1
        THEN RETURN (TRUE) ;
      ELSE
      -- рекурсивная проверка на допустимость основных кривых для типа curve_replica
      IF ('AIC_SHELL_BASED_WIREFRAME.CURVE_REPLICA') IN TYPEOF (crv)
       THEN RETURN (valid_wireframe_edge_curve
                          (crv \ curve_replica.parent_curve)) ;
      ELSE
          -- рекурсивная проверка на допустимость основных кривых для типа offset_curve
          IF('AIC_SHELL_BASED_WIREFRAME.OFFSET_CURVE_3D') IN TYPEOF (crv)
            THEN RETURN (valid_wireframe_edge_curve
                               (crv \ offset_curve_3d.basis_curve)) ;
          END_IF ;
      END_IF ;
     END_IF ;
     RETURN (FALSE) ;
  END_FUNCTION ;
(*
```

Определение аргумента

crv — заданная кривая, которая должна быть проверена на допустимость. Допустимой кривой является кривая типов **line**, **conic**, **b_spline_curve**, **offset_curve_3d**, **polyline** или **curve_replica**. Если кривая имеет тип **offset_curve_3d** или **curve_replica**, то кривая, на которую дается ссылка как на базовую (**basis_curve**) или порождающую (**parent_curve**), также должна быть допустимой кривой.

4.4.2 Функция **valid_wireframe_vertex_point** проверяет, действительно ли заданная аргументом точка является допустимой для использования в представлении формы, определенной топологически ограниченным каркасом.

EXPRESS спецификация

```
*)
FUNCTION valid_wireframe_vertex_point (pnt : point) : BOOLEAN ;
        -- проверка на допустимость основных типов точек
        IF ('AIC_SHELL_BASED_WIREFRAME.CARTESIAN_POINT' ) IN TYPEOF (pnt)
          THEN RETURN (TRUE) ;
        ELSE
        -- рекурсивная проверка на допустимость основных типов точек как порождающих для
        -- типа point_replica

        IF ('AIC_SHELL_BASED_WIREFRAME.POINT_REPLICA' ) IN TYPEOF (pnt)
          THEN RETURN (valid_wireframe_vertex_point
                                (pnt \ point_replica.parent_pt)) ;
        END_IF;
      END_IF;
      RETURN (FALSE);
  END_FUNCTION;
(*
```

Определение аргумента

pnt — заданная точка, которая должна быть проверена на допустимость. Допустимой точкой является точка типа либо **cartesian_point**, либо **point_replica**. Если точка имеет тип **point_replica**, то порождающая точка (**parent_point**) также должна быть допустимой точкой.

```
*)
END_SCHEMA ;
(*
```

Приложение А
(обязательное)

Сокращенное наименование объекта

Сокращенное наименование объекта, установленного в настоящем стандарте, приведено в таблице А.1. Требования к использованию сокращенных наименований содержатся в методах реализации, описанных в соответствующих стандартах комплекса ИСО 10303.

Таблица А.1 — Сокращенное наименование объекта

Полное наименование объекта	Сокращенное наименование
SHELL_BASED_WIREFRAME_SHAPE_REPRESENTATION	SBWSR

Приложение В
(обязательное)

Регистрация информационного объекта

В.1 Обозначение документа

Для обеспечения однозначного обозначения информационного объекта в открытой системе настоящему стандарту присвоен следующий идентификатор объекта:

{ iso standard 10303 part(502) version(1) }

Смысл данного обозначения установлен в ИСО/МЭК 8824-1 и описан в ИСО 10303-1.

В.2 Обозначение схемы

Для обеспечения однозначного обозначения в открытой системе схеме aic_shell_based_wireframe_schema (см. раздел 4) присвоен следующий идентификатор объекта:

{ iso standard 10303 part(502) version(1) object(1) aic-shell-based-wireframe-schema(1) }

Смысл данного обозначения установлен в ИСО/МЭК 8824-1 и описан в ИСО 10303-1.

Приложение С
(справочное)

EXPRESS-G диаграммы

EXPRESS диаграммы, представленные на рисунках С.1 — С.6, получены из сокращенного листинга, представленного в разделе 4, с использованием спецификаций интерфейса стандарта ИСО 10303-11. В диаграммах использована графическая нотация EXPRESS-G языка EXPRESS. Описание EXPRESS-G установлено в ИСО 10303-11, приложение D.

Примечание — Выбранные типы geometric_set_select, transformation, trimming_select, reversible_-topology, vector_or_direction и wireframe_select импортируются в расширенный листинг ПИК в соответствии с правилами неявных интерфейсов по ИСО 10303-11. В настоящем стандарте другие объекты не ссылаются на эти выбранные типы.

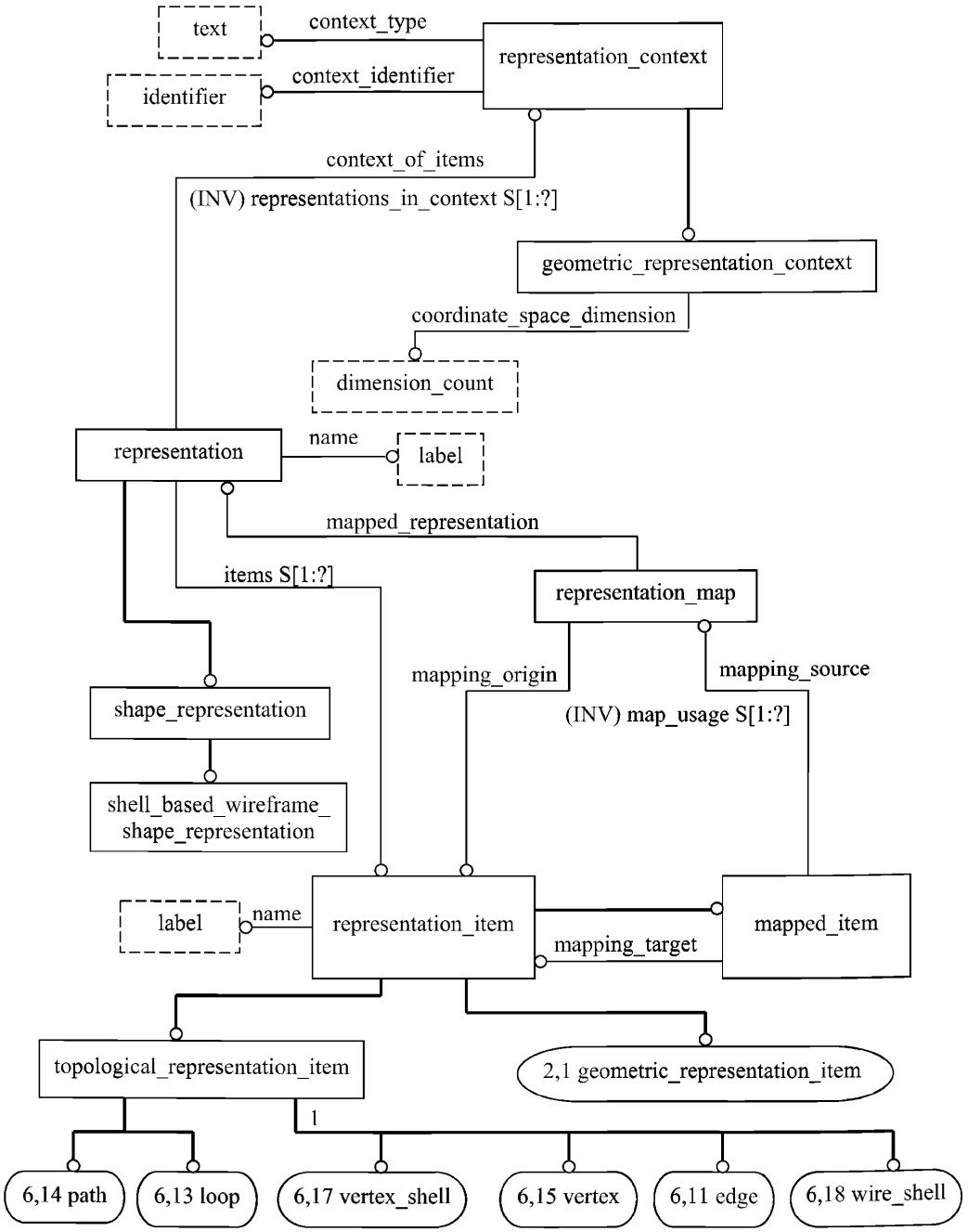

Рисунок С.1 — **aic_shell_based_wireframe** — EXPRESS-G диаграмма 1 из 6

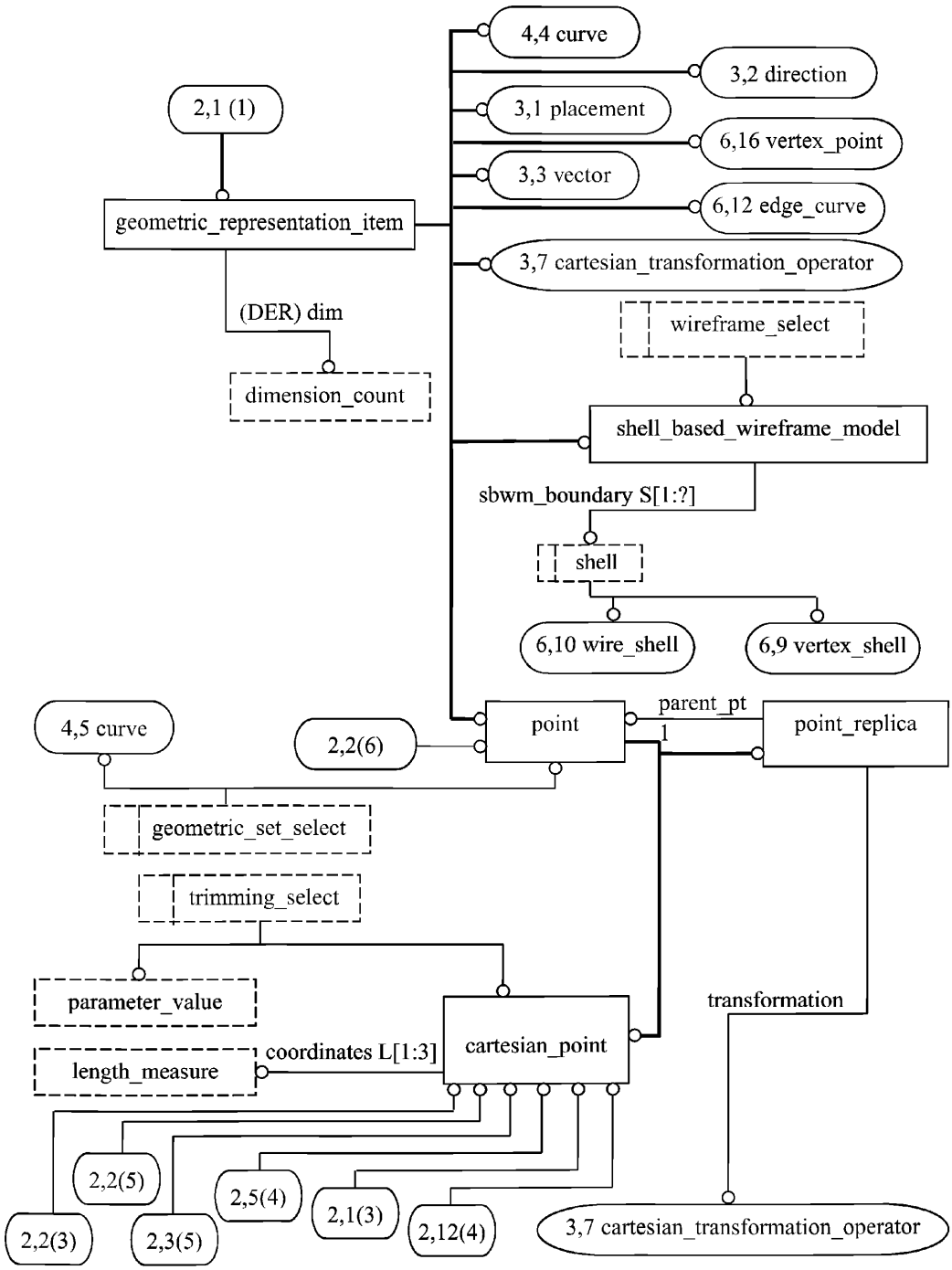

Рисунок С.2 — **aic_shell_based_wireframe** — EXPRESS-G диаграмма 2 из 6

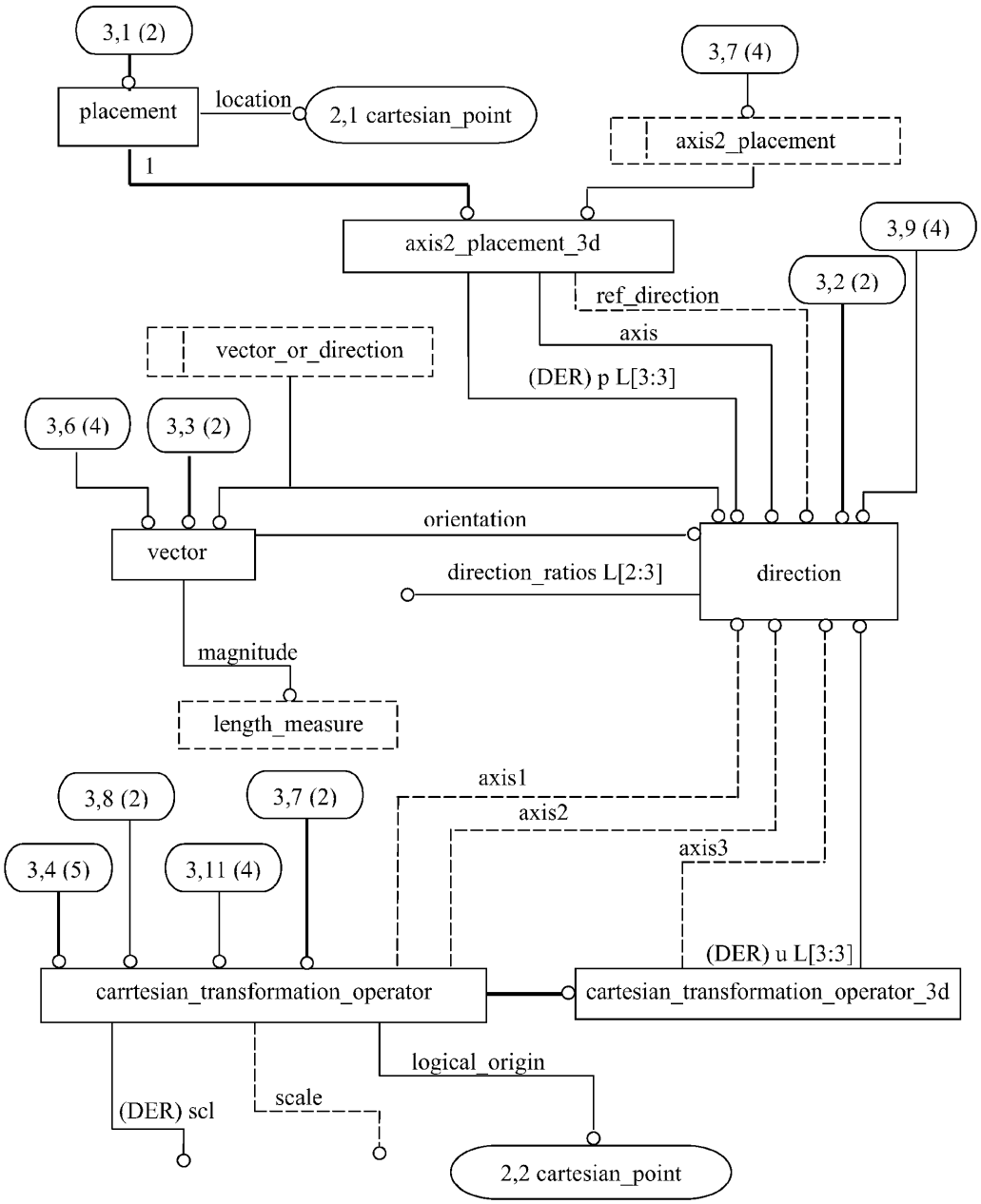

Рисунок С.3 — **aic_shell_based_wireframe** — EXPRESS-G диаграмма 3 из 6

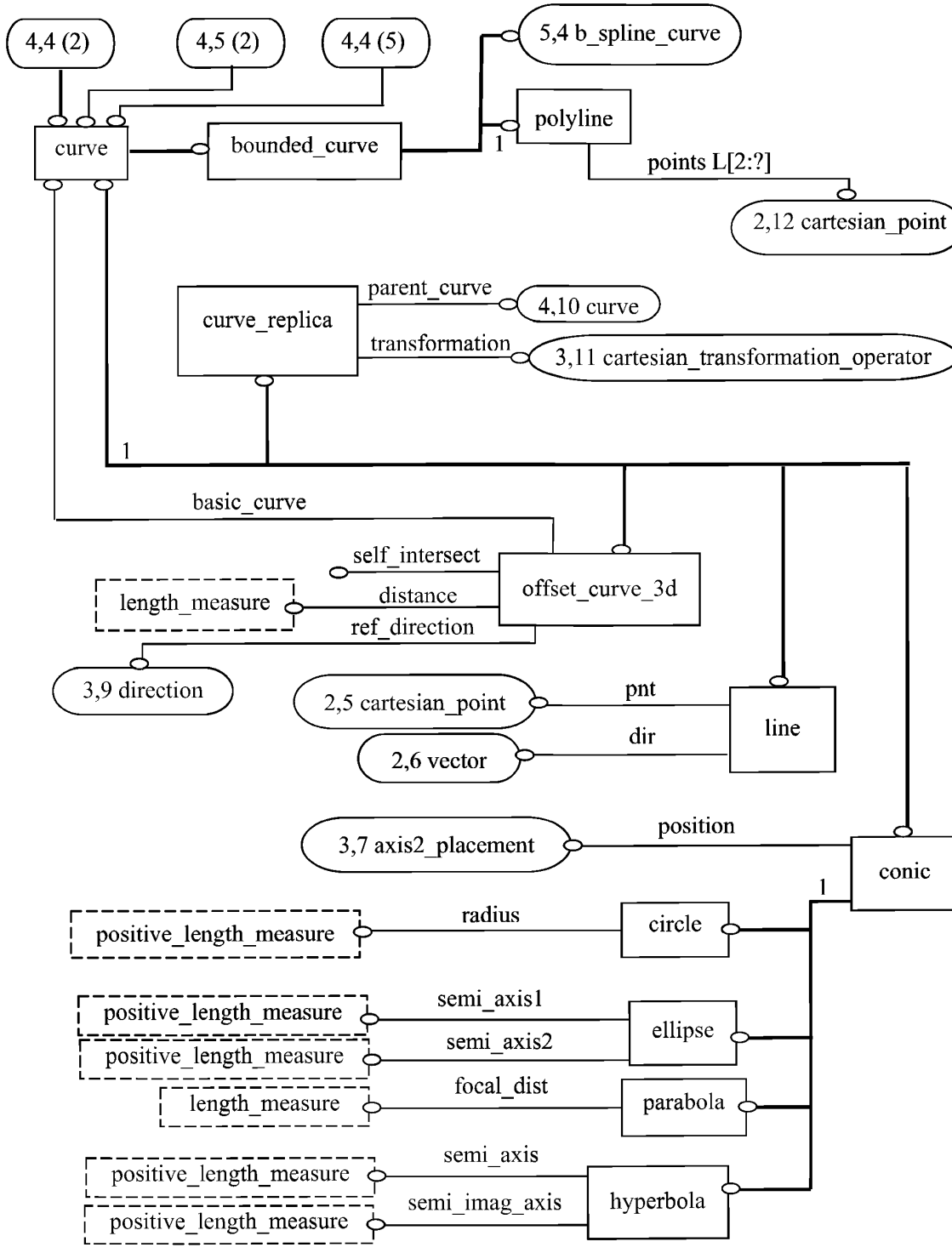

Рисунок С.4 — **aic_shell_based_wireframe** — EXPRESS-G диаграмма 4 из 6

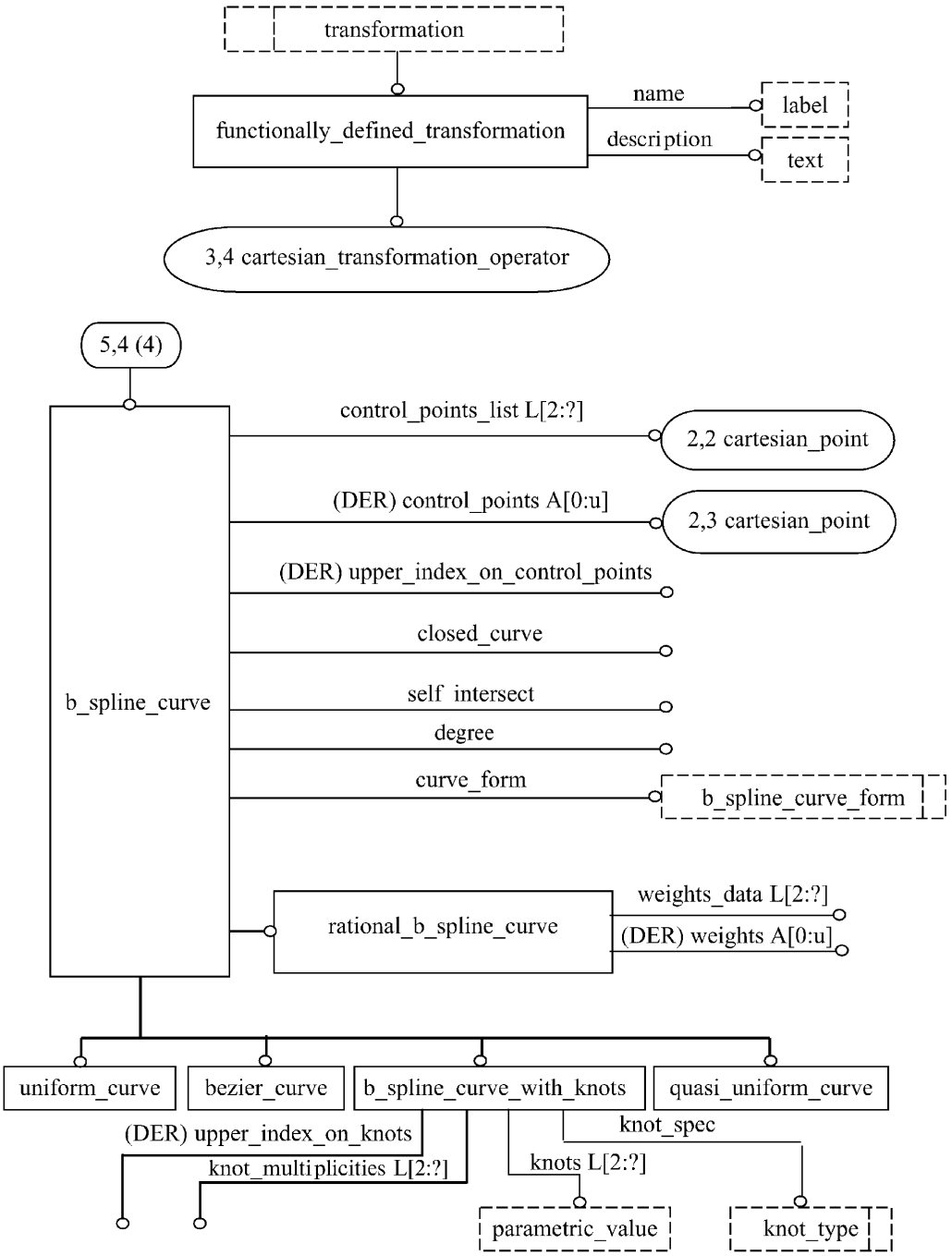

Рисунок С.5 — **aic_shell_based_wireframe** — EXPRESS-G диаграмма 5 из 6

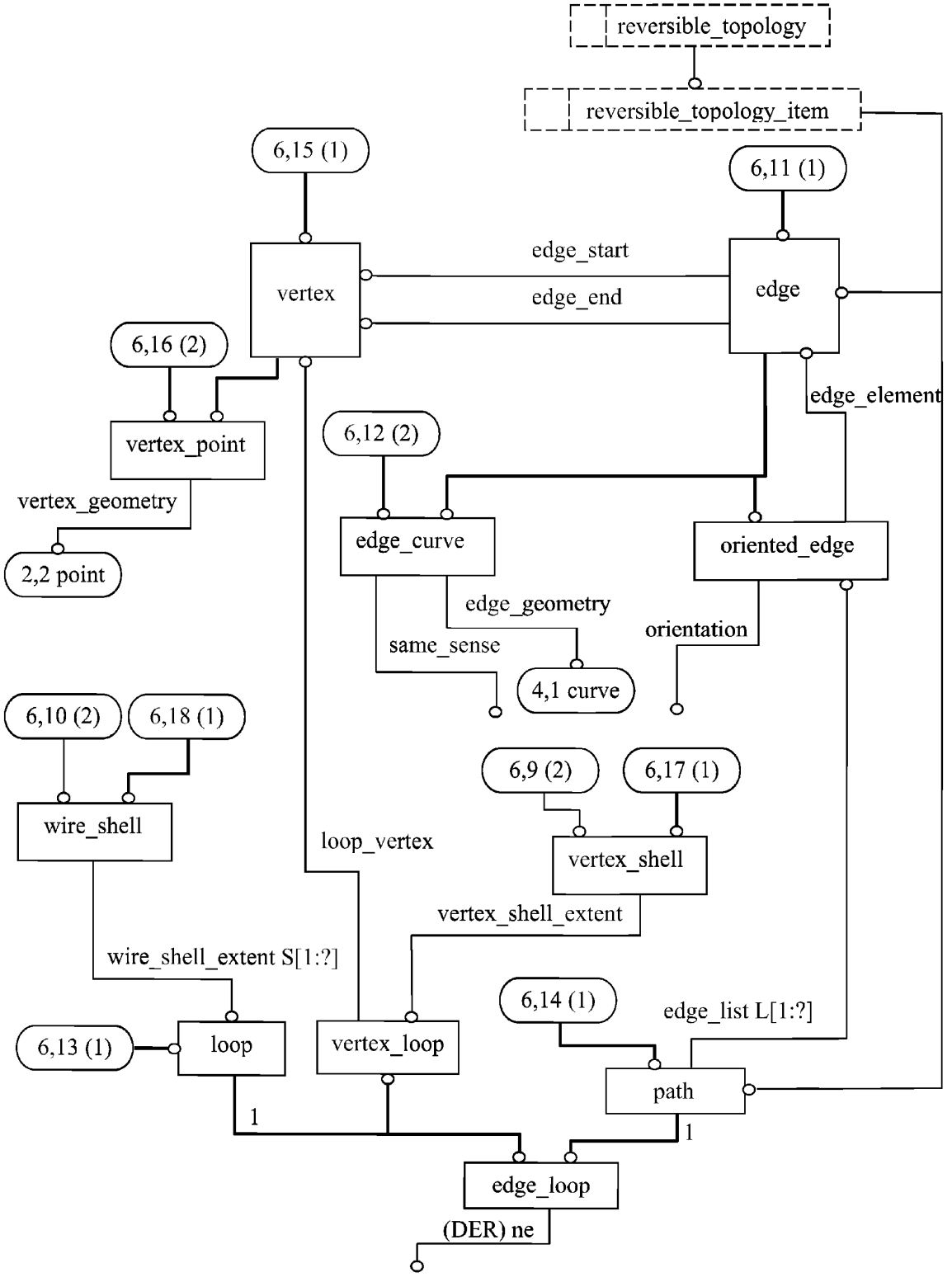

Рисунок С.6 — **aic_shell_based_wireframe** — EXPRESS-G диаграмма 6 из 6

Приложение D
(справочное)

Машинно-интерпретируемые листинги

В данном приложении приведены ссылки на сайты, на которых находятся листинги наименований объектов на языке EXPRESS и соответствующих сокращенных наименований, установленных в настоящем стандарте. На этих же сайтах находятся листинги всех EXPRESS-схем, установленных в настоящем стандарте без комментариев и другого поясняющего текста. Эти листинги доступны в машинно-интерпретируемой форме и могут быть получены по следующим адресам URL:

Сокращенные наименования: gost.e-ecolog.ru
EXPRESS: gost.e-ecolog.ru

При невозможности доступа к этим сайтам необходимо обратиться в центральный секретариат ИСО или непосредственно в секретариат ИСО ТК184/ПК4 по адресу электронной почты: gost.e-ecolog.ru

Примечание — Информация, представленная в машинно-интерпретированном виде на указанных выше URL, является справочной. Обязательным является текст настоящего стандарта.

Приложение Е
(справочное)

Сведения о соответствии национальных стандартов Российской Федерации ссылочным международным стандартам

Таблица Е.1

Обозначение ссылочного международного стандарта	Обозначение и наименование соответствующего национального стандарта
ИСО/МЭК 8824-1:1995	ГОСТ Р ИСО/МЭК 8824-1—2001 Информационная технология. Абстрактная синтаксическая нотация версии один (АСН.1). Часть 1. Спецификация основной нотации
ИСО 10303-1:1994	ГОСТ Р ИСО 10303-1—99 Системы автоматизации производства и их интеграция. Представление данных об изделии и обмен этими данными. Часть 1. Общие представления и основополагающие принципы
ИСО 10303-11:1994	ГОСТ Р ИСО 10303-11—2000 Системы автоматизации производства и их интеграция. Представление данных об изделии и обмен этими данными. Часть 11. Методы описания. Справочное руководство по языку EXPRESS
ИСО 10303-41:1994	ГОСТ Р ИСО 10303-41—99 Системы автоматизации производства и их интеграция. Представление данных об изделии и обмен этими данными. Часть 41. Интегрированные обобщенные ресурсы. Основы описания и поддержки изделий
ИСО 10303-42:1994	*
ИСО 10303-43:1994	ГОСТ Р ИСО 10303-43—2002 Системы автоматизации производства и их интеграция. Представление данных об изделии и обмен этими данными. Часть 43. Интегрированные обобщенные ресурсы. Структуры представлений
ИСО 10303-202:1996	*

* Соответствующий национальный стандарт отсутствует. До его утверждения рекомендуется использовать перевод на русский язык данного международного стандарта. Перевод данного международного стандарта находится в Федеральном информационном фонде технических регламентов и стандартов.

УДК 656.072:681.3:006.354 ОКС 25.040.40 П87 ОКСТУ 4002

Ключевые слова: автоматизация, средства автоматизации, прикладные автоматизированные системы, промышленные изделия, данные, представление данных, обмен данными, прикладные конструкции, формы, каркасное представление, оболочки

Редактор *В.Н. Копысов*
Технический редактор *В.Н. Прусакова*
Корректор *М.И. Першина*
Компьютерная верстка *И.А. Налейкиной*

Сдано в набор 19.02.2007. Подписано в печать 27.03.2007. Формат 60 × 84 1/8. Бумага офсетная. Гарнитура Ариал.
Печать офсетная. Усл. печ. л. 2,79. Уч.-изд. л. 2,20. Тираж 167 экз. Зак. 260. С 3849.

ФГУП «Стандартинформ», 123995 Москва, Гранатный пер., 4.
gost.e-ecolog.ru gost.e-ecolog.ru
Набрано во ФГУП «Стандартинформ» на ПЭВМ.
Отпечатано в филиале ФГУП «Стандартинформ» — тип. «Московский печатник», 105062 Москва, Лялин пер., 6.

gost.e-ecolog.ru