ГОСТ Р ИСО 10303-46—2002

ГОСУДАРСТВЕННЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

Системы автоматизации производства и их интеграция

ПРЕДСТАВЛЕНИЕ ДАННЫХ ОБ ИЗДЕЛИИ И ОБМЕН ЭТИМИ ДАННЫМИ

Часть 46

Интегрированные обобщенные ресурсы Визуальное представление

Издание официальное

БЗ 9—2002/186

ГОССТАНДАРТ РОССИИ
Москва

gost.e-ecolog.ru

Предисловие

1 РАЗРАБОТАН Научно-исследовательским центром (НИЦ) CALS-технологий «Прикладная логистика» и Всероссийским научно-исследовательским институтом стандартизации (ВНИИстандарт)

ВНЕСЕН Техническим комитетом по стандартизации ТК 431 «CALS-технологии»

2 ПРИНЯТ И ВВЕДЕН В ДЕЙСТВИЕ Постановлением Госстандарта России от 20 декабря 2002 г. № 499-ст

3 Настоящий стандарт представляет собой полный аутентичный текст международного стандарта ИСО 10303-46—94 «Системы автоматизации производства и их интеграция. Представление данных об изделии и обмен этими данными. Часть 46. Интегрированные обобщенные ресурсы. Визуальное представление» с учетом Поправки № 1 (1999 г.)

4 ВВЕДЕН ВПЕРВЫЕ

© ИПК Издательство стандартов, 2003

Настоящий стандарт не может быть полностью или частично воспроизведен, тиражирован и распространен в качестве официального издания без разрешения Госстандарта России

Содержание

1 Область применения . . . 1
2 Нормативные ссылки . . . 2
3 Определения и сокращения . . . 2
3.1 Термины, определенные в ГОСТ Р ИСО 10303-1 . . . 2
3.2 Термины, определенные в настоящем стандарте . . . 2
3.3 Сокращения . . . 3
4 Структура представления . . . 3
4.1 Введение . . . 4
4.2 Фундаментальные понятия и допущения . . . 6
4.3 Определения типов схемы структуры представления . . . 9
4.4 Определения объектов схемы представления: иерархия представления . . . 10
4.5 Определение объектов схемы структуры представления: модель камеры и проекция . . . 17
4.6 Определение объектов схемы структуры представления: уровни . . . 25
4.7 Определения объектов схемы структуры представления: связь представления с моделью изделия . . . 27
4.8 Определение правила схемы структуры представления . . . 27
4.9 Определение функции схемы структуры представления . . . 28
5 Описание представления . . . 29
5.1 Введение . . . 30
5.2 Фундаментальные понятия и допущения . . . 30
5.3 Определения типов схемы описания представления . . . 31
5.4 Определение объектов схемы описания представления: примитивы комментариев . . . 33
5.5 Определения объектов схемы описания представления: экземпляры аннотаций . . . 46
5.6 Определения функций схемы описания представления . . . 50
6 Вид представления . . . 53
6.1 Введение . . . 54
6.2 Фундаментальные понятия и допущения . . . 54
6.3 Определение типов схемы вида представления . . . 57
6.4 Определение объектов схемы вида представления: задание стиля . . . 69
6.5 Определения объектов схемы вида представления: стили представления для точек . . . 72
6.6 Определения объектов схемы вида представления: стили представления для кривых линий . . . 73
6.7 Определения объектов схемы вида представления: стили представления для заполненных областей . . . 77
6.8 Определения объектов схемы вида представления: стили представления для поверхностей . . . 82
6.9 Определения объектов схемы вида представления: стили представления для текста . . . 87
6.10 Определение объектов схемы вида представления: стили представления для символов 90
6.11 Определение объектов схемы вида представления: допуски аппроксимации . . . 91
6.12 Определения объектов схемы вида представления: сокрытие и видимость . . . 92
6.13 Определение функции схемы вида представления . . . 93
7 Схема ресурсов представления . . . 94
7.1 Введение . . . 95
7.2 Определения типов схемы ресурсов представления . . . 95
7.3 Определения объектов схемы ресурсов представления . . . 95
Приложение А Сокращенные наименования объектов . . . 103

Приложение B Регистрация информационного объекта 107
B.1 Обозначение документа 107
B.2 Обозначение схемы 107
Приложение C Машинно-интерпретируемые листинги 108
Приложение D Технические вопросы 108
D.1 Символы, используемые в формулах отражающей способности 108
D.2 Рекомендуемые формулы отражающей способности 109
Приложение E EXPRESS-G диаграммы 110
Приложение F Библиография 149
Тематический указатель 150

Введение

Стандарты серии ГОСТ Р ИСО 10303 распространяются на машинно-ориентированное представление данных об изделии и обмен этими данными. Целью является создание механизма, позволяющего описывать данные об изделии на протяжении всего его жизненного цикла независимо от конкретной системы. Характер такого описания делает его пригодным не только для обмена инвариантными файлами, но также и для создания баз данных об изделиях, коллективного пользования этими базами и архивирования соответствующих данных.

Стандарты серии ГОСТ Р ИСО 10303 представляют собой набор отдельно издаваемых стандартов (частей). Части данной серии стандартов относятся к одной из следующих тематических групп: методы описания, интегрированные ресурсы, прикладные протоколы, комплекты абстрактных тестов, методы реализации и аттестационное тестирование. Группы стандартов данной серии описаны в ГОСТ Р ИСО 10303-1. Настоящий стандарт входит в группу интегрированных ресурсов. Основными разделами настоящего стандарта являются:

- структура представления;
- описание представления;
- вид представления;
- ресурсы представления.

Настоящий стандарт определяет интегрированные ресурсы для визуализации свойств изделий, воспроизводимых на экране дисплея.

Информация, задаваемая в четырех схемах, определенных в настоящем стандарте, достаточна для подробного описания правил и порядка визуального представления информации об изделии в принимающей системе. Информацию о представлении, описанную в настоящем стандарте, следует использовать только вместе с подлежащей визуализации информацией об изделии. Информация о представлении не может быть отображена на экране без связи с соответствующей информацией об изделии.

Схема структуры представления (**presentation_organization_schema**) описывает иерархию и частичную рекурсию структуры наборов, областей и видов представлений, при помощи которых информацию об изделии отображают на экране. Данная схема уточняет как компоненты изображения информации об изделии и пояснения к ним компонуют в отображаемые объекты и располагают в контексте представлений. Схема также определяет процесс построения геометрических проекций посредством модели камеры и требования к моделям распространения света и затенения.

Схема описания представления (**presentation_definition_schema**) служит для определения, как отдельные геометрические и негеометрические компоненты информации об изделии должны быть выбраны, скомпонованы в группы представлений и связаны со стилями представлений.

Схема вида представления (**presentation_appearance_schema**) определяет видовые атрибуты, которые могут быть выбраны для описания желаемого визуального вида отображаемых элементов информации об изделии посредством перечисления доступных стилей графического представления.

Схема ресурсов представления (**presentation_resource_schema**) определяет основные графические возможности, например описание текстовых шрифтов, символов и цветов.

Характеристики визуального представления, описанные в настоящем стандарте, часто связаны с другими стандартами по обобщенным ресурсам, особенно в части геометрических и топологических представлений. Порядок совместного использования этих стандартов определяют в прикладных протоколах. Приложения, в которых используют обобщенные ресурсы из настоящего стандарта, должны определять информацию об изделии, подлежащую визуальному представлению, и семантический смысл такого представления. Подобными приложениями, например, являются: изображаемые виды формы изделия, визуализация научных результатов, технические чертежи, схемы, диаграммы и графики для научно-технических изданий.

В настоящем стандарте ряд понятий, отображающих конструктивы на языке EXPRESS, набран полужирным шрифтом (например, **product_definition**).

Примечание — Основная часть стандарта дополнена следующими приложениями:

- А — содержащим сокращенные наименования объектов;
- В — описывающим идентификаторы информационных объектов, присвоенные объектам настоящего стандарта;

- C — описывающим порядок получения машинно-интерпретируемых листингов объектов, определенных в настоящем стандарте;
- D — обсуждающем ряд технических вопросов;
- E — содержащим схематические описания диаграмм объектов настоящего стандарта на языке EXPRESS-G;
- F — содержащим библиографию дополнительных публикаций, связанных с настоящим стандартом.

Связь с графическими стандартами

Интегрированные ресурсы, описанные в настоящем стандарте, обеспечивают визуальное представление свойств изделий. Для создания визуальных изображений с использованием данных, определяемых этими ресурсами, необходимо применение соответствующей системы отображения. В настоящем стандарте определены исходные данные для таких систем, а также необходимые структуры и конструкции, связывающие данные о представлении с другими аспектами данных об изделии.

Большинство существующих систем отображения соответствует стандартам ИСО по машинной графике, например ИСО/МЭК 8805 [1] и серии ИСО/МЭК 9592 [2—5]. В настоящем стандарте учтены основные положения и терминология этих стандартов, а также предусмотрены возможности использования других стандартов по машинной графике.

ГОСУДАРСТВЕННЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

Системы автоматизации производства и их интеграция

ПРЕДСТАВЛЕНИЕ ДАННЫХ ОБ ИЗДЕЛИИ И ОБМЕН ЭТИМИ ДАННЫМИ

Часть 46

Интегрированные обобщенные ресурсы
Визуальное представление

Industrial automation systems and integration. Product data representation and exchange.
Part 46. Integrated generic resources. Visual presentation

Дата введения 2003—07—01

1 Область применения

Настоящий стандарт устанавливает интегрированные ресурсы для визуализации информации об изделии, подлежащей отображению на экране дисплея. Данные о представлении в соответствии с настоящим стандартом связывают с данными об изделии и обмениваются этими данными между системами, могущими создавать ряд изображений информации об изделии, пригодных для восприятия человеком.

Стандарт определяет обобщенные ресурсы, необходимые для описания требуемых видов визуальной информации об изделии, подлежащей представлению в качестве изображений. Вопросы практического создания конкретного изображения на основе информации об изделии и связанных с ней данных о представлении оставлены на усмотрение соответствующей принимающей системы. Фактический вид изображения может отличаться от заданного вследствие ограничений функциональных возможностей графических систем.

Информация об изделии может быть визуализирована двумя методами: либо посредством создания реальных изображений в соответствии с правилами проекционной геометрии, распространения и отражения света, либо в символическом виде согласно стандартам и условным обозначениям, принятым для черчения. Настоящий стандарт обеспечивает возможности реализации обоих типов представлений. Оба типа процессов визуализации требуют графических преобразований различных видов, которые могут сочетаться в одном и том же изображении.

Настоящий стандарт устанавливает:

- связи между данными об изделии, определенными в других стандартах серии ГОСТ Р ИСО 10303, данными о представлении;
- методы обеспечения графических функциональных возможностей в соответствии с действующими графическими стандартами;
- определения атрибутов стилей представления для реальных и символических отображений геометрических и негеометрических элементов изображения информации об изделии;
- пределы допустимых отклонений (допусков) для элементов графических представлений;
- методы определения видов знаков и символов в шрифтах;
- методы введения внешне определенных шрифтов и символов;
- контроль изображения посредством уровневого механизма;
- правила вложенности областей представлений.

Настоящий стандарт не устанавливает:

- описание информации об изделии;

Издание официальное

- правила обмена исключительно графической информацией вне связи с информацией об изделии;
- описание содержания библиотек шрифтов и символов.

2 Нормативные ссылки

В настоящем стандарте использованы ссылки на следующие стандарты:

ГОСТ Р ИСО/МЭК 8824-1—2001 Информационная технология. Абстрактная синтаксическая нотация версии один (АСН.1). Часть 1. Спецификация основной нотации

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

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

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

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 10303-42—94* Системы автоматизации производства и их интеграция. Представление данных об изделии и обмен этими данными. Часть 42. Интегрированные обобщенные ресурсы. Геометрическое и топологическое представления

3 Определения и сокращения

3.1 Термины, определенные в ГОСТ Р ИСО 10303-1

В настоящем стандарте использованы следующие термины:

- **данные об изделии;**
- **данные;**
- **изделие;**
- **интегрированный ресурс;**
- **информация об изделии;**
- **информация;**
- **обмен данными;**
- **обобщенный ресурс;**
- **представление;**
- **приложение;**
- **структура.**

3.2 Термины, определенные в настоящем стандарте

В настоящем стандарте использованы следующие термины с соответствующими определениями:

3.2.1 **аннотация** (annotation): Текст и (или) символика, используемые при обмене информацией об изделии.

3.2.2 **отображаемая информация об изделии** (displayable product information): Факты, положения или инструкции по изделию, отображаемые посредством процесса визуализации.

Пример 1 — Отображаемой информацией об изделии являются его свойства, например форма, размеры, допуски и материалы.

---

* Оригинал международного стандарта — во ВНИИКИ Госстандарта России.

3.2.3 **уровень** (layer): Набор отображаемых элементов, предназначенных для управления стилем отображения и представления.

3.2.4 **изображение** (picture): Двумерное графическое представление свойств изделия, предназначенное для восприятия человеком.

3.2.5 **информация о представлении** (presentation information): Информация, предназначенная для создания представления об изделии посредством визуализации. Информация данного вида имеет смысловое значение только при непосредственной увязке с информацией об изделии.

3.2.6 **реальное представление свойств** (realistic presentation of properties): Тип визуализации, позволяющий применять физические законы и математические принципы для создания правдоподобных образов изделия. Представления данного вида создают посредством применения преобразований перспективы, вычислений отражений, теневых эффектов и цветового оформления.

3.2.7 **статическая переменная** (state variable): Переменная, представляемая в количественном виде, например температура.

3.2.8 **символ** (symbol): Метка или знаки, трактуемые как условное обозначение некоторого предмета, идей, функции или процесса.

3.2.9 **символическое представление свойств** (symbolic presentation of properties): Тип визуализации, предназначенный для создания аннотации посредством использования технического опыта и соответствующих принципов.

3.2.10 **искусственная модель камеры** (synthetic camera model): Модель, описывающая процесс отображения формы изделия в двумерном пространстве. В этой модели используют теоретический процесс, описывающий работу фотокамеры.

3.2.11 **визуализация** (visualization): Процесс использования отображаемой информации об изделии и информации о представлении для создания отображения.

3.3 Сокращения

В настоящем стандарте использованы следующие сокращения:

CIE — Международная комиссия по цветопередаче (Comission Internationale de l'Eclairage). Используют для ссылок на универсальную систему CIE для определения цвета;

HLS — цветовое пространство цвет — яркость — насыщенность (Hue, Lightness, Saturation color space);

HSV — цветовое пространство цвет — насыщенность — значение (Hue, Saturation, Value color space);

RGB — цветовое пространство красный — зеленый — синий (Red, Green, Blue color space).

4 Структура представления

Следующее описание на языке EXPRESS открывает **presentation_organization_schema** и определяет необходимые внешние ссылки.

EXPRESS-спецификация

```
*)
SCHEMA presentation_organization_schema;
REFERENCE FROM presentation_definition_schema
    (annotation_occurence,
     symbol_representation,
     symbol_representation_relationship);
REFERENCE FROM presentation_appearance_schema
    (styled_item);
REFERENCE FROM presentation_resource_schema
    (colour,
     planar_box,
     presentation_scaled_placement);
REFERENCE FROM geometry_schema
    (axis2_placement_2d,
```

```
        axis2_placement_3d,
        cartesian_point,
        curve,
        direction,
        dot_product,
        geometric_representation_context,
        geometric_representation_item,
        plane
        );
REFERENCE FROM representation_schema
        (founded_item,
        item_defined_transformation,
        item_in_context,
        mapped_item,
        representation,
        representation_item,
        representation_map,
        representation_relationship,
        representation_relationship_with_transformation);
REFERENCE FROM measure_schema
        (length_measure,
        positive_plane_angle_measure);
REFERENCE FROM support_resource_schema
        (identifier,
        label,
        text,
        bad_to_set);
(*
```

Примечания

1 Схемы, на которые выше даны ссылки, можно найти в следующих стандартах:

presentation_definition_schema	раздел 5 настоящего стандарта;
presentation_appearance_schema	раздел 6 настоящего стандарта;
presentation_resource_schema	раздел 7 настоящего стандарта;
geometry_schema	ИСО 10303-42;
representation_schema	ГОСТ Р ИСО 10303-43;
measure_schema	ГОСТ Р ИСО 10303-41;
support_resource_schema	ГОСТ Р ИСО 10303-41.

2 Графическое представление определяемой схемы приведено в приложении Е с использованием нотации языка EXPRESS-G.

4.1 Введение

Схема **representation_organization_schema** представляет структуру для управления самим изображением и его компонентами. Она также устанавливает отношение между свойствами изделия и их представлениями в изображении. Компоненты изображения могут быть связаны друг с другом либо путем взаимоувязки двух компонентов, являющихся в других случаях независимыми, либо компонентами, один из которых является элементом определения другого. Эти отношения позволяют создавать сложные структуры изображений и их компонентов.

Компоненты изображения могут быть организованы в соответствии с иерархией, построенной на наборах представлений, областях представлений, аннотациях представлений, связанных с их областями, видах представлений, аннотациях представлений, связанных с их видами, и видами представлений данных об изделии. Данная иерархия состоит из следующих четырех уровней.

Уровень 1 — набор представлений (presentation set) множества (коллекции) независимых изображений, относящихся к одному и тому же субъекту. Примерами наборов представлений являются

комплект чертежей или коллекция экранных изображений. Набор образует одну или несколько областей представления.

Уровень 2 — область представления (presentation area), являющаяся обобщенным описанием дисплея, на экране которого представляется одна из возможных областей. Область представления может содержать любое число вложенных областей представлений, видов представлений и аннотированных представлений, связанных с соответствующими областями.

Уровень 3 — вид представления (presentation view), являющийся двумерным представлением формы изделия, в которое входит любая аннотация, связанная с этим видом. Вид представления может содержать любое количество видов представления данных об изделии и представлений аннотаций, связанных с этими видами.

Уровень 4:

- вид представления данных об изделии (product data representation view), являющийся двумерным представлением формы изделия, включающим в себя любые аннотации, связанные с формой изделия;

- **представление аннотации, связанное с видом представления** (view-dependent annotation representation).

Фактическая иерархия представления может состоять из более чем четырех уровней. Единая область представления может быть образована из нескольких областей. Иерархия представления может состоять из менее чем четырех уровней, потому что отдельные компоненты иерархии могут быть не востребованы.

Примечание — На рисунках 1 и 2 показаны отношения между уровнями иерархии представления.



Рисунок 1 — Иерархия представления

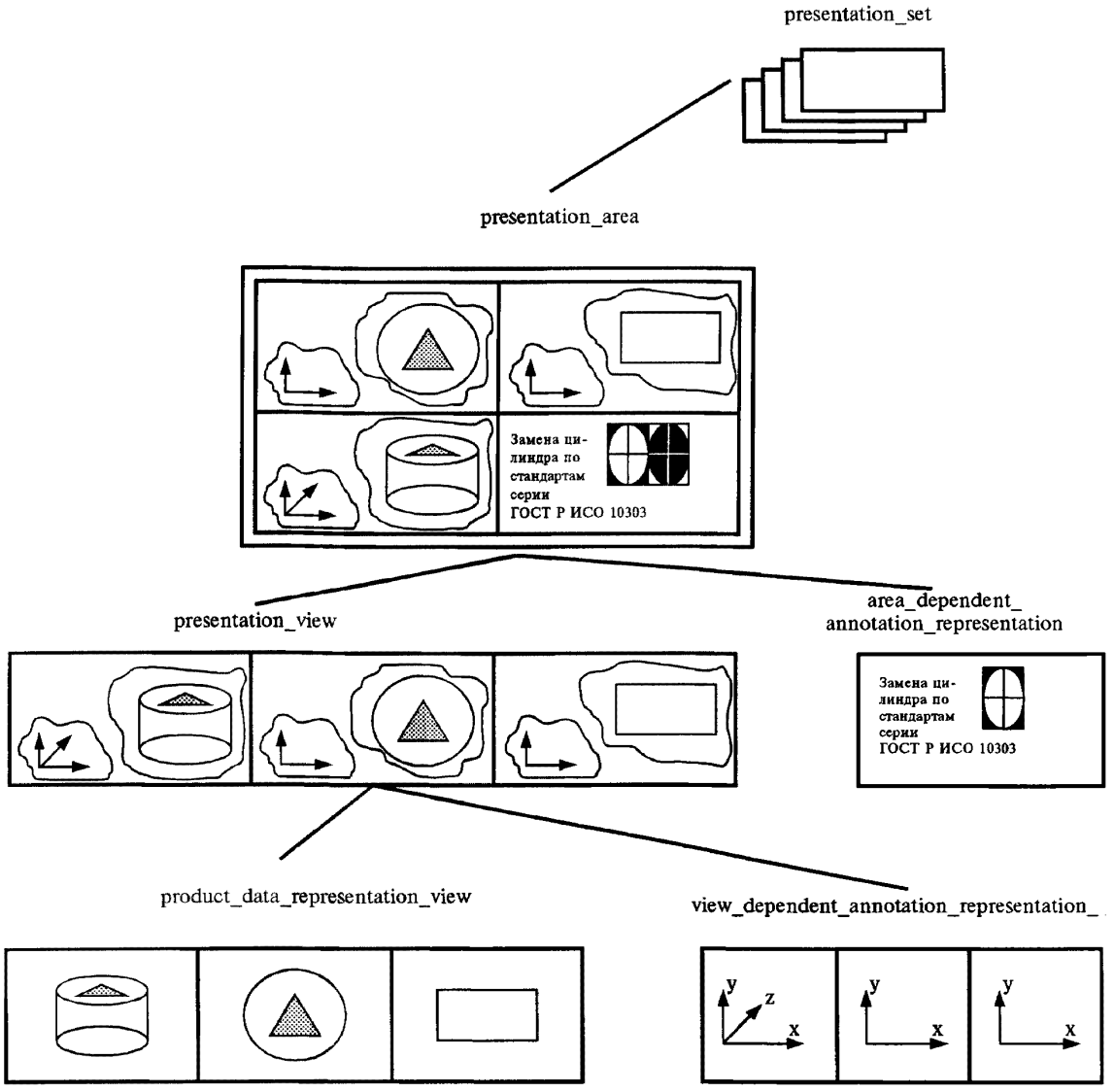


Рисунок 2 — Пример иерархии представления

4.2 Фундаментальные понятия и допущения

4.2.1 Иерархия представления

В рамках иерархии представления только область представления содержит информацию, достаточную для однозначной генерации изображения на устройствах вывода, например экране дисплея, принтере или плоттере.

Другие элементы иерархии не содержат информации, необходимой для размещения изображения на соответствующем устройстве вывода. Виды представления или другие компоненты низших уровней иерархии могут быть отображены только при условии их прямой или косвенной увязки с областью представления.

Схема **presentation_organization_schema** описывает иерархию представления посредством различных элементов иерархии и соотношений между ними.

Примечание 1 — Концепции представлений и отношения между ними описаны в ГОСТ Р ИСО 10303-43.

Каждый элемент иерархии представления должен быть описан посредством объекта **presentation_representation** с использованием подтипа **representation**. Подтипы объекта **presentation_representation** описывают различные элементы иерархии.

Примечание 2 — Объект **representation** определен в ГОСТ Р ИСО 10303-43.

Контекст каждого элемента иерархии представления должен быть описан посредством объекта **geometric_representation_context,** ограниченного двумерным пространством.

Примечание 3 — Объект **geometric_representation_context** определен в ИСО 10303-42.

Контекст каждого элемента иерархии представления следует описывать посредством набора атрибутов **items** каждого объекта **presentation_representation.** Конкретные **items** определяют двумерную геометрию или аннотацию, связанные с данным элементом, или отражают результаты включения других элементов в данный.

Некоторые элементы иерархии могут быть ограничены по содержанию или в отношении их роли в других элементах. Семантики конкретных элементов должны быть связаны с этими ограничениями; например объект **product_data_representation_view** описывает элемент иерархии, отражающий результаты использования трехмерной геометрии или соответствующей аннотации.

Наивысший уровень иерархии представления должен быть отображен объектами **presentation_set.presentation_area** и **area_in_set.** Объект **area_in_set** обеспечивает многочисленные отношения между наборами представлений и соответствующими областями.

Другие элементы иерархии представления, связанные друг с другом, используют объект **presentation_representation_relationship** или объекты **mapped_item** и **representation_map.**

Примечание 4 — Объекты **mapped_item** и **representation_map** определены в ГОСТ Р ИСО 10303-43.

Связь между двумя независимыми элементами иерархии определяют посредством описаний применения объекта **presentation_representation_relationship**. Тем самым описывают отношение между двумя экземплярами объекта **presentation_representation**; в данном отношении один из этих объектов является «родителем», а другой — «потомком». В данное отношение должно быть включено описание соответствующего преобразования, отражающего геометрическое отношение между атрибутами **items** «родительского» объекта **presentation_representation** и соответствующими атрибутами «потомственного» объекта **presentation_representation**.

Пример 2 — Для определения иерархии, содержащей единственную **presentation_area**, два **presentation_view** и единственное **area_dependent_annotation_representation**, необходимы три экземпляра **presentation_representation_relationship**, как показано на рисунке 3.

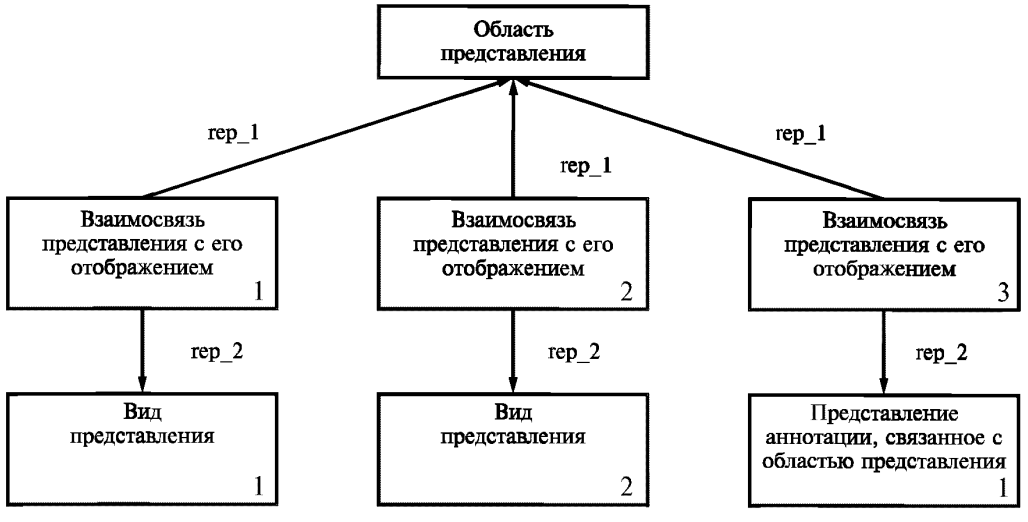


Рисунок 3 — Отображение иерархии представления на экземпляры объектов

Связь между двумя элементами иерархии, один из которых входит в определение другого, следует описывать с использованием объектов **mapped_item** и **representation_map**.

Такая связь может быть описана экземпляром объекта **mapped_item**. Данный экземпляр может быть включен в качестве одного из атрибутов **items** конкретного объекта **presentation_representation**, содержащего другой аналогичный объект. На второй объект **presentation_representation** ссылаются посредством атрибута **mapped_representation** объекта **representation_map**, который определен в качестве атрибута **mapping_source** объекта **mapped_item**. Преобразование, описывающее геометрическое отношение между атрибутами **items** двух объектов **presentation_representation**, следует определять посредством атрибута **mapping_target объекта mapped_item** и атрибута **mapping_origin** объекта **representation_map**.

Пример 3 — Для включения объекта **presentation_view** в **presentation_area** необходимы: один экземпляр объекта **mapped_item**, один экземпляр объекта **representation_map** и два экземпляра объекта **axis2_placement_2d**, как показано на рисунке 4.

Примечание 5 — Объект **axis2_placement_2d** определен в ИСО 10303-42.

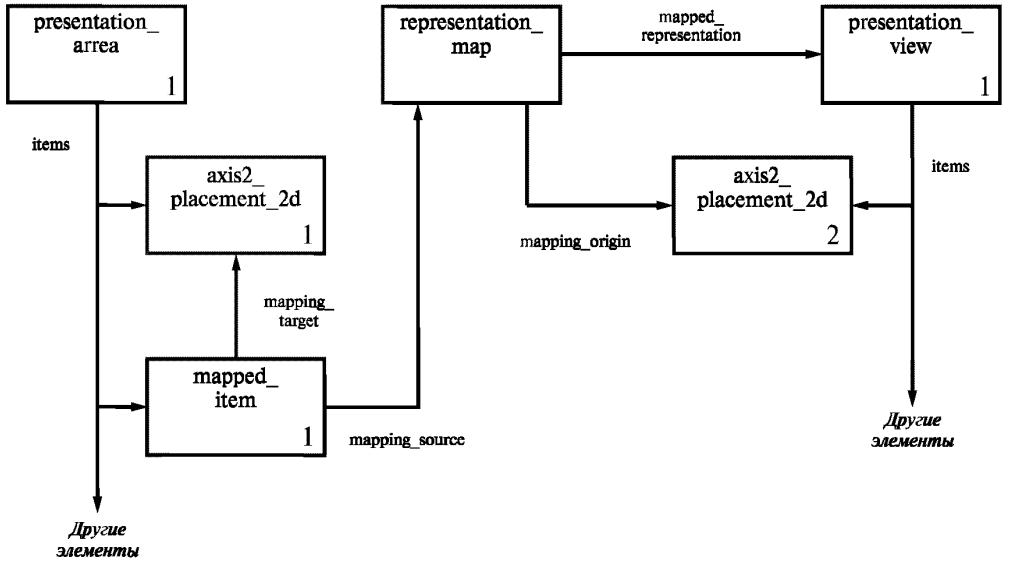


Рисунок 4 — Связь между **presentation_view** и **presentation_area** с использованием **mapped_item**

4.2.2 Модель камеры и проекции

Каждый уровень иерархии представления содержит только двумерный объект или аннотацию. Дополнительно должна быть определена искусственная модель трехмерной камеры для установления связи представления с трехмерной формой изделия или плоскостной аннотацией в трехмерном пространстве. Данная модель должна определять правила проектирования трехмерных геометрических объектов и аннотации в двумерном пространстве для системы отображения. Искусственная модель трехмерной камеры должна быть описана в соответствии с требованиями стандартов трехмерной машинной графики. Более подробные сведения об искусственной модели камеры могут быть найдены в [1, 2—5] и в литературе по машинной графике, например в [6] или [7]. Для обеспечения реализации требований к черчению также должна быть описана модель двумерной камеры. Данная модель должна обеспечивать масштабирование и перевод формы изделия в двумерное пространство.

Объект **product_data_representation_view** является только компонентом иерархии представления, могущим содержать двумерные проекции, связанные с трехмерной формой изделия, или аннотации в трехмерном пространстве. Так как настоящий стандарт не определяет спроектированное изображение, но устанавливает всю информацию, необходимую для вычисления проекции, в содержание **product_data_representation_view** включена только метка данного изображения. Данная метка названа **camera_image** и ссылается на модель камеры и дву- или трехмерную форму изделия или аннотацию.

Для более реалистичного представления трехмерных предметов в трехмерной модели камеры может быть определен порядок переноса скрытых линий и поверхностей, а также источников освещения.

4.2.3 Уровни

Уровень является набором (коллекцией) элементов формы изделия, аннотации или компонентов иерархии представления, созданным с целью управления видимостью и стилем представления. Набор элементов, связанных с уровнем, может быть определен видимым или невидимым в компоненте иерархии представления. Единственный элемент может быть связан с несколькими уровнями, а единственный уровень может быть использован в нескольких компонентах иерархии представления. Для элемента, связанного с несколькими уровнями, могут быть определены различные стили его представления на каждом уровне. Уровень должен быть определен посредством объекта **presentation_layer_assignment.** Видимость и стиль заданного уровня в компоненте иерархии представления определяют посредством объекта **presentation_layer_usage**.

4.2.4 Связь представления с моделью изделия

Настоящий стандарт обеспечивает связь информации о представлении с информацией о представляемом изделии. Связь осуществляется посредством использования объекта **presented_item_representation**. Данный объект связывает конкретный элемент с его представлением. Конкретный атрибут **presented_item** должен быть определен в прикладных протоколах.

Пример 4 — Элементами представления являются корпус автомобиля, расположение в нем электронных схем и его архитектура.

4.3 Определения типов схемы структуры представления

4.3.1 Тип presentation_size_assignment_select

Тип **presentation_size_assignment_select** определяет предметы, для которых могут быть указаны размеры.

EXPRESS-спецификация

```
*)
TYPE presentation_size_assignment_select = SELECT
    (presentation_view,
     presentation_area,
     area_in_set);
END_TYPE;
(*
```

4.3.2 Тип area_or_view

Тип **area_or_view** указывает предметы, которые могут быть выделены основным (фоновым) цветом.

EXPRESS-спецификация

```
*)
TYPE area_or_view = SELECT
    (presentation_area,
     presentation_view);
END_TYPE;
(*
```

4.3.3 Тип central_or_parallel

Тип **central_or_parallel** определяет тип преобразования перспективы, используемый объектом **camera_model.** Данными типами являются либо параллельное проецирование на рассматриваемую плоскость, либо центральное проектирование из объекта **projection_point** на рассматриваемую плоскость (см. рисунки 7 и 8).

EXPRESS-спецификация

```
*)
TYPE central_or_parallel = ENUMERATION OF
    (central,
     parallel);
END_TYPE;
(*
```

Определения перечисляемых элементов

central — проекция выполняется по линиям, исходящим из объекта **projection_point** и связанным с объектом **view_window**;

parallel — проекция выполняется параллельно линии, исходящей из объекта **projection_point**, на геометрический цент объекта **view_window**.

4.3.4 Тип layered_item

Тип **layered_item** определяет элементы, сгруппированные в уровнях посредством объекта **presentation_layer_assigment**.

EXPRESS-спецификация

```
*)
TYPE layered_item = SELECT
    (presentation_representation,
     representation_item);
END_TYPE;
(*
```

4.3.5 Тип presentation_representation_select

Тип **presentation_represantation_select** предназначен для обеспечения связи представления с представляемым элементом посредством объекта **presentation_item_representation**.

EXPRESS-спецификация

```
*)
TYPE presentation_representation_select = SELECT
    (presentation_representation,
     presentation_set);
END_TYPE;
(*
```

4.4 Определения объектов схемы представления: иерархия представления

4.4.1 Объект presentation_set

Объект **presentation_set** является набором (коллекцией) объектов **presentation_areas**.

EXPRESS-спецификация

```
*)
ENTITY presentation_set;
INVERSE
    areas: SET [1 : ?] OF area_in_set FOR in_set;
END_ENTITY;
(*
```

Описание атрибута

areas — набор объектов **presentation_area**, образующих **presentation_set**.

4.4.2 Объект presentation_representation

Объект **presentation_representation** содержит описание изображения, генерируемого системой, отображающей информацию о представлении. Данное изображение связано с предметом, представляемым в виде дву- или трехмерного геометрического объекта или аннотации. Ссылка на геометрические объекты или аннотацию может быть введена косвенно посредством включения в иерархию объекта **presentation_representation**. Нижние уровни подобной иерархии могут быть ограничены только по составу геометрических изображений или аннотаций.

Примечание 1 — Описание иерархии представления приведено в 4.1.

EXPRESS-спецификация

```
*)
ENTITY presentation_representation
    SUBTYPE OF (representation);
WHERE
    WR1: SELF \representation.
            context_of_items\geometric_representation_context.
            coordinate_space_dimension = 2;
    WR2: 'GEOMETRY_SCHEMA.GEOMETRIC_REPRESENTATION_CONTEXT'
```

```
        IN TYPEOF (SELF\representation.context_of_items);
END_ENTITY;
(*
```

Формальные утверждения

WR1 — изображение должно иметь размерность, равную 2;

WR2 — объект **presentation_representation** должен содержать геометрический контекст.

Примечания

2 При представлении трехмерного объекта следует использовать его двумерные проекции. Объект **presentation_representation** содержит изображения спроектированных предметов.

3 В настоящем стандарте не описаны реальные геометрические проекции. Вместе с тем в модель входит объект **presentation_representation**, определяющий правила геометрического проецирования и информацию, необходимую для создания проекции.

4.4.3 Объект presentation_area

Объект **presentation_area** представляет изображение, могущее содержать другие вложенные изображения, а также входить в другое изображение, представленное аналогичным объектом **presentation_area**.

Размер объекта **presentation_area** определяют через объект **presentation_size**. В каждом объекте **presentation_area** размер задают прямо, посредством использования объекта **presentation_size**, или косвенно с использованием объектов **area_in_set**. Выделение изображения в размерах, заданных объектом **presentation_area**, осуществляет система, создающая реальное изображение на основе информации о представлении. Если данный объект **presentation_area** входит в другой аналогичный объект, тогда ограничения, определяющие формат данного объекта, могут быть использованы для его выделения из другого объекта.

Основной (фоновый) может быть определен путем использования **presentation_area** в объекте **background_colour.** Только в одном из объектов **background_colour** может быть использован какой-либо объект **presentation_area**.

EXPRESS-спецификация

```
*)
ENTITY presentation_area
    SUBTYPE OF (presentation_representation);
WHERE
    WR1: ((SIZEOF (QUERY (ais<* USEDIN (SELF, 'PRESENTATION_ORGANIZATION_SCHEMA.'
                                          +'AREA_IN_SET.AREA')|
            SIZEOF (USEDIN (ais, 'PRESENTATION_ORGANIZATION_SCHEMA.' +
                                  'PRESENTATION_SIZE.UNIT')) = 1)) > 0) OR
        (SIZEOF (USEDIN (SELF, 'PRESENTATION_ORGANIZATION_SCHEMA.' +
                                  'PRESENTATION_SIZE.UNIT')) = 1));
END_ENTITY;
(*
```

Формальное утверждение

WR1 — объект **presentation_area** должен входить в **presentation_set** посредством включения в атрибут **area_in_set.area**, в котором объект **area_in_set** входит в атрибут **presentation_size.unit**, или размер **presentation_area** должен быть определен непосредственно из **presentation_size**.

Неформальные утверждения

IP1 — любой объект **presentation_representation** должен быть представлен в контексте «дерева», «корнем» которого является **presentation_area;**

IP2 — размер **presentation_area** должен быть определен только однократно.

Примечание — Ограничения по применению рассматриваемого объекта установлены правилами для объекта **presentation_representation_relationship.**

4.4.4 Объект area_in_set

Объект **area_in_set** определяет вхождение объекта **presentation_area** в **presentation_set**. Объект

presentation_area может входить в несколько объектов **presentation_set**; в конкретный **presentation_set** должен входить по крайней мере один **presentation_area**.

EXPRESS-спецификация

```
*)
ENTITY area_in_set;
    area      :  presentation_area;
    in_set    :  presentation_set;
END_ENTITY;
(*
```

Описания атрибутов

area — объект **presentation_area**, входящий в **presentation_set**;

in_set — объект **presentation_set**, в который входит **presentation_area**.

4.4.5 Объект presentation_view

Объект **presentation_view** является изображением, могущим содержать другие изображения или входить в другое изображение. Данный объект не представляет полное изображение и не может быть отображен без включения в объект **presentation_area**.

Размер, установленный в объекте **presentation_view**, может быть указан посредством ссылки на него из объекта **presentation_size**. Выделение изображения в размерах, заданных объектом **presentation_view**, выполняет система, создающая реальное изображение на основе информации о представлении. Если в объекте **presentation_view** не установлены соответствующие размеры, тогда выделение изображения выполняют только на основе размеров, заданных в объекте **presentation_area**, в который входит данный **presentation_view**.

Основной (фоновый) цвет может быть указан посредством объекта **background_colour**.

EXPRESS-спецификация

```
*)
ENTITY presentation_view
    SUBTYPE OF (presentation_representation);
END_ENTITY;
(*
```

Примечание — Ограничения по применению рассматриваемого объекта установлены правилами для объекта **presentation_representation_relationship.**

4.4.6 Объект area_dependent_annotation_representation

Объект **area_dependent_annotation_representation** является изображением, которое может входить в изображение, описанное объектом **presentation_area.** Данное изображение состоит из элементов, определенных в объектах **annotation_occurrence.** Рассматриваемый объект может быть связан только с объектом **presentation_area**.

EXPRESS-спецификация

```
*)
ENTITY area_dependent_annotation_representation
    SUBTYPE OF (presentation_representation);
WHERE
    WR1: SIZEOF (QUERY (item<* (SELF \representation.items |
         NOT (SIZEOF (['PRESENTATION_DEFINITION_SCHEMA.' +
                       'ANNOTATION_OCCURRENCE',
                       'GEOMETRY_SCHEMA.AXIS2_PLACEMENT']*
               TYPEOF (item)) = 1
         ))) = 0;
    WR2: SIZEOF (QUERY (item<* (SELF \representation.items |
         ('PRESENTATION_DEFINITION_SCHEMA.ANNOTATION_OCCURRENCE' IN
           TYPEOF (item))
         )) > = 1;
END_ENTITY;
(*
```

Формальные утверждения

WR1 — видами атрибутов **representation_items**, входящими в рассматриваемый объект, могут быть только объекты **annotation_occurrence** или **axis2_placement**;

WR2 — одним из элементов в объекте **area_dependent_annotation_representation** должен быть **annotation_occurrence**.

Неформальное утверждение

IP1 — при наличии объекта **area_dependent_annotation_representation** он должен входить в **presentation_area**.

Примечание — Ограничения по применению рассматриваемого объекта установлены правилами для объекта **presentation_representation_relationship**.

4.4.7 Объект product_data_representation_view

Объект **product_data_representation_view** представляет собой изображение, состоящие из двумерных геометрических проекций, аннотации или того и другого. В данное изображение могут входить другие изображения, или данное изображение может быть включено в другое изображение. Данный объект не содержит полного изображения и не может быть отображен без представления его в объект **presentation-view**.

EXPRESS-спецификация

```
*)
ENTITY product_data_representation_view;
    SUBTYPE OF (presentation_representation);
WHERE
    WR1: SIZEOF (QUERY (item<* SELF \representation.item |
        NOT (SIZEOF (['PRESENTATION_ORGANIZATION_SCHEMA.CAMERA_IMAGE',
                      'GEOMETRY_SCHEMA.AXIS2_PLACEMENT']*
                      TYPEOF (item)) = 1
        ))) = 0;
    WR2: SIZEOF (QUERY (item<* (SELF \representation.items |
        ('PRESENTATION_ORGANIZATION_SCHEMA.CAMERA_IMAGE' IN
         TYPEOF (item))
END_ENTITY;
(*
```

Формальные утверждения

WR1 — элементом **product_data_representation_view** должны быть объекты **camera_image** или **axis2_placement**;

WR2 — одним из элементов в **product_data_representation_view** должен быть объект **camera_image**.

Неформальное утверждение

IP1 — при наличии объекта **product_data_representation_view** он должен входить в **presentation_view.**

Примечание — Ограничения по применению рассматриваемого объекта установлены правилами для объекта **presentation_representation_relationship**.

4.4.8 Объект view_dependent_annotation_representation

Объект **view_dependent_annotation_representation** является изображением, которое может быть размещено в изображении, представленном объектом **presentation_view**. Данное изображение включает только элементы объектов **annotation_occurrence.** Рассматриваемый объект может быть связан только с объектом **presentation_view**.

EXPRESS-спецификация

```
*)
ENTITY view_dependent_annotation_representation
    SUBTYPE OF (presentation_representation);
WHERE
    WR1: SIZEOF (QUERY (item<* (SELF \representation.items |
        NOT (SIZEOF (['PRESENTATION_DEFINITION_SCHEMA.' +
                      'ANNOTATION_OCCURRENCE',
                      'GEOMETRY_SCHEMA.AXIS2_PLACEMENT']*
                      TYPEOF (item)) = 1
        ))) = 0;
```

```
    WR2: SIZEOF (QUERY (item<* (SELF \representation.items |
            ('PRESENTATION_DEFINITION_SCHEMA.ANNOTATION_OCCURRENCE' IN
             TYPEOF (item))
            )) > = 1;
END_ENTITY;
(*
```

Формальные утверждения

WR1 — в набор элементов объекта **view_dependent_annotation_representation** должны входить виды **representation_items**, представляющие объекты **annotation_occurrence** или **axis2_placement**;

WR2 — одним из элементов в **view_dependent_annotation_representation** должен быть объект **annotation_occurrence**.

Неформальное утверждение

IP1 — при наличии объекта **view_dependent_annotation_representation** он должен входить в **presentation_view**.

Примечание — Ограничения по применению рассматриваемого объекта установлены правилами для объекта **presentation_representation_relationship**.

4.4.9 Объект presentation_size

Объект **presentation_size** используют для определения размера объектов **presentation_area** или **presentation_view**. Размер объекта **presentation_area** может быть задан непосредственно или присвоен в зависимости от объекта **presentation_set**, содержащего **presentation_area**. Данный метод позволяет назначать различные размеры для одного объекта **presentation_area** в каждом **presentation_set**, описывающем определенную область.

EXPRESS-спецификация

```
*)
ENTITY presentation_size;
    unit : presentation_size_assignment_select;
    size : planar_box
WHERE
    WR1: (('PRESENTATION_ORGANIZATION_SCHEMA.PREZENTATION_REPREZENTATION'
              IN TYPEOF (SELF.unit)) AND
              item_in_context (SELF.size,
                              SELF.unit\representation.context_of_items)
          )
              OR
          (
             ('PRESENTATION_ORGANIZATION_SCHEMA.AREA_IN_SET'
                 IN TYPEOF (SELF.unit)) AND
           (SIZEOF (QUERY (ais<* SELF.unit\area_in_set.in_set.areas |
                    NOT item_in_context (SELF.size, ais.area\representation.
                                                    context_of_items) )) = 0)
          );
END_ENTITY;
(*
```

Описания атрибутов

unit — объекты **presentation_view**, **presentation_area** или **area_in_set**, определяющие связь **presentation_area** с **presentation_set**;

size — объект **planar_box**, описывающий размер блока.

Формальное утверждение

WR1 — если блоком (**unit**) является объект **presentation_representation,** тогда **planax_box** должен быть элементом представления, имеющего тот же контекст, что и **unit**. Если блоком (**unit**) является объект **area_in_set**, тогда **planar_box** должен быть элементом в контексте, связанном с данной областью.

4.4.10 Объект background_colour

Объект **background_colour** определяет цвет, используемый для фонов представления области или вида.

EXPRESS-спецификация

```
*)
ENTITY background_colour
    SUBTITY OF (colour);
    presentation : area_or_view;
UNIQUE
    UR1 : presentation;
END_ENTITY;
(*
```

Описание атрибута

presentation — объекты **presentation_area** или **presentation_view**, для которых установлен фоновый цвет.

Формальное утверждение

UR1 — для любого объекта **presentation_area** или **presentation_view** может быть задан только один цвет фона.

4.4.11 Объект presentation_representation_relationship

Объект **presentation_representation_relationship** определяет отношение между объектами **presentation_representation**. Данное отношение является направленным в том смысле, что представление «потомка» (**rep_2**) преобразуется в представление «родителя» (**rep_1**) и не предусматривает обратного преобразования. Рассматриваемый объект ограничивает отношение между некоторыми объектами **presentation_representation** для обеспечения их представления в иерархии (см. 4.1).

EXPRESS-спецификация

```
*)
ENTITY presentation_representation_relationship
    SUBTYPE OF (representation_relationship_with_transformation);
WHERE
    WR1 : 'PRESENTATION_DEFENITION_SCHEMA.PRESENTATION_REPRESENTATION'IN
            TYPEOF (SELF\representation_relationship_rep_1);
    WR2 : 'PRESENTATION_DEFENITION_SCHEMA.PRESENTATION_REPRESENTATION'IN
            TYPEOF (SELF\representation_relationship_rep_2);
    WR3 :  acyclic_presentation_representation_relationship  (SELF,
            [SELF\representation_relationship_rep_2]);
    WR4 : NOT (('PRESENTATION_ORGANIZATION_SCHEMA.PRESENTATION_AREA'IN
            TYPEOF (SELF\representation_relationship_rep_1))
                    AND
          NOT (SIZEOF (['PRESENTATION_ORGANIZATION_SCHEMA.' +
                        'PRODUCT_DATA_REPRESENTATION'_VIEW',
                        'PRESENTATION_ORGANIZATION_SCHEMA.' +
                        'VIEW_DEPENDENT_ANNOTATION_REPRESENTATION']*
            TYPEOF (SELF\representation_relationship_rep_2))  = 0));
    WR5 : NOT (('PRESENTATION_ORGANIZATION_SCHEMA.PRESENTATION_VIEW"
            IN TYPEOF (SELF\representation_relationship_rep_1))
                    AND
          NOT (SIZEOF (['PRESENTATION_ORGANIZATION_SCHEMA.' +
                        'PRESENTATION_AREA',
                        'PRESENTATION_ORGANIZATION_SCHEMA.' +
                        'PRESENTATION_VIEW',
                        'PRESENTATION_ORGANIZATION_SCHEMA.' +
                        'AREA_DEPENDENT_ANNOTATION_REPRESENTATION_']*
                TYPEOF (SELF\representation_relationship_rep_2))  = 0));
    WR6 : NOT (('PRESENTATION_ORGANIZATION_SCHEMA.PRESENTATION_VIEW" IN
            TYPEOF (SELF\representation_relationship_rep_2)))
                XOR
            ('PRESENTATION_ORGANIZATION_SCHEMA.PRESENTATION_AREA' IN
            TYPEOF (SELF\representation_relationship_rep_1));
```

```
    WR7 : (NOT ('PRESENTATION_ORGANIZATION_SCHEMA.' +
                'PRODUCT_DATA_REPRESENTATION_VIEW' IN
              (TYPEOF (SELF\representation_relationship_rep_1) +
               TYPEOF (SELF\representation_relationship_rep_2))))
                                XOR
            ('PRESENTATION_ORGANIZATION_SCHEMA.PRESENTATION_VIEW' IN
              TYPEOF (SELF\representation_relationship_rep_1))
                                AND
            ('PRESENTATION_ORGANIZATION_SCHEMA'. PRODUCT_DATA_
            REPRESENTATION_VIEW' IN
              TYPEOF (SELF\representation_relationship_rep_2));
    WR8 : 'PRESENTATION_ORGANIZATION_SCHEMA.GRAPHICAL_TRANSFORMATION' IN
              TYPEOF (SELF\representation_relationship_with_transformation.
                                transformation_operator);
END_ENTITY;
(*
```

Определения атрибутов

SELF\representation_relationship.rep_1 — объект **presentation_representation**, играющий роль «родителя» в «дереве» **presentation_representation**;

SELF\representation_relationship.rep_2 — объект **presentation_representation**, играющий роль «потомка» в «дереве» **presentation_representation**.

Формальные учреждения

WR1 — атрибутом **rep_1** должен быть объект **presentation_representation**;

WR2 — атрибутом **rep_2** должен быть объект **presentation_representation**;

WR3 — объект **presentation_representation_relationship** не должен входить в «дерево» объектов **presentation_representation**, «корнем» которого является «лист» собственного «дерева» рассматриваемого объекта;

WR4 — изображения, представленные объектами **product_data_representation_view** или **view_dependent_annotation_representation**, не должны быть связаны с изображением, представленным объектом **presentation_area**;

WR5 — изображения, представленные объектами **presentation_area**, **presentation_view** или **area_dependent_annotation_representation**, не должны быть связаны с изображением, представленным объектом **presentation_view**;

WR6 — изображение, представленное объектом **presentation_view**, может быть связано только с изображением, представленным объектом **presentation_area**;

WR7 — объект **product_data_representation_view** должен входить только в объект **presentation_representation_relationship**, если атрибутом **rep_1** является объект **presentation_view**;

WR8 — Связь данных изображений с другими осуществляется посредством объектов **graphical_transformation.**

4.4.12 Объект graphical_transformation

Объект **graphical_transformation** является преобразованием, позволяющим осуществить произвольное двумерное позиционирование, вращение и однородное масштабирование. Данный объект используется объектом **presentation_representation_relationship** для определения преобразования между связанными объектами **presentation_representations**. Данное преобразование определяют следующим образом.

Атрибут **transform_item_1.location** преобразуют в атрибут **transform_item_2.placement_location**. Дополнительно ось *x*, определенную в объекте **transform_item_1.p[1]**, отображают в ось *x* атрибута **transform_item_2.placement.p[1].** Подобное отображение должно быть выполнено для оси *y* атрибутов **transform_item_1.p[2]** и **transform_item_2.placement.p[2].** По окончании используют масштабирование, определенное атрибутом **transform_item_2.scaling.**

Примечание 1 — На рисунке 5 показан механизм использования объекта **graphical_transformation.**

EXPRESS-спецификация

```
*)
ENTITY graphical_transformation
    SUBTYPE OF (item_defined_transformation);
```

```
WHERE
    WR1 : 'GEOMETRY_SCHEMA.AXIS2_PLACEMENT_2D'IN
            TYPEOF (SELF\item_defined_transformation.transform_item_1);
    WR2 :  'PRESENTATION_RESOURCE_SCHEMA.PRESENTATION_SCALED_PLACEMENT' IN
            TYPEOF (SELF\item_defined_transformation.transform_item_2);
END_ENTITY;
(*
```

Описания атрибутов

SELF\item_defined_transformation.transform_item_1 — объект **axis2_placement_2d,** определяющий исходные преобразуемые координаты;

SELF\item_defined_transformation.transform_item_2 — объект **presentation_scaled_placement,** определяющий конечные координаты преобразования.

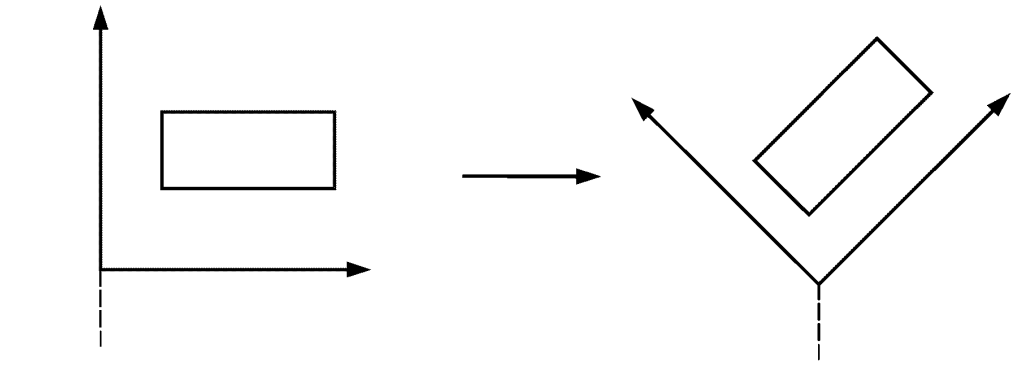


Рисунок 5 — Графическое преобразование

Формальные утверждения

WR1 — атрибутом **transform_item_1** объекта **graphical_transformation** должен быть объект **axis2_placement_2d;**

WR2 — атрибутом **transform_item_2** объекта **graphical_transformation** должен быть объект **presentation_ scaled_placement.**

Примечание 2 — Атрибуты **transform_item_1** и **transform_item_2** определены в ГОСТ Р ИСО 10303-43.

4.5 Определение объектов схемы структуры представления: модель камеры и проекция

4.5.1 Объект camera_model

Объект **camera_model** должен содержать информацию, необходимую для создания проекции или отображения представления в изображение данного представления.

EXPRESS-спецификация

```
*)
ENTITY camera_model
    SUPERTYPE OF (ONEOF (camera_model_d2, camera_model_d3))
    SUPERTYPE OF (geometric_representation_item);
WHERE
    WR1 : (SIZEOF (USEDIN (SELF, 'REPRESENTATION_SCHEMA.' +
                                 'ITEM_DEFINED_TRANSFORMATION.' +
                                 'TRANSFORM_ITEM_1')) +
           SIZEOF (USEDIN (SELF, 'REPRESENTATION_SCHEMA.' +
                                 'REPRESENTATION_MAP.MAPPING_ORIGIN'))
           ) > 0;
```

```
    WR2 : SIZEOF (USEDIN (SELF, 'REPRESENTATION_APPEARANCE_SCHEMA.' +
                                'STYLED_ITEM.ITEM')) = 0;
END_ENTITY;
(*
```

Формальные утверждения

WR1 — объект **camera_model** должен определять проекцию по крайней мере одного объекта **representation**;

WR2 — объект **camera_model** не должен быть связан с объектом **styled_item**.

4.5.2 Объект camera_model_d2

Объект **camera_model_d2** должен содержать информацию, необходимую для создания двумерного отображения данного представления на его изображение.

EXPRESS-спецификация

```
*)
ENTITY camera_model_d2
    SUBTYPE OF (camera_model);
    view_window                 : planar_box;
    view_window_clipping        : BOOLEAN;
WHERE
    WR1 : SELF\geometric_representation_item.dim = 2
END_ENTITY;
(*
```

Описания атрибутов

view_window — прямоугольные границы в координатном пространстве объекта **representation**, отображаемого в объект **product_data_representation**. Преобразование и возможное неоднородное масштабирование объекта **view_window** должны быть выполнены так, чтобы контуры данного объекта совпадали с контурами атрибута **mapping_target** в объекте **camera_image**;

view_window_clipping — определяет проведение или не проведение оконтуривания объекта **view_window**. Значение TRUE данного атрибута указывает на выполнение оконтуривания объекта **view_window**, а значение FALSE — на его отсутствие.

Примечания

1 Если атрибут **view_window_clipping** имеет значение FALSE, двумерная проекция может быть расширена за границы, определенные атрибутом **mapping_target** в объекте **camera_image.**

2. Оконтуривание должно быть проведено до создания проекции.

Формальное утверждение

WR1 — объект **camera_model_d2** должен быть двумерным.

Примечание 3 — Механизм работы объекта **camera_model_d2** представлен на рисунке 6.

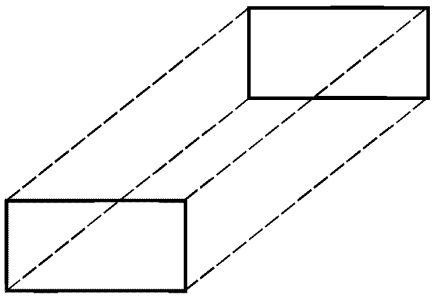


Рисунок 6 — Модель двумерной камеры

4.5.3 Объект camera_model_d2_shape_clipping

Объект **camera_model_d2_shape_clipping** является объектом **camera_model_d2** с дополнительными условиями по выделению (оконтуриванию) отображаемого представления.

EXPRESS-спецификация

```
*)
ENTITY camera_model_d2_shape_clipping
    SUBTYPE OF (camera_model_d2);
    shape_clipping      : curve;
END_ENTITY;
(*
```

Описание атрибута

shape_clipping — замкнутая кривая, определяющая пространство проектируемого представления.

Примечание — Если атрибут **view_window_clipping** имеет значение TRUE, проектируемое представление выделяется из **view_window** и области, определенной объектом **shape_clipping.**

Неформальное утверждение

IP1 — кривая, определенная в атрибуте **shape_clipping**, должна быть замкнутой и не самопересекающейся.

4.5.4 Объект camera_model_d3

Объект **camera_model_d3** должен содержать информацию, необходимую для создания проекции из трехмерного представления на его двумерное изображение (подробности проецирования см. в 4.5.5).

EXPRESS-спецификация

```
*)
ENTITY camera_model_d3
    SUBTYPE OF (camera_model);
    view_reference_system       : axis2_placement_3d;
    perspective_of_volume       : view_volume;
WHERE
    WR1 : (dot_product (SELF.view_reference_system.p[3]
              SELF.perspective_of_volume.view_window.placement.p[3] = 1.0)
              AND
            (SELF.view_reference_system.location.coordinates[3] =
              SELF.perspective_of_volume.view_window.
                    placement.location.coordinates[3]);
    WR2:  SELF\geometric_representation_item.dim = 3;
END_ENTITY;
(*
```

Описания атрибутов

view_reference_system — вспомогательное трехмерное координатное пространство в координатном пространстве проецируемого отображения;

perspective_of_volume — информация, необходимая для проецирования геометрической формы отображения. Данный атрибут определяют во вспомогательном трехмерном координатном пространстве объекта **view_reference_system**.

Формальные утверждения

WR1 — прямоугольник, заданный атрибутом **view_window** объекта **perspective_of_volume**, должен быть расположен в той же плоскости, что и оси *x* и *y* объекта **axis2_placement_3d**, указанного подтипом **view_reference_system**;

WR2 — объект **camera_model_d3** является трехмерным.

4.5.5 Объект view_volume

Объект **view_volume** определяют в объекте **view_reference_system** модели камеры, использующей объемные представления. Рассматриваемый объект определяет объемное представление, которое проецируется на демонстрационное окно объекта **product_data_representation_view**. Данным объемным представлением может быть усеченная пирамида или параллелепипед. Предмет, охваченный данным объемом, проецируется на прямоугольник, заданный атрибутом **view_window**, который задает отображение на демонстрационное окно.

Если тип проецирования — параллельный (**parallel**), проекция создается параллельно линии, направленной из **projection_point** в геометрический центр **view_window**, а объектом **view_volume** является параллелепипед. Если тип проецирования — центральный (**central**), проекция создается вдоль линий исходящих из **projection_point** и пересекающихся с **view_window**, а объектом **view_volume** является усеченная пирамида. Соответствующие иллюстрации приведены на рисунках 7 и 8.

Примечание — Так как объект **view_volume** не является подтипом объекта **geometric_representation_item**, экземпляры **cartesian_point**, представленные атрибутом **projection_point**, и **planar_box**, представленные атрибутом **view_window**, не связывают обычным методом с атрибутом **geometric_representation_context** каждого объекта **representation** посредством объекта **camera_model_d3**, содержащего данный **view_volume**. Данная связь выполняется посредством супертипа **founded_item**.

EXPRESS-спецификация

```
*)
ENTITY view_volume
    SUBTYPE OF (founded_item);
```

```
    projection_type              : central_or_parallel;
    projection_point             : cartesian_point;
    view_plane_distance          : length_measure;
    front_plane_distance         : length_measure;
    front_plane_clipping         : BOOLEAN;
    back_plane_distance          : length_measure;
    back_plane_clipping          : BOOLEAN;
    view_volume_sides_clipping   : BOOLEAN;
    view_window                  : planar_box;
END_ENTITY;
(*
```

Определения атрибутов

projection_type — указатель типа проекции (центральная или параллельная);

projection_point — центр проекции, то есть место из которого рассматривают предметы. При параллельном проецировании данная точка определяет линию от центра проекции до геометрического центра объекта **view_window**. При центральном проецировании данная точка является вершиной усеченной пирамиды;

view_plane_distance — расстояние вдоль оси z объекта **axis2_placement**, заданное атрибутом **view_reference_system**. Данное расстояние определяет положение оригинала объекта **view_window**;

front_plane_distance — заданное расстояние вдоль оси z объекта **axis2_placement**, указанное атрибутом **view_reference_system**. Данное расстояние указывает плоскость, параллельную плоскости объекта **view_window**. Это расстояние задают либо от вершины усеченной пирамиды, либо от передней грани параллелепипеда, в зависимости от типа проекции;

front_plane_clipping — указатель вычерчивания (или нет) геометрической формы проецируемого отображения на плоскости, заданной атрибутом **front_plane_distance**. Значение TRUE указывает на выполнение вычерчивания, а FALSE — на его отсутствие.

Примечание 1 — Если **front_plane_clipping** имеет значение FALSE, объекты за точкой проецирования могут быть видимыми. При этом подразумевается направление, обратное по отношению к направлению от точки проецирования к плоскости отображения;

back_plane_distance — расстояние вдоль оси z объекта **axis2_placement**, заданное атрибутом **view_reference_system**. Данное расстояние определяет плоскость, параллельную плоскости объекта **view_window**. Этой плоскостью является основание усеченной пирамиды или задняя грань параллелепипеда, в зависимости от типа проекции;

back_plane_clipping — указатель вычерчивания (или нет) геометрической формы проецируемого отображения на плоскости, заданной атрибутом **back_plane_distance**. Значение TRUE указывает на выполнение вычерчивания, а FALSE — на его отсутствие;

view_volume_sides_clipping — указатель вычерчивания (или нет) геометрической формы проецируемого отображения на плоскости, которые являются сторонами объемного представления, заданного объектом **view_window**. Значение TRUE указывает на выполнение вычерчивания, а FALSE — на его отсутствие.

Примечание 2 — Если **view_volume_sides_clipping** имеет значение FALSE, тогда двумерная проекция может быть расширена за границы демонстрационного окна (**viewport**), заданного объектом **camera_image**;

view_window — прямоугольник из **view_plane**. Представление отображается на этот прямоугольник, который в свою очередь отображают на **viewport** объекта **camera_image**.

Неформальные утверждения

IP1 — плоскость, представленная атрибутом **front_plane_distance**, должна быть расположена к объекту **projection_point** ближе, чем плоскость, представленная атрибутом **back_plane_distance**;

IP2 — прямоугольник, заданный атрибутом **view_window**, должен быть расположен в рассматриваемой плоскости (view_plane);

IP3 — объект **projection_point** не должен располагаться в рассматриваемой плоскости (view_plane).

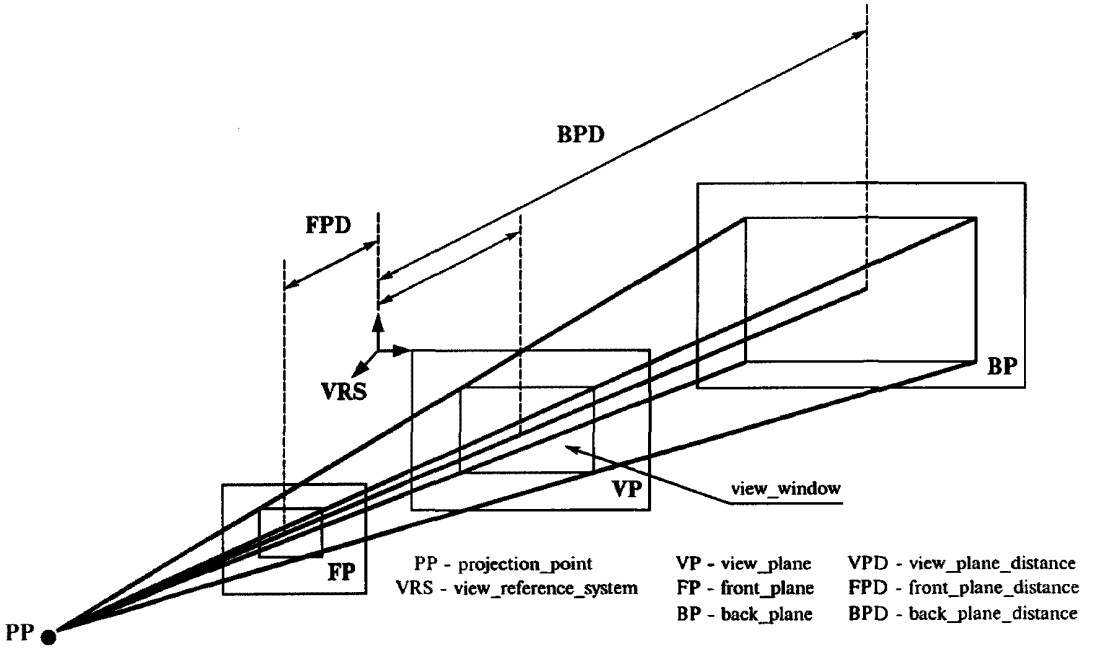


Рисунок 7 — Вид объема, тип проекции — CENTRAL

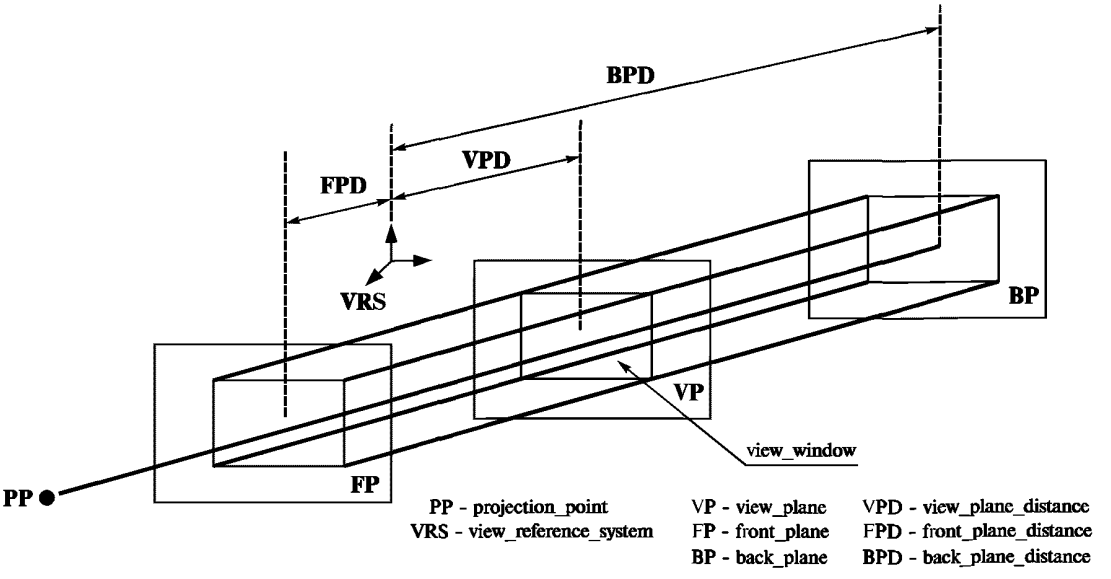


Рисунок 8 — Вид объема, тип проекции — PARALLEL

4.5.6 Объект camera_model_d3_with_hlhsr

Объектом **camera_model_d3_with_hlhsr** является объект **camera_model_d3**, указывающий на необходимость удаления из изображения невидимых линий и поверхностей.

EXPRESS-спецификация

```
*)
ENTITY camera_model_d3_with_hlhsr
    SUBTYPE OF (camera_model_d3);
    hidden_line_surface_removal : BOOLEAN;
END_ENTITY;
(*
```

Определение атрибута

hidden_line_surface_removal — указывает на необходимость удаления из изображения невидимых линий и поверхностей при проецировании трехмерного изображения. Значение TRUE данного атрибута указывает на то, что невидимые линии и поверхности при проецировании должны быть удалены, а значение FALS — на необходимость их отображения.

4.5.7 Объект camera_model_d3_multi_clipping

Объект **camera_model_d3_multi_clipping** должен содержать информацию, необходимую для создания проекции отображения на его изображение с использованием плоскостей координатного пространства проецируемого отображения для оконтуривания (выделения) геометрической формы данного отображения до его проецирования.

EXPRESS-спецификация

```
*)
ENTITY camera_model_d3_multi_clipping
    SUBTYPE OF (camera_model_d3);
    shape_clipping : SET [1:?] OF plane;
END_ENTITY;
(*
```

Определение атрибута

shape_clipping — плоскости, ограничивающие выделяемую область. Каждая плоскость указывает определенную область, которая является бесконечным полупространством, заданным этой плоскостью и плоскостью, перпендикулярной к ней. Выделяемая (оконтуриваемая) область должна быть определена на основе взаимопересечения всех заданных областей.

Примечания

1 Выделяемая область, заданная атрибутом **shape_clipping**, может быть бесконечной.

2 Выделение выполняют до проецирования.

4.5.8 Объект camera_model_with_light_sources

Объект **camera_model_with_light_sources** должен содержать информацию, необходимую для создания проекции отображения на его изображение, с дополнительными сведениями об источниках освещения, используемых для создания теневых эффектов.

EXPRESS-спецификация

```
*)
ENTITY camera_model_with_light_sources
    SUBTYPE OF (camera_model_d3);
    sources : SET [1:?] OF light_sources;
END_ENTITY;
(*
```

Определение атрибута

sources — набор объектов **light_sources**, определяющих объемную модель проецируемого отображения.

4.5.9 Объект light_source

Объект **light_source** связан с экранным изображением поверхностей. Подсветка (объемное сканирование) связана с конкретной поверхностью, то есть не отражает взаимосвязей отдельных поверхностей.

EXPRESS-спецификация

```
*)
ENTITY light_source
    SUBTYPE OF (ONEOF(light_source_ambient,
                      light_source_directional,
                      light_source_positional,
                      light_source_spot))
    SUBTYPE OF (geometric-representation_item);
    light_colour : colour;
```

```
WHERE
    WR1: SIZEOF(USEDIN (SELF, 'PRESENTATION_APPEARANCE_SCHEMA.'+
                'STYLED_ITEM.ITEM')) = 0;
END_ENTITY;
(*
```

Определение атрибута

light_colour — цвет источника освещения, используемый для создания теневых эффектов.

Формальное утверждение

WR — атрибут **light_source** не должен быть связан с объектом **styled_item**.

4.5.10 Объект light_source_ambient

Объект **light_source_ambient** описывает поверхность независимо от ее ориентации и расположения.

EXPRESS-спецификация

```
*)
ENTITY light_source_ambient
    SUBTYPE OF (light_source);
END_ENTITY;
(*
```

4.5.11 Объект light_source_directional

Объект **light_source_directional** описывает ориентацию поверхности независимо от ее местоположения. Все лучи от источника света должны быть направлены параллельно выбранному направлению.

Примечание — На рисунке 9 проиллюстрирован объект **light_source_directional**.

EXPRESS-спецификация

```
*)
ENTITY light_source_directional
    SUBTYPE OF (light_source);
    orientation : direction;
END_ENTITY;
(*
```

Определение атрибута

orientation — направление, заданное объектом **light_source** в координатном пространстве проецируемого отображения.

4.5.12 Объект light_source_positional

Объект **light_source_positional** описывает поверхность с точки зрения ее ориентации и местоположения.

Примечание 1 — На рисунке 10 проиллюстрирован объект **light_source_positional**.

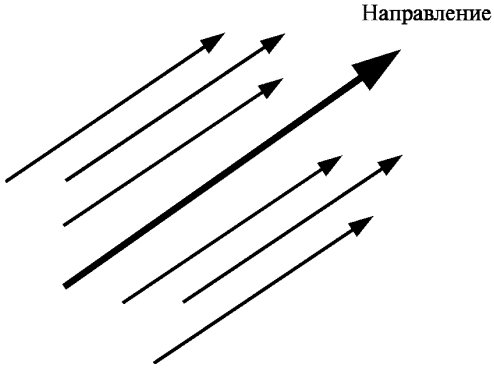


Рисунок 9 — Направление источника света

EXPRESS-спецификация

```
*)
ENTITY light_source_positional;
    SUBTYPE OF (light_source);
    position                : cartesian_point;
    constant_attenuation    : REAL;
    distance_attenuation    : REAL;
END_ENTITY;
(*
```

Определения атрибутов

position — местоположение объекта **light_source** в координатном пространстве проецируемого отображения;

constant_attenuation — значение затухания в формуле отражающей способности, выраженное константой;

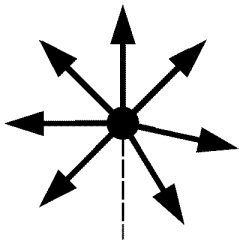


Рисунок 10 — Местоположение источника света (объект **light_source**)

distance_attenuation — значение затухания в формуле отражающей способности, пропорциональное расстоянию от объекта **light_source**.

Примечание 2 — Примеры формул отражающей способности приведены в приложении D.

4.5.13 Объект light_source_spot

Объект **light_source_spot** описывает поверхность на основе ее местоположения и направления с учетом распространения потока от источника света. Данный поток определяют на основе местоположения, ориентации и значения атрибута **spread_angle** объекта **light_source_spot**. В объекте **light_source_spot** следует описывать только те части (детали) предмета, которые попадают внутрь светового потока от источника.

Примечание 1 — На рисунке 11 проиллюстрирован объект **light_source_spot**.

EXPRESS-спецификация

```
*)
ENTITY light_source_spot;
    SUBTYPE OF (light_source);
    position                : cartesian_point;
    orientation             : direction;
    concentration_exponent  : REAL;
    constant_attenuation    : REAL;
    distance_attenuation    : REAL;
    spread_angle            : positive_plane_angle_measure;
END_ENTITY;
(*
```

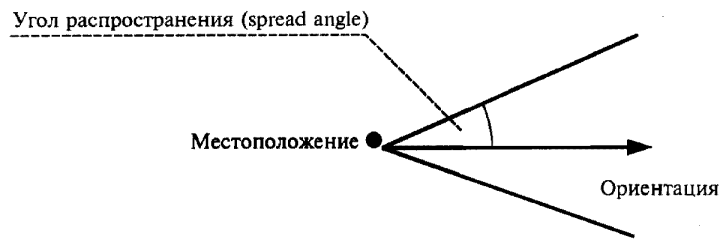


Рисунок 11 — Распространение потока от источника света (объект **light_source_spot**)

Определения атрибутов

position — местоположение объекта **light_source** в координатном пространстве проецируемого отображения;

orientation — направление оси светового потока от объекта **light_source** в координатном пространстве проецируемого отображения;

concentration_exponent — косинус угла между линией, исходящей из местоположения объекта **light_source** в направлении его ориентации, и линией, исходящей из местоположения объекта **light_source** и пересекающей выделяемую (затеняемую) поверхность. Положение данной поверхности, входящее в формулу отражающей способности, задается атрибутом **surface_style_rendering_properties**;

constant_attenuation — значение затухания в формуле отражающей способности, выраженное константой;

distance_attenuation — значение затухания в формуле отражающей способности, пропорциональное расстоянию от объекта **light_source**;

spread_angle — половина телесного угла светового потока от источника.

Неформальное утверждение

IP1 — угол распространения не должен превышать 180°.

Примечание 2 — примеры формул отражающей способности приведены в приложении D.

4.5.14 Объект camera_image

Объект **camera_image** отражает результат проецирования дву- или трехмерной геометрической формы, отображаемой объектом **camera_model** на демонстрационный экран.

EXPRESS-спецификация

```
*)
ENTITY camera_image
    SUBTYPE OF (mapped_item);
WHERE
    WR1: 'PRESENTATION_ORGANIZATION_SCHEMA.CAMERA_USAGE'
            IN TYPEOF (SELF\mapped_item_mapping_source);
    WR2: 'PRESENTATION_RESOURCE_SCHEMA.PLANAR_BOX'
            IN TYPEOF (SELF\mapped_item.mapping_target);
    WR3:  'GEOMETRY_SCHEMA.GEOMETRUC_REPRESENTATION_ITEM'
            IN TYPEOF (SELF);
END_ENTITY;
(*
```

Определения атрибутов

SELF\mapped_item_mapping_source — объект **camera_usage**, содержащий проецируемый объект **representation**, и объект **camera_model**, являющийся исходным при отображении проекции;

SELF\mapped_item.mapping_target — объект **planar_box**, на который проецируется демонстрационное окно, связанное с дву- или трехмерным объектом **camera_model**.

Формальные утверждения

WR1 — исходным для отображения должен быть объект **camera_usage**;

WR2 — целью отображения должно быть создание объекта **planar_box**;

WR3 — объектом **camera_image** должен быть объект **geometric_representation_item**.

4.5.15 Объект camera_usage

Объект **camera_usage** устанавливает связь между дву- или трехмерным отображением и его источником (оригиналом). Рассматриваемый объект является исходным при отображении проекции.

EXPRESS-спецификация

```
*)
ENTITY camera_usage
    SUBTYPE OF (representation_map);
WHERE
    WR1: NOT ('PRESENTATION_ORGANIZATION_SCHEMA.
                PRESENTATION_REPRESENTATION
          IN TYPEOF (SELF\representation_map.mapped_representation);
    WR2: 'PRESENTATION_ORGANIZATION_SCHEMA.CAMERA_MODEL'
          IN TYPEOF (SELF\representation_map.mapping_origin);
END_ENTITY;
(*
```

Определение атрибута

SELF\representation_map.mapping_origin — объект **camera_model**, являющийся исходным при отображении проекции.

Формальные утверждения

WR1 — объектом **mapped_representation** не должен быть объект **presentation_representation**;

WR2 — исходным для объекта **camera_usage** должен быть объект **camera_model**.

4.6 Определение объектов схемы структуры представления: уровни

4.6.1 Объект presentation_layer_assignment

Объект **presentation_layer_assignment** является идентификатором (указателем) набора объектов **presentation_representation** или **representation_item**. Данные наборы содержат изображения целиком или их элементы, связанные с данным уровнем. Объектом **representation_item**, связанным с данным уровнем, должен быть конкретный объект **styled_item** или несколько аналогичных объектов. В случае использования объектов **presentation_representation** значение идентификатора уровня должно быть связано с каждым перечисленным в нем или связанным с ним объектом **representation_item**, если

данное значение не ограничено объектом **representation_item_dependent_layer_assignment**. В случае использования объектов **representation_item** значение идентификатора уровня также должно быть связано с каждым перечисленным в нем объектом **representation_item**, ссылающимся (прямо или косвенно) на конкретный элемент, если данное значение не ограничено объектом **representation_item_dependent_layer_assignment**.

Примечание — Содержимое данного объекта используют при определении видимости и стиля изображения, представленного объектом **presentation_representation**, с использованием рассматриваемого объекта.

EXPRESS-спецификация

```
*)
ENTITY presentation_layer_assignment;
    name              : label;
    description       : text;
    assigned_items    : SET [1:?] OF layered_item;
END_ENTITY;
(*
```

Определения атрибутов

name — слово или группа слов, определяющих конкретный уровень при ссылке на него из объекта **presentation_layer_assignment**;

description — текст, описывающий характер объекта **presentation_layer_assignment**;

assigned_items — набор элементов, относящихся к уровню, определенному в объекте **presentation_layer_assignment**.

4.6.2 Объект representation_item_dependent_layer_assignment

Объектом **representation_item_dependent_layer_assignment** является объект **presentation_layer_assignment**, определяющий уровень объекта **presentation_representation** или конкретного объекта **representation_item** только с точки зрения их участия в описании объекта **representation_item**. Содержимое данного объекта пригодно для описания любого уровня, присвоенного объекту **presentation_representation** или конкретному **representation_item**, с точки зрения их участия в описании объекта **representation_item**.

Пример 5 — При черчении определенная размерность связана с некоторыми кривыми и текстом. При этом саму размерность определяют уровнем 'DIMENSION', а пояснительный текст — уровнем 'DIMENSION VALUE'. Для указания, что данная размерность связана с уровнем 'DIMENSION' в объекте **presentation_layer_assignment**, она должна быть включена в качестве элемента набора объектов **assignment_item**. Для указания, что конкретный текст связан с уровнем 'DIMENSION VALUE', он должен быть включен в объект **representation_item_dependent_layer_assignment** в качестве элемента набора объектов **assignment_item** и определять данную размерность в контексте рассматриваемого объекта.

EXPRESS-спецификация

```
*)
ENTITY representation_item_dependent_layer_assignment;
    SUBTYPE OF (presentation_layer_assignment);
    item_context : representation_item
END_ENTITY;
(*
```

Определение атрибута

item_context — объект **representation_item**, определяющий контекст объектов **assignment_items** на заданном уровне.

Неформальное утверждение

IP1 — объекты **assignment_items** должны входить в описание объекта **item_context**.

4.6.3 Объект presentation_layer_usage

Объект **presentation_layer_usage** связывает объекты **presentation_layer_assignment** и **presentation_representation,** определяя контекст стиля и видимости элементов уровня. Ряд объектов **presentation_representation** может быть связан с одним объектом **presentation_layer_assignment**, а один объект **presentation_representation** — с несколькими объектами **presentation_layer_assignment**.

Примечание — Данный метод позволяет размещать на одном уровне различные изображения или одно изображение на разных уровнях.

EXPRESS-спецификация
*)
```
ENTITY presentation_layer_usage;
    assignment      : presentation_layer_assignment;
    presentation    : presentation_representation;
UNIQUE
    UR1: assignment, presentation;
END_ENTITY;
```
(*

Определения атрибутов

assignment — на уровень, определенный в объекте **presentation_representation**, следует ссылаться посредством атрибута **presentation**;

presentation — объект **presentation_representation**, содержащий отображение уровня, на который ссылаются посредством атрибута **assignment**.

Формальное утверждение

UR1 — пара атрибутов (**assignment**, **presentation**) должна быть уникальной. Заданный уровень не может быть представлен в изображении более одного раза.

4.7 Определения объектов схемы структуры представления: связь представления с моделью изделия

4.7.1 Объект presented_item_representation

Объект **presented_item_representation** устанавливает связь объектов **presentation_representation** или **presentation_set** с соответствующим элементом изображения.

Примечание — Это позволяет описать множественные отношения между представлением и его отображением.

EXPRESS-спецификация
*)
```
ENTITY presented_item_representation;
    presentation    : presentation_representation_select;
    item            : presented_item;
END_ENTITY;
```
(*

Определения атрибутов

presentation — объект **presentation_representation** или **presentation_set**;

item — элемент, представленный в изображении.

4.7.2 Объект presented_item

Объект **presented_item** является обозначением изображаемого предмета. Спецификация конкретного предмета (реального объекта) должна быть приведена в соответствующем прикладном протоколе.

Пример 6 — В прикладном протоколе представляют свойства конкретного изделия. Данный протокол должен полностью определять конструкцию объекта **presented_item** с точки зрения его увязки с объектом **product_definition**.

EXPRESS-спецификация
*)
```
ENTITY presented_item
    ABSTRACT SUPERTYPE;
END_ENTITY;
```
(*

4.8 Определение правила схемы структуры представления

4.8.1 Правило symbol_representation_rule

Правило **symbol_representation_rule** гарантирует, что объект **presentation_representation_relationship,** связывающий два примитива **symbol_representation**, является примитивом **symbol_representation_relationship**.

EXPRESS-спецификация
*)
```
RULE symbol_representation_rule
```

```
    FOR (presentation_representation_relationship);
WHERE
    WR1: SIZEOF (QUERY(each_1 <* presentation_representation_relationship |
                    NOT ('PRESENTATION_DEFINITION_SCHEMA.' +
                    'SYMBOL_REPRESENTATION_RELATIONSHIP'+IN TYPEOF (each_1)) AND
            (SIZEOF (QUERY(each_2 <* [each_1\representation_relationship.rep_1,
                                each_1\representation_relationship.rep_2] |
            'PRESENTATION_DEFINITION_SCHEMA. SYMBOL_REPRESENTATION'IN
            TYPEOF (each_2)
            )) > 0)
        )) = 0;
END_RULE;
(*
```

Формальное утверждение

WR1 — объекты **presentation_representation_relationship**, не являющиеся примитивами **symbol_representation_relationship**, не могут быть связаны с примитивами **symbol_representations**.

4.9 Определение функции схемы структуры представления

4.9.1 Функция acyclic_presentation_representation_relationship

Функция **acyclic_presentation_representation_relationship** проверяет, является ли ацикличным граф, определенный объектом **presentation_representation_relationship**, связанным объектом **presentation_representation** и всеми другими объектами **presentation_representation_relationship**, ссылающимися на данный **presentation_representation**. Функция возвращает значение TRUE, если граф является ацикличным, и значение FALSE — в противном случае.

EXPRESS-спецификация

```
*)
FUNCTION acyclic_presentation_representation_relationship
    ( relation      : presentation_representation_relationship;
      children      : SET OF presentation_representation) : BOOLEAN
    LOCAL
      x             : SET OF presentation_representation_relationship;
      local_children    : SET OF presentation_representation
    END_LOCAL;
    REPEAT i: = 1 TO HIINDEX (children);
      IF relation\representation_relationship.rep_1 :=: children[i] THEN
        RETURN(FALSE);
        END_IF;
    END_REPEAT;
    x := bag_to_set (USEDIN (relation\representation_relationship.rep_1,
                    'REPRESENTATION_SCHEMA.'+
                    'REPRESENTATION_RELATIONSHIP.REP_2));
    local_children := children + relation\representation_relationship.rep_1;
    IF SIZEOF (x) > 0 THEN
      REPEAT i:=1 TO HIINDEX (x);
            IF NOT acyclic_presentation_representation_relationship
                    (x[i], local_children) THEN
              RETURN (FALSE);
            END_IF;
      END_REPEAT;
    END_IF;
    RETURN (TRUE);
END_FUNCTION;
(*
```

Определения аргументов

relation — проверяемый объект **presentation_representation_relationship**. Данный объект является исходным для рассматриваемой функции;

children — объекты **presentation_representation**, связанные посредством объекта **relation**. Они служат исходными данными для рассматриваемой функции. В качестве исходных данных набор этих объектов содержит только атрибут **rep_2** объекта **relation**.

EXPRESS-спецификация

```
*)
END_SCHEMA; - - presentation_organization_schema
(*
```

5 Описание представления

Следующее описание на языке EXPRESS открывает **presentation_definition_schema** и определяет необходимые внешние ссылки.

EXPRESS-спецификация

```
*)
SCHEMA presentation_definition_schema;
REFERENCE FROM external_reference_schema
        (externally_defined_item,
         pre_defined_item;
REFERENCE FROM geometry_schema
        (axis2_placement,
         curve,
         geometric_representation_item,
         point
         );
REFERENCE FROM measure_schema
        (positive_ratio_measure);
REFERENCE FROM presentation_appearance_schema
        (styled_item);
REFERENCE FROM presentation_resource_schema
        (character_glyph_symbol,
         planar_box,
         planar_extent,
         font_select,
         presentable_text,
         text_font);
REFERENCE FROM representation_schema
        (item_in_context,
         mapped_item,
         representation,
         representation_item,
         representation_map,
         representation_relationship,
         representation_relationship_with_transformation,
         using_representations);
REFERENCE FROM support_resource_schema
        (label,
         text,
         bag_to_set);
(*
```

Примечания

1 Схемы, на которые выше даны ссылки, можно найти в следующих стандартах:

external_reference_schema	ГОСТ Р ИСО 10303-41;
geometry_schema	ИСО 10303-42;
presentation_appearance_schema	Раздел 6 настоящего стандарта;
presentation_resource_schema	Раздел 7 настоящего стандарта;
representation_schema	ГОСТ Р ИСО 10303-43.

2 Графическое представление определяемой схемы приведено в приложении Е с использованием нотации языка EXPRESS-G.

5.1 Введение

Схема **presentation_definition_schema** устанавливает структуру для описания аннотации и присвоения стилей соответствующим примитивам. Аннотации, определяемые в настоящем стандарте, всегда двумерны (планарны), но могут быть размещены в трехмерном пространстве.

5.2 Фундаментальные понятия и допущения

Примитивы аннотаций являются элементами, из которых создают символические представления аннотаций. Данными примитивами, определенными в настоящей схеме, являются аннотируемые точки, кривые, заполняемые области, тексты, символы и таблицы.

Аннотируемой является точка, представляемая с использованием объекта **point_style**. Более подробная информация об этом объекте приведена в разделе 6. Схема вида представления не определяет специального объекта для аннотируемых точек, потому что объект **point**, определенный в ИСО 10303-42, достаточен в смысле аннотирования.

Аннотируемая кривая является плоской (планарной) кривой, представляемой с использованием объекта **curve_style**. Более подробная информация об этом объекте приведена в разделе 6. Схема вида представления не определяет специального объекта для аннотируемых кривых, потому что объект **curve**, определенный в ИСО 10303-42, достаточен с точки зрения аннотирования.

Аннотируемый текст представляет собой набор (коллекцию) символов, простых символьных строк, сочетаний различных строк и более сложных наборов строк и символов (знаков). Аннотированный текст определяют посредством объекта **annotation_text**, в котором использовано понятие объекта **mapped_item**. Более подробная информация об объекте **mapped_item** приведена в ГОСТ Р ИСО 10303-43. Объект **annotation_text** определяет местоположение и ориентацию набора символов, простых символьных строк, сочетаний различных строк и более сложных наборов строк и символов, заданных в примитиве **text_string_representation**. Несколько объектов **annotation_text** могут ссылаться на один и тот же примитив **text_string_representation**. Данный примитив представляет собой набор объектов **annotation_text**, **defined_character_glyph**, **annotation_text_character**, **text_literal** или **composite_text**. Подобная структура обеспечивает рекурсивную конструкцию объекта **annotation_text**. Символы (знаки) могут быть заданы заранее (предопределены), определены из вне или при соответствующем обмене согласно положениям настоящего стандарта. В последнем случае объект **annotation_text_character** должен ссылаться на объект **character_glyph_symbol**. Этот объект содержит геометрическое представление символа. Примитив **text_literal** позволяет определить текстовую строку в качестве части объекта **annotation_text**. Сложный набор конструктивов может быть размещен и стилизован как единое целое посредством объекта **composite_text**. Дальнейшая детализация объекта **annotation_text** позволяет дополнительно определить в нем для заданного текста границы затенения, обрамления, выделения (подчеркивания, надчеркивания) или внешние границы. Объект **annotation_text** представляют посредством объекта **text_style**. Более подробные сведения об объекте **text_style** приведены в разделе 6.

Аннотируемым символом является предопределенный или внешне определяемый символ или набор объектов **representation_item**, образующий графический символ (например, мнемограмму). Аннотируемый символ определяют посредством объектов **defined_symbol** или **annotation_symbol**. Объект **defined_symbol** масштабирует, позиционирует и ориентирует неявное описание символа. Объект **annotation_symbol** использует положения объекта **mapped_item**. Более полная информация об объекте **mapped_item** приведена в ГОСТ Р ИСО 10303-43. Объект **annotation_symbol** масштабирует, позиционирует и ориентирует набор (коллекцию) объектов **representation_item**, заданный в объекте **symbol_representation**. Несколько объектов **annotation_symbol** могут ссылаться на один и тот же объект **symbol_representation**. Объекты **representation_item**, образующие символ, могут быть элементами геометрической формы (см. ИСО 10303-42), примитивами или экземплярами аннотации. Подобная структура позволяет рекурсивно конструировать символы. Кроме того, сам объект **symbol_representation** может быть встроен в другие аналогичные объекты. Подобную иерархию определяют посредством отношения между двумя объектами **symbol_representation**. Это отношение задают посредством объекта типа **symbol_representation_relationship**, ссылающегося на пару отображений. Одно из них называют **rep_1**, а другое — **rep_2**; отношение направлено от родителя (**rep_1**) к потомку (**rep_2**). Данное отношение дополнительно определяет преобразование. Это преобразование должно быть выполнено для перевода элементов из **rep_2** в систему координат **rep_1**. Тогда один объект **symbol_representation**

представляет собой набор всех объектов **representation_item** данного отображения, а также всех объектов **symbol_representation**, являющихся узлами в «дереве» отношений, связывающих данное отображение. Символы представляют посредством объекта **symbol_style**, представляющего собой набор стилей представления. Более подробная информация об объекте **symbol_style** приведена в разделе 6.

Аннотируемой таблицей является специальный тип примитива **annotation_symbol**, отображающий таблицу. Аннотируемую таблицу определяют посредством объекта **annotation_table**, использующего положения объекта **mapped_item**. Более подробная информация об объекте **mapped_item** приведена в ГОСТ Р ИСО 10303-43. Объект **annotation_table** масштабирует, позиционирует и ориентирует таблицу, заданную в объекте **table_representation**. Несколько объектов **annotation_table** могут ссылаться на один и тот же объект **table_representation**. Объект **table_representation** может быть встроен в объекты **table_record_representation**, которые могут образовывать ряд объектов **table_record_field_representation**. Объект **table_record_field_representation** обычно соответствует столбцу или строке таблицы, а объект **table_record_field_representation** — одной ячейке таблицы. Однако могут быть созданы более сложные структуры, потому что объекты **table_record_field_representation** могут входить в другие подобные объекты. Подобная иерархия представляется объектом **table_representation**, содержащим соответствующие компоненты из объектов **table_representation_relationship**, уточняющих специализацию объекта **symbol_representation_relationship**. Объект **annotation_table** определяет только пустую таблицу. Для внесения текста в таблицу следует отдельно определить экземпляры объекта **annotation_text** в зависимости от их расположения в таблице. Для обеспечения связи этих текстов с конкретными полями таблицы может быть использован объект **table_text_relationship**.

Примитивы аннотаций могут быть описаны только вместе с заданными для них стилями. Аннотируемый примитив вместе с его стилем называют объектом **annotation_occurrence**. Для каждого типа примитива должен быть определен специальный объект **annotation_occurrence**, определяющий ограничения стиля представления для данного типа. Объект **annotation_occurrence_relationship** позволяет определить отношение между двумя объектами **annotation_occurrence**. Объект **table_text_relationship** связывает объект **annotation_text_occurrence** с объектом **table_record_field_representation** аннотируемой таблицы.

5.3 Определения типов схемы описания представления

5.3.1 Тип text_delineation

Тип **text_delineation** предназначен для контроля за выделениями в тексте.

Примечание — Соответствующие значения **text_delineation** и их точный смысл могут быть установлены в прикладных протоколах.

EXPRESS-спецификация

```
*)
TYPE text_delineation = label;
END_TYPE;
(*
```

Пример 7 — В прикладном протоколе могут быть заданы только следующие допустимые значения данного типа: 'underline' (подчеркивание) и 'overline' (надчеркивание) и соответствующие им выделения текста (см. рисунок 12).

Надчеркнутый текст **Подчеркнутый текст**

Рисунок 12 — Примеры выделения текста

5.3.2 Тип defined_symbol_select

Тип **defined_symbol_select** задает неявное описание примитива **defined_symbol**.

EXPRESS-спецификация

```
*)
TYPE defined_symbol_select = SELECT;
    (pre_defined_symbol,
     externally_defined_symbol);
END_TYPE;
(*
```

5.3.3 Тип text_or_character

Тип **text_or_character** определяет элементы, которые могут быть использованы в примитивах **annotation_text** или **composite_text**.

EXPRESS-спецификация

```
*)
TYPE text_or_character = SELECT;
    (annotation_text,
     annotation_text_character,
     defined_character_glyph,
     composite_text,
     text_literal);
END_TYPE;
(*
```

5.3.4 Тип text_alignment

Тип **text_alignment** предназначен для контроля за выравниванием текста.

Примечание — Соответствующие значения **text_alignment** и их точный смысл могут быть установлены в прикладных протоколах.

EXPRESS-спецификация

```
*)
TYPE text_alignment = label;
END_TYPE;
(*
```

Пример 8 — В прикладном протоколе могут быть заданы только следующие допустимые значения данного типа: 'left' (левостороннее), 'centr' (центральное), 'right' (правостороннее) и соответствующее им расположения текста относительно точки его выравнивания (см. рисунок 13).

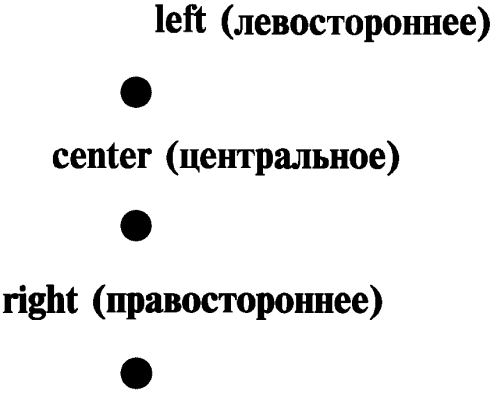


Рисунок 13 — Примеры выравнивания текста

5.3.5 Тип defined_glyph_select

Объект **defined_glyph_select** определяет выбор одного из примитивов: **pre_defined_character_glyph** или **externally_defined_character_glyph**.

EXPRESS-спецификация

```
*)
TYPE defined_glyph_select = SELECT;
    (pre_defined_character_glyph,
     externally_defined_character_glyph);
END_TYPE;
(*
```

5.3.6 Тип text_path

Тип **text_path** определяет расположение следующего символа текста относительно текущего символа. Следующий символ может быть расположен слева, справа, ниже или выше текущего символа.

EXPRESS-спецификация

```
*)
TYPE text_path = ENUMERATION OF
    (left,
     right,
     up,
     down);
END_TYPE;
(*
```

Определения перечисленных элементов

left — следующий символ располагают слева от текущего;

right — следующий символ располагают справа от текущего;

up — следующий символ располагают выше текущего;

down — следующий символ располагают ниже текущего.

5.4 Определение объектов схемы описания представления: примитивы комментариев

5.4.1 Примитив annotation_fill_area

Примитив **annotation_fill_area** представляет собой набор замкнутых кривых, поле которых должно быть заполнено штриховкой, затенением, цветом или мозаикой. Данный примитив описывают посредством границ непересекающихся или несамопересекающихся замкнутых кривых. Данные кривые определяют границы планарных областей, заполняемых в соответствии со стилем, заданным в **annotation_fill_area**. Заполнение областей проводят в соответствии со следующими правилами:

- кривая, не охваченная любой другой кривой, является границей между незаполняемой внешней и заполняемой внутренней областями.

Примечание 1 — См. рисунок 14a;

- кривая, окруженная незаполненной областью, если она охватывает другую внутреннюю кривую, определяет заполняемую область.

Примечание 2 — См. рисунок 14b;

- если третья кривая расположена внутри второй, то она охватывает заполняемую область.

Примечание 3 — См. рисунок 14c;

- для каждой последующей кривой применяют описанную выше процедуру в том же порядке.

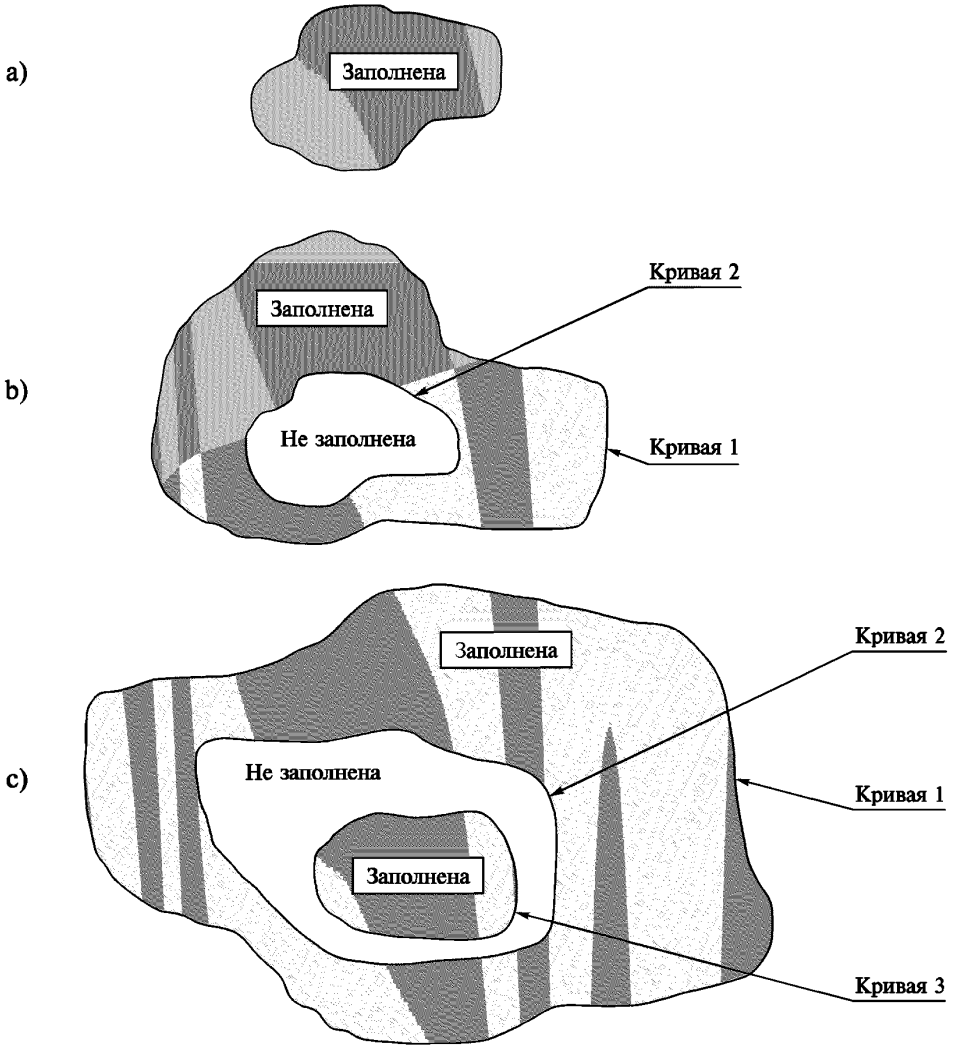


Рисунок 14 — Заполнение аннотируемых областей

EXPRESS-спецификация
```
*)
ENTITY annotation_fill_area
    SUBTYPE OF (geometric_representation_item);
    boundaries    :  SET (1:?] OF curve;
END_ENTITY;
(*
```
Определение атрибута

boundaries — набор кривых, определяющих области заполнения.

Неформальные утверждения

IP1 — все кривые в наборе, заданном атрибутом **boundaries**, должны быть замкнутыми и планарными;

IP2 — если в набор, заданный атрибутом **boundaries**, входит не менее двух кривых, все они должны быть компланарными (расположенными в одной плоскости), и любая пара кривых не должна пересекаться между собой.

5.4.2 Примитив defined_symbol

Примитив **defined_symbol** представляет собой неявное определение символа посредством примитивов **pre_defined_symbol** или **externally_defined_symbol**.

EXPRESS-спецификация
```
*)
ENTITY defined_symbol
    SUBTYPE OF (geometric_representation_item);
    definition    :  defined_symbol_select;
    target        :  symbol_target;
END_ENTITY;
(*
```
Определения атрибутов

definition — неявное описание символа (предопределенное или заданное из вне);

target — описание масштабирования, местоположения и ориентации примитива **defined_symbol**.

5.4.3 Примитив defined_table

Примитив **defined_table** является примитивом **defined_symbol**, определяющим неявное описание таблицы.

EXPRESS-спецификация
```
*)
ENTITY defined_table
    SUBTYPE OF (defined_symbol);
END_ENTITY;
(*
```
5.4.4 Примитив symbol_target

Примитив **symbol_target** является указателем преобразования, определяющего местоположение и ориентацию примитивов **symbol_representation** или **defined_symbol**, используемых в качестве примитива **annotation_symbol**. Рассматриваемый примитив определяет ориентированное местоположение и масштабирование по осям *x* и *y* указанных в нем примитивов.

EXPRESS-спецификация
```
*)
ENTITY symbol_target
    SUBTYPE OF (geometric_representation_item);
    placement     :  axis2_placement;
    x-scale       :   positive_ratio_measure;
    y-scale       :   positive_ratio_measure;
END_ENTITY;
(*
```
Определения атрибутов

placement — местоположение и ориентация примитивов **symbol_representation** или **defined_symbol**, используемых в качестве **annotation_symbol**;

x-scale — масштаб примитивов **symbol_representation** или **defined_symbol** по оси *x*;

y-scale — масштаб примитивов **symbol_representation** или **defined_symbol** по оси *y*.

5.4.5 Примитив pre_defined_symbol

Примитив **pre_defined_symbol** указывает на возможность обмена информацией для определения символа, используемого в конкретном приложении. Фактическое представление данного символа должно быть определено в прикладном протоколе.

EXPRESS-спецификация

```
*)
ENTITY pre_defined_symbol
    SUBTYPE OF (pre_defined_item);
END_ENTITY;
(*
```

5.4.6 Примитив externally_defined_symbol

Примитивом **externally_defined_symbol** является объект **externally_defined_item**, имеющий внешнюю ссылку на данный символ.

EXPRESS-спецификация

```
*)
ENTITY externally_defined_symbol
    SUBTYPE OF (externally_defined_item);
END_ENTITY;
(*
```

5.4.7 Примитив annotation_symbol

Примитивом **annotation_symbol** является отображение примитива **symbol_representation** в виде объекта **geometric_representation_item**, указывающее на использование примитива **symbol_representation** в качестве элемента изображения.

Примечание — Примеры аннотируемых символов приведены на рисунке 15.

EXPRESS-спецификация

```
*)
ENTITY annotation_symbol
    SUBTYPE OF (mapped_item);
WHERE:
    WR1: 'PRESENTATION_DEFINITION_SCHEMA.SYMBOL_REPRESENTATION_MAP' IN
         TYPEOF (SELF\mapped_item_mapping_source);
    WR2: 'PRESENTATION_DEFINITION_SCHEMA.SYMBOL_TARGET' IN
         TYPEOF (SELF\mapped_item_mapping_target);
    WR3: 'GEOMETRY_SCHEMA.GEOMETRIC_REPRESENTATION_ITEM' IN
         TYPEOF (SELF);
END_ENTITY;
(*
```

Определения атрибутов

SELF\mapped_item_mapping_source — примитив **symbol_representation_map**, отображающий **symbol_representation**;

SELF\mapped_item_mapping_target — примитив **symbol_target**, определяющий местоположение символа.

Формальные утверждения

WR1 — объектом **mapping_source** должен быть примитив **symbol_representation_map**;

WR2 — объектом **mapping_target** должен быть примитив **symbol_target**;

WR3 — экземпляром примитива **annotation_symbol** должен быть экземпляр объекта **geometric_representation_item**.

5.4.8 Примитив annotation_table

Примитив **annotation_table** является отображением примитива **table_representation** для представления его в качестве элемента изображения.

EXPRESS-спецификация

```
*)
```

```
ENTITY annotation_table
    SUBTYPE OF (annotation_symbol);
WHERE:
    WR1:  'PRESENTATION_DEFINITION_SCHEMA.TABLE_REPRESENTATION' IN
          TYPEOF (SELF\mapped_item_mapping_source.mapped_representation);
END_ENTITY;
(*
```

Формальное утверждение

WR1 — отображаемым объектом **representation** должен быть примитив **table_representation**.

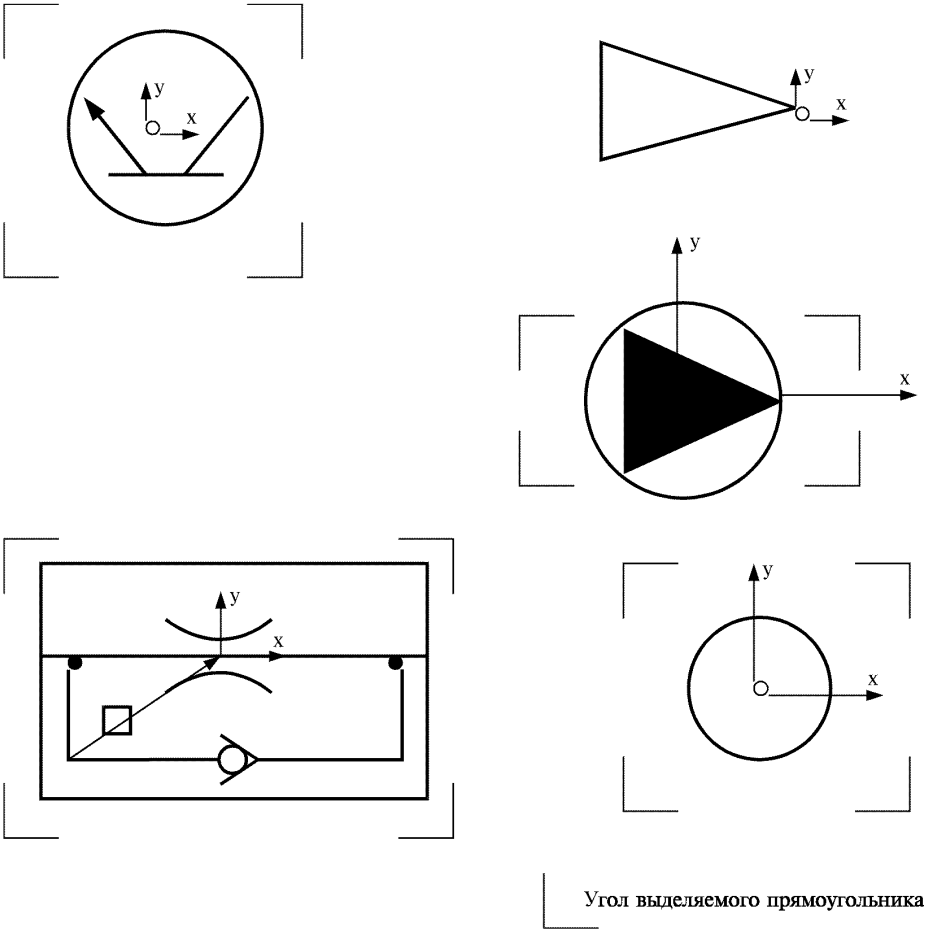


Рисунок 15 — Примеры аннотируемых символов

5.4.9 Примитив symbol_representation_map

Примитивом **symbol_representation_map** является объект **representation_map**, отображающий примитив **symbol_representation** в **annotation_symbol**.

EXPRESS-спецификация

```
*)
ENTITY symbol_representation_map
    SUBTYPE OF (representation_map);
WHERE:
    WR1: 'PRESENTATION_DEFINITION_SCHEMA.SYMBOL_REPRESENTATION' IN
          TYPEOF (SELF\representation_map.mapped_representation);
```

```
    WR2: 'GEOMETRY_SCHEMA.AXIS2_PLACEMENT' IN
         TYPEOF (SELF\representation_map.mapping_origin);
END_ENTITY;
(*
```

Определения атрибутов

SELF\representation_map.mapped_representation — примитив **symbol_representation**, входящий в **annotation_symbol**;

SELF\representation_map.mapping_origin — объект **axis2_placement**, определяющий отображаемый оригинал.

Формальные утверждения

WR1 — объектом **mapped_representation** должен быть примитив **symbol_representation**;

WR2 — объектом **mapping_origin** должен быть объект **axis2_placement.**

5.4.10 Примитив symbol_representation

Примитив **symbol_representation** является видом объекта **representation**, используемым при формировании символа. Это позволяет определить примитивы **annotation_table**, конкретизирующие примитив **annotation_symbol**. Примитивы **symbol_representation** могут быть связаны друг с другом только посредством примитивов **symbol_representation_relationship**.

EXPRESS-спецификация

```
*)
ENTITY symbol_representation
    SUBTYPE OF (representation);
END_ENTITY;
(*
```

5.4.11 Примитив symbol_representation_with_blanking_box

Примитивом **symbol_representation_with_blanking_box** является примитив **symbol_representation**, определяющий прямоугольник для отображения символа, не содержащий других элементов.

EXPRESS-спецификация

```
*)
ENTITY symbol_representation_with_blanking_box
    SUBTYPE OF (symbol_representation);
    blanking : planar_box
WHERE
    WR1: item_in_context (SELF.blanking, SELF\representation_context_of_items);
END_ENTITY;
(*
```

Определение атрибута

blanking — прямоугольник, в котором размещен только отображаемый символ.

Формальное утверждение

WR1 — данный прямоугольник должен быть представлен в контексте атрибута **SELF.context_of_items**.

5.4.12 Примитив table_representation

Примитив **table_representation** определяет вид символа, используемый при формировании таблицы. Данные примитивы могут быть связаны между собой только посредством объектов **table_representation_relationship**. Если примитив **table_representation** играет роль **rep_1** в **table_representation_relationship**, тогда роль **rep_2** может играть только примитив **table_record_representation**.

EXPRESS-спецификация

```
*)
ENTITY table_representation
    SUBTYPE OF (symbol_representation);
END_ENTITY;
(*
```

5.4.13 Примитив table_record_representation

Примитив **table_record_representation** представляет собой вид символа, используемый для создания записей в таблицах. Данные примитивы могут быть связаны между собой только посредством

примитивов **table_representation_relationship**. Если рассматриваемый примитив играет роль **rep_1** в **table_representation_relationship**, тогда только **table_record_field_representation** может играть роль **rep_2**.

EXPRESS-спецификация

```
*)
ENTITY table_record_representation
    SUBTYPE OF (symbol_representation)
WHERE
    WR1: (SIZEOF(USEDIN(SELF, 'REPRESENTATION_SCHEMA.' +
                              'REPRESENTATION_RELATIONSHIP.REP_2')) > 0)
                          OR
         (SIZEOF(QUERY(map_item <* USEDIN(SELF, 'REPRESENTATION_SCHEMA.' +
                                                'REPRESENTATION_MAP.' +
                                                'MAPPED_REPRESENTATION') |
          'PRESENTATION_DEFINITION_SCHEMA.TABLE_REPRESENTATION' IN
          TYPEOF (using_representation (map_item)) )) > 0);
END_ENTITY;
(*
```

Формальное утверждение

WR1 — примитив **table_record_representation** должен быть использован как **rep_2** в **table_representation_relationship** или отображен в **table_representation**.

5.4.14 Примитив table_record_field_representation

Примитив **table_record_field_representation** представляет собой вид символа, используемого при создании полей в записях таблицы. Рассматриваемые примитивы могут быть связаны между собой только посредством примитивов **table_representation_relationship**. Если рассматриваемый примитив играет роль **rep_1** в **table_representation_relationship**, тогда только **table_record_field_representation** может играть роль **rep_2**.

EXPRESS-спецификация

```
*)
ENTITY table_record_field_representation
    SUBTYPE OF (symbol_representation);
WHERE
    WR1: (SIZEOF(USEDIN(SELF, 'REPRESENTATION_SCHEMA.' +
                              'REPRESENTATION_RELATIONSHIP.REP_2')) > 0)
                          OR
         (SIZEOF(QUERY(map_item <* USEDIN(SELF, 'REPRESENTATION_SCHEMA.' +
                                                'REPRESENTATION_MAP.' +
                                                'MAPPED_REPRESENTATION') |
                                   'PRESENTATION_DEFINITION_SCHEMA.' +
                                   'TABLE_RECORD_REPRESENTATION' IN
          TYPEOF (using_representation (map_item)) )) > 0);
END_ENTITY;
(*
```

Формальное утверждение

WR1 — примитив **table_record_field_representation** должен быть использован как **rep_2** в **table_representation_relationship** или отображен в **table_record_representation**.

5.4.15 Примитив table_record_field_representation_with_clipping_box

Примитивом **table_record_field_representation_with_clipping_box** является примитив **table_record_field_representation**, включающий ограничивающий прямоугольник, очищенный от всех внешних элементов.

EXPRESS-спецификация

```
*)
ENTITY table_record_field_representation_with_clipping_box
    SUBTYPE OF (table_record_field_representation);
    clipping_box: planar_box;
```

```
WHERE
    WR1: item_in_context (SELF.clipping_box,
                          SELF\representation.context_of_items);
END_ENTITY;
(*
```

Определение атрибута

clipping_box — объект **planar_box**, определяющий границы выделения содержимого примитива **table_record_field_representation**.

Формальное утверждение

WR1 — атрибут **clipping_box** должен быть представлен в контексте примитива **table_record_field_representation_with_clipping_box**.

5.4.16 Примитив symbol_representation_relationship

Примитивом **symbol_representation_relationship** является вид объекта **representation_relationship_with_transformation**, используемый для связи объектов **symbol_representation**.

EXPRESS-спецификация

```
*)
ENTITY symbol_representation_relationship
    SUBTYPE OF (representation_relationship_with_transformation);
WHERE
    WR1: acyclic symbol_representation_relationship (SELF,
                                                     [SELF\representation_relationship.rep_2]);
    WR2: 'PRESENTATION_DEFINITION_SCHEMA.SYMBOL_REPRESENTATION'IN
         TYPEOF (SELF\representation_relationship.rep_1);
    WR3: 'PRESENTATION_DEFINITION_SCHEMA.SYMBOL_REPRESENTATION'IN
         TYPEOF (SELF\representation_relationship.rep_2);
END_ENTITY;
(*
```

Определения атрибутов

SELF\representation_relationship.rep_1 — объект **symbol_representation**, играющий роль «родителя (корня)» в «дереве» объектов **symbol_representation**;

SELF\representation_relationship.rep_2 — объект **symbol_representation**, играющий роль «потомка (листа)» в «дереве» объектов **symbol_representation**.

Формальные утверждения

WR1 — объект **symbol_representation_relationship** не должен входить в «дерево» объектов **symbol_representation**, «корнем» которого является «лист дерева» данных объектов;

WR2 — атрибутом **rep_1** объекта **symbol_representation_relationship** должен быть объект **symbol_representation**;

WR3 — атрибутом **rep_2** объекта **symbol_representation_relationship** должен быть объект **symbol_representation**.

Примечание — Атрибуты **rep_1** и **rep_2** определены в ГОСТ Р ИСО 10303-43.

5.4.17 Примитив table_representation_relationship

Примитивом **table_representation_relationship** является вид объекта **symbol_representation_relationship**, используемый для связи с объектами **table_representation**, **table_record_representation** и **table_record_field_representation**.

EXPRESS-спецификация

```
*)
ENTITY table_representation_relationship
    SUBTYPE OF (sembol_representation_relationship);
WHERE
    WR1: NOT('PRESENTATION_DEFINITION_SCHEMA. TABLE_RECORD_REPRESENTATION'IN
              TYPEOF (SELF\representation_relationship_rep_1);
         XOR
('PRESENTATION_DEFINITION_SCHEMA.TABLE_RECORD_FIELD_REPRESENTATION'IN
              TYPEOF (SELF\representation_relationship.rep_2));
```

```
    WR2: NOT ('PRESENTATION_DEFINITION_SCHEMA.TABLE_REPRESENTATION'IN
              TYPEOF (SELF\representation_relationship.rep_1))
             XOR
         ('PRESENTATION_DEFINITION_SCHEMA.TABLE_RECORD_REPRESENTATION'IN
              TYPEOF (SELF\representation_relationship.rep_2));
    WR3: NOT ('PRESENTATION_ DEFINITION_SCHEMA. TABLE_RECORD_FIELD_
              REPRESENTATION'IN
              TYPEOF (SELF\representation_relationship.rep_1))
             XOR
          ('PRESENTATION_DEFINITION_SCHEMA. TABLE_RECORD_ FIELD_
            REPRESENTATION'IN
              TYPEOF (SELF\representation_relationship.rep_2));
END_ENTITY:
(*
```

Формальные утверждения

WR1 — если примитив **table_record_representation** играет роль атрибута **SELF\representation_relationship.rep_1**, тогда объект **table_record_field_representation** должен выполнять роль атрибута **SELF\representation_relationship.rep_2**;

WR2 — если примитив **table_representation** выполняет роль атрибута **SELF\representation_relationship.rep_1**, тогда примитив **table_record_representation** должен выполнять роль атрибута **SELF\representation_relationship.rep_2**;

WR3 — если примитив **table_record_field_representation** выполняет роль атрибута **SELF\representation_relationship.rep_1**, тогда данный примитив также должен выполнять роль атрибута **SELF\representation_relationship.rep_2**.

5.4.18 Примитив annotation_text

Примитив **annotation_text** представляет собой отображение примитива **text_string_representation**, содержащее объекты **annotation_text_character**, **text_literal**, **composite_text** или **annotation_text** в качестве объекта **geometric_representation_item**, определяющего, что **text_string_representation** является частью изображения. Данная структура позволяет представить **annotation_text** в виде простых символьных строк, набора подобных строк или более сложных наборов строк и символов.

EXPRESS-спецификация

```
*)
ENTITY annotation_text
    SUBTYPE OF (mapped_item);
WHERE
    WR1: 'GEOMETRY_SCHEMA.AXIS2_PLACEMENT' IN
             TYPEOF (SELF\mapped_item.mapping_target);
    WR2: 'PRESENTATION_DEFINITION_SCHEMA.TEXT_STRING_REPRESENTATION'IN
             TYPEOF (SELF\mapped_item.mapping_source.mapped_representation);
    WR3: 'GEOMETRY_SCHEMA.GEOMETRIC_REPRESENTATION_ITEM' IN
             TYPEOF (SELF);
END_ENTITY;
(*
```

Определения атрибутов

SELF\mapped_item.mapping_source — объект **representation_map**, отображающий примитив **text_string_representation**, содержащий текст, символы или объекты **text_literal_symbol** для примитива **annotation_text**;

SELF\mapped_item.mapping_target — объект **axis2_placement**, определяющий местоположение и ориентацию объекта **annotation_text_map**.

Формальные утверждения

WR1 — объектом **mapping_target** должен быть объект **axis2_placement**;

WR2 — объектом **mapped_representation** должен быть примитив **text_string_representation**;

WR3 — экземпляром объекта **annotation_text** должен быть экземпляр объекта **geometric_representation_item**.

5.4.19 Примитив annotation_text_with_extent

Примитивом **annotation_text_with_extent** является примитив **annotation_text**, объем (протяженность) которого четко задан.

EXPRESS-спецификация

```
*)
ENTITY annotation_text_with_extent
    SUBTYPE OF (annotation_text);
    extent : planar_extent;
END_ENTITY;
(*
```

Определение атрибута

extent — протяженность примитива **annotation_text** по осям *x* и *y* системы локальных координат, заданной атрибутом **placement**.

5.4.20 Примитив annotation_text_with_delineation

Примитивом **annotation_text_with_delineation** является примитив **annotation_text**, определяющий выделение текста. Тип выделения и его влияние на вид текста задают в прикладном протоколе.

EXPRESS-спецификация

```
*)
ENTITY annotation_text_with_delineation
    SUBTYPE OF (annotation_text);
    delineation : text_delineation;
END_ENTITY;
(*
```

Определение атрибута

delineation — требования к характеристикам выделения текста.

5.4.21 Примитив annotation_text_with_blanking_box

Примитивом **annotation_text_with_blanking_box** является примитив **annotation_text**, определяющий габариты прямоугольника, обрамляющего текст.

EXPRESS-спецификация

```
*)
ENTITY annotation_text_with_blanking_box
    SUBTYPE OF (annotation_text);
    blanking :  planar_box;
END_ENTITY;
(*
```

Определение атрибута

blanking — объект **planar_box**, задающий прямоугольную область, внутри которой может быть представлен только примитив **annotation_text**.

5.4.22 Примитив annotation_text_with_associated_curves

Примитивом **annotation_text_with_associated_curves** является примитив **annotation_text**, содержащий одну или несколько кривых (**curve**).

EXPRESS-спецификация

```
*)
ENTITY annotation_text_with_associated_curves
    SUBTYPE OF (annotation_text);
    associated_curves :  SET[1:?] of curves;
END_ENTITY;
(*
```

Определение атрибута

associated_curves — набор кривых (**curve**), связанных с примитивом **annotation_text**.

Примечание — Если имеются кривые (**curve**), связанные с примитивом **annotation_text_with_associated_curves**, стиль их представления может быть задан объектом **annotation_curve_occurrence**.

5.4.23 Примитив text_string_representation

Примитивом **text_string_representation** является объект **representation**, содержащий набор (коллекцию) текстовых строк и символов, отображенных в объекте **annotation_text**.

```
EXPRESS-спецификация
*)
ENTITY text_string_representation
    SUBTYPE OF (representation);
WHERE
    WR1: SIZEOF (
            QUERY (item <* SELF\representation.items |
                SIZEOF (['PRESENTATION_DEFINITION_SCHEMA.TEXT_LITERAL',
                        'PRESENTATION_DEFINITION_SCHEMA.ANNOTATION_TEXT', '
                        'PRESENTATION_DEFINITION_SCHEMA.'ANNOTATION_TEXT_
                        CHARACTER",
                        'PRESENTATION_DEFINITION_SCHEMA.'DEFINED_CHARACTER_
                        GLYPH',
                        'PRESENTATION_DEFINITION_SCHEMA.COMPOSITE_TEXT', '
                        'GEOMETRY_SCHEMA.AXIS2_PLACEMENT]*TYPEOF (item)) = 0
            )) = 0
    WR2: SIZEOF (
            QUERY (item <* SELF\representation.items |
                NOT (SIZEOF (['PRESENTATION_DEFINITION_SCHEMA.TEXT_LITERAL',
                            'PRESENTATION_DEFINITION_SCHEMA. 'ANNOTATION_TEXT', '
                            'PRESENTATION_DEFINITION_SCHEMA. 'ANNOTATION_TEXT_
                            CHARACTER",
                            'PRESENTATION_DEFINITION_SCHEMA.'DEFINED_CHARACTER_
                            GLYPH',
                            'PRESENTATION_DEFINITION_SCHEMA.COMPOSITE_TEXT',]*
                        TYPEOF (item)) = 0)
            )) > = 1;
    WR3: SIZEOF (
            QUERY (a2p <*
                QUERY (item <* SELF\representation.items |
                    'GEOMETRY_SCHEMA.AXIS2_PLACEMENT' IN TYPEOF (item)) |
                NOT ((SIZEOF (
                    QUERY (at <*
                        QUERY (item <* SELF\representation.items |
                            'PRESENTATION_DEFINITION_SCHEMA.'+
                            'ANNOTATION_TEXT' IN TYPEOF (item)) |
                        (at\mapped_item.mapping_target :=: a2p))) >= 1) OR
                SIZEOF (
                  QUERY (atc <*
                      QUERY (item <* SELF\representation.items |
                            'PRESENTATION_DEFINITION_SCHEMA.'+
                            'ANNOTATION_TEXT_CHARACTER' IN TYPEOF (item)) |
                        (atc\mapped_item.mapping_target :=: a2p))) >= 1)
            ))) = 0
END_ENTITY;
(*
```

Определение атрибута

SELF\representation.items — набор объектов **item**, переопределенный в виде набора строк и компоновок.

Формальные утверждения

WR1 — каждым элементом примитива **text_string_representation** должны быть объекты **annotation_text**, **annotation_text_character**, **text_literal**, **composite_text** или **axis2_placement**;

WR2 — примитив **text_string_representation** должен содержать один или несколько объектов **annotation_text**, **annotation_text_character**, **text_literal** или **composite_text**;

WR3 — каждым объектом **axis2_placement** в примитиве **text_string_representation** должен быть **mapping_target** примитива **annotation_text** или **annotation_text_character**.

5.4.24 Примитив annotation_text_character

Примитивом **annotation_text_character** является объект **mapped_item**, содержащий мнемограмму (символ) соответствующего **mapping_source**. Данный символ является глифом в рамках соответствующего обмена данными.

EXPRESS-спецификация

```
*)
ENTITY annotation_text_character
    SUBTYPE OF (mapped_item);
    alignment : text_alignment;
WHERE
    WR1: 'PRESENTATION_RESOURCE_SCHEMA.CHARACTER_GLYPH_SYMBOL' IN
            TYPEOF (SELF\mapped_item.mapping_source.mapped_representation);
    WR2: 'GEOMETRY_SCHEMA.AXIS2_PLACEMENT' IN
            TYPEOF (SELF\mapped_item.mapping_target);
    WR3: 'GEOMETRY_SCHEMA.GEOMETRIC_REPRESENTATION_ITEM' IN
            TYPEOF (SELF);
END_ENTITY;
(*
```

Определения атрибутов

alignment — указывает местоположение символа;

SELF\mapped_item.mapping_source — глиф символа в рамках соответствующего обмена данными;

SELF\mapped_item.mapping_target — объект **axis2_placement**, получаемый в результате отображающего преобразования.

Формальные утверждения

WR1 — источником примитива **annotation_text_character** должен быть объект **character_glyph_symbol**;

WR2 — адресатом примитива **annotation_text_character** должен быть объект **axis2_placement**;

WR3 — примитивом **annotation_text_character** должен быть объект **geometric_representation_item**.

5.4.25 Примитив defined_character_glyph

Примитивом **defined_character_glyph** является глиф символа, неявно заданный посредством примитива **pre_defined_character_glyph** или **externally_defined_character_glyph**.

EXPRESS-спецификация

```
*)
ENTITY defined_character_glyph
    SUBTYPE OF (geometric_representation_item);
    definition   :  defined_glyph_select;
    placement    :  axis2_placement;
END_ENTITY;
(*
```

Определения атрибутов

definition — неявное описание глифа символа (заданное ранее или определенное извне);

placement — описание местоположения и ориентации глифа символа.

5.4.26 Примитив externally_defined_character_glyph

Примитивом **externally_defined_character_glyph** является объект **externally_defined_item**, имеющий ссылку на глиф символа.

EXPRESS-спецификация

```
*)
ENTITY externally_defined_character_glyph
    SUBTYPE OF (externally_defined_item);
END_ENTITY;
(*
```

5.4.27 Примитив pre_defined_character_glyph

Примитив **pre_defined_character_glyph** обеспечивает обмен данными для определения прикладной специфики глифа символа. Фактический глиф символа должен быть задан в прикладном протоколе.

EXPRESS-спецификация

```
*)
ENTITY pre_defined_character_glyph
    SUBTYPE OF (pre_defined_item);
END_ENTITY;
(*
```

5.4.28 Примитив text_literal

Примитив **text_literal** является описанием текстовой строки посредством строкового литерала.

EXPRESS-спецификация

```
*)
ENTITY text_literal
    SUBTYPE OF (geometric_representation_item);
    literal      : presentable_text;
    placement    : axis2_placement;
    alignment    : text_alignment;
    path         : text_path;
    font         : font_select;
END_ENTITY;
(*
```

Определения атрибутов

literal — представляемый текстовый литерал;

placement — объект **axis2_placement**, определяющий местоположение и ориентацию представляемой строки. Ось *y* является эталонным направлением для типов **box_rotate_angle** и **box slant_angle**.

Примечание — Если типы **box_rotate_angle** и **box slant_angle** имеют нулевое значение, основание символьного прямоугольника параллельно оси *x*, а его вертикальная сторона — оси *y* (см. 6.3.29 рисунок 17);

alignment — выравнивание текстового литерала относительно его местоположения;

path — предписанное направление текстового литерала;

font — заранее заданный или внешне определенный шрифт, который должен быть использован при представлении **text_literal**.

5.4.29 Примитив text_literal_with_extent

Примитивом **text_literal_with_extent** является примитив **text_literal** с четко заданной протяженностью.

EXPRESS-спецификация

```
*)
ENTITY text_literal_with_extent
    SUBTYPE OF (text_literal);
    extent     : planar_extent;
END_ENTITY;
(*
```

Определение атрибута

extent — протяженность объекта **text_literal_symbol** по осям *x* и *y*.

5.4.30 Примитив text_literal_with_delineation

Примитивом **text_literal_with_delineation** является примитив **text_literal**, задающий выделение текста. Тип выделения и его влияние на вид текста задают в прикладном протоколе.

EXPRESS-спецификация

```
*)
ENTITY text_literal_with_delineation
    SUBTYPE OF (text_literal);
    delineation     : text_delineation;
END_ENTITY;
(*
```

Определение атрибута

delineation — требования к характеристикам выделения текста.

5.4.31 Примитив text_literal_with_blanking_box

Примитивом **text_literal_with_blanking_box** является примитив **text_literal**, определяющий габариты прямоугольника, обрамляющего текст.

EXPRESS-спецификация

```
*)
ENTITY text_literal_with_blanking_box
    SUBTYPE OF (text_literal);
    blanking      : planar_box;
END_ENTITY;
(*
```

Определение атрибута

blanking — объект **planar_box**, задающий прямоугольную область, внутри которой может быть представлен только примитив **text_literal**.

5.4.32 Примитив text_literal_with_associated_curves

Примитивом **text_literal_with_associated_curves** является примитив **text_literal**, содержащий одну или несколько кривых (**curve**).

EXPRESS-спецификация

```
*)
ENTITY text_literal_with_associated_curves
    SUBTYPE OF (text_literal);
    associated_curves      : SET[1:?] of curve;
END_ENTITY;
(*
```

Определение атрибута

associated_curves — набор кривых (**curve**), связанных с примитивом **text_literal**.

Примечание — Если имеются кривые (**curve**), связанные с примитивом **text_literal_with_associated_curves**, стиль их представления может быть задан объектом **annotation_ curve_occurrence**.

5.4.33 Примитив composite_text

Примитивом **composite_text** является набор (коллекция) примитивов **text_literal**, **annotation_text**, **annotation_text_character**, **defined_character_glyph** или другой примитив **composite_text**, который может быть задан и оформлен в виде коллекции.

EXPRESS-спецификация

```
*)
ENTITY composite_text
    SUBTYPE OF (geometric_representation_item);
    collected_text      : SET[1:?] of text_or_character;
WHERE
    WR1: acyclic_composite_text(SELF, SELF.collected_text);
END_ENTITY;
(*
```

Определение атрибута

collected_text — набор примитивов **text_literal**, **annotation_text**, **annotation_text_character** или другой примитив **composite_text,** заданный и оформленный в виде коллекции.

Формальное утверждение

WR1 — сам сложный текст не должен участвовать в описании рассматриваемого примитива.

5.4.34 Примитив composite_text_with_extent

Примитивом **composite_text_with_extent** является примитив **composite_text** с четко заданной протяженностью.

EXPRESS-спецификация

```
*)
ENTITY composite_text_with_extent
    SUBTYPE OF (composite_text);
    extent      : planar_extent;
```

```
END_ENTITY;
(*
```

Определение атрибута

extent — протяженность примитива **composite_text_with_extent** по осям *x* и *y*.

5.4.35 Примитив composite_text_with_delineation

Примитивом **composite_text_with_delineation** является примитив **composite_text**, задающий выделение текста. Тип выделения и его влияние на вид текста задают в прикладном протоколе.

EXPRESS-спецификация

```
*)
ENTITY composite_text_with_delineation
    SUBTYPE OF (composite_text);
    delineation    : text_delineation;
END_ENTITY;
(*
```

Определение атрибута

delineation — требования к характеристикам выделения текста.

5.4.36 Примитив composite_text_with_blanking_box

Примитивом **composite_text_with_blanking_box** является примитив **composite_text**, определяющий габариты прямоугольника, обрамляющего текст.

EXPRESS-спецификация

```
*)
ENTITY composite_text_with_blanking_box
    SUBTYPE OF (composite_text);
    blanking       :   planar_box;
END_ENTITY;
(*
```

Определение атрибута

blanking — объект **planar_box**, задающий прямоугольную область, внутри которой может быть представлен только примитив **composite_text**.

5.4.37 Примитив composite_text_with_associated_curves

Примитивом **composite_text _with_associated_curves** является примитив **composite_text**, содержащий одну или несколько кривых (**curve**).

EXPRESS-спецификация

```
*)
ENTITY composite_text_with_associated_curves
    SUBTYPE OF (composite_text);
    associated_curves      :  SEN[1:?] of curve;
END_ENTITY;
(*
```

Определение атрибута

associated_curves — набор кривых (**curve**), связанных с примитивом **composite_text**.

Примечание — Если имеются кривые (**curve**), связанные с примитивом **composite_text_with_associated_curves**, стиль их представления может быть задан объектом **annotation_curve_occurrence**.

5.5 Определения объектов схемы описания представления: экземпляры аннотаций

5.5.1 Объект annotation_occurrence

Объект **annotation_occurrence** определяет экземпляры аннотации посредством комбинирования двумерной геометрической формы или элементов аннотации в стиле ее представления. Более подробное описание стиля представления приведено в разделе 6. Объекты **annotation_occurrence** должны быть использованы только в объектах **representations**, определяющих назначение аннотации, то есть в объектах **area_dependent_annotation_representation**, **view_dependent_annotation_representation**, **curve_style_curve_pattern**, **fill_area_style_tile_curve_with_style** или **fill_area_style_tile_coloured_region**.

EXPRESS-спецификация

```
*)
ENTITY annotation_occurrence
    SUPERTYPE OF (ONEOF(annotation_point_occurrence,
```

```
                                annotation_curve_occurrence,
                                annotation_fill_area_occurrence,
                                annotation_text_occurrence,
                                annotation_symbol_occurrence))
    SUBTYPE OF (styled_item);
WHERE
    WR1: 'GEOMETRY_SCHEMA.GEOMETRIC_REPRESENTATION_ITEM' IN
            TYPEOF (SELF);
END_ENTITY;
(*
```

Формальное утверждение

WR1 — объектом **annotation_occurrence** должен быть объект geometric_representation_item.

5.5.2 Объект annotation_point_occurrence

Объектом **annotation_point_occurrence** является объект **point** с заданным стилем представления.

EXPRESS-спецификация

```
*)
ENTITY annotation_point_occurrence
    SUBTYPE OF (annotation_occurrence);
WHERE
    WR1: 'GEOMETRY_SCHEMA.POINT' IN TYPEOF (SELF\styled_item.item);
END_ENTITY;
(*
```

Формальное утверждение

WR1 — элементом, определяющим стиль представления, должен быть объект **point**.

5.5.3 Объект annotation_curve_occurrence

Объектом **annotation_curve_occurrence** является объект **curve** с заданным стилем представления.

EXPRESS-спецификация

```
*)
ENTITY annotation_curve_occurrence
    SUBTYPE OF (annotation_occurrence);
WHERE
    WR1: 'GEOMETRY_SCHEMA.CURVE' IN TYPEOF (SELF\styled_item.item);
END_ENTITY;
(*
```

Формальное утверждение

WR1 — элементом, определяющим стиль представления, должен быть объект **curve**.

5.5.4 Объект annotation_fill_area_occurrence

Объект **annotation_fill_area_occurrence** задает стиль объекта **annotation_fill_area**; он также описывает исходную позицию для объекта **fill_area**.

EXPRESS-спецификация

```
*)
ENTITY annotation_fill_area_occurrence
    SUBTYPE OF (annotation_occurrence);
    fill_style_target : point;
WHERE
    WR1: 'PRESENTATION_DEFINITION_SCHEMA.ANNOTATION_FILL_AREA' IN
            TYPEOF (SELF.item);
END_ENTITY;
(*
```

Определение атрибута

fill_style_target — точка, определяющая исходное местоположение объекта **fill_area_style**, заданное для объекта **annotation_fill_area_occurrence**.

Формальное утверждение

WR1 — элементом, определяющим стиль представления, должен быть объект **annotation_fill_area**.

5.5.5 Объект annotation_text_occurrence

Объектом **annotation_text_occurrence** являются примитивы **text_literal**, **annotation_text**, **annotation_text_character**, **defined_character_glyph** или **composite_text** с заданными стилями их представления.

EXPRESS-спецификация

```
*)
ENTITY annotation_text_occurrence
    SUBTYPE OF (annotation_occurrence);
WHERE
    WR1: SIZEOF (
            ['PRESENTATION_DEFINITION_SCHEMA.TEXT_LITERAL',
             'PRESENTATION_DEFINITION_SCHEMA.ANNOTATION_TEXT',
             'PRESENTATION_DEFINITION_SCHEMA.ANNOTATION_TEXT_CHARACTER',
             'PRESENTATION_DEFINITION_SCHEMA.DEFINED_CHARACTER_GLYPH',
             'PRESENTATION_DEFINITION_SCHEMA.COMPOSITE_TEXT']*
            TYPEOF (SELF\styled_item.item)) > 0;
END_ENTITY;
(*
```

Формальное утверждение

WR1 — элементом с заданным стилем должен быть по крайней мере один из примитивов **text_literal**, **annotation_text**, **annotation_text_character**, **defined_character_glyph** или **composite_text**.

5.5.6 Объект annotation_symbol_occurrence

Объектом **annotation_symbol_occurrence** являются объекты **annotation_symbol** или **defined_symbol** с заданным стилем их представления.

EXPRESS-спецификация

```
*)
ENTITY annotation_symbol_occurrence
    SUBTYPE OF (annotation_occurrence);
WHERE
    WR1: SIZEOF (
            ['PRESENTATION_DEFINITION_SCHEMA.ANNOTATION_SYMBOL',
             'PRESENTATION_DEFINITION_SCHEMA.DEFINED_SYMBOL']*
            TYPEOF (SELF\styled_item.item)) > 0;
END_ENTITY;
(*
```

Формальное утверждение

WR1 — элементом, определяющим стиль представления, должен быть объект **annotation_symbol** или **defined_symbol**.

5.5.7 Объект annotation_table_occurrence

Объектом **annotation_table_occurrence** являются объекты **annotation_table** или **defined_table** с заданным стилем их представления.

EXPRESS-спецификация

```
*)
ENTITY annotation_table_occurrence
    SUBTYPE OF (annotation_symbol_occurrence);
WHERE
    WR1: SIZEOF (
            ['PRESENTATION_DEFINITION_SCHEMA.ANNOTATION_TABLE',
             'PRESENTATION_DEFINITION_SCHEMA.DEFINED_TABLE']*
            TYPEOF (SELF\styled_item.item)) > 0;
END_ENTITY;
(*
```

Формальное утверждение

WR1 — элементом, определяющим стиль представления, должен быть объект **annotation_table** или **defined_table**.

5.5.8 Объект annotation_occurrence_relationship

Объект annotation_occurrence_relationship задает связь между двумя объектами annotation_occurrence.

EXPRESS-спецификация

```
*)
ENTITY annotation_occurrence_relationship;
    name                              : lable;
    description                       : text;
    relating_annotation_occurrence    : annotation_occurrence;
    related_annotation_occurrence     : annotation_occurrence;
END_ENTITY;
(*
```

Определения атрибутов

name — слово или группа слов, посредством которых ссылаются на **annotation_occurrence_relationship**;

description — текст, определяющий характер объекта **annotation_occurrence_relationship**;

relating_annotation_occurrence — первый из двух связываемых объектов **annotation_occurrence**;

related_annotation_occurrence — второй из двух связываемых объектов **annotation_occurrence**.

Примечание — Порядок объектов **annotation_occurrence** не имеет значения. Разные имена атрибутов используют только для различения этих объектов.

5.5.9 Объект table_text_relationship

Объект **table_text_relationship** устанавливает отношение между соответствующими полями в объектах **annotation_table_occurrence** и **annotation_text_occurrence**.

EXPRESS-спецификация

```
*)
ENTITY table_text_relationship
    SUBTYPE OF (annotation_occurrence_relationship);
    field : table_record_field_representation;
WHERE
    WR1: 'PRESENTATION_DEFINITION_SCHEMA.ANNOTATION_TABLE_OCCURRENCE"
         IN TYPEOF (SELF\annotation_occurrence_relationship.
                    relating_annotation_occurrence);
    WR2: 'PRESENTATION_DEFINITION_SCHEMA.ANNOTATION_TABLE"
         IN TYPEOF (SELF\annotation_occurrence_relationship.
                    relating_annotation_occurrence\styled_item.item);
    WR3: 'PRESENTATION_DEFINITION_SCHEMA.ANNOTATION_TEXT_OCCURRENCE»
         IN TYPEOF (SELF\annotation_occurrence_relationship.
                    related_annotation_occurrence);
    WR4:  field_in_table (SELF\field,
                    SELF\annotation_occurrence_relationship.
                    relating_annotation_occurrence);
END_ENTITY;
(*
```

Определения атрибутов

field — конкретное поле таблицы, в котором размещают соответствующий текст;

SELF\annotation_occurrence_relationship.relating_annotation_occurrence — объект **annotation_table_occurrence,** содержащий текст;

SELF\annotation_occurrence_relationship.related_annotation_occurrence — объект **annotation_text_occurrence,** размещенный в таблице.

Формальные утверждения

WR1 — объектом **relating_annotation_occurrence** должен быть объект **annotation_table_occurrence**;

WR2 — объектом **relating_annotation_occurrence** должен представлять объект **annotation_table**;

WR3 — объектом **related_annotation_occurrence** должен быть объект **annotation_text_occurrence**;

WR4 — конкретное поле должно соответствовать полю таблицы, в котором размещен данный текст.

5.6 Определения функций схемы описания представления

5.6.1 Функция acyclic_composite_text

Функция **acyclic_composite_text** проверяет экземпляр объекта **composite_text** с точки зрения его участия в самоопределении. Она возвращает значении TRUE, если экземпляр **composite_text** является ацикличным (не входит в собственное определение), и значение FALSE — в противном случае.

Данная функция изначально проводит проверку набора объектов **collected_text** и возвращает значение FALSE при его наличии. Затем функция создает локальный набор всех экземпляров **composite_text** в собственном наборе объектов **collected_text**. Далее вносит в данный набор все экземпляры **composite_text**, которые ссылаются на примитив **text_string_representation**, использованный объектом **representation_map** из любого примитива **annotation_text**, входящего в набор объектов **collected_text**. Созданный локальный набор, внесенный в заданный набор экземпляров, всегда проверяется функцией. Если общий проверенный набор экземпляров не увеличился, это означает завершение проверки функцией всех возможных вариантов, и она возвращает значение TRUE. В противоположном случае данная функция вызывается повторно для дальнейшей рекурсивной проверки.

EXPRESS-спецификация

```
*)
FUNCTION acyclic_composite_text (start_composite : composite_text;
                                 child_text : SET [1:?] OF
                                 text_or_character) : LOGICAL;
  LOCAL
    i : INTEGER
    local_composite_text  : SET [0:?] OF composite_text;
    local_annotation_text : SET [0:?] OF annotation_text;
    local_children        : SET [0:?] OF text_or_character;
  END_LOCAL;
  local_composite_text := QUERY (child <* child_text |
                          ('PRESENTATION_DEFINITION_SCHEMA.COMPOSITE_TEXT'
                            IN TYPEOF (child)));
  IF (SYZEOF (local_composite_text) > 0 THEN
     REPEAT i := 1 TO HIINDEX (local_composite_text);
        IF (start_composite :=: local_composite_text[i]) THEN
           RETURN (FALSE);
        END_IF;
     END_REPEAT;
    END_IF;
  local_children := child_text;
  IF (SIZEOF (local_composite_text) > 0 THEN
     REPEAT i := 1 TO HIINDEX (local_composite_text);
        local_children :=  local_children +
                           local composite text[i].collected_text;
     END_REPEAT;
  END_IF;
  local_annotation_text := QUERY (child <* child_text |
                           ('PRESENTATION_DEFINITION_SCHEMA.ANNOTATION_TEXT'
                             IN TYPEOF (child)));
  IF (SYZEOF (local_annotation_text) > 0 THEN
     REPEAT i := 1 TO HIINDEX (local_annotation_text);
        local_children :=  local_children +
        QUERY (item <* local_annotation_text[i]\mapped_item.
                           mapping_source.mapped_representation.items |
          SIZEOF (['PRESENTATION_DEFINITION_SCHEMA.ANNOTATION_TEXT'
                   'PRESENTATION_DEFINITION_SCHEMA.COMPOSITE_TEXT']*
                    TYPEOF (item)) > 0);
```

```
      END_REPEAT;
    END_IF;
    IF (local_children :<>: child_text) THEN
      RETURN (acyclic_composite_text (start_composite, local_children));
    ELSE
      RETURN (TRUE);
    END_IF;
END_FUNCTION;
(*
```

Определения аргументов

start_composite — проверяемый примитив **composite_text**. Он служит исходными данными для функции;

child_text — типы **text_or_character**, на которые ссылается объект **start_composite**. Они являются исходными данными для функции. Изначально набор этих типов содержит в качестве элемента только объект **collected_text** из **start_composite**.

5.6.2 Функция acyclic_symbol_representation_relationship

Функция **acyclic_symbol_representation_relationship** определяет, использован ли объект **symbol_representation** в «дереве» объектов **symbol_representation_relationship**, определяющих данный **symbol_representation**. Функция возвращает значение TRUE, если данный объект не использован (функция ациклична), и FALSE- в противном случае.

EXPRESS-спецификация

```
*)
FUNCTION acyclic_symbol_representation_relationship
    (relation       : symbol_representation_relationship;
    children        : SET OF symbol_representation) : BOOLEAN
    LOCAL
      x             : SET OF symbol_representation_relationship;
      local_children    : SET OF symbol_representation;
    END_LOCAL;
    REPEAT i: = 1 TO HIINDEX (children);
      IF relation\representation_relationship.rep_1 :=: children[i] THEN
        RETURN(FALSE);
      END_IF;
    END_REPEAT;
    x := bag_to_set (USEIN (relation\representation_relationship.rep_1
                    'REPRESENTATION_SCHEMA.'+ '
                    'REPRESENTATION_RELATIONSHIP.' + 'REP_2'));
    local_children := children + relation\representation_relationship.rep_1;
    IF (SIZEOF (x) > 0 THEN
      REPEAT i :=1 TO HIINDEX (x);
          IF NOT acyclic_symbol_representation_relationship(x[i],
                                                  local_children) THEN
              RETURN (FALSE);
          END_IF;
      END_REPEAT;
    END_IF;
    RETURN (TRUE);
END_FUNCTION;
(*
```

Определения аргументов

relation — проверяемый объект **symbol_representation_relationship**. Он служит исходными данными для функции;

children — примитивы **symbol_representations**, ссылающиеся на объект **relation**. Они служат исходными данными для функции. Изначально набор этих примитивов содержит в качестве элемента только атрибут **rep_2** объекта **relation**.

5.6.3 Функция field_in_table

Функция **field_in_table** проверяет примитив **table_record_field_representation** объекта **annotation_table_occerrence**; она возвращает значение TRUE, если поле, заданное в них, входит в таблицу, и значение FALSE — в противном случае.

Изначально функция находит примитив **table_representation**, отображающий объект **annotation_table_occerrence**. Затем все примитивы **table_record_representation**, связанные с объектом **table_representation_entity** посредством примитива **symbol_representation_relationship**, или включенные в примитив **table_representation** через объект **mapped_item**. Функция возвращает значение FALSE при отсутствии примитивов **table_record_representation**, связанных с заданным полем посредством примитива **symbol_representation_relationship** или входящих в это поле через объект **mapped_item**, и значение TRUE — в противном случае.

EXPRESS-спецификация

```
*)
FUNCTION field_in_table (field : table_record_field_representation;
                         table : annotation_table_occurrence) : BOOLEAN;
  LOCAL
    table_rep            : table_representation;
    symbol_rep_rel_set   : SET OF symbol_representation_relationship;
    mapped_item_set      : SET OF mapped_item;
    table_record_rep_set   : SET OF table_record_representation := [];
  END_LOCAL;
    table_rep    : table\styled_item.item\mapped_item.mapping_sourse.
      mapped_representation;
    mapped_item_set : = QUERY (item <* table_rep.items |
                        ('REPRESENTATION_SCHEMA.MAPPED_ITEM' IN
                          TYPEOF (item))
                                AND
                         ('RESENTATION_DEFINITION _SCHEMA.'+
                          'TABLE_RECORD_REPRESENTATION' IN
                            TYPEOF (item\mapped_item.mapping_source.
                                    mapped_representation))
                        );

  REPEAT i: = 1 TO HIINDEX (mapped_item_set);
     table_record_rep_set : = table_record_rep_set +
           mapped_item_set[i].mapping_source.mapped_representation;
  END_REPEAT;
    symbol_rep_rel_set : = bag_to_set (USEDIN(table_rep,
                                 'REPRESENTATION_SCHEMA.'+'
                                 'REPRESENTATION_RELATIONSHIP.REP_1'));
  REPEAT i :=1 TO HIINDEX (symbol_rep_rel_set);
    table_record_rep_set : = table_record_rep_set +
            symbol_rep_rel_set[i]\representation_relationship.rep_2;
  END_REPEAT;
  IF SIZEOF(QUERY(table_record_rep <* table_record_rep_set |
          (SIZEOF(QUERY(rep_rel <* USEDIN(table_record_rep,
                          'REPRESENTATION_SCHEMA.'+
                          'REPRESENTATION_RELATIONSHIP.REP_1') |
                        '(PRESENTATION_DEFINITION_SCHEMA.'+
                        'SYMBOL_REPRESENTATION_RELATIONSHIP' IN
                        TYPEOF (rep_rel))
                        AND
                        (rep_rel.rep_2 :=: field)
                        )) > 0)
                        OR
```

```
            (SIZEOF(QUERY(item <* table_record_rep.items |
                        'REPRESENTATION_SCHEMA.MAPPED_ITEM' IN
                        TYPEOF (item))
                                AND
                        (field : = : item\mapped_item.mapping_source.
                                mapped_representation)
                        )) > 0)
                )) = 0 THEN
        RETURN (FALSE);
    END_IF;
    RETURN (TRUE);
END_FUNCTION;
(*
```

Определения аргументов

field — полем записи в таблицу должен быть объект **table**. Данный объект является исходным для функции;

table — аннотируемая таблица, содержащая объект **field.** Данный объект является исходным для функции.

EXPRESS-спецификация

```
*)
END_SCHEMA; - - presentation_definition_schema
(*
```

6 Вид представления

Следующее описание на языке EXPRESS открывает **presentation_appearance_schema** и определяет необходимые внешние ссылки.

EXPRESS-спецификация

```
*)
SCHEMA presentation_appearance_schema;
REFERENCE FROM external_reference_schema
        (externally_defined_item,
         pre_defined_item);
REFERENCE FROM geometry_schema
        (axis2_placement,
         cartesian_point,
         curve,
         geometric_representation_item,
         point,
         vector);
REFERENCE FROM group_schema
        (group);
REFERENCE FROM measure_schema
        (descri ptive_measure,
         length_measure,
         measure_with_unit,
         plane_angle_measure,
         positive_length_measure,
         ratio_measure,
         positive_ratio_measure);
REFERENCE FROM presentation_organization_schema
        (area_dependent_annotation_representation,
         presentation_area,
         presentation_layer_assignment,
         presentation_layer_usage,
         presentation_representation,
```

```
        presentation_set,
        presentation_view,
        product_data_representation_view,
        view_dependent_annotation_representation);
REFERENCE FROM presentation_definition_schema
      (annotation_curve_occurrence,
        annotation_fill_area,
        annotation_symbol_occurrence,
        annotation_text_with_delineation,
        symbol_representation_with_blanking_box);
REFERENCE FROM presentation_resource_schema
      (character_glyph_symbol_outline,
        character_glyph_symbol_stroke,
        colour);
REFERENCE FROM representation_schema
      (mapped_item,
        representation,
        representation_item,
        representation_map,
        using_representation);
REFERENCE FROM support_resource_schema
      (label,
        bag_to_set);
(*
```

Примечания

1 Схемы, на которые выше даны ссылки, можно найти в следующих стандартах:

external_reference_schema	ГОСТ Р ИСО 10303-41;
geometry_schema	ИСО 10303-42;
management_resource_schema	ГОСТ Р ИСО 10303-41;
measure_schema	ГОСТ Р ИСО 10303-41;
presentation_organization_schema	раздел 4 настоящего стандарта;
presentation_definition_schema	раздел 5 настоящего стандарта;
presentation_resource_schema	раздел 7 настоящего стандарта;
representation_schema	ГОСТ Р ИСО 10303-43;
support_resource_schema	ГОСТ Р ИСО 10303-41.

2 Графическое представление определяемой схемы приведено в приложении Е с использованием нотации языка EXPRESS-G.

6.1 Введение

Целью **presentation_appearance_schema** является определение соответствующего графического вида представляемого изображения. Данная схема также определяет механизм, позволяющий увязать атрибуты данного вида представления с объектами **annotation_occurrence** в контексте объекта **presentation_representation**. Таким образом контекст **presentation_representation** может быть использован для определения видов различных элементов изображения.

6.2 Фундаментальные понятия и допущения

6.2.1 Задание стиля представления

Схема **presentation_appearance_schema** позволяет увязать виды атрибутов с выбранными аннотируемыми примитивами и элементами формы изделия. В соответствии с настоящим стандартом стиль представления может быть задан для любого объекта **representation_item**. Стиль определяют посредством конкретизации объекта **styled_item**, ссылающегося на объект **representation_item**, и соответствующего объекта **presentation_style_assignment**. Последний объект представляет собой набор (коллекцию) различных стилей представления (например, стиль точки, кривой или текста). Стилизацию нестилизованного объекта **representation_item** проводят посредством определения нового объекта **representation_item** с заданным стилем представления. Объект **presentation_style_assignment**, указанный в объекте **styled_item**, определяет вид ссылочного объекта **representation_item**, а также вид всех объектов **representation_item**, прямо или косвенно ссылающихся на данный элемент. При этом могут быть указаны только те объекты **representation_item**, стиль которых еще не задан. Это означает, что

стилизация ранее стилизованных объектов **representation_item** не проводится, а стилизуются только нестилизованные объекты **representation_item**, влияющие на вид представления данного элемента в целом. Представлены могут быть только стилизованные объекты **representation_item**. Их фактически представляют в зависимости от ряда других факторов, например уровня видимости и объекта **invisibility**, наличия скрытых линий или поверхностей, а также «обрезки» объекта. В настоящем стандарте не приведены рекомендации по устранению конфликтов при определении стиля. Конфликт при описании стиля может, например, возникнуть, когда объект **representation_item** используется в нескольких объектах **styled_items**.

Объект **presentation_style_assignment** используют для определения стиля объекта **representation_item** независимо от контекста его представления. Подтип данного объекта **presentation_style_by_context** позволяет задать стиль конкретного представления контекста. Контекстом представления может быть любой объект **presentation_set_representation** или **representation_item**. В последнем случае стиль может быть задан для элемента в целом, а для различных его частей заданы другие стили.

Заданный стиль может быть переопределен через объект **over_riding_styled_item**. Подтип данного объекта (**over_riding_styled_item**) **context_dependent_over_riding_styled_item** позволяет переопределить стиль конкретного представления контекста.

6.2.2 Типы стилей представления

Для каждого аннотируемого примитива, заданного в схеме **presentation_definition_schema**, существует конкретная группа стилей. Эти группы определяют стили точек, кривых, заполняемых областей, поверхностей, текстов и символов. В настоящем стандарте не заданы ограничения по определению стилей элементов поверхности. Это позволяет, например, задать стиль описания поверхности для точки. Однако только соответствующие стили определяют вид элемента формы изделия, то есть стили поверхностей связаны только с поверхностями, стили областей заполнения и кривых — с кривыми, стили точек — только с точками, кривыми и поверхностями. Для общих элементов формы изделия (например, пространственной модели) при определении видов элементов используют прикладные протоколы. Однако присвоение стилей аннотируемым примитивам несколько более консервативно. Более подробно это описано в разделе 5 для объекта **annotation_occurrence**.

Стили представлений могут быть определены с использованием ресурсов данной схемы, заданы извне или предопределены в прикладных протоколах. Стили представлений, определенные в настоящей схеме, охватывают ряд нижеописанных объектов.

Стили точек задают ресурсы для визуальных отображений точек. Данные стили позволяют определить маркировочные символы, размеры и цвета, используемые при представлении точек. Стили точек могут быть заданы объектом **point_style**.

Стили кривых задают ресурсы для визуальных отображений кривых. Данные стили позволяют определить соответствующие шрифты, толщину кривых, виды концов и углов кривых, цвета или шаблоны для заполнения видимых сегментов кривых. Шрифт кривой определяет необходимость ее вычерчивания с использованием сплошных, пунктирных или штрих-пунктирных линий. Рассматриваемая схема позволяет задать шаблоны шрифтов кривых и использовать заданные извне или заранее предопределенные шрифты кривых. При определении шрифтов кривых используют объекты **externally_defined_curve_font, pre_defined_curve_font, curve_style_font, curve_style_font_patten** и **curve_style_font_and_scaling**. Толщина кривой может быть определена на основе измерения или задана заранее. Концы и углы кривой могут быть заданы окружностями или прямоугольниками и сжаты или расширены для концов кривой. Объектами, входящими в данную структуру, являются **curve_style_with_ends_and_corners** и **curve_style_with_extension**. Объект **curve_style_ wide** позволяет определить стиль области заполнения, используемый для заполнения видимых сегментов кривой. Это позволяет задать соответствующие требования для вычерчивания кривых. Объект **curve_style_rendering** определяет изображение кривых на поверхности, а также определяет стили поверхностей.

Стили заполняемых областей обеспечивают ресурсы, определяющие виды визуального представления для аннотируемых заполняемых областей. Эти стили позволяют определить основной цвет, штриховки или виды ячеистой структуры заполняемых областей. Объект **fill_area_style_colour** может быть использован для задания основного цвета при представлении заполняемой области. Штриховка данных областей может быть задана в виде набора параллельных линий с разными углами наклона. Параметры штриховки могут быть заданы заранее или определены извне. Штриховка может быть

задана объектами **fill_area_style_hatching**, **pre_defined_hatch_style**, **externally_defined_hatch_style** и **one_direction_repeat_factor**. Ячеестые структуры заполняемых областей могут быть описаны кривыми, цветовыми пятнами или символами. Параметры ячеек могут быть заданы заранее или определены извне. Ячейки заполняют соответствующую область двумерно по направлениям, заданным соответствующими векторами. Ячеестая структура может быть задана объектами **pre_defined_tile_style**, **externally_defined_tile_style**, **fill_area_style_tiles**, **fill_area_style_tile_curve_with_style**, **fill_area_style_tile_coloured_region**, **fill_area_style_tile_symbol_with_style**, **pre_defined_tile externally_defined_tile** и **two_direction_repeat_factor.**

Стили поверхностей обеспечивают ресурсы, определяющие виды визуального представления для аннотируемых поверхностей. Отдельные стили поверхности могут быть использованы для каждой ее стороны отдельно или обеих сторон. Конкретный стиль поверхности может определять описание одной или обеих ее сторон. Объектами, определяющими стиль сторон поверхности, являются **surface_style_usage** и **surface_side_style**. Стиль описания стороны поверхности может быть представлен в виде комбинации стилей заполняемых областей, границ, силуэтов, сегментов кривых, решеток, параметров линий или отображений (визуализаций).

Стиль заполняемой области определяет основной цвет, штриховку или ячеестую структуру для описания стороны поверхности.

Стиль границы определяет стиль кривой или метод отображения и свойства для представления ограниченных кривых на поверхности. Если стиль границ не задан, ограниченные кривые не могут быть представлены.

Стиль силуэта определяет стиль кривой или метод отображения и свойства для представления тонируемых (силуэтных) кривых на поверхности. Если стиль силуэта не задан, тонируемые кривые не могут быть представлены.

Стиль сегментации кривой определяет стиль кривой или метод отображения и свойства для представления сегментированной кривой на поверхности. Данный стиль связан только с сегментированной кривой на поверхности. Если стиль этой кривой не задан, данные кривые не могут быть представлены.

Стиль опорного «скелета» определяет стиль кривой или метод отображения и свойства для представления сетки опорных точек для описания поверхности. Данный стиль относится только к поверхностям, задаваемым сеткой опорных точек. Если стиль опорного «скелета» не задан, данный «скелет» не может быть представлен.

Стиль строки параметров определяет стиль кривой или метод отображения и свойства для представления строки параметров на поверхности. Данный стиль должен определить число строк параметров в каждом направлении. Если стиль строки параметров не задан, данные строки не могут быть представлены.

Объектами, определяющими стили описания поверхностей, являются: **surface_fill_area, surface_style_boundary, surface_style_silhouette, surface_style_segmentation_curve, surface_style_control_grid** и **surface_style_parameter_line**.

Объекты **curve_style_rendering** и **surface_rendering_properties** определяют метод отображения и свойства для представления кривой на поверхности. Стиль отображения задает метод, используемый для отображения поверхности. Определение данного стиля позволяет использовать его при вычислениях отражающей способности поверхности для ее представления. При определении данного стиля должен быть задан стиль отображаемой поверхности. Кроме того, в данном стиле могут быть заданы дополнительные свойства отображения поверхности, включая коэффициенты ее прозрачности, окружения, рассеивания и отражения. Объектами, определяющими стили описания поверхностей, являются: **surface_style_rendering**, **surface_style_rendering_with_properties, surface_style_transparent, surface_style_reflectance_ambient, surface_style_reflectance_ambient_diffuse** и **surface_style_reflectance_ambient_diffuse_specular.** Помимо вышеперечисленных, стили представления поверхностей могут быть заданы заранее.

Стили текстов обеспечивают ресурсы, определяющие виды визуального представления аннотируемых текстов. Правила выравнивания текста, характеристики параллелограмма («ящика»), типы пробелов и ось зеркального отображения влияют на вид текста в целом, а стиль глифа символа — на вид отдельных знаков или символов, образующих данный текст. Выравнивание текста задает его расположение в отведенной области. Дополнительные типы размещения текста могут быть определены в прикладных протоколах. «Ящиком» символа является параллелограмм, задающий ширину,

высоту, угол наклона и поворота символа. Пробелы текста определяют расстояние между смежными «ящиками» символов в строке в дополнение к символу пробела, заданному в описании шрифта. При зеркальном отображении текста задают ось его зеркального отображения. Зеркальное отображение выполняют в последнюю очередь (после реализации остальных стилей текста). Объектами, определяющими стили текста, являются: **text_style, ext_style_with_spacing, text_style_with_mirror, text_style_with_justification** и **text_style_with_box_characteristics**. Стиль глифа символа может определять начертание, шаг или только цвет текста. Стилем начертания является стиль кривой, используемый для представления начертания символа (его глифа). Дополнительно может быть задан стиль области заполнения, определяющий «заливку» очерченного символа. Стиль начертания влияет только на глифы символов, определяемые их очертанием. Стиль шага влияет только на глифы символов, заданные шагами. Для предварительно или внешне заданных шрифтов должен быть определен только их цвет.Объектами,определяющими стили глифов символов, являются: **character_glyph_style_stroke, character_glyph_style_outline, character_glyph_style_outline_with_characteristics** и **text_style_for_-defined_font**.

Стили знаков обеспечивают ресурсы, определяющие виды визуального представления аннотируемых знаков. Стиль знака состоит из произвольного набора (коллекции) стилей точек, кривых, заполняемых областей, поверхностей и текстов. Объектами, определяющими стили знаков, являются: **symbol_style, symbol_element_style** и **symbol_colour**.

6.2.3 Допуски аппроксимации

Допуски аппроксимации определяют разрешенные отклонения между формами представленных элементов и их математически заданными позициями и формами. Данная аппроксимация применяется к кривым и поверхностям, и может быть задана в параметрическом пространстве, пространстве формы изделия или области представления. Пространство с заданными допусками зависит от используемого метода аппроксимации. При методах хордовой аппроксимации (по длине и отклонению) допуск может быть задан в пространстве формы изделия или области представления. При параметрической аппроксимации допуск может быть задан в параметрическом пространстве.

6.2.4 Сокрытие и невидимость

В рассматриваемой схеме предусмотрены ресурсы для определения видов элементов, расположенных друг за другом и перекрывающихся в двумерном пространстве. Для представления подобных элементов используют иерархию, описанную в разделе 4, на основе аннотируемых областей заполнения, текстов с выделениями, глифов символов и знаков. При этом объект **occlusion_precedence** может быть задан для любых двух подобных элементов.

Дополнительно рассматриваемая схема задает ресурсы для определения невидимости объектов **styled_item** или уровней.

6.3 Определение типов схемы вида представления

6.3.1 Тип style_context_select

Тип **style_context_select** определяет выбор одного из объектов, могущих задавать контекст для назначения или переопределения стиля представления.

EXPRESS-спецификация

```
*)
TYPE style_context_select = SELECT
    (group,
     presentation_layer_assignment,
     representation,
     representation_item,
     presentation_set);
END_TYPE;
(*
```

6.3.2 Тип presentation_style_select

Тип **presentation_style_select** используется объектом **presentation_style_assignment** для увязки стиля с объектом **representation_item**. Для каждого стилизуемого объекта **representation_item** должен быть задан свой стиль.

EXPRESS-спецификация

```
*)
TYPE presentation_style_select = SELECT
```

```
    (pre_defined_presentation_style,
     point_style,
     curve_style,
     surface_style_usage,
     symbol_style,
     fill_area_style,
     text_style,
     approximation_tolerance,
     externally_defined_style,
     null_style);
END_TYPE;
(*
```

6.3.3 Тип null_style

Тип **null_style** указывает на отсутствие конкретного стиля, непосредственно присвоенного конкретному для его представления. Стиль(и), используемый(е) при представлении элемента, задают непосредственно в определении этого элемента. При отсутствии указания стиля в определении элемента, данный элемент не может быть представлен.

Пример 9 — В определении знака указаны два объекта **annotation_curve_occurrence**, стиль представления одного из которых — красный, а другого — синий. Если экземпляр символа задан типом **null_style**, тогда данный экземпляр может быть представлен с использованием цветов, указанных в его определении.

EXPRESS-спецификация

```
*)
TYPE null_style = ENUMERATION OF
    (null);
END_TYPE;
(*
```

Определение элемента перечисления

null — объект **representation_item**, при представлении которого должен быть использован(ы) стиль(и), заданный(е) в его определении.

6.3.4 Тип marker_select

Тип **marker_select** определяет выбор конкретного или ранее заданного маркера для представления точек.

EXPRESS-спецификация

```
*)
TYPE marker_select = SELECT
    (marker_type,
     pre_defined_marker);
END_TYPE;
(*
```

6.3.5 Тип marker_type

Тип **marker_type** определяет форму представления точек.

EXPRESS-спецификация

```
*)
TYPE marker_type = ENUMERATION OF
    (dot,
     x,
     plus,
     asterisk,
     ring,
     square,
     triangle);
END_TYPE;
(*
```

Определения элементов перечисления

dot — точки, представленные знаком «точка» (.);
x — точки, представленные знаком «диагональный крест» (Ч);
plus — точки, представленные знаком «плюс» (+);
asterisk — точки, представленные знаком «звездочка» (*);
ring — точки, представленные знаком «кружок» (○);
square — точки, представленные знаком «квадрат» (□);
triangle — точки, представленные знаком «треугольник» (Δ).

6.3.6 Тип size_select

Тип **size_select** используют для определения размеров маркировочных знаков или толщины кривых.

EXPRESS-спецификация

```
*)
TYPE size_select = SELECT
    (positive_length_measure,
     measure_with_unit,
     descriptive_measure,
     pre_defined_size);
END_TYPE;
(*
```

6.3.7 Тип curve_font_or_scaled_curve_font_select

Тип **curve_font_or_scaled_curve_font_select** определяет выбор одного из объектов (типов) **curve_style_font_select** или **curve_style_font_and_scaling**.

EXPRESS-спецификация

```
*)
TYPE curve_font_or_scaled_curve_font_select = SELECT
    (curve_style_font_select,
     curve_style_font_and_scaling);
END_TYPE;
(*
```

6.3.8 Тип curve_style_font_selest

Тип **curve_style_font_select** определяет выбор одного из объектов **curve_style_font**, **pre_defined_curve_font** или **externally_defined_curve_font**. Данный тип используют для задания немасштабированного шрифта при представлении кривой.

EXPRESS-спецификация

```
*)
TYPE curve_style_font_select = SELECT
    (curve_style_font,
     pre_defined_curve_font,
     externally_defined_curve_font);
END_TYPE;
(*
```

6.3.9 Тип squared_or_rounded

Тип **squared_or_rounded** задает виды кривых с точки зрения их угловых и концевых положений.

Примечание — См. рисунок 16.

EXPRESS-спецификация

```
*)
TYPE squared_or_rounded = ENUMERATION OF
    (squared,
     rounded);
END_TYPE;
(*
```

Определения элементов перечисления

squared — кривая, позиционированная с точки зрения ее угловых и концевых положений;

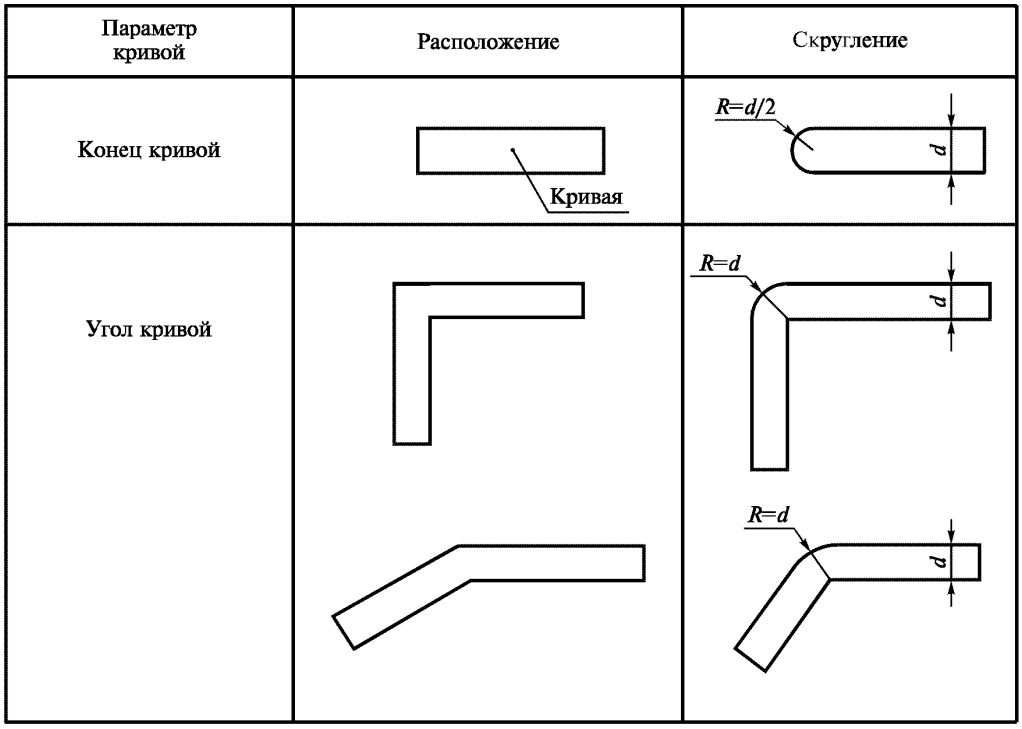
rounded — полудуга диаметром, равным толщине (ширине) кривой, вычерченная на концах и углах кривой.

6.3.10 Тип fill_style_select

Тип **fill_style_select** определяет выбор различных стилей заполнения.

EXPRESS-спецификация

```
*)
TYPE fill_style_select = SELECT
    (fill_area_style_colour,
     pre_defined_tile_style,
     externally_defined_tile_style,
     fill_area_style_tiles,
     pre_defined_hatch_style,
     externally_defined_hatch_style,
     fill_area_style_hatching);
END_TYPE;
(*
```



d — толщина (ширина) кривой

Рисунок 16 — Спрямление или скругление

6.3.11 Тип fill_area_style_tile_shape_select

Тип **fill_area_style_tile_shape_select** используют для определения объектов **fill_area_style_tiles**. Данный тип обеспечивает выбор различных форм и исходных данных для выбора заполняющих ячеек.

EXPRESS-спецификация

```
*)
TYPE fill_area_style_tile_shape_select = SELECT
    (fill_area_style_tile_curve_with_style,
     fill_area_style_tile_coloured_region,
     fill_area_style_tile_symbol_with_style,
     pre_defined_tile,
     externally_defined_tile);
```

```
END_TYPE;
(*
```

6.3.12 Тип curve_or_annotation_curve_occurrence

Тип **curve_or_annotation_curve_occurrence** описывает объекты **curve** или **annotation_curve_occurrence**, определяющие границу объекта **fill_area_style_tile_coloured_region**.

EXPRESS-спецификация

```
*)
TYPE curve_or_annotation_curve_occurrence = SELECT
    (curve,
     annotation_curve_occurrence);
END_TYPE;
(*
```

6.3.13 Тип surface_side

Тип **surface_side** используется объектом **surface_style_usage** при указании сторон поверхности, для которых применяется конкретный стиль.

EXPRESS-спецификация

```
*)
TYPE surface_side = ENUMERATION OF
    (positive,
     negative
     both);
END_TYPE;
(*
```

Определения элементов перечисления

positive — сторона поверхности, соответствующая направлению нормали к ней;

negative — сторона поверхности, противоположная направлению нормали к ней;

both — обе стороны поверхности.

6.3.14 Тип surface_side_style_select

Тип **surface_side_style_select** определяет выбор одного из объектов **surface_side_style** или **pre_-defined_surface_side_style**.

EXPRESS-спецификация

```
*)
TYPE surface_side_style_select = SELECT
    (surface_side_style,
     pre_defined_surface_side_style);
END_TYPE;
(*
```

6.3.15 Тип surface_style_element_select

Тип **surface_style_element_select** определяет выбор различных стилей описания поверхности, используемых при представлении сторон этой поверхности.

EXPRESS-спецификация

```
*)
TYPE surface_style_element_select = SELECT
    (surface_style_fill_area,
     surface_style_boundary,
     surface_style_silhouette,
     surface_style_segmentation_curve,
     surface_style_control_grid,
     surface_style_parameter_line,
     surface_style_rendering);
END_TYPE;
(*
```

6.3.16 Тип curve_or_render

Тип **curve_or_render** используют для описания нескольких стилей поверхности при задании видов кривых на ней.

EXPRESS-спецификация
*)
```
TYPE curve_or_render = SELECT
    (curve_style,
     curve_style_rendering);
END_TYPE;
```
(*

6.3.17 Тип shading_curve_method

Тип **shading_curve_method** определяет метод, подлежащий использованию при затенении кривых. Затенение кривых (посредством соответствующих полилиний) связано с колористическим выделением определенных сегментов в соответствии с цветовой гаммой отдельных точек кривой.

EXPRESS-спецификация
*)
```
TYPE shading_curve_method = ENUMERATION OF
    (constant_colour,
     linear_colour);
END_TYPE;
```
(*

Определения элементов перечисления

constant_colour — если для кривой в форме таблицы задана цветовая гамма и имеется *i*-й сегмент, то его колористика должна соответствовать таблице и задаваться сменой цветов на каждой границе сегментов. При этом каждый *i*-й сегмент должен быть оттенен цветом, заданным для него;

linear_colour — если для кривой в форме таблицы задана цветовая гамма и имеется *i*-й сегмент, то его колористика должна соответствовать таблице и задаваться сменой цветов на каждой границе сегментов. При этом каждый *i*-й сегмент должен быть оттенен цветами, заданными для каждой границы сегмента.

6.3.18 Тип direction_count_select

Тип **direction_count_select** определяет выбор одного из объектов **u_direction_count** или **v_direction_count**, используемых для представления параметризованных линий на поверхности.

EXPRESS-спецификация
*)
```
TYPE direction_count_select = SELECT
    (u_direction_count,
     v_direction_count);
END_TYPE;
```
(*

6.3.19 Тип u_direction_count

Тип **u_direction_count** является положительным целым числом, указывающим количество параметризованных кривых в направлении *u* параметрической поверхности.

EXPRESS-спецификация
*)
```
TYPE u_direction_count = INTEGER
WHERE
    WR1:SELF > 1;
END_TYPE;
```
(*

Формальное утверждение

WR1 — номер направления *u* должен быть больше единицы.

6.3.20 Тип v_direction_count

Тип **v_direction_count** является положительным целым числом, указывающим количество параметризованных кривых в направлении *v* параметрической поверхности.

EXPRESS-спецификация
*)
```
TYPE v_direction_count = INTEGER
WHERE
```

```
    WR1:SELF > 1;
END_TYPE;
(*
```

Формальное утверждение

WR1 — номер направления *v* должен быть больше единицы.

6.3.21 Тип shading_surface_method

Тип **shading_surface_method** определяет метод затенения поверхностей.

Примечания

1 Описания различных типов затенения поверхности в одних случаях связаны с методами интерполирования результатов ее освещения, а в других — с вычислением ее отражающей способности. В связи с этим метод затенения может быть реализован посредством выбора конвейерной обработки графики в системе изображения с использованием соответствующей интерполяции.

2 В результате использования методов затенения должны быть получены эффекты, соответствующие нижеуказанным элементам перечисления. Необходимые эффекты трудно получить, если силуэт поверхности является самопересекающимся или пересекается с другим силуэтом той же поверхности. В этом случае описание данного эффекта связано с конкретной реализацией.

3 Соответствующие методы затенения реализуются посредством PHIGS PLUS (см. приложения F и D).

EXPRESS-спецификация

```
*)
TYPE shading_surface_method = ENUMERATION OF
    (constant_shading,
     colour_shading,
     dot_shading,
     normal_shading);
END_TYPE;
(*
```

Определения элементов перечисления

constant_shading — вычисленная отражающая способность каждой фасеты (грани) аппроксимируемой поверхности для определения единого отраженного цвета данной фасеты. Выбор опорной точки на фасете при выполнении этого вычисления определяют в конкретной реализации. Цвет, используемый при вычислении отражающей способности, задают посредством атрибута **surface_colour** соответствующего объекта **surface_style_rendering**;

colour_shading — вычисленная отражающая способность каждой вершины (узла) фасеты аппроксимируемой формы изделия с использованием атрибута **surface_colour** и нормалей к поверхности в каждой вершине фасеты;

dot_shading — любые элементы изделий, используемые в формуле отражающей способности, определяемые нормалями к соответствующим поверхностям в виде набора их местоположений на конкретной поверхности. Эти элементы интерполируют линейно в местах их пересечения с данной поверхностью. Вычисление отражающей способности выполняют для каждой интерполируемой позиции поверхности для указания ее цвета на основе заданных элементов изделий и значений атрибута **surface_colour** соответствующего объекта **surface_style_rendering**;

normal_shading — нормали, интерполированные линейно по отношению к местам их пересечения с данной поверхностью. Вычисление отражающей способности выполняют для каждой интерполируемой позиции поверхности для указания ее цвета на основе заданных нормалей и значений атрибута **surface_colour** соответствующего объекта **surface_style_rendering**.

Если атрибут **surface_colour** задают через объект **colour_specification**, выбор соответствующих цветов должен быть выполнен в рамках цветовой модели, определенной в этом объекте. В противоположном случае данная интерполяция может быть выполнена на основе произвольно выбранной модели.

Примечания

4 Результат выбора цвета зависит от колористики цветовой модели, в рамках которой выполнена соответствующая интерполяция.

5 Примерами цветовых моделей являются RGB, HSV и HLS.

6 Более подробная информация о цветовых моделях и выборе соответствующих цветов приведена в [7].

6.3.22 Тип rendering_properties_select

Тип **rendering_properties_select** определяет выбор одного из свойств поверхности (отражение или прозрачность).

EXPRESS-спецификация

```
*)
TYPE rendering_properties_select = SELECT
    (surface_style_reflectance_ambient,
     surface_style_transparent);
END_TYPE;
(*
```

6.3.23 Тип character_style_select

Тип **character_style_select** определяет выбор одного из объектов **character_glyph_style_stroke**, **character_glyph_style_outline** или **text_style_for_defined_font**.

EXPRESS-спецификация

```
*)
TYPE character_style_select = SELECT
    (character_glyph_style_stroke,
     character_glyph_style_outline,
     text_style_for_defined_font);
END_TYPE;
(*
```

6.3.24 Тип text_justification

Тип **text_justification** обеспечивает управление выравниванием текста.

Примечание — В прикладных протоколах должны быть заданы соответствующие значения **text_justification** и пояснен их смысл.

EXPRESS-спецификация

```
*)
TYPE text_justification = lable;
END_TYPE;
(*
```

6.3.25 Тип box_characteristic_select

Тип **box_characteristic_select** определяет выбор одного из типов **box_height, box_width**, **box_slant_angle** и **box_rotate_angle**.

EXPRESS-спецификация

```
*)
TYPE box_characteristic_select = SELECT
    (box_height,
     box_width,
     box_slant_angle,
     box_rotate_angle);
END_TYPE;
(*
```

6.3.26 Тип box_height

Тип **box_height** задает максимальный масштабируемый размер глифа символа.

EXPRESS-спецификация

```
*)
TYPE box_height = positive_ratio_measure;
END_TYPE;
(*
```

6.3.27 Тип box_width

Тип **box_width** определяет коэффициент масштабирования ширины в определении глифа символа.

EXPRESS-спецификация

```
*)
TYPE box_width = positive_ratio_measure;
END_TYPE;
(*
```

6.3.28 Тип box_slant_angle

Тип **box_slant_angle** определяет угол наклона «ящика» глифа символа, представляемого в виде параллелограмма; данный угол определяет наклон вертикальной границы «ячейки» символа к перпендикуляру к нижней границе данной «ячейки».

Примечание — Пояснение **box_slant_angle** приведено на рисунке 17.

EXPRESS-спецификация

```
*)
TYPE box_slant_angle = plane_angle_measure;
END_TYPE;
(*
```

Неформальное утверждение

IP1: тип **box_slant_angle** должен иметь значение от 0° до 90°.

6.3.29 Тип box_rotate_angle

Тип **box_rotate_angle** задает угол поворота «ячейки» глифа символа относительно горизонтального положения текстовой строки; данный угол определяет наклон вертикальной границы глифа к перпендикуляру к горизонтали строки.

Примечание — Пояснение **box_rotate_angle** приведено на рисунке 17.

EXPRESS-спецификация

```
*)
TYPE box_rotate_angle = plane_angle_measure;
END_TYPE;
(*
```

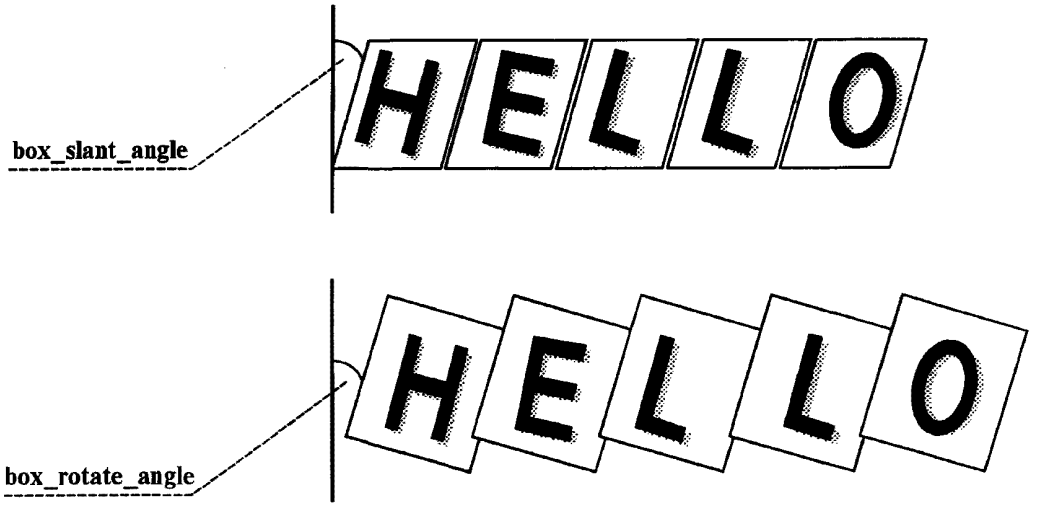


Рисунок 17 — Наклон и поворот ячейки

6.3.30 Тип character_spacing_select

Тип **character_spacing_select** определяет выбор одного из объектов **length_measure**, **ratio_measure**, **measure_with_unit**, **descriptive_measure** и **pre_defined_character_spacing**. Данный тип используют для определения пробелов между смежными символами в текстовой строке.

EXPRESS-спецификация

```
*)
TYPE character_spacing_select = SELECT
    (length_measure,
     ratio_measure,
```

```
    measure_with_unit,
    descriptive_measure,
    pre_defined_character_spacing);
END_TYPE;
(*
```

6.3.31 Тип symbol_style_select

Тип **symbol_style_select** определяет выбор одного из объектов **symbol_element_style** или **symbol_colour**. Данный тип используют для определения стиля элементов символа или цвета данного символа в целом.

EXPRESS-спецификация

```
*)
TYPE symbol_style_select = SELECT
    (symbol_element_style,
    symbol_colour);
END_TYPE;
(*
```

6.3.32 Тип tolerance_select

Тип **tolerance_select** используется объектом **approximation_tolerance** для выбора одного из объектов **approximation_tolerance_deviation** или **approximation_tolerance_parameter**.

EXPRESS-спецификация

```
*)
TYPE tolerance_select = SELECT
    (approximation_tolerance_deviation,
    approximation_tolerance_parameter);
END_TYPE;
(*
```

6.3.33 Тип approximation_method

Тип **approximation_method** используют для перечисления двух возможных методов создания ячеестых структур кривых и поверхностей посредством линейных сегментов (линейной аппроксимации) или ячеек из плоских многоугольников.

EXPRESS-спецификация

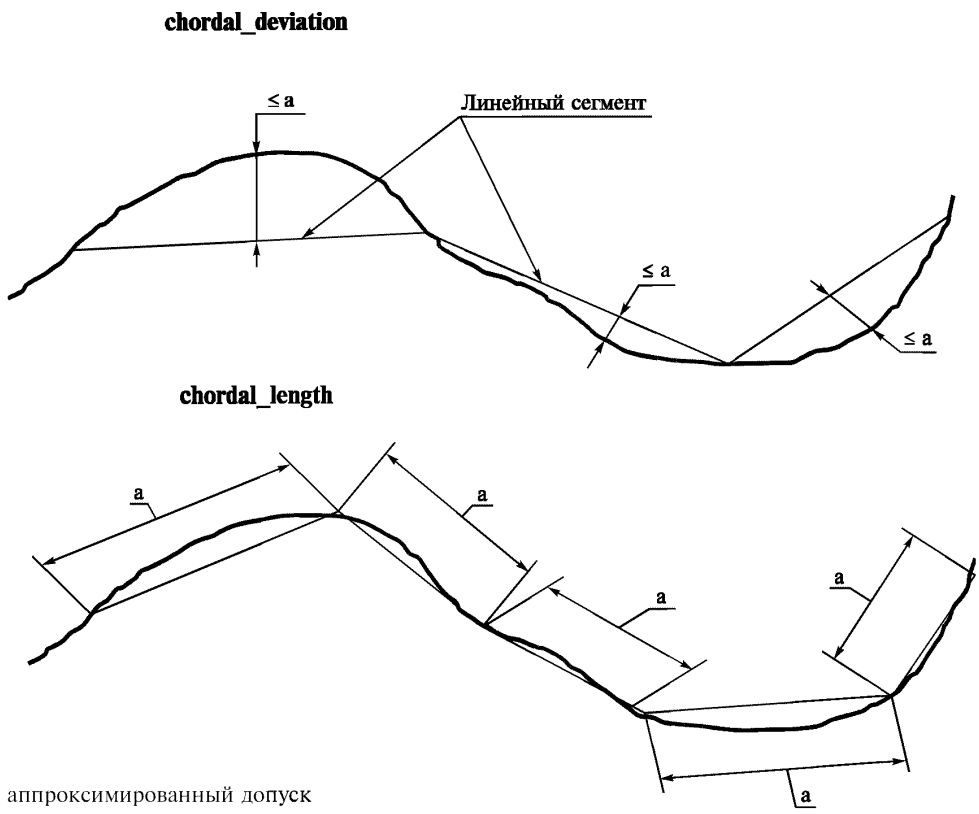
```
*)
TYPE approximation_method = ENUMERATION OF
    (chordal_deviation,
    chordal_length);
END_TYPE;
(*
```

Определения элементов перечисления

chordal_deviation — кривые, аппроксимированные так, что расстояние между данной кривой и аппроксимирующим ее линейным сегментом не превышает установленного значения. Поверхности, аппроксимированные так, что расстояние между данной поверхностью и аппроксимирующими ее многоугольниками не превышает установленного значения. Данные расстояния измеряют в направлении нормали, возведенной к данному сегменту или многоугольнику. Данный тип измеряют в единицах пространства формы изделия или единицах объекта **presentation_area**, заданных через объект **approximation_tolerance_deviation**;

chordal_length — кривые, аппроксимированные так, что соответствующие линейные сегменты имеют одинаковую длину. Поверхности, аппроксимированные так, что грани соответствующих плоских многоугольников имеют одинаковую длину. Данный тип измеряют в единицах пространства формы изделия или единицах объекта **presentation_area**, заданных через объект **approximation_tolerance_deviation**.

Примечание — На рисунке 18 проиллюстрированы элементы **chordal_deviation** и **chordal_length**.



a — аппроксимированный допуск

Рисунок 18 — Отклонение и длина хорды

6.3.34 Тип tolerance_deviation_select

Тип **tolerance_deviation_select** используется объектом **approximation_tolerance_deviation** для выбора одного из типов **curve_tolerance_deviation** или **surface_tolerance_deviation**.

EXPRESS-спецификация

```
*)
TYPE tolerance_deviation_select = SELECT
    (curve_tolerance_deviation,
     surface_tolerance_deviation);
END_TYPE;
(*
```

6.3.35 Тип curve_tolerance_deviation

Тип **curve_tolerance_deviation** определяет аппроксимированный допуск для кривой через значение его отклонения.

EXPRESS-спецификация

```
*)
TYPE curve_tolerance_deviation = positive_length_measure;
END_TYPE;
(*
```

6.3.36 Тип surface_tolerance_deviation

Тип **surface_tolerance_deviation** определяет аппроксимированный допуск для поверхности через значение его отклонения.

EXPRESS-спецификация

```
*)
TYPE surface_tolerance_deviation = positive_length_measure;
```

```
END_TYPE;
(*
```

6.3.37 Тип product_or_presentation_space

Тип **product_or_presentation_space** используется объектом **approximation_tolerance_deviation** для определения пространства, в котором заданы значения допусков.

EXPRESS-спецификация

```
*)
TYPE product_or_presentation_space = ENUMERATION OF
    (product_shape_space,
     presentation_area_space);
END_TYPE;
(*
```

Определения элементов перечисления

product_shape_space — заданное пространство для представления элемента формы изделия;

presentation_area_space — координатное пространство, в котором размещен объект **presentation_area**.

6.3.38 Тип tolerance_parameter_select

Тип **tolerance_parameter_select** используется объектом **approximation_tolerance_parameter** для выбора одного из объектов **curve_tolerance_parameter** или **surface_tolerance_parameter**.

EXPRESS-спецификация

```
*)
TYPE tolerance_parameter_select = SELECT
    (curve_tolerance_parameter,
     surface_tolerance_parameter);
END_TYPE;
(*
```

6.3.39 Тип curve_tolerance_parameter

Тип **curve_tolerance_parameter** задает аппроксимированный допуск для кривых в единицах параметризованного пространства.

EXPRESS-спецификация

```
*)
TYPE curve_tolerance_parameter = REAL;
END_TYPE;
(*
```

6.3.40 Тип surface_tolerance_parameter

Тип **surface_tolerance_parameter** задает аппроксимированный допуск для поверхностей в единицах параметризованного пространства.

EXPRESS-спецификация

```
*)
TYPE surface_tolerance_parameter = REAL;
END_TYPE;
(*
```

6.3.41 Тип hiding_or_blanking_select

Тип **hidingor_blanking_select** определяет объекты, которые могут быть скрыты или пропущены посредством объектов, входящих в данное представление.

EXPRESS-спецификация

```
*)
TYPE hiding_or_blanking_select = SELECT
    (presentation_area,
     presentation_view,
     product_data_representation_view,
     annotation_fill_area,
     area_dependent_annotation_representation,
     view_dependent_annotation_representation,
     annotation_text_with_delineation,
     character_glyph_symbol_stroke,
```

```
    character_glyph_symbol_outline,
    symbol_representation_with_blanking_box);
END_TYPE;
(*
```

6.3.42 Тип invisibility_context

Тип **invisibility_context** задает контекст, определяющий возможные невидимые элементы изображения.

EXPRESS-спецификация

```
*)
TYPE invisibility_context = SELECT
    (presentation_layer_usage,
    presentation_representation
    presentation_set);
END_TYPE;
(*
```

6.3.43 Тип invisible_item

Тип **invisible_item** выбирает элементы изображения, обозначаемые как невидимые.

EXPRESS-спецификация

```
*)
TYPE invisible_item = SELECT
    (styled_item,
    presentation_layer_assignment,
    presentation);
END_TYPE;
(*
```

6.4 Определение объектов схемы вида представления: задание стиля

6.4.1 Объект styled_item

Объект **styled_item** является объектом **representation_item** с заданным стилем его представления.

EXPRESS-спецификация

```
*)
ENTITY styled_item
  SUBTYPE OF (representation_item);
  styles   : SET [1:?] OF presentation_style_assignment;
  item     : representation_item;
WHERE
  WR1 : (SIZEOF(SELF.styles) = 1)
                XOR
        (SIZEOF(QUERY(pres_style <* SELF.styles |
          NOT ('PRESENTATION_APPEARANCE_SCHEMA.' +
                PRESENTATION_STYLE_BY_CONTEXT' IN
          TYPEOF(pres_style))
          )) = 0);
END_TYPE;
(*
```

Определения атрибутов

styles — стили, заданные для конкретного элемента;

item — элемент с заданными стилями.

Формальное утверждение

WR1— в набор объектов **style** должен входить только один стиль, или данный набор должен содержать объекты **presentation_style_by_context**.

Примечание — Это гарантирует отсутствие конфликтов между стилями; кроме того, конкретный стиль может быть задан только в определенном контексте.

6.4.2 Объект over_riding_styled_item

Объектом **over_riding_styled_item** является объект **styled_item**, стиль которого задан через другой ранее определенный стиль. Это происходит, когда атрибут **over_ridden_style.styled_item** и данный **over_riding_styled_item** одновременно прямо или косвенно входят в одно и тоже представление.

Пример 10 — Экземпляр окружности входит в объект **geometric_curve_set**, являющийся элементом объекта **representation**. Экземпляр объекта **styled_item** является элементом того же объекта **representation**. Рассматриваемый **styled_item** включает в качестве составляющих данный **geometric_curve_set** и объект **presentation_style_assignment** с объектом **curve_style**, имеющим значение цвета «синий». Экземпляр объекта **over_riding_styled_item** также является элементом того же представления. Элементами данного **over_riding_styled_item** являются: экземпляр окружности, экземпляр объекта **styled_item** в качестве **over_ridden_style** и объект **presentation_style_assignment** с объектом **curve_style**, имеющим значение цвета «красный». В этом случае красный цвет окружности является преимущественным по отношению к синему, заданному **geometric_curve_set** при его представлении.

EXPRESS-спецификация

```
*)
ENTITY over_riding_styled_item
   SUBTYPE OF (styled_item);
   over_ridden_style     : styled_item;
END_ENTITY;
(*
```

Определение атрибута

over_ridden_style — объект **styled_item**, стиль которого переопределен.

6.4.3 Объект context_dependent_over_riding_styled_item

Объектом **context_dependent_over_riding_styled_item** является объект **over_riding_styled_item**, стиль которого задан через другой ранее определенный атрибутом **representation.representation_item** или комбинацию объектов **representation** и **representation_item**, посредством которой стилизуют данный элемент.

Пример 11 — Экземпляр окружности использован в экземпляре 1 объекта **geometric_curve_set**, являющегося элементом экземпляра 1 объекта **representation**. В настоящем примере окружность ограничивает головку винта дверной петли. Экземпляр объекта **styled_item** является элементом экземпляра 1 объекта **representation**. В данный **styled_item** в качестве элементов входят **geometric_curve_set** и **presentation_style_assignment** с объектом **curve_style**, задающим синий цвет. Экземпляр 1 объекта **representation** входит в экземпляр 2 объекта **representation** посредством экземпляров 1 объектов **representation_map** и **mapped_item** в качестве одной из дверных петель. Экземпляр 1 объекта **representation** входит в различные конфигурации экземпляра 2 объекта **representation** посредством экземпляров 2 объектов **representation_map** и **mapped_item** в качестве второй дверной петли. Экземпляр **context_dependent_over_riding_styled_item** также является элементом экземпляра 2 объекта **representation**. При этом элементами **context_dependent_over_riding_styled_item** являются: экземпляр окружности, экземпляр **styled_item** в качестве **over_ridden_style**, **presentation_style_assignment** с объектом **curve_style**, задающим красный цвет, и **style_context** экземпляра 1 объекта **mapped_item**. Красный цвет окружности является преимущественным по сравнению с синим, заданным объектом **geometric_curve_set** для **representation**, так как он входит в данное представление посредством экземпляра 1 **mapped_item**. Представление экземпляра 2 объекта **representation** может включать две различные позиции **geometric_curve_set**: первую, имеющую головку винта красного цвета, а все другие кривые — синего, и вторую, где все кривые синего цвета.

EXPRESS-спецификация

```
*)
ENTITY context_dependent_over_riding_styled_item
   SUBTYPE OF (over_riding_styled_item);
   style_context     : SET [1:2] OF style_context_select;
WHERE
   WR1 : (SIZEOF(QUERY (sc <* SELF.style_context |
         'RE'PRESENTATION_SCHEMA.REPRESENTATION' IN
           TYPEOF(sc))) = 1)
                  AND
         (SIZEOF(QUERY(sc <* SELF.style_context |
         REPRESENTATION_SCHEMA.REPRESENTATION_ITEM' IN
           TYPEOF(sc))) = 1);
END_ENTITY;
(*
```

Определение атрибута

style_context — набор из одного или двух контекстов для переопределения соответствующего стиля.

Формальное утверждение

WR1 — в набор **style_context** не должно входить более одного объекта **representation** и **representation_item**.

6.4.4 Объект presentation_style_assignment

Объект **presentation_style_assignment** представляет собой набор стилей, заданных в объекте **representation_item** с целью представления конкретного элемента. Определения этих стилей влияют только на вид элемента конкретного типа. Стиль поверхности связан только с поверхностями. Стили области заполнения и кривой связаны только с кривыми и поверхностями. Стиль точки относится к точкам, кривым и поверхностям. Стиль текста связан только с аннотируемым текстом. Стиль знаков связан только со знаками.

Пример 12 — В случае задания для линии стиля кривой, его следует использовать. В случае задания для линии стилей кривой и точки следует использовать оба стиля.

EXPRESS-спецификация

```
*)
ENTITY presentation_style_assignment
   styles    : SET [1:?] OF presentation_style_select;
WHERE
   WR1 : (SIZEOF (QUERY (style1 <* SELF.styles |
            NOT (SIZEOF (QUERY (style2 <* (SELF.styles — style 1) |
               NOT ((TYPEOF (style1) <> TYPEOF (style2)) OR
                  (SIZEOF (['PRESENTATION_APPEARANCE_SCHEMA.' +
                              'SURFACE_STYLE_USAGE',
                              'PRESENTATION_APPEARANCE_SCHEMA.' +
                              'EXTERNALLY_DEFINED_STYLE'] *
                           TYPEOF(style1)) = 1)
               )) = 0
            ))) = 0;
WR2: (SIZEOF (QUERY(style1 <* SELF.styles |
       'PRESENTATION_APPEARANCE_SCHEMA.SURFACE_STYLE_USAGE' IN
         TYPEOF (style1)
         )) < = 2;
END_ENTITY;
(*
```

Определение атрибута

styles — набор стилей представления, заданных в объекте **representation_item**.

Формальные утверждения

WR1 — один и тот же стиль не должен использоваться дважды, за исключением объектов **externally_defined_style** и **surface_style_usage**;

WR2 — в наборе стилей объект **surfage_style_usage** не должен использоваться более двух раз.

Неформальные утверждения

IP1 — внешне определенный стиль не должен противоречить другим стилям того же объекта **presentation_style_assignment**, включая другие внешне определенные стили.

Примечание — При конфликте одного стиля с другим это связано с различными стилями описания одинаковой характеристики (такой, как цвет или ширина). Например, для одного объекта могут быть заданы два стиля цвета: синий или зеленый;

IP2 — тип каждого стиля должен быть уникальным;

IP3 — если в наборе стилей имеются два экземпляра объекта **surface_style_usage**, каждый из них должен определять стилизацию противоположных сторон поверхности.

6.4.5 Объект presentation_style_by_context

Объектом **presentation_style_by_context** является объект **presentation_style_assignment**, заданный для объекта **representation_item** и используемый только в контексте конкретного представления.

EXPRESS-спецификация
```
*)
ENTITY presentation_style_by_context
    SUBTYPE OF (presentation_style_assignment);
    style_context     : style_context_select;
END_ENTITY;
(*
```

Определение атрибута

style_context — контекст представления, стиль которого задан объектом **representation_item**.

6.4.6 Объект pre_defined_presentation_style

Объект **pre_defined_presentation_style** может быть использован для определения некоторых прикладных характеристик видов атрибутов, определенных в рассматриваемой схеме.

Примечание — Порядок применения данного объекта задают в прикладных ресурсах или протоколах.

EXPRESS-спецификация
```
*)
ENTITY pre_defined_presentation_style
    SUBTYPE OF (pre_defined_item);
END_ENTITY;
(*
```

6.4.7 Объект externally_defined_style

Объектом **externally_defined_style** является внешняя ссылка на стиль представления.

EXPRESS-спецификация
```
*)
ENTITY externally_defined_style
    SUBTYPE OF (externally_defined_item);
END_ENTITY;
(*
```

6.5 Определения объектов схемы вида представления: стили представления для точек

6.5.1 Объект point_style

Объект **point_style** определяет визуальное представление точек.

EXPRESS-спецификация
```
*)
ENTITY point_style
    name              : lable;
    marker            : marker_select;
    marker_size       : size_select;
    marker_colour     : colour;
END_ENTITY;
(*
```

Определения атрибутов

name — слово или группа слов, посредством которых ссылаются на **point_style**;

marker — вид маркера, который следует использовать для представления точки;

marker_size — размер, в единицах объекта **presentation_area**, используемый для начертания маркера;

marker_colour — цвет маркера.

6.5.2 Объект pre_defined_marker

Объект **pre_defined_marker** может быть использован для определения прикладных знаков маркера.

Примечание — Порядок применения данного объекта задают в прикладных ресурсах или протоколах.

EXPRESS-спецификация
```
*)
ENTITY pre_defined_marker
```

```
    SUBTYPE OF (pre_defined_item);
END_ENTITY;
(*
```

6.5.3 Объект pre_defined_size

Объект **pre_defined_size_marker** может быть использован для определения прикладных размеров маркера.

Примечание — Порядок применения данного объекта задают в прикладных ресурсах или протоколах.

EXPRESS-спецификация

```
*)
ENTITY pre_defined_size
    SUBTYPE OF (pre_defined_item);
END_ENTITY;
(*
```

6.6 Определения объектов схемы вида представления: стили представления для кривых линий

6.6.1 Объект curve_style

Объект **curve_style** определяет визуальный вид кривой.

EXPRESS-спецификация

```
*)
ENTITY curve_style
    name          : lable;
    curve_font    : curve_font_or_scaled_curve_font_select;
    curve_width   : size_select;
    curve_colour  : colour;
END_ENTITY;
(*
```

Определения атрибутов

name — слово или группа слов, посредством которых ссылаются на **curve_style**;

curve_font — объект **curve_style_font**, масштабированный объект **curve_style_font**, объект **pre_-defined_curve_font**, масштабированный объект **pre_defined_curve_font**, объект **externally_defined_curve_font** или масштабированный **externally_defined_curve_font**, используемые для представления кривой;

curve_width — ширина (толщина) видимой части представленной кривой, выраженная в единицах объекта **presentation_area**;

curve_colour — цвет видимой части кривой.

6.6.2 Объект curve_style_with_ends_and_corners

Объект **curve_style** задает визуальный вид окончаний и угловых положений кривой

EXPRESS-спецификация

```
*)
ENTITY curve_style_with_ends_and_corners
    SUBTYPE OF (curve_style);
    curve_ends     : squared_or_rounded;
    curve_corners  : squared_or_rounded;
END_ENTITY;
(*
```

Определения атрибутов

curve_ends — атрибут, определяющий представление концов кривой;

curve_corners — атрибут, определяющий представление угловых положений кривой.

6.6.3 Объект curve_style_with_extension

Объектом **curve_style_with_extension** является объект **curve_style**, показывающий отображение концов кривой в представлении (удлиненное или укороченное).

EXPRESS-спецификация

```
*)
ENTITY curve_style_with_extension
    SUBTYPE OF (curve_style);
```

```
    curve_extension    : length_measure;
END_ENTITY;
(*
```

Определение атрибута

curve_extension — объект **length_measure**, показывающий, как удлинить или укоротить концы кривой. Если объект **length_measure** положителен, тогда оба конца кривой должны быть расширены в направлении их касательной на величину, заданную единицами измерения объекта **presentation_area**. Если объект **length_measure** отрицателен, тогда кривая должна быть укорочена на абсолютное значение длины, заданное единицами длины объекта **presentation_area**.

Примечание — См. рисунок 19.

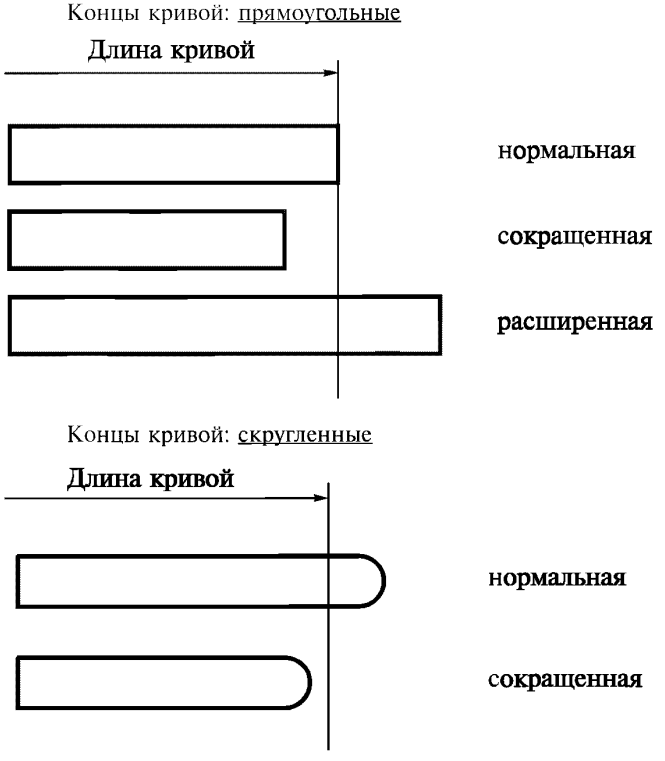


Рисунок 19 — Стиль кривой линии с расширением

6.6.4 Объект pre_defined_curve_font

Объект **pre_defined_curve_font** может быть использован для определения прикладных шрифтов кривой.

Примечание — Порядок применения данного объекта задают в прикладных ресурсах или протоколах.

EXPRESS-спецификация

```
*)
ENTITY pre_defined_curve_font
    SUBTYPE OF (pre_defined_item);
END_ENTITY;
(*
```

6.6.5 Объект externally_defined_curve_font

Объект **externally_defined_curve_font** задает внешнюю ссылку на шрифт кривой.

EXPRESS-спецификация

```
*)
```

```
ENTITY externally_defined_curve_font
   SUBTYPE OF (externally_defined_item);
END_ENTITY;
(*
```

6.6.6 Объект curve_style_font

Объект **curve_style_font** описывает шаблон из нескольких объектов **curve_style_font_patterns**. Данный шаблон применяют на всем протяжении кривой.

EXPRESS-спецификация

```
*)
ENTITY curve_style_font
   name          : lable;
   pattern_list  : LIST [1:?] OF curve_style_font_pattern;
END_ENTITY;
(*
```

Определения атрибутов

name — слово или группа слов, посредством которых ссылаются на **curve_style_font**;

pattern_list — список объектов **curve_style_font_pattern**, содержащий шаблоны, используемые для вычерчивания кривых. Данные шаблоны реализуют в порядке их перечисления в списке.

Неформальные утверждения

IP1 — кривая всегда начинается с полного шаблона;

IP2 — шаблон шрифта ограничивает конец кривой и может быть неполным.

6.6.7 Объект curve_style_font_pattern

Объект **curve_style_font_pattern** представляет собой пару видимых и невидимых сегментов кривой, заданных в единицах объекта **presentation_area**.

EXPRESS-спецификация

```
*)
ENTITY curve_style_font_pattern
   visible_segment_length      : positive_length_measure;
   invisible_segment_length    : positive_length_measure;
END_ENTITY;
(*
```

Определения атрибутов

visible_segment_length — длина видимого сегмента шаблона, заданная в единицах измерения объекта **presentation_area**;

invisible_segment_length — длина невидимого сегмента шаблона, заданная в единицах измерения объекта **presentation_area**.

6.6.8 Объект curve_style_wide

Объект **curve_style_wide** определяет стиль заполнения видимых сегментов кривой. Его используют при стилистическом оформлении кривых с ячеестой структурой или штрих-пунктирных линий, а также при оформлении границ сегментов кривой.

EXPRESS-спецификация

```
*)
ENTITY curve_style_wide
   SUBTYPE OF (curve_style_font);
   interior_style      : fill_area_style;
END_ENTITY;
(*
```

Определение атрибута

interior_style — стиль заполнения видимых сегментов кривой посредством «ячеек» или штрихов.

6.6.9 Объект curve_style_curve_pattern_set

Объект **curve_style_curve_pattern_set** определяет стиль заполнения видимых сегментов кривой в соответствии с повторяющимися шаблонами. Данный объект дублируется на всем протяжении видимых сегментов кривой.

EXPRESS-спецификация

```
*)
ENTITY curve_style_curve_pattern_set
    SUBTYPE OF (curve_style_font;
                geometric_representation_item);
    pattern_set     : SET [1:?] OF curve_style_curve_pattern;
END_ENTITY;
(*
```

Определение атрибута

pattern_set — повторяемый объект **pattern_set**, содержащий набор объектов **curve_style_curve_pattern**.

Неформальные утверждения

IP1 — кривая всегда начинается с полного шаблона;

IP2 — шаблон кривой ограничивает конец кривой и может быть неполным.

6.6.10 Объект curve_style_curve_pattern

Объект **curve_style_curve_pattern** определяет шаблон кривой, используемый в объекте **curve_style_curve_pattern_set**. Объект **curve_style_curve_pattern** задают в виде локального шаблона в определенной системе координат, распространяемого на видимые сегменты исходной кривой. Проецируемая кривая касательна к оси *x* и нормальна к оси y системы координат, определяющей данный локальный шаблон.

Примечание — Иллюстрация объекта **curve_style_curve_pattern** приведена на рисунке 20.

EXPRESS-спецификация

```
*)
ENTITY curve_style_curve_pattern
    SUBTYPE OF (geometric_representation_item);
    pattern           : annotation_curve_occurrence;
    pattern_length    : positive_length_measure;
END_ENTITY;
(*
```

Определения атрибутов

pattern — дублируемый шаблон для заполнения кривой. Данный шаблон определяют как объект **annotation_curve_occurrence**, поэтому он связан с объектом **presentation_style**;

pattern_length — длина шаблона в единицах, заданных объектом **presentation_area**.

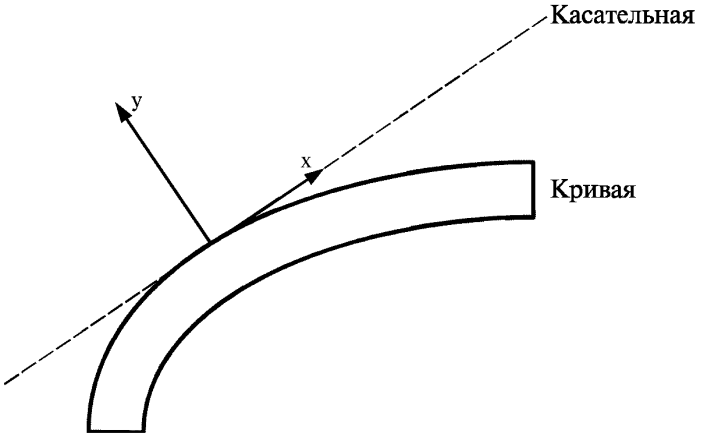


Рисунок 20 — Стиль изображения кривой

6.6.11 Объект curve_style_font_and_scaling

Объект **curve_style_font_and_scaling** используют для масштабирования конкретного объекта **curve_style_font**.

EXPRESS-спецификация

```
*)
ENTITY curve_style_font_and_scaling
    name                 : lable;
    curve_font           : curve_style_font_select;
    curve_font_scaling   : REAL;
END_ENTITY;
(*
```

Определения атрибутов

name — слово или группа слов, посредством которых ссылаются на объект **curve_style_font_and_scaling**;

curve_font — масштабируемый объект **curve_font**;

curve_font_scaling — коэффициент масштабирования.

6.7 Определения объектов схемы вида представления: стили представления для заполненных областей

6.7.1 Объект fill_area_style

Объект **fill_area_style** задает стиль для заполнения видимых сегментов кривой, аннотируемых заполняемых областей или поверхностей в виде ячеек или штриховки.

EXPRESS-спецификация

```
*)
ENTITY fill_area_style
    name          : lable;
    fill_styles   : SET [1:?] OF fill_style_select;
WHERE
    WR1 : SIZEOF(QUERY (fill_style <* SELF.fill_styles |
                'PRESENTATION_APPEARANCE_SCHEMA.'+
                'FILL_AREA_STYLE_COLOUR' IN
                TYPEOF(fill_style)
                )) <= 1;
END_ENTITY;
(*
```

Определения атрибутов

name — слово или группа слов, посредством которых сслаются на **fill_area_style**;

fill_styles — набор стилей заполнения областей для представления видимых сегментов кривой, аннотируемых заполняемых областей или поверхностей.

Формальное утверждение

WR1 — в наборе объектов **fill_styles** не должно быть более одного объекта **fill_area_style_colour**.

6.7.2 Объект fill_area_style_colour

Объект **fill_area_style_colour** определяет цвет, используемый для сплошного заполнения видимых сегментов кривой, аннотируемых заполняемых областей или поверхностей.

EXPRESS-спецификация

```
*)
ENTITY fill_area_style_colour
    name          : lable;
    fill_colour   : colour;
END_ENTITY;
(*
```

Определения атрибутов

name — слово или группа слов, посредством которых ссылаются на **fill_area_style_colour**;

fill_colour — цвет, используемый для заливки конкретной области.

6.7.3 Объект pre_defined_hatch_style

Объект **pre_defined_hatch_style** является стилем штриховки, заданным для прикладных протоколов, определяющих стиль штриховки сплошными или пунктирными линиями.

Примечание — Порядок применения данного объекта задают в прикладных ресурсах или протоколах.

EXPRESS-спецификация

```
*)
ENTITY pre_defined_hatch_style
   SUBTYPE OF (pre_defined_item, geometric_representation_item);
END_ENTITY;
(*
```

6.7.4 Объект externally_defined_hatch_style

Объект **externally_defined_hatch_style** задает внешнюю ссылку на стиль штриховки.

EXPRESS-спецификация

```
*)
ENTITY externally_defined_hatch_style
   SUBTYPE OF (externally_defined_item, geometric_representation_item);
END_ENTITY;
(*
```

6.7.5 Объект fill_area_style_hatching

Объект **fill_area_style_hatching** определяет стилизуемые отрезки для штриховки видимых сегментов кривых, аннотируемых заполняемых областей или поверхностей.

EXPRESS-спецификация

```
*)
ENTITY fill_area_style_hatching
   SUBTYPE OF (geometric_representation_item);
   hatch_line_appearance          : curve_style;
   start_of_next_hatch_line       : one_direction_repeat_factor;
   point_of_reference_hatch_line  : cartesian_point;
   pattern_start                  : cartesian_point;
   hatch_line_angle               : plane_angle_measure;
END_ENTITY;
(*
```

Определения атрибутов

hatch_line_appearance — объект **curve_style** для линий штриховки. Любой **curve_style** должен быть привязан к началу линии штриховки. Начало эталонной штриховой линии задают посредством атрибута **pattern_start**. Начало любой другой линии штриховки задают посредством включения ряда атрибутов **start_of_next_hatch_line** в атрибут **pattern_start**;

start_of_next_hatch_line — сдвиг между смежными линиями штриховки, заданный в виде вектора;

point_of_reference_hatch_line — начало отображения объекта **fill_area_style_hatching** на кривую, аннотируемую заполняемую область или поверхность;

pattern_start — исходная точка объекта **curve_style** для **reference_hatch_line**;

hatch_line_angle — угол, задающий направление параллельных линий штриховки.

Примечание — Объект **fill_area_style_hatching** проиллюстрирован на рисунке 21.

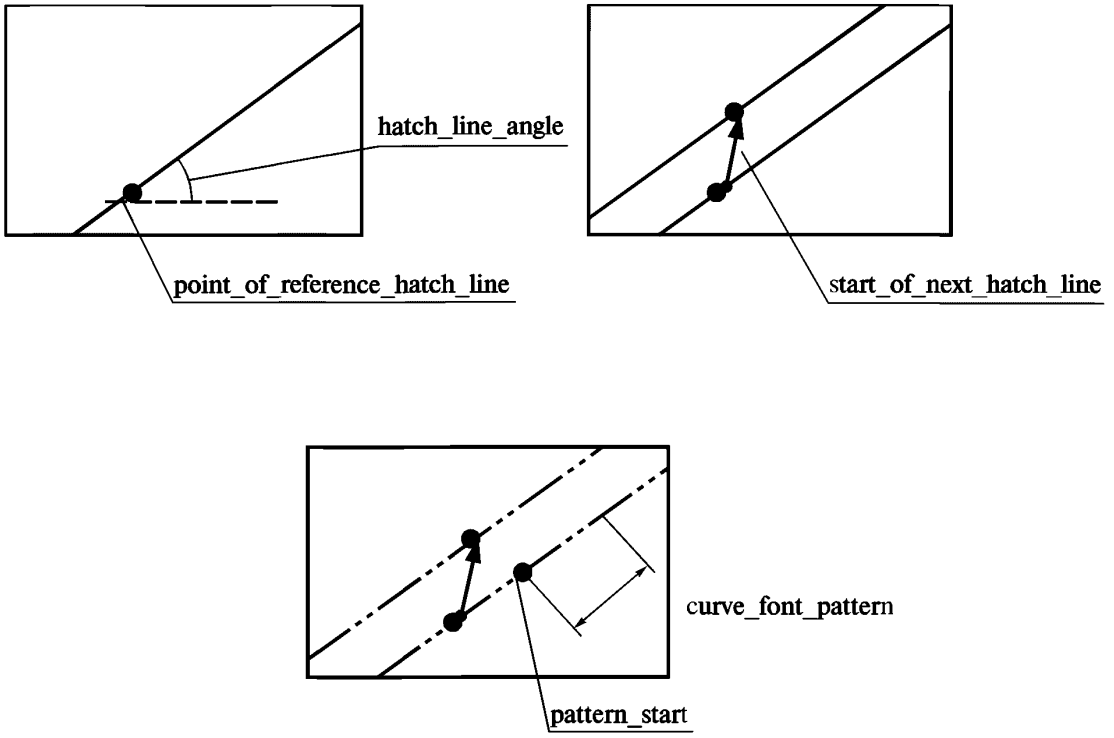


Рисунок 21 — Стиль штриховки заполняемой области

6.7.6 Объект pre_defined_tile_style

Объект **pre_defined_tile_style** является стилем «ячейки (мозаики)», заданным для прикладных протоколов, определяющих конкретный мозаичный стиль.

Примечание — Порядок применения данного объекта задают в прикладных ресурсах или протоколах.

EXPRESS-спецификация

```
*)
ENTITY pre_defined_tile_style
   SUBTYPE OF (pre_defined_item, geometric_representation_item);
END_ENTITY;
(*
```

6.7.7 Объект externally_defined_tile_style

Объект **externally_defined_tile_style** задает внешнюю ссылку на мозаичный стиль.

EXPRESS-спецификация

```
*)
ENTITY externally_defined_tile_style
   SUBTYPE OF (externally_defined_item, geometric_representation_item);
END_ENTITY;
(*
```

6.7.8 Объект fill_area_style_tiles

Объект **fill_area_style_tiles** определяет типовую двумерную «ячейку», используемую для мозаичного заполнения аннотируемых заполняемых или других ограниченных (замкнутых) областей. Вид (структуру) мозаики задают посредством набора объектов **tiles**, а местоположение каждой ячейки — посредством объекта **tiling_pattern**, определяющего взаиморасположение смежных ячеек. Ячейки или их части, выходящие за пределы заполняемой области, должны быть обрезаны по границам данной области.

EXPRESS-спецификация
```
*)
ENTITY fill_area_style_tiles
    SUBTYPE OF (geometric_representation_item);
    tiling_pattern      : two_direction_repeat_factor;
    tiles               : SET [1:?] OF fill_area_style_tile_shape_select;
    tiling_scale        : positive_ratio_measure;
END_ENTITY;
(*
```

Определения атрибутов

tiling_pattern — объект **two_direction_repeat_factor**, определяющий форму и взаимоположение ячеек;

tiles — набор компонентов ячейки;

tiling_scale — коэффициент масштабирования каждой ячейки при ее определении в объекте **annotation_fill_area**.

6.7.9 Объект fill_area_style_tile_with_style

Объект **fill_area_style_tile_with_style** определяет стилизованную кривую, являющуюся компонентом объекта **fill_area_style_tiles**.

EXPRESS-спецификация
```
*)
ENTITY fill_area_style_tile_with_style
    SUBTYPE OF (geometric_representation_item);
    styled_curve        : annotation_curve_occurrence;
END_ENTITY;
(*
```

Определение атрибута

styled_curve — двумерная стилизованная кривая, заданная в локальной системе координат ячейки заполняемой области.

6.7.10 Объект fill_area_style_tile_coloured_region

Объект **fill_area_style_tile_coloured_region** определяет замкнутую кривую, окрашенную определенным цветом и являющуюся компонентом объекта **fill_area_style_tiles**.

EXPRESS-спецификация
```
*)
ENTITY fill_area_style_tile_coloured_region
    SUBTYPE OF (geometric_representation_item);
    closed_curve        : curve_or_annotation_curve_occurrence;
    region_colour       : colour;
END_ENTITY;
(*
```

Определения атрибутов

closed_curve — замкнутая кривая, ограничивающая закрашиваемую область. Данная эталонная кривая также может быть задана в определенном стиле;

region_colour — цвет области, ограниченной замкнутой кривой.

Неформальное утверждение

IP1 — объект **closed_curve** должен быть замкнутым и не самопересекающимся.

6.7.11 Объект fill_area_style_tile_symbol_with_style

Объект **fill_area_style_tile_symbol_with_style** представляет собой символ (знак), являющийся компонентом объекта **fill_area_style_tiles**.

EXPRESS-спецификация
```
*)
ENTITY fill_area_style_tile_symbol_with_style
    SUBTYPE OF (geometric_representation_item);
    symbol    : annotation_symbol_occurrence;
END_ENTITY;
(*
```

Определение атрибута

symbol — стилизованный аннотируемый символ.

6.7.12 Объект pre_defined_tile

Объект **pre_defined_tile** может быть использован для определения применяемой ячейки.

Примечание — Порядок применения данного объекта задают в прикладных ресурсах или протоколах.

EXPRESS-спецификация

```
*)
ENTITY pre_defined_tile
   SUBTYPE OF (pre_defined_item);
END_ENTITY;
(*
```

6.7.13 Объект externally_defined_tile

Объектом **externally_defined_tile** является внешне определенная эталонная ячейка.

EXPRESS-спецификация

```
*)
ENTITY externally_defined_tile
   SUBTYPE OF (externally_defined_item);
END_ENTITY;
(*
```

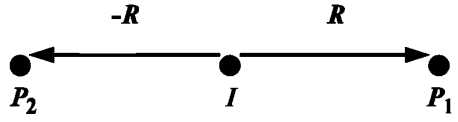
6.7.14 Объект one_direction_repeat_factor

Объектом **one_direction_repeat_factor** является вектор, используемый в объекте **fill_area_style_hatching** для определения положения повторяющейся штриховой линии по отношению к исходной линии штриховки. При заданном исходном положении I какой-либо линии штриховки объект **one_direction_repeat_factor** определяет два новых положения R в соответствии с формулой

$$I + k \cdot R,$$

где $k = -1,1$.

Примечание — На рисунке 22 показаны позиции, задаваемые объектом **one_direction_repeat_factor**.



I — исходное положение; $P_1 = I + P$
P — коэффициент повторения. $P_2 = I - P$

Рисунок 22 — Коэффициент повторения одномерного изображения

EXPRESS-спецификация

```
*)
ENTITY one_direction_repeat_factor
   SUBTYPE OF (geometric_representation_item);
   repeat_factor     : vector;
END_ENTITY;
(*
```

Определение атрибута

repeat_factor — вектор, определяющий относительное положение штриховых линий.

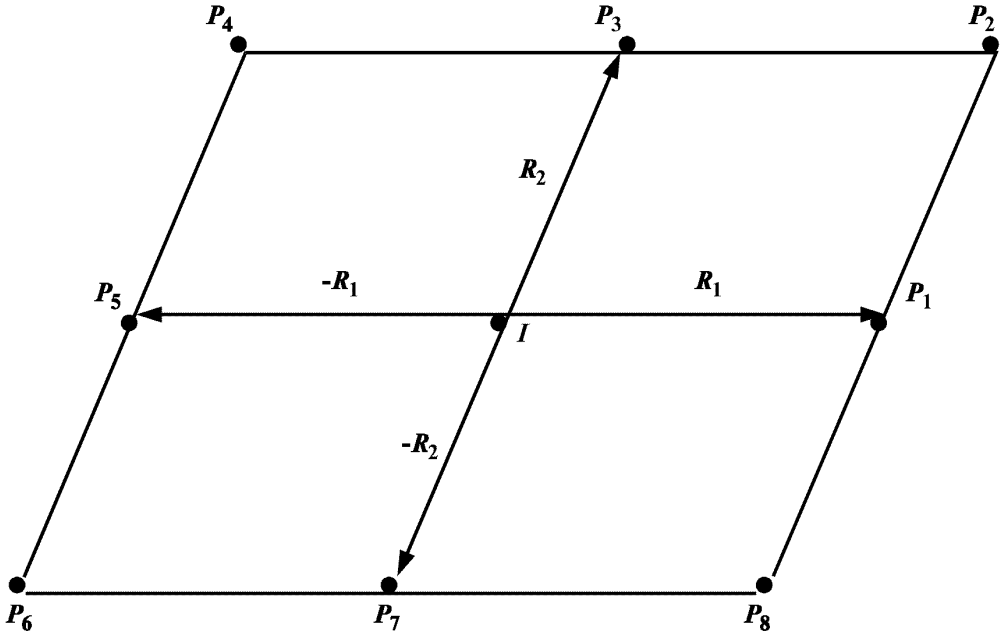
6.7.15 Объект two_direction_repeat_factor

Объект **two_direction_repeat_factor** является комбинацией двух векторов, используемых в объекте **fill_area_style_tiles** для определения формы и взаимоположения ячеек. При заданном исходном положении I какой-либо ячейки, объект **two_direction_repeat_factor** задает восемь новых позиций посредством вектора $R = (R_1, R_2)$ в соответствии с формулой

$$I = k_1 \cdot R_1 + k_2 \cdot R_2,$$

где $k_1, k_2 = -1, 0, 1, k^2_1 + k^2_2 \neq 0$.

Примечание — На рисунке 23 показаны позиции, задаваемые объектом **two_direction_repeat_factor**.



I — исходное положение;
R_1 — коэффициент повторения;
R_2 — второй коэффициент повторения.

$P_1 = I + R_1$.
$P_2 = I + R_1 + R_2$.
$P_3 = I + R_2$.
$P_4 = I - R_1 + R_2$.
$P_5 = I - R_1$.
$P_6 = I - R_1 - R_2$.
$P_7 = I - R_2$.
$P_8 = I + R_1 - R_2$.

Рисунок 23 — Коэффициент повтора двумерного изображения

EXPRESS-спецификация

```
*)
ENTITY two_direction_repeat_factor
    SUBTYPE OF (one_direction_repeat_factor);
    second_repeat_factor     : vector;
END_ENTITY;
(*
```

Определение атрибута

second_repeat_factor — вектор, определяющий относительное положение ячеек в двух направлениях.

6.8 Определения объектов схемы вида представления: стили представления для поверхностей

6.8.1 Объект surface_style_usage

Объект **surface_style_usage** определяет использование объекта **surface_side_style_select** для лицевой, оборотной или обеим сторонам поверхности.

EXPRESS-спецификация

```
*)
ENTITY surface_style_usage
    side      : surface_side;
    style     : surface_side_style_select;
END_ENTITY;
(*
```

Определения атрибутов

side — определяет сторону поверхности, для которой используют заданный стиль;

style — стиль, который следует использовать при описании поверхности.

6.8.2 Объект pre_defined_surface_side_style

Объект **pre_defined_surface_side_style** может быть использован при определении прикладного применения объектов **surface_side_style**.

Примечание — Порядок применения данного объекта задают в прикладных ресурсах или протоколах.

EXPRESS-спецификация

```
*)
ENTITY pre_defined_surface_side_style
   SUBTYPE OF (pre_defined_item);
END_ENTITY;
(*
```

6.8.3 Объект surface_side_style

Объект **surface_side_style** представляет собой набор (коллекцию) стилей, используемых при представлении конкретной стороны поверхности.

EXPRESS-спецификация

```
*)
ENTITY surface_side_style;
   name      : lable;
   styles       : SET [1:?] OF surface_style_element_select;
WHERE
   WR1 : SIZEOF(QUERY (style1 <* SELF.styles |
            SIZEOF(QUERY (style2 <* SELF.style1 |
               TYPEOF(style1) = TYPEOF(style2)
               )) > 0
            )) = 0;
END_ENTITY;
(*
```

Определения атрибутов

name — слово или группа слов, посредством которых ссылаются на **surface_side_style**;

styles — набор (коллекция) различных стилей поверхности.

Формальное утверждение

WR1 — все стили должны быть различных типов.

6.8.4 Объект surface_style_fill_area

Объект **surface_style_fill_area** определяет стиль поверхности при отображении на нее заполняемой области.

EXPRESS-спецификация

```
*)
ENTITY surface_style_fill_area;
   fill_area    : fill_area_style;
END_ENTITY;
(*
```

Определение атрибута

fill_area — объект **fill_area_style**, связанный с двумерным параметризованным пространством поверхности и отображаемый на эту поверхность.

6.8.5 Объект surface_style_boundary

Объект **surface_style_boundary** задает стиль поверхности, используемый для кривых, ограничивающих данную поверхность.

EXPRESS-спецификация

```
*)
ENTITY surface_style_boundary;
   style_of_boundary    : curve_or_render;
END_ENTITY;
(*
```

Определение атрибута

style_of_boundary — стиль кривых, ограничивающих поверхность.

6.8.6 Объект curve_style_rendering

Объект **curve_style_rendering** позволяет визуалировать кривые на поверхности посредством применения соответствующих методов.

EXPRESS-спецификация

```
*)
ENTITY curve_style_rendering;
    rendering_method        : shading_curve_method;
    rendering_properties    : surface_rendering_properties;
END_ENTITY;
(*
```

Определения атрибутов

rendering_method — определяет метод, используемый для интерполяции цветов кривых на поверхности;

rendering_properties — определяет свойства визуализации поверхности, содержащей заданные кривые.

6.8.7 Объект surface_rendering_properties

Объект **surface_rendering_properties** определяет свойства поверхности, необходимые для вычисления фактических параметров визуализации поверхностей на основе соответствующих методов.

EXPRESS-спецификация

```
*)
ENTITY surface_rendering_properties;
    rendered_colour     : colour;
END_ENTITY;
(*
```

Определение атрибута

rendered_colour — цвет поверхности, используемый при ее визуализации.

6.8.8 Объект surface_style_silhouette

Объектом **surface_style_silhouette** является стиль поверхности, используемый при описании кривых, ограничивающих ее силуэт.

EXPRESS-спецификация

```
*)
ENTITY surface_style_silhouette;
    style_of_silhouette     : curve_or_render;
END_ENTITY;
(*
```

Определение атрибута

style_of_silhouette — стиль кривых, описывающих силуэт поверхности.

6.8.9 Объект surface_style_segmentation_curve

Объектом **surface_style_segmentation_curve** является стиль поверхности, используемый при описании кривых, ограничивающих сегменты данной поверхности.

EXPRESS-спецификация

```
*)
ENTITY surface_style_segmentation_curve;
    style_of_segmentation_curve     : curve_or_render;
END_ENTITY;
(*
```

Определение атрибута

style_of_segmentation_curve — стиль кривых, ограничивающих сегменты поверхности.

Примечание — Данный стиль относится только к сегментированным поверхностям. Таковыми являются поверхности следующих типов:

- «кусочные» (B-spline);
- прямоугольно выровненные, ограниченные кривыми, образованные из прямоугольков, трассированные, смещенные и дублируемые (связанные с «кусочными»).

6.8.10 Объект surface_style_control_grid

Объектом **surface_style_control_grid** является стиль поверхности, используемый для сети опорных точек, определяющих поверхность.

EXPRESS-спецификация

```
*)
ENTITY surface_style_control_grid;
   style_of_control_grid     : curve_or_render;
END_ENTITY;
(*
```

Определение атрибута

style_of_control_grid — стиль опорной сети поверхности.

Примечание — Данный стиль относится только к поверхностям, заданным через сеть опорных точек. Таковыми являются поверхности следующих типов:

- «кусочные» (B-spline);
- прямоугольно выровненные, ограниченные кривыми, образованные из прямоугольков, трассированные, смещенные и дублируемые (связанные с «кусочными»).

6.8.11 Объект surface_style_parameter_line

Объектом **surface_style_parameter_line** является стиль поверхности, используемый для представления на поверхности линий, параметризованных в соответствии с принятыми стандартами ИСО.

EXPRESS-спецификация

```
*)
ENTITY surface_style_parameter_line;
   style_of_parameter_lines    : curve_or_render;
   direction_counts            : SET [1:2] OF direction_count_select;
WHERE
   WR1 : (HIINDEX(SELF.direction_counts) = 1)
                              XOR
         (TYPEOF(SELF.direction_counts[1]) <>
            TYPEOF(SELF.direction_counts[2]) ));
END_ENTITY;
(*
```

Определения атрибутов

style_of_parameter_lines — стиль представления на поверхности кривых, параметризованных в соответствии с принятыми стандартами ИСО;

direction_counts — набор объектов **u_direction_count** и **v_direction_count**, определяющий число параметризованных кривых в направлениях *u* и *v*.

Формальное утверждение

WR1 — если в атрибут **direction_counts** включены два элемента, они не должны быть однотипными.

6.8.12 Объект surface_style_rendering

Объект **surface_style_rendering** позволяет визуализировать реальные поверхности посредством применения соответствующих методов. Результаты вычисления отражающей способности поверхности должны быть интерполированы к системе координат, в которой задан объект **camera_model**.

EXPRESS-спецификация

```
*)
ENTITY surface_style_rendering;
   rendering_method     : shading_surface_method;
   surface_colour       : colour;
END_ENTITY;
(*
```

Определения атрибутов

rendering_method — метод интерполяции цветов через различные поверхности;

surface_colour — цвет, применяемый при визуализации данной поверхности.

6.8.13 Объект surface_style_rendering_with_properties

Объект **surface_style_rendering_with_properties** позволяет визуализировать реальные поверхности с учетом характеристик их прозрачности и отражающей способности.

EXPRESS-спецификация

```
*)
ENTITY surface_style_rendering_with_properties;
    SUBTYPE OF (surface_style_rendering);
    properties          : SET [1:2] OF rendering_properties_select;
WHERE
    WR1 : (HIINDEX(SELF.properties) = 1)
                                XOR
          (TYPEOF(SELF.properties[1]) <> TYPEOF(SELF.properties[2]) ));
END_ENTITY;
(*
```

Определение атрибута

properties — набор (коллекция) визуализируемых свойств поверхности.

Формальное утверждение

WR1 — все свойства должны быть различных типов.

6.8.14 Объект surface_style_reflectance_ambient

Объект **surface_style_reflectance_ambient** определяет внешнее проявление отражающей способности поверхности.

Примечания

1 Вычисление отражающей способности проводят в ряде освещаемых или затеняемых точек поверхности с указанием цветов этих точек. Исходными данными для этих вычислений являются: позиция, для которой проводят вычисление, нормаль к поверхности, цвет поверхности в данной точке, источники освещения и трехмерная модель камеры.

2 Рекомендуемые формулы для вычисления отражающей способности приведены в приложении D.

EXPRESS-спецификация

```
*)
ENTITY surface_style_reflectance_ambient;
    ambient_reflectance     : REAL;
END_ENTITY;
(*
```

Определение атрибута

ambient_reflectance — коэффициент отражения во внешнюю среду в формуле отражающей способности.

6.8.15 Объект surface_style_reflectance_ambient_diffuse

Объект **surface_style_reflectance_ambient_diffuse** определяет рассеяние при отражении от поверхности.

Примечание — Рекомендуемые формулы для вычисления отражающей способности приведены в приложении D.

EXPRESS-спецификация

```
*)
ENTITY surface_style_reflectance_ambient_diffuse
    SUBTYPE OF (surface_style_reflectance_ambient);
    diffuse_reflectance         : REAL;
END_ENTITY;
(*
```

Определения атрибутов

diffuse_reflectance — коэффициент рассеяния в формуле отражающей способности.

6.8.16 Объект surface_style_reflectance_ambient_diffuse_specular

Объект **surface_style_reflectance_ambient_diffuse_specular** определяет зеркальное отражение от поверхности.

Примечание — Рекомендуемые формулы для вычисления отражающей способности приведены в приложении D.

EXPRESS-спецификация
*)
```
ENTITY surface_style_reflectance_ambient_diffuse_specular
    SUBTYPE OF (surface_style_reflectance_ambient_diffuse);
    specular_reflectance      : REAL;
    specular_exponent         : REAL;
    specular_colour           : colour;
END_ENTITY;
```
(*

Определения атрибутов

specular_reflectance — коэффициент зеркального отражения в формуле отражающей способности;

specular_exponent — показатель зеркального отражения в формуле отражающей способности;

specular_colour — цвет зеркального отражения в формуле отражающей способности.

6.8.17 Объект surface_style_transparent

Объект **surface_style_transparent** определяет свойство визуалируемой поверхности, связанное со степенью ее прозрачности.

EXPRESS-спецификация
*)
```
ENTITY surface_style_transparent
    transparency      : REAL;
WHERE
    WR1 : {0.0 <= transparency <= 1.0}
END_ENTITY;
```
(*

Определение атрибута

transparency — степень прозрачности, заданная в процентах светового потока, проходящего через поверхность.

Формальное утверждение

WR1 — значение прозрачности должно лежать в пределах 0.0—1.0.

6.9 Определения объектов схемы вида представления: стили представления для текста

6.9.1 Объект text_style

Объект **text_style** задает стиль представления аннотируемого текста.

EXPRESS-спецификация
*)
```
ENTITY text_style
    name                      : label;
    character_appearance      : character_style_select;
END_ENTITY;
```
(*

Определения атрибутов

name — слово или группа слов, посредством которых ссылаются на **text_style**;

character_appearance — стиль символов, используемый для представления текста.

6.9.2 Объект character_glyph_style_stroke

Объект **character_glyph_style_stroke** определяет стиль глифа символа для текста, сверстанного преимущественно с использованием кривых линий (криволинейных строк).

EXPRESS-спецификация
*)
```
ENTITY character_glyph_style_stroke
    stroke_style      : curve_style;
END_ENTITY;
```
(*

Определение атрибута

stroke_style — объект **curve_style**, используемый при определении кривых, входящих в объект **character_glyph_symbol_stroke**.

6.9.3 Объект character_glyph_style_outline

Объект **character_glyph_style_outline** определяет стиль глифа символа для текста, сверстанного преимущественно с использованием замкнутых зон.

EXPRESS-спецификация

```
*)
ENTITY character_glyph_style_outline
    outline_style      : curve_style;
END_ENTITY;
(*
```

Определение атрибута

outline_style — объект **curve_style**, используемый при определении кривых, входящих в объект **character_glyph_symbol_outline**.

6.9.4 Объект character_glyph_style_outline_with_characteristics

Объектом **character_glyph_style_outline_with_characteristics** является объект **character_glyph_style_outline** с дополнительными характеристиками зон, определенных в **character_glyph_style_outline**.

EXPRESS-спецификация

```
*)
ENTITY character_glyph_style_outline_with_characteristics
    SUBTYPE OF (character_glyph_style_outline);
    characteristics        : fill_area_style;
END_ENTITY;
(*
```

Определение атрибута

characteristics — характеристики зон, связанных с глифом символа.

6.9.5 Объект text_style_for_defined_font

Объект **text_style_for_defined_font** определяет стиль глифа символа для шрифтов текста, заданных заранее (предопределенных) или задаваемых извне.

EXPRESS-спецификация

```
*)
ENTITY text_style_for_defined_font
    text_colour            : colour;
END_ENTITY;
(*
```

Определение атрибута

text_colour — цвет, используемый для представления текста.

6.9.6 Объект text_style_with_justification

Объектом **text_style_with_justification** является объект **text_style**, определяющий выравнивание текста.

EXPRESS-спецификация

```
*)
ENTITY text_style_with_justification
    SUBTYPE OF (text_style);
    justification          : text_justification;
END_ENTITY;
(*
```

Определение атрибута

justification — метод выравнивания текста по строке.

6.9.7 Объект text_style_with_box_characteristics

Объектом **text_style_with_box_characteristics** является объект text_style, определяющий параметры «ящиков» символов в тексте.

EXPRESS-спецификация

```
*)
ENTITY text_style_with_box_characteristics
    SUBTYPE OF (text_style);
    characteristics      : SET [1:4] OF box_characteristics_select;
```

```
WHERE
   WR1 :  SIZEOF(QUERY (c1 <* SELF.characteristics |
             SIZEOF(QUERY (c2 <* SELF.characteristics  — c1 |
                TYPEOF(c1) = TYPEOF(c2)
                )) > 0
             )) = );
END_ENTITY;
(*
```

Определение атрибута

characteristics — характеристики «ящиков» символов. Данные характеристики определяют высоту, ширину, углы поворота и наклона «ящиков» символов.

Формальное утверждение

WR1 — характеристики должны иметь различные типы.

6.9.8 Объект text_style_with_spacing

Объектом **text_style_with_spacing** является объект **text_style**, определяющий пробелы между символами.

EXPRESS-спецификация

```
*)
ENTITY text_style_with_spacing
   SUBTYPE OF (text_style);
   character_spacing    : character_spacing_select;
END_ENTITY;
(*
```

Определение атрибута

character_spacing — расстояние между смежными символами.

6.9.9 Объект pre_defined_character_spacing

Объектом **pre_defined_character_spacing** является пробел между символами, определяемый в конкретных приложениях.

Примечание — Порядок применения данного объекта задают в прикладных ресурсах или протоколах.

EXPRESS-спецификация

```
*)
ENTITY pre_defined_character_spacing
   SUBTYPE OF (pre_defined_item);
END_ENTITY;
(*
```

6.9.10 Объект text_style_with_mirror

Объектом **text_style_with_mirror** является объект **text_style**, определяющий оси зеркального отображения символов.

Примечание — На рисунке 24 проиллюстрирован объект **text_style_with_mirror**.

EXPRESS-спецификация

```
*)
ENTITY text_style_with_mirror
   SUBTYPE OF (text_style);
   mirror_placement    : axis2_placement;
END_ENTITY;
(*
```

Определение атрибута

mirror_placement — местоположение и ориентация оси зеркального отображения. Данная ось пересекает объект **location** из **axis2_placement** и параллельна оси *x*, указанной в объекте **axis2_placement**.

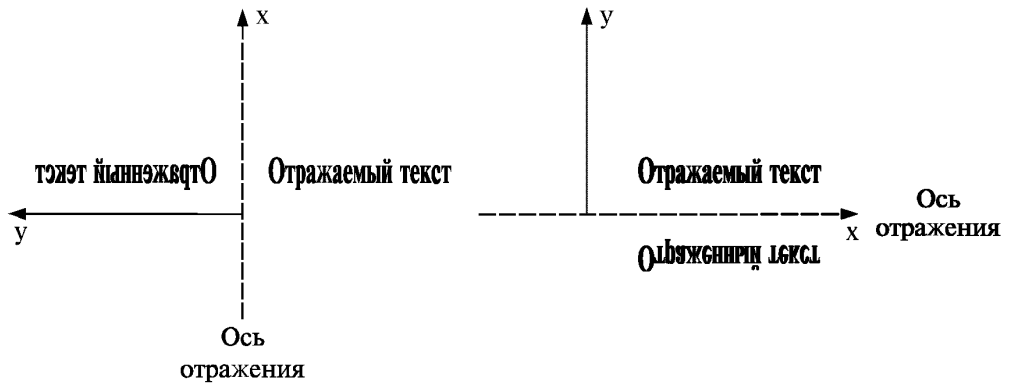


Рисунок 24 — Стиль текста с зеркальным отображением

6.10 Определение объектов схемы вида представления: стили представления для символов

6.10.1 Объект symbol_style

Объектом **symbol_style** является объект **presentation_style**, определяющий визуальные виды объектов **annotation_symbol**. В данном объекте задают один или несколько стилей для компонентов символа или их цветового представления.

EXPRESS-спецификация

```
*)
ENTITY symbol_style;
    name              : label;
    style_of_symbol   : symbol_style_select;
END_ENTITY;
(*
```

Определения атрибутов

name — слово или группа слов, посредством которых ссылаются на **symbol_style**;

style_of_symbol — объект **symbol_element_style** или **symbol_colour**, определяющий визуальный вид символа.

6.10.2 Объект symbol_element_style

Объектом **symbol_element_style** является стиль представления, определяющий визуальный вид компонентов объектов **annotation_symbol**.

EXPRESS-спецификация

```
*)
ENTITY symbol_element_style;
    style_of_symbol : presentation_style_assignment;
WHERE
    WR1 :  SIZEOF(QUERY (style <* SELF.style_of_symbol.styles |
           'PRESENTATION_APPEARANCE_SCHEMA.SYMBOL_STYLE' IN
           TYPEOF(style))) = 0;
    WR2 : NOT ('PRESENTATION_APPEARANCE_SCHEMA.' +
                'PRESENTATION_STYLE_BY_CONTEXT' IN
                TYPEOF(SELF.style_of_symbol)
                );
END_ENTITY;
(*
```

Определение атрибута

style_of_symbol — объект **presentation_style_assignment** для компонентов символа.

Формальные утверждения

WR1 — в объект **style_of_symbol** не должен входить объект **symbol_style**;

WR2 — объект **style_of_symbol** не должен зависеть от контекста.

6.10.3 Объект symbol_colour

Объектом **symbol_colour** является объект **presentation_style**, задающий цвет объекта **annotation_symbol**.

EXPRESS-спецификация

```
*)
ENTITY symbol_colour;
   colour_of_symbol     : colour;
END_ENTITY;
(*
```

Определение атрибута

colour_of_symbol — объект **colour** для символа.

6.11 Определение объектов схемы вида представления: допуски аппроксимации

6.11.1 Объект approximation_tolerance

Объектом **approximation_tolerance** являются визуализируемые требования к каждому изображаемому элементу. Данный объект определяет допустимое положение и форму каждого элемента изображения по отношению к его математически точно проецируемому положению и форме.

EXPRESS-спецификация

```
*)
ENTITY approximation_tolerance;
   tolerance        : tolerance_select;
END_ENTITY;
(*
```

Определение атрибута

tolerance — допуски, используемые при аппроксимации кривых и поверхностей.

Примечание — Если объект **approximation_tolerance** не задан, точность изображения зависит от конкретной реализации.

6.11.2 Объект approximation_tolerance_deviation

Объект **approximation_tolerance_deviation** задает размер отклонения при аппроксимации кривых и поверхностей. Величина отклонения может быть задана в пространстве формы изделия или области представления.

EXPRESS-спецификация

```
*)
ENTITY approximation_tolerance_deviation;
   tessellation_type     : approximation_method;
   tolerances            : SET [1:2] OF tolerance_deviation_select;
   definition_space      : product_or_presentation_space;
WHERE
   WR1 : (HIINDEX(SELF.tolerances) = 1)
                  XOR
         (TYPEOF(SELF.tolerances [1]) <> TYPEOF(SELF.tolerances [2]) ));
END_ENTITY;
(*
```

Определения атрибутов

tessellation_type — выбранный **approximation_method**, определяющий вид мозаичной аппроксимации кривых и поверхностей посредством графических примитивов;

tolerances — набор допусков, определяющий максимально допустимые отклонения при аппроксимации кривых и поверхностей;

definition_space — координатное пространство, в котором заданы допуски. Допуски могут быть заданы в интервале отклонений соответствующей кривой или поверхности или в интервале отклонений объекта **presentation_area**, содержащего кривую или поверхность.

Формальное утверждение

WR1 — при наличии в наборе допусков двух элементов они должны быть разного типа.

6.11.3 Объект approximation_tolerance_parameter

Объект **approximation_tolerance_parameter** определяет допуски аппроксимации кривых и поверхностей в единицах параметрического пространства.

EXPRESS-спецификация

```
*)
ENTITY approximation_tolerance_parameter;
   tolerances              : SET [1:2] OF tolerance_parameter_select;
WHERE
   WR1 : (HIINDEX(SELF.tolerances) = 1)
                    XOR
         (TYPEOF(SELF.tolerances [1]) <> TYPEOF(SELF.tolerances [2]) ));
END_ENTITY;
(*
```

Определение атрибута

tolerances — набор допусков, используемых при аппроксимации кривых и поверхностей. Кривые и поверхности аппроксимируют на основе однородных операций (шагов), заданных в параметрическом пространстве. Соответствующие длины измеряют в единицах параметрического пространства.

Формальное утверждение

WR1 — при наличии в наборе допусков двух элементов последние должны быть разного типа.

6.12 Определения объектов схемы вида представления: сокрытие и видимость

6.12.1 Объект occlusion_precedence

Объектом **occlusion_precedence** является отношение между двумя объектами, один из которых может закрывать или заменять другой. Данное отношение устанавливают, когда один из объектов скрывает или заменяет другой при их перекрытии в представлении.

Данное отношение является транзитивным. Если объект *A* закрывает объект *B*, а объект *B* закрывает объект *C*, тогда объект *A* также закрывает объект *C*.

Данное отношение применяют только для двух объектов, входящих в одно отображение.

Примечание — Если два таких объекта перекрываются и не связаны отношением **occlusion_precedence**, трудно представить, какой из объектов является «верхним» в данной реализации.

EXPRESS-спецификация

```
*)
ENTITY occlusion_precedence;
   higher_precedence       : hiding_or_blanking_select;
   lower_precedence        : hiding_or_blanking_select;
   occlusion_context       : representation;
WHERE
   WR1 : acyclic_occlusion_precedence (SELF, [SELF.lower_precedence]);
END_ENTITY;
(*
```

Определения атрибутов

higher_precedence — объект, который может заменять или скрывать объект **lower_precedence**;

lower_precedence — объект, который может быть заменен или скрыт объектом **higher_precedence**;

occlusion_context — объект **representation**, в котором содержится данное представление.

Формальное утверждение

WR1 — объект **occlusion_precedence** не должен входить в «дерево» объектов **hiding_or_blanking_select**, «корнем» которого является «лист дерева» самих объектов.

6.12.2 Объект invisibility

Объект **invisibility** определяет, что набор (коллекция) из одного или нескольких объектов **styled_items**, элементы, заданные уровню посредством объекта **presentation_style_assignment**, или элементы объекта **presentation_representation** не должны входить в представление.

EXPRESS-спецификация

```
*)
ENTITY invisibility;
   invisible_items   : SET [1:?] OF invisible_item;
END_ENTITY;
(*
```

Определение атрибута

items — набор объектов **styled_item, presentation_layer_assignment** или **presentation_representation**, заданных как невидимые.

6.12.3 Объект context_dependent_invisibility

Объектом **context_dependent_invisibility** является объект **invisibility**, используемый в контексте изображения или уровня. Элементы, заданные как невидимые, являются таковыми только в контексте определенных объектов **presentation_set**, **presentation_representation** или **presentation_layer_usage.**

Пример 13 — Символ входит в качестве элемента в два различных вида, но будет представлен только в первом виде. Объект **context_dependent_invisibility** использован для указания невидимости символа во втором виде посредством использования данного вида в качестве контекста невидимости.

EXPRESS-спецификация

```
*)
ENTITY context_dependent_invisibility;
SUBTYPE OF (invisibility);
    presentation_context          : invisibility_context;
END_ENTITY;
(*
```

Определение атрибута

presentation_context — объект **presentation_set**, **presentation_representation** или **presentation_layer_usage**, содержащий контекст, определяющий невидимость.

6.13 Определение функции схемы вида представления

6.13.1 Функция acyclic_occlusion_precedence

Функция **acyclic_occlusion_precedence** проверяет наличие в любом «дереве» набора объектов **hiding_or_blanking_select** и заданного **occlusion_precedence**, содержащего **hiding_or_blanking_select**, являющегося одновременно «корнем» и «листом» одного «поддерева». Функция возвращает значение TRUE при отсутствии такого «поддерева» и FALSE — в противном случае.

EXPRESS-спецификация

```
*)
FUNCTION acyclic_occlusion_precedence
    ( relation              : occlusion_precedence;
      set_of_lower          : SET OF hiding_or_blanking_select) : BOOLEAN;
    LOCAL
        x                       : SET OF occlusion_precedence;
        local_set_of_lower      : SET OF hiding_or_blanking_select;
    END_LOCAL
    REPEAT i: = 1 TO HIINDEX (set_of_lower);
        IF relation.higher_precedence :=: set_of_lower [i] THEN
            RETURN(FALSE);
        END_IF;
    END_REPEAT;
        x  := bag_to_set (USEDIN (relation.higher_precedence,
                        'PRESENTATION_APPEARANCE_SCHEMA.'+
                  'OCCLUSION_PRECEDENCE.LOWER_PRECEDENCE'));
      local_set_of_lower := set_of_lower + relation.higher_precedence;
      IF SIZEOF (x) > 0 THEN
        REPEAT i:=1 TO HIINDEX (x);
            IF NOT acyclic_occlusion_precedence(x[i],
                                                local_set_of_lower) THEN
                RETURN (FALSE);
          END_IF;
      END_REPEAT;
    END_IF;
    RETURN (TRUE);
END_FUNCTION;
(*
```

Определения аргументов

relation — проверяемый объект **occlusion_precedence**. Он служит исходными данными для функции;

set_of_lower — набор элементов, на которые прямо или косвенно ссылаются посредством атрибута **lower_precedence** объекта **relation**. Данный аргумент служит исходными данными для функции. Изначально данный набор может содержать только один соответствующий элемент (атрибут **lower_precedence** объекта **relation**).

EXPRESS-спецификация

```
*)
END_SCHEMA; - - presentation_appearance_schema
(*
```

7 Схема ресурсов представления

Следующее описание на языке EXPRESS открывает **presentation_resource_schema** и определяет необходимые внешние ссылки.

EXPRESS-спецификация

```
*)
SCHEMA presentation_resource_schema
REFERENCE FROM external_reference_schema
        (externally_defined_item,
         pre_defined_item);
REFERENCE FROM geometry_schema
        (axis2_placement,
         curve,
         geometric_representation_item
         );
REFERENCE FROM measure_schema
        (length_measure,
         positive_length_measure,
         positive_ratio_measure,
         ratio_measure);
REFERENCE FROM presentation_definition_schema
        (annotation_fill_area,
         symbol_representation);
REFERENCE FROM representation_schema
        (item_in_context,
         representation);
REFERENCE FROM support_resource_schema
        (identifier,
         lable,
         text);
(*
```

Примечания

1 Схемы, на которые выше даны ссылки, можно найти в следующих стандартах:

external_reference_schema	ГОСТ Р ИСО 10303-41;
geometry_schema	ИСО 10303-42;
management_resource_schema	ГОСТ Р ИСО 10303-41;
measure_schema	ГОСТ Р ИСО 10303-41;
presentation_definition_schema	раздел 5 настоящего стандарта;
representation_schema	ГОСТ Р ИСО 10303-43;
support_resource_schema	ГОСТ Р ИСО 10303-41.

2 Графическое представление определяемой схемы приведено в приложении Е с использованием нотации языка EXPRESS-G.

7.1 Введение

Схема **presentation_resource_schema** определяет основные ресурсы для реализации представления. В данной схеме заданы три типа соответствующей информации:

- ресурсы шрифтов текста;
- ресурсы определения цвета;
- геометрические ресурсы.

Настоящая схема определяет ресурсы, необходимые для конструирования шрифтов символов и аннотационных знаков (например, мнемограмм). Данные шрифты задают в локальной системе координат. Они могут быть масштабированы и преобразованы в соответствии с прикладными задачами.

Существуют два типа ресурсов определения цвета. Первый, прямо определяемый на основе цветовой модели RGB, второй — реализующий таблицу отображения цвета, связывающую изменение цвета с соответствующей шкалой.

Геометрическими ресурсами являются геометрические элементы, применяемые в настоящем стандарте для обеспечения различных элементов конструкции изображения.

7.2 Определения типов схемы ресурсов представления

7.2.1 Тип staircase_or_linear

Тип **staircase_or_linear** определяет метод интерполяции цветов в объекте **colour_association_table**.

EXPRESS-спецификация

```
*)
TYPE staircase_or_linear = ENUMERATION OF
    (staircase,
     linear);
END_TYPE;
*)
```

Определения элементов перечисления

staircase — цвета, интерполируемые посредством «эскалаторной» функции;

linear — цвета, интерполируемые линейно.

7.2.2 Тип presentable_text

Объектом **presentable_text** является любая представляемая строка.

EXPRESS-спецификация

```
*)
TYPE presentable_text = STRING;
END_TYPE;
*)
```

Неформальное утверждение

IP1 — конкретная строка не должна содержать управляющих символов.

Пример 14 — В соответствии с **IP1** в объекте **presentable_text** запрещено использование символов «перевод строки» и «возврат каретки».

7.2.3 Тип font_select

Тип **font_select** используют для определения объектов **text_literal** и **character_glyph_symbol**. Данный тип позволяет выбрать один из исходных шрифтов текста.

EXPRESS-спецификация

```
*)
TYPE font_select = SELECT;
    (pre_defined_text_font
     externally_defined_text_font);
END_TYPE;
*)
```

7.3 Определения объектов схемы ресурсов представления

7.3.1 Объект character_glyph_symbol

Объект **character_glyph_symbol** содержит геометрическое отображение символа.

EXPRESS-спецификация

```
*)
ENTITY character_glyph_symbol
```

```
    SUBTYPE OF (symbol_representation);
    character_box       : planar_extent;
    baseline_ratio      : ratio_measure;
DERIVE
    box_height : length_measure      := character_box.size_in_y;
WHERE
    WR1: {0.0 <= baseline_ratio <= 1.0};
    WR2: item_in_context(SELF.character_box,
                         SELF\representation.context_of_items);
    WR3: 'MEASURE_SCHEMA.POSITIVE_LENGTH_MEASURE'
         IN TYPEOF (SELF.box_height);
END_ENTITY;
(*
```

Определения атрибутов

character_box — прямоугольный «ящик», ограничивающий область глифа символа. В данном атрибуте может быть задан наклон глифов символов посредством объекта **text_style**;

baseline_ratio — положение опорной линии глифа символа относительно атрибута **character_box**. Когда глифы символов представляются в форме текстового литерала, опорные линии смежных глифов выстраивают по одной строке. Опорная линия символа параллельна оси *x* атрибута **character_box**. Расстояние между осью *x* атрибута **character_box** и опорной линией пропорционально значению атрибута **box_height**.

Примечание — Ось *x* и площадь «ящика» символа задают посредством объекта **planar_extent** (см. 7.3.17).

Пример 15 — Значение 0,0 атрибута **baseline_ratio** указывает, что опорная линия глифа символа и ось *x* совпадают, а значение 0,5 — что опорная линия расположена посередине высоты глифа, заданной для атрибута **character_box**;

box_height — высота атрибута **character_box**.

Формальные утверждения

WR1 — значения атрибута **baseline_ratio** должны располагаться в диапазоне 0,0—1,0;

WR2 — атрибут **character_box** должен быть представлен в контексте объекта **text_symbol**;

WR3 — атрибут **box_height** должен иметь тип **positive_length_measure**.

Примечание — На рисунке 25 показаны типы и описания знаков глифов символов.

7.3.2 Объект character_glyph_symbol_stroke

Объектом **character_glyph_symbol_stroke** является объект **character_glyph_symbol**, в котором геометрия глифа описана набором кривых.

EXPRESS-спецификация

```
*)
ENTITY character_glyph_symbol_stroke
    SUBTYPE OF (character_glyph_symbol);
    strokes     : SET [1:?] OF curve;
WHERE
    WR1: SELF.strokes SELF\representation_items;
END_ENTITY;
(*
```

Определение атрибута

strokes — набор объектов **curve**, определяющих геометрию глифа символа.

Формальное утверждение

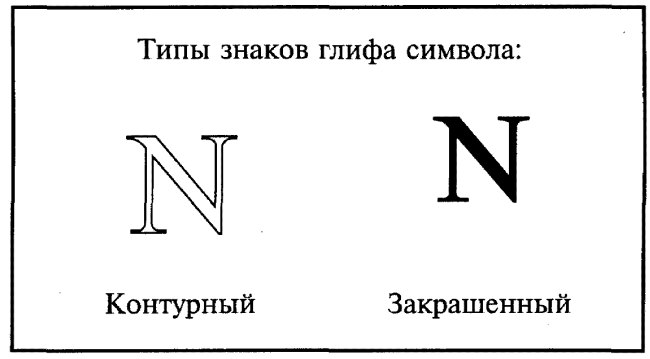
WR1 — все кривые, образующие глиф символа, должны входить в набор соответствующих элементов.

7.3.3 Объект character_glyph_symbol_outline

Объектом **character_glyph_symbol_outline** является объект **character_glyph_symbol**, в котором описана геометрия глифа через набор объектов **annotation_fill_area**.

EXPRESS-спецификация

```
*)
```



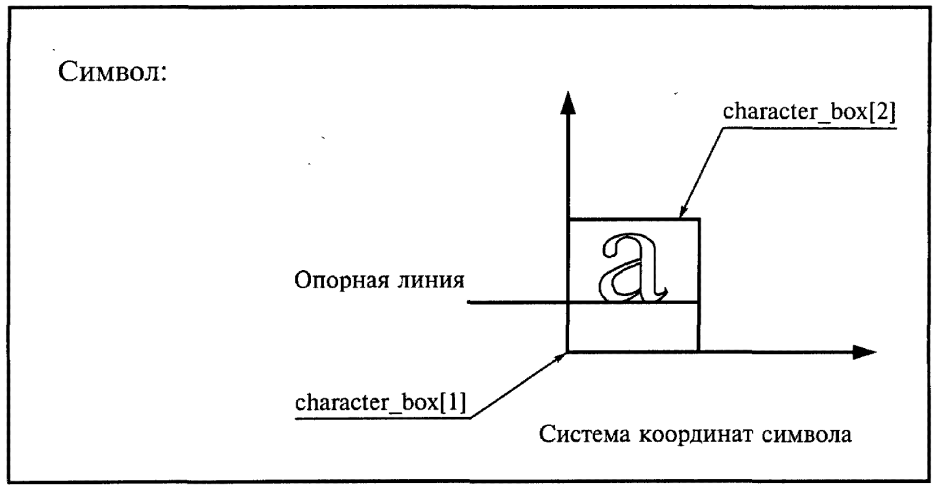


Рисунок 25 — Знаки глифа символа

```
ENTITY character_glyph_symbol_outline
    SUBTYPE OF (character_glyph_symbol);
    outlines      : SET [1:?] OF annotation_fill_areas;
WHERE
    WR1: SELF.outlines SELF\representation_items;
END_ENTITY;
(*
```

Определение атрибута

outlines — набор объектов **annotation_fill_area**, определяющих геометрию глифа символа.

Формальное утверждение

WR1 — Все заполняемые области, образующие глиф символа, должны входить в набор соответствующих элементов.

7.3.4 Объект character_glyph_font_usage

Объект **character_glyph_font_usage** определяет наличие объекта **character_glyph_symbol** в объекте **text_font**.

EXPRESS-спецификация

```
*)
ENTITY character_glyph_font_usage
    character     : character_glyph_symbol;
```

```
    font        : text_font;
END_ENTITY;
(*
```

Определения атрибутов

character — объект **character_glyph_symbol**, являющийся частью конкретного шрифта;

font — объект **text_font,** для которого задан **character_glyph_symbol**.

7.3.5 Объект text_font

Объект **text_font** определяет конкретный шрифт символа.

EXPRESS-спецификация

```
*)
ENTITY text_font
    id              : identifier;
    name            : lable;
    description     : text;
INVERSE
    glyphs      : SET [1:?] OF character_glyph_font_usage FOR font;
END_ENTITY;
(*
```

Определения атрибутов

id — обозначение объекта **text_font**;

name — слово или группа слов, посредством которых ссылаются на **text_font**;

description — текст, описывающий характер объекта **text_font**;

glyphs — набор объектов **character_glyph_font_usage**, определяющий объект **character_glyph_-symbol**, входящий в конкретный объект **text_font**.

Пример 16 — Примерами шрифтов символов являются Courier 12, Times 10, Helvetica Bold 14, 0815 и ABC.

7.3.6 Объект text_font_family

Объект **text_font_family** обозначает соответствующий набор (коллекцию) объектов **text_font**.

EXPRESS-спецификация

```
*)
ENTITY text_font_family
    id              : identifier;
    name            : lable;
    description     : text;
INVERSE
    fonts       : SET [1:?] OF text_font_in_familiy FOR family;
END_ENTITY;
(*
```

Определения атрибутов

id — обозначение объекта **text_font_family**;

name — слово или группа слов, посредством которых ссылаются на **text_font_family**.

description — текст, описывающий характер объекта **text_font_family**;

fonts — набор объектов **text_font_in_family**, определяющий объекты **texts_font**, входящие в объект **text_font_family**.

7.3.7 Объект text_font_in_family

Объект **text_font_in_family** определяет вхождение объекта **text_font** в объект **text_font_family**.

EXPRESS-спецификация

```
*)
ENTITY text_font_in_family
    font        : text_font;
    family      : text_font_family;
END_ENTITY;
(*
```

Определения атрибутов

font — объект **text_font**, входящий в соответствующее семейство;

family — объект **text_font_family**, для которого задан соответствующий **text_font**.

Пример 17 — Семействами шрифтов, например, являются Courier и Helvetica. Данные семейства включают соответственно шрифты Courier 12 и Helvetica Bold 14.

7.3.8 Объект externally_defined_text_font

Объект **externally_defined_text_font** определяет внешнюю ссылку на шрифт текста.

EXPRESS-спецификация

```
*)
ENTITY externally_defined_text_font
   SUBTYPE OF (externally_defined_item);
END_ENTITY;
(*
```

7.3.9 Объект pre_defined_text_font

Объектом **pre_defined_text_font** является шрифт текста, используемый при определении прикладного шрифта.

Примечание — Порядок применения данного объекта задают в прикладных ресурсах или протоколах.

EXPRESS-спецификация

```
*)
ENTITY pre_defined_text_font
   SUBTYPE OF (pre_defined_item);
END_ENTITY;
(*
```

7.3.10 Объект colour

Объект **colour** определяет основное свойство конкретного элемента, связанное с его способностью отражать сетовой поток.

EXPRESS-спецификация

```
*)
ENTITY colour
END_ENTITY;
(*
```

7.3.11 Объект colour_specification

Объект **colour_specification** определяет цвет, прямо связанный с заданным цветовым интервалом (пространством).

Пример 18 — Цветовыми пространствами являются RGB, HLS, HSV и CIE. Более подробно эти вопросы рассмотрены в [7].

Примечание — В настоящем стандарте использованы только цвета модели RGB. Данная модель может быть расширена.

EXPRESS-спецификация

```
*)
ENTITY colour_specification
   SUBTYPE OF (colour);
   name      : lable;
END_ENTITY;
(*
```

Определение атрибута

name — слово или группа слов, посредством которых ссылаются на **colour_specification**.

7.3.12 Объект colour_rgb

Объект **colour_rgb** определяет цвет посредством задания яркости красного, зеленого и синего цветов.

EXPRESS-спецификация

```
*)
```

```
ENTITY colour_rgb
    SUBTYPE OF (colour_specification);
    red     : REAL;
    green   : REAL;
    blue    : REAL;
WHERE
    WR1: {0.0 <= red <= 1.0};
    WR2: {0.0 <= green <= 1.0};
    WR3: {0.0 <= blue <= 1.0};
END_ENTITY;
(*
```

Определения атрибутов

red — яркость красного компонента цвета;

green — яркость зеленого компонента цвета;

blue — яркость синего компонента цвета.

Формальные утверждения

WR1 — яркость красного компонента цвета должна иметь значение от 0,0 до 1,0;

WR2 — яркость зеленого компонента цвета должна иметь значение от 0,0 до 1,0;

WR3 — яркость синего компонента цвета должна иметь значение от 0,0 до 1,0.

7.3.13 Объект colour_associated

Объект **colour_associated** определяет цвет для визуализации одномерных статических переменных при создании изображения. Цвет кривых или поверхностей в конкретных положениях задают значениями статических переменных и объектом **colour_association_table**. Положения (кривых или поверхностей) с выбранными цветами зависят от объектов **shading_curve_method** или **shading_surface_method**, заданных соответственно для кривой или поверхности. Выбранные цвета интерполируют соответственно атрибуту вида метода затенения.

Примечания

1 Статические переменные часто используют для задания физических величин.

2 Интерполяция цветов кривых и поверхностей описана в 6.3.17 и 6.3.21.

EXPRESS-спецификация

```
*)
ENTITY colour_associated
    SUBTYPE OF (colour);
    name                    : label;
    variable_to_be_shown    : SET [1:?]  OF REAL;
    mapping                 : colour_association_table;
END_ENTITY;
(*
```

Определения атрибутов

name — слово или группа слов, посредством которых ссылаются на **colour_associated**;

variable_to_be_shown — набор статических переменных, имеющих одномерные значения, предназначенный для визуализации цветов;

mapping — объект **colour_association_table**, используемый для выбора конкретного цвета.

7.3.14 Объект colour_association_table

Объект **colour_association_table** определяет отображение пространства одномерных статических переменных в пространство цветов.

Объект **colour_association_table** содержит значения статических переменных, связанных с заданием цветов. Если конкретное значение статической переменной находится между двумя заданными значениями статических переменных, тогда конкретный цвет должен быть вычислен с использованием методов ступенчатой или линейной интерполяции.

EXPRESS-спецификация

```
*)
ENTITY colour_association_table
    discrete_states_with_colours    : LIST [1:?]  OF state_variable_with_colour;
    interpolation_type              : straircase_or_linear;
```

```
END_ENTITY;
(*
```

Определения атрибутов

discrete_states_with_colours — список значений одномерных статических переменных, соответствующих определенным цветам;

interpolation_type — используемый метод интерполяции.

7.3.15 Объект state_variable_with_colour

Объект **state_variable_with_colour** связывает одну заданную статическую переменную с объектом **colour_specification**.

EXPRESS-спецификация

```
*)
ENTITY state_variable_with_colour
    state_variable          : REAL;
    associated_colour       : colour_specification;
END_ENTITY;
(*
```

Определения атрибутов

state_variable — значение одномерной статической переменной (объект **state_variable**).

Пример 19 — Физическими статическими переменными являются температура и компоненты нагрузки;

associated_colour — объект **colour_specification**, связанный с объектом **state_variable**.

7.3.16 Объект pre_defined_colour

Объект **pre_defined_colour** предназначен для обеспечения возможности задания цветов в конкретном приложении.

Примечание — Порядок применения данного объекта задают в прикладных ресурсах или протоколах. Объект **pre_defined_colour** позволяет устанавливать значения цветов или их компонентов в прикладных ресурсах или протоколах.

EXPRESS-спецификация

```
*)
ENTITY pre_defined_colour
    SUBTYPE OF (pre_defined_item, colour);
END_ENTITY;
(*
```

7.3.17 Объект planar_extent

Объект **planar_extent** определяет протяженность по обоим направлениям двумерной системы координат.

EXPRESS-спецификация

```
*)
ENTITY planar_extent
    SUBTYPE OF (geometric_representation_item);
    size_in_x       : length_measure;
    size_in_y       : length_measure;
END_ENTITY;
(*
```

Определения атрибутов

size_in_x — протяженность в направлении оси x;

size_in_y — протяженность в направлении оси y.

7.3.18 Объект planar_box

Объект **planar_box** определяет произвольный прямоугольник («ящик») и его расположение в двумерной декартовой системе координат.

EXPRESS-спецификация

```
*)
ENTITY planar_box
    SUBTYPE OF (planar_extent);
```

```
    placement       : axis2_placement;
END_ENTITY;
(*
```

Определение атрибута

placement — положение и ориентация нижнего левого угла прямоугольника. Длину сторон прямоугольника в направлении осей *x* и *y* задают посредством атрибутов супертипа.

7.3.19 Объект presentation_scaled_placement

Объектом **presentation_scaled_placement** является объект **geometric_representation_item**, обеспечивающий определение объекта **graphical_transformations**.

Примечание — Объект **graphical_transformation** описан в 4.4.12.

EXPRESS-спецификация

```
*)
ENTITY presentation_scaled_placement
    SUBTYPE OF (geometric_representation_item);
    placement       : axis2_placement;
    scaling         : positive_ratio_measure;
END_ENTITY;
(*
```

Определения атрибутов

placement — местоположение, используемое при определении поворота и перемещения;

scaling — масштабирование, являющееся частью преобразования.

EXPRESS-спецификация

```
*)
END_SCHEMA; - - presentation_resource_schema
(*
```

ПРИЛОЖЕНИЕ А
(обязательное)

Сокращенные наименования объектов

Таблица А.1 содержит сокращенные наименования объектов, установленных в настоящем стандарте. Требования по использованию сокращенных наименований содержатся в методах реализации, входящих в стандарты серии ГОСТ Р ИСО 10303.

Т а б л и ц а А.1 — Сокращенные наименования объектов

Наименование объекта	Сокращенное наименование
ANNOTATION_CURVE_OCCURRENCE	ANCROC
ANNOTATION_FILL_AREA	ANFLAR
ANNOTATION_FILL_AREA_OCCURRENCE	AFAO
ANNOTATION_OCCURRENCE	ANNOCC
ANNOTATION_OCCURRENCE_RELATIONSHIP	ANOCRL
ANNOTATION_POINT_OCCURRENCE	ANPNOC
ANNOTATION_SYMBOL	ANNSYM
ANNOTATION_SYMBOL_OCCURRENCE	ANSYOC
ANNOTATION_TABLE	ANNTBL
ANNOTATION_TABLE_OCCURRENCE	ANTBOC
ANNOTATION_TEXT	ANNTXT
ANNOTATION_TEXT_CHARACTER	ANTXCH
ANNOTATION_TEXT_OCCURRENCE	ANTXOC
ANNOTATION_TEXT_WITH_ASSOCIATED_CURVES	ATWAC
ANNOTATION_TEXT_WITH_BLANKING_BOX	ATWBB
ANNOTATION_TEXT_WITH_DELINEATION	ATWD
ANNOTATION_TEXT_WITH_EXTENT	ATWE
APPROXIMATION_TOLERANCE	APPTLR
APPROXIMATION_TOLERANCE_DEVIATION	APTLDV
APPROXIMATION_TOLERANCE_PARAMETER	APTLPR
AREA_DEPENDENT_ANNOTATION_REPRESENTATION	ADAR
AREA_IN_SET	ARINST
BACKGROUND_COLOUR	BCKCLR
CAMERA_IMAGE	CMRIMG
CAMERA_MODEL	CMRMDL
CAMERA_MODEL_D2	CMMDD2
CAMERA_MODEL_D2_SHAPE_CLIPPING	CMDSC
CAMERA_MODEL_D3	CMMDD3
CAMERA_MODEL_D3_MULTI_CLIPPING	CMDMC
CAMERA_MODEL_D3_WITH_HLHSR	CMDWH

Продолжение таблицы А.1

Наименование объекта	Сокращенное наименование
CAMERA_MODEL_WITH_LIGHT_SOURCES	CMWLS
CAMERA_USAGE	CMRUSG
CHARACTER_GLYPH_FONT_USAGE	CGFU
CHARACTER_GLYPH_STYLE_OUTLINE	CGO
CHARACTER_GLYPH_STYLE_OUTLINE_WITH_CHARACTERISTICS	CGSOWC
CHARACTER_GLYPH_STYLE_STROKE	CGS
CHARACTER_GLYPH_SYMBOL	CHGLSY
CHARACTER_GLYPH_SYMBOL_OUTLINE	CGSO
CHARACTER_GLYPH_SYMBOL_STROKE	CGSS
COLOUR	COLOUR
COLOUR_ASSOCIATED	CLRASS
COLOUR_ASSOCIATION_TABLE	CLASTB
COLOUR_RGB	CLRRGB
COLOUR_SPECIFICATION	CLRSPC
COMPOSITE_TEXT	CMPTXT
COMPOSITE_TEXT_WITH_ASSOCIATED_CURVES	CTWAC
COMPOSITE_TEXT_WITH_BLANKING_BOX	CTWBB
COMPOSITE_TEXT_WITH_DELINEATION	CTWD
COMPOSITE_TEXT_WITH_EXTENT	CTWE
CONTEXT_DEPENDENT_INVISIBILITY	CNDPIN
CONTEXT_DEPENDENT_OVER_RIDING_STYLED_ITEM	CDORSI
CURVE_STYLE	CRVSTY
CURVE_STYLE_CURVE_PATTERN	CSCP
CURVE_STYLE_CURVE_PATTERN_SET	CSCPS
CURVE_STYLE_FONT	CRSTFN
CURVE_STYLE_FONT_AND_SCALING	CSFAS
CURVE_STYLE_FONT_PATTERN	CSFP
CURVE_STYLE_RENDERING	CRSTRH
CURVE_STYLE_WIDE	CRSTWD
CURVE_STYLE_WITH_ENDS_AND_CORNERS	CSWEAC
CURVE_STYLE_WITH_EXTENSION	CSWE
DEFINED_CHARACTER_GLYPH	DFCHGL
DEFINED_SYMBOL	DFNSYM
DEFINED_TABLE	DFNTBL
EXTERNALLY_DEFINED_CHARACTER_GLYPH	EDCG
EXTERNALLY_DEFINED_CURVE_FONT	EDCF
EXTERNALLY_DEFINED_HATCH_STYLE	EDHS
EXTERNALLY_DEFINED_STYLE	EXDFST

Продолжение таблицы А.1

Наименование объекта	Сокращенное наименование
EXTERNALLY_DEFINED_SYMBOL	EXDFSY
EXTERNALLY_DEFINED_TEXT_FONT	EDTF
EXTERNALLY_DEFINED_TILE	EXDFTL
EXTERNALLY_DEFINED_TILE_STYLE	EDTS
FILL_AREA_STYLE	FLARST
FILL_AREA_STYLE_COLOUR	FASC
FILL_AREA_STYLE_HATCHING	FASH
FILL_AREA_STYLE_TILES	FAST
FILL_AREA_STYLE_TILE_COLOURED_REGION	FASTCR
FILL_AREA_STYLE_TILE_CURVE_WITH_STYLE	FASTCW
FILL_AREA_STYLE_TILE_SYMBOL_WITH_STYLE	FASTSW
GRAPHICAL_TRANSFORMATION	GRPTRN
INVISIBILITY	INVSBL
LIGHT_SOURCE	LGHSRC
LIGHT_SOURCE_AMBIENT	LGSRAM
LIGHT_SOURCE_DIRECTIONAL	LGSRDR
LIGHT_SOURCE_POSITIONAL	LGSRPS
LIGHT_SOURCE_SPOT	LGSRSP
OCCLUSION_PRECEDENCE	OCCPRC
ONE_DIRECTION_REPEAT_FACTOR	ODRF
OVER_RIDING_STYLED_ITEM	ORSI
PLANAR_BOX	PLNBX
PLANAR_EXTENT	PLNEXT
POINT_STYLE	PNTSTY
PRESENTATION_AREA	PRSAR
PRESENTATION_LAYER_ASSIGNMENT	PRLYAS
PRESENTATION_LAYER_USAGE	PRLYUS
PRESENTATION_REPRESENTATION	PRSRPR
PRESENTATION_REPRESENTATION_RELATIONSHIP	PRRPRL
PRESENTATION_SCALED_PLACEMENT	PRSCPL
PRESENTATION_SET	PRSST
PRESENTATION_SIZE	PRSSZ
PRESENTATION_STYLE_ASSIGNMENT	PRSTAS
PRESENTATION_STYLE_BY_CONTEXT	PSBC
PRESENTATION_VIEW	PRSVW
PRESENTED_ITEM	PRSITM
PRESENTED_ITEM_REPRESENTATION	PRITRP
PRE_DEFINED_CHARACTER_GLYPH	PDCG

Продолжение таблицы А.1

Наименование объекта	Сокращенное наименование
PRE_DEFINED_CHARACTER_SPACING	PDCS
PRE_DEFINED_COLOUR	PRDFCL
PRE_DEFINED_CURVE_FONT	PDCF
PRE_DEFINED_HATCH_STYLE	PDHS
PRE_DEFINED_MARKER	PRDFMR
PRE_DEFINED_PRESENTATION_STYLE	PDPS
PRE_DEFINED_SIZE	PRDFSZ
PRE_DEFINED_SURFACE_SIDE_STYLE	PDSSS
PRE_DEFINED_SYMBOL	PRDFSY
PRE_DEFINED_TEXT_FONT	PDTF
PRE_DEFINED_TILE	PRDFTL
PRE_DEFINED_TILE_STYLE	PDTS
PRODUCT_DATA_REPRESENTATION_VIEW	PDRV
REPRESENTATION_ITEM_DEPENDENT_LAYER_ASSIGNMENT	RIDLA
STATE_VARIABLE_WITH_COLOUR	SVWC
STYLED_ITEM	STYITM
SURFACE_RENDERING_PROPERTIES	SRRNPR
SURFACE_SIDE_STYLE	SRSDST
SURFACE_STYLE_BOUNDARY	SRSTBN
SURFACE_STYLE_CONTROL_GRID	SSCG
SURFACE_STYLE_FILL_AREA	SSFA
SURFACE_STYLE_PARAMETER_LINE	SSPL
SURFACE_STYLE_REFLECTANCE_AMBIENT	SSRA
SURFACE_STYLE_REFLECTANCE_AMBIENT_DIFFUSE	SSRAD
SURFACE_STYLE_REFLECTANCE_AMBIENT_DIFFUSE_SPECULAR	SSRADS
SURFACE_STYLE_RENDERING	SRSTRN
SURFACE_STYLE_RENDERING_WITH_PROPERTIES	SSRWP
SURFACE_STYLE_SEGMENTATION_CURVE	SSSC
SURFACE_STYLE_SILHOUETTE	SRSTSL
SURFACE_STYLE_TRANSPARENT	SRSTTR
SURFACE_STYLE_USAGE	SRSTUS
SYMBOL_COLOUR	SYMCLR
SYMBOL_ELEMENT_STYLE	SYELST
SYMBOL_REPRESENTATION	SYMRPR
SYMBOL_REPRESENTATION_MAP	SYRPMP
SYMBOL_REPRESENTATION_RELATIONSHIP	SYRPRL
SYMBOL_REPRESENTATION_WITH_BLANKING_BOX	SRWBB
SYMBOL_STYLE	SYMSTY

Окончание таблицы А.1

Наименование объекта	Сокращенное наименование
SYMBOL_TARGET	SYMTRG
TABLE_RECORD_FIELD_REPRESENTATION	TRFR
TABLE_RECORD_FIELD_REPRESENTATION_WITH_CLIPPING_BOX	TRFRWC
TABLE_RECORD_REPRESENTATION	TBRCRP
TABLE_REPRESENTATION	TBLRPR
TABLE_REPRESENTATION_RELATIONSHIP	TBRPRL
TABLE_TEXT_RELATIONSHIP	TBTXRL
TEXT_FONT	TXTFNT
TEXT_FONT_FAMILY	TXFNFM
TEXT_FONT_IN_FAMILY	TFIF
TEXT_LITERAL	TXTLTR
TEXT_LITERAL_WITH_ASSOCIATED_CURVES	TLWAC
TEXT_LITERAL_WITH_BLANKING_BOX	TLWBB
TEXT_LITERAL_WITH_DELINEATION	TLWD
TEXT_LITERAL_WITH_EXTENT	TLWE
TEXT_STRING_REPRESENTATION	TXSTRP
TEXT_STYLE	TXTSTY
TEXT_STYLE_FOR_DEFINED_FONT	TSFDF
TEXT_STYLE_WITH_BOX_CHARACTERISTICS	TSWBC
TEXT_STYLE_WITH_JUSTIFICATION	TSWJ
TEXT_STYLE_WITH_MIRROR	TSWM
TEXT_STYLE_WITH_SPACING	TSWS
TWO_DIRECTION_REPEAT_FACTOR	TDRF
VIEW_DEPENDENT_ANNOTATION_REPRESENTATION	VDAR
VIEW_VOLUME	VWVLM

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(46) version(2) }

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

В.2 Обозначение схемы

В.2.1 Обозначение presentation_organisation_schema

Для обеспечения однозначного обозначения **presentation_organisation_schema** в открытой системе ей присвоен следующий идентификатор объекта:

{ iso standard 10303 part(46) version(2) object(1) presentation-organisation_schema(1) }

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

B.2.2 Обозначение presentation_definition_schema

Для обеспечения однозначного обозначения **presentation_definition_schema** в открытой системе ей присвоен следующий идентификатор объекта:

{ iso standard 10303 part(46) version(2) object(1) presentation-definition_schema(2) }

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

B.2.3 Обозначение presentation_appearance_schema

Для обеспечения однозначного обозначения **presentation_appearance_schema** в открытой системе ей присвоен следующий идентификатор объекта:

{ iso standard 10303 part(46) version(2) object(1) presentation-appearance_schema(3) }

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

B.2.4 Обозначение presentation_resource_schema

Для обеспечения однозначного обозначения **presentation_resource_schema** в открытой системе ей присвоен следующий идентификатор объекта:

{ iso standard 10303 part(46) version(2) object(1) presentation-resource_schema(4) }

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

ПРИЛОЖЕНИЕ C
(справочное)

Машинно-интерпретируемые листинги

Настоящее приложение представляет листинг (распечатку) сокращенных наименований и описаний на языке EXPRESS объектов, установленных в настоящем стандарте, а также листинги каждой EXPRESS-схемы, описанной в настоящем стандарте, без комментариев и пояснений. Эти листинги предоставляются в машинно-интерпретируемой форме и могут быть получены по следующим адресам унифицированного указателя ресурсов (URL):

Сокращенные наименования: ≤gost.e-ecolog.ru ≥

EXPRESS: ≤gost.e-ecolog.ru ≥

Если доступ по данным адресам затруднен, то материал может быть получен через Центральный секретариат ИСО или через секретариат ИСО ТК 184/ПК4 по адресу: gost.e-ecolog.ru

Примечание — Информация, представленная в машинно-ориентированной форме, является справочной; обязательным является текст, содержащийся в настоящем стандарте.

ПРИЛОЖЕНИЕ D
(справочное)

Технические вопросы

D.1 Символы, используемые в формулах отражающей способности

Таблица D.1 взята из приложения E к ИСО/МЭК 9592-4 [5].

Таблица D.1 — Описания переменных и их источники

Символ	Описание	Тип данных	Источник данных
$\vec{L}_d$	Описание источника света	3xR	(1)
L_c	Цвет источника света	COLRV	(1)
$\vec{L}_p$	Местоположение источника света	3xR	(1)
L_e	Коэффициент концентрации источника света	R	(1)

Окончание таблицы D.1

Символ	Описание	Тип данных	Источник данных
$C_1 \cdot C_2$	Коэффициенты затухания	R	(1)
A_s	Угол рассеивания	R	(1)
L_a	Затухание света	R	(2)
$\vec{O}_p$	Местоположение предмета	3xR	(3)
O_d	Рассеиваемый цвет предмета	COLRV	(4)
O_s	Отражаемый цвет предмета	COLRV	(5)
O_e	Коэффициент отражения от предмета	R	(5)
K_a	Коэффициент общего отражения	R	(5)
K_d	Коэффициент диффузионного отражения	R	(5)
K_s	Коэффициент зеркального отражения	R	(5)
$\vec{V}_e$	Единичный вектор от предмета к точке взгляда	3xR	(2)
$\vec{V}_r$	Единичный вектор отражения от предмета	3xR	(2)
$\vec{V}_l$	Единичный вектор от предмета к источнику света	3xR	(2)
$\vec{V}_n$	Единичный вектор, перпендикулярный к предмету	NORM	(3)
C_a	Общая часть от источника света	COLRV	(2)
C_d	Рассеиваемая часть от источника света	COLRV	(2)
C_s	Отражаемая часть от источника света	COLRV	(2)

Описание источника данных:
1 Отображение источника света.
2 Вычисление.
3 Заданные или полученные из геометрии предмета.
4 Таблица цветов, прямой цвет, цвет вершины, внутренний фоновый цвет.
5 Свойства поверхности или обратной стороны предмета.

D.2 Рекомендуемые формулы отражающей способности

Данные формулы взяты из приложения Е к ИСО/МЭК 9592-4 [5].

Результатом вычисления отражающей способности отдельной точки является общий цвет, определяемый суммой отдельных компонентов $C_a + C_d + C_s$ от всех действующих источников света

$$\sum_{i=1}^{N} (C_{a_i} + C_{d_i} + C_{s_i}),$$

где N — общее число источников света.

Для общих источников света:

$$C_a = K_a L_c O_d;$$
$$C_d = 0;$$
$$C_s = 0.$$

Для прямых источников света:

$$C_a = 0;$$
$$C_d = K_d L_c O_d (\vec{V}_n \cdot \vec{V}_l);$$
$$C_s = K_s O_s L_c (\vec{V}_e \cdot \vec{V}_r)^{O_e}.$$

Для позиционированных источников света:

$$C_a = 0;$$

$$C_d = K_d O_d L_c \, (\vec{V}_n \cdot \vec{V}_l) \, L_a;$$

$$C_s = K_s O_s L_c \, (\vec{V}_e \cdot \vec{V}_r)^{O_e} \, L_a \, .$$

Для местного источника света:

(Влияние местных источников света может быть нулевым, если $\vec{V}_l$ находится вне потока от конкретного источника света).

$$C_a = 0;$$

$$C_d = K_d O_d L_c \, (\vec{V}_n \cdot \vec{V}_l) \, (\vec{L}_d \cdot (-\vec{V}_l))^{L_c} \, L_a \, ;$$

$$C_s = K_s O_s L_c \, (\vec{V}_e \cdot \vec{V}_r)^{O_e} \, (\vec{L}_d \cdot (-\vec{V}_l))^{L_e} \, L_a \, .$$

Затухание света может быть вычислено по формуле

$$L_a = \frac{1}{C_1 + C_2 \left\| \vec{O}_p - \vec{L}_p \right\|}.$$

Вектор отражения света может быть вычислен по формуле

$$\vec{V}_r = 2 (\vec{V}_n \cdot \vec{V}_l) \vec{V}_n - \vec{V}_l.$$

Во всех случаях отдельные результаты, имеющие отрицательные значения, должны быть заменены нулевыми значениями.

ПРИЛОЖЕНИЕ Е
(справочное)

EXPRESS-G ДИАГРАММЫ

Рисунки, представленные в настоящем приложении, соответствуют описаниям схем настоящего стандарта. В диаграммах использована графическая нотация EXPRESS-G языка EXPRESS. Правила построения EXPRESS-G диаграмм установлены в приложении D ГОСТ Р ИСО 10303-11.

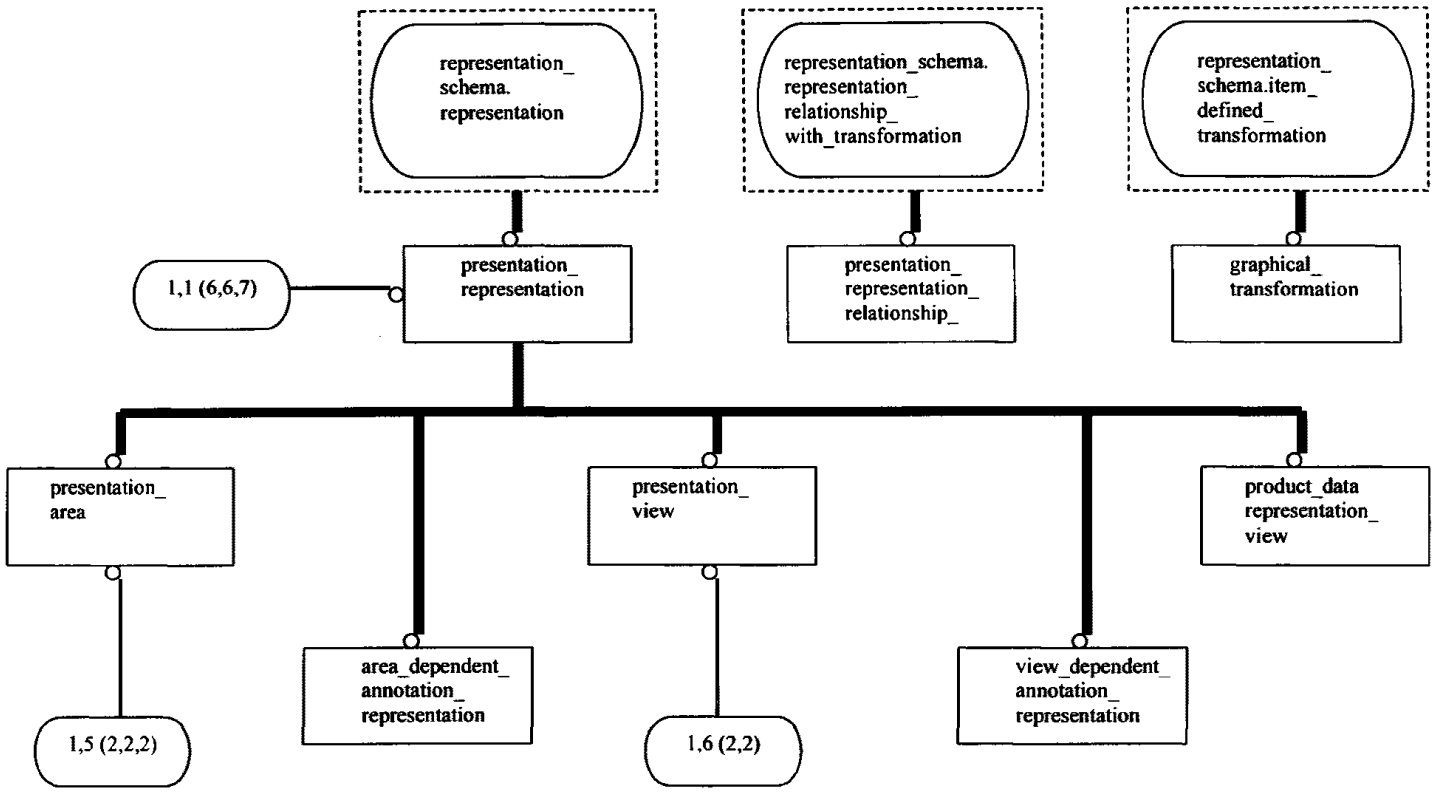


Рисунок Е.1 — EXPRESS-G диаграмма presentation_organization_schema 1 из 7

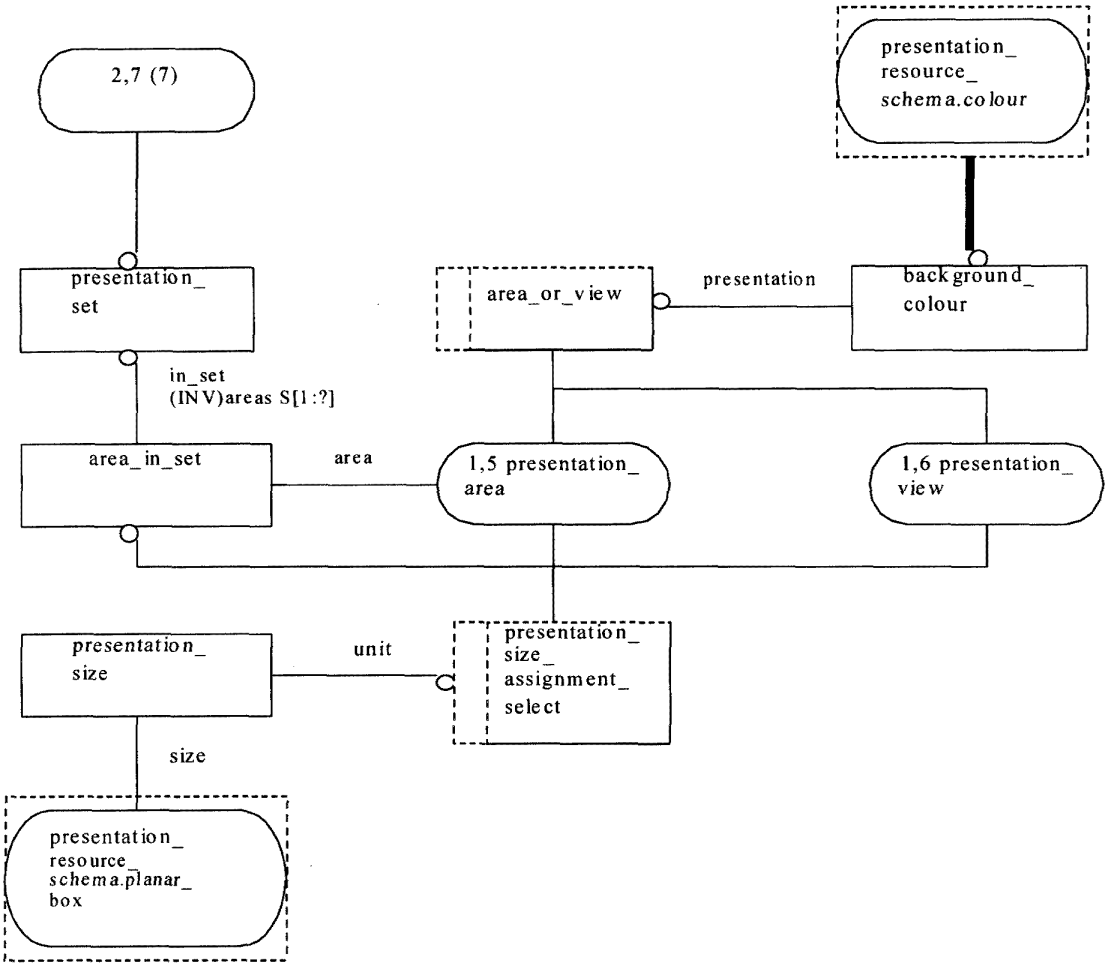


Рисунок Е.2 — EXPRESS-G диаграмма presentation_organization_schema 2 из 7

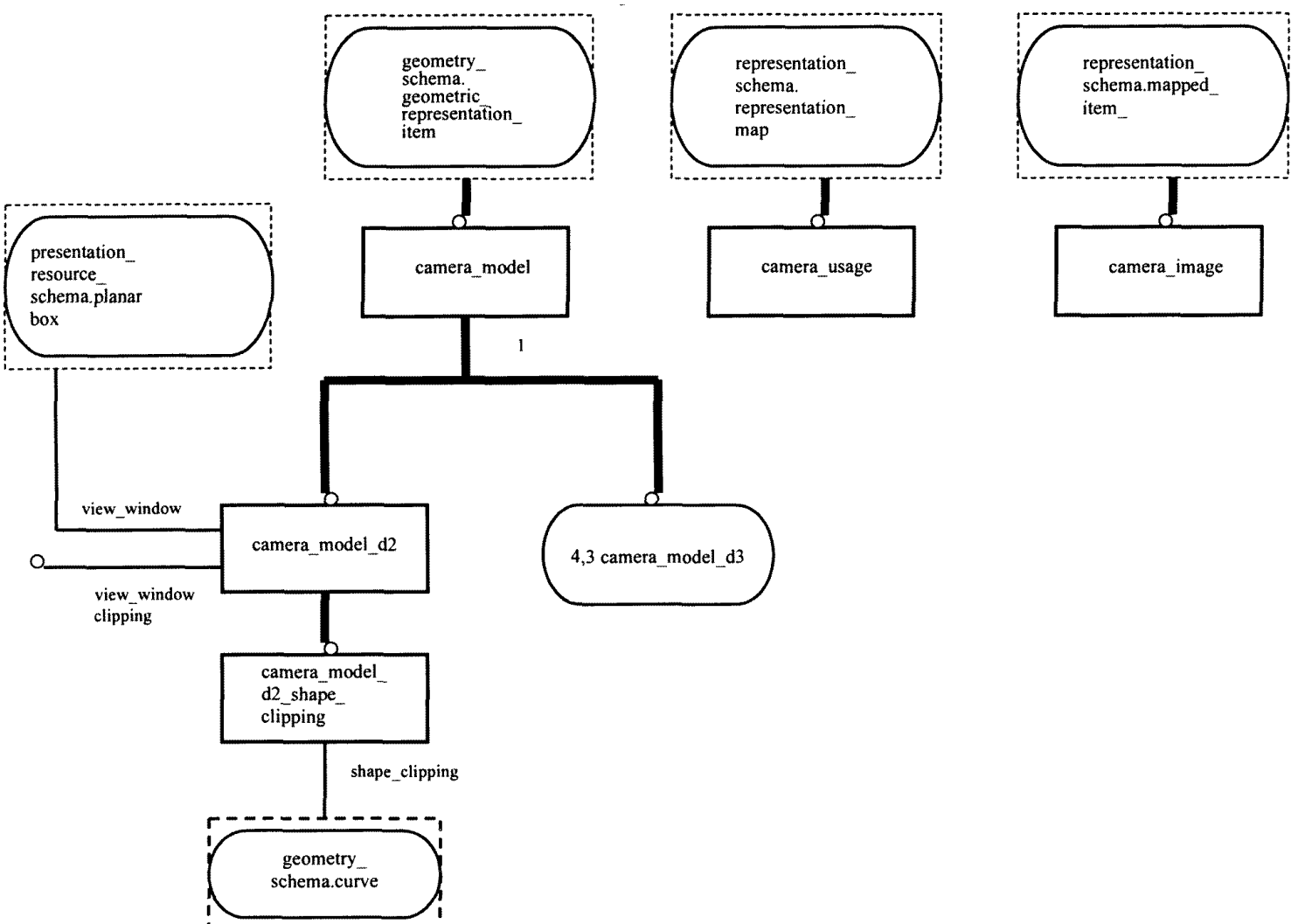


Рисунок Е.3 — EXPRESS-G диаграмма presentation_organization_schema 3 из 7

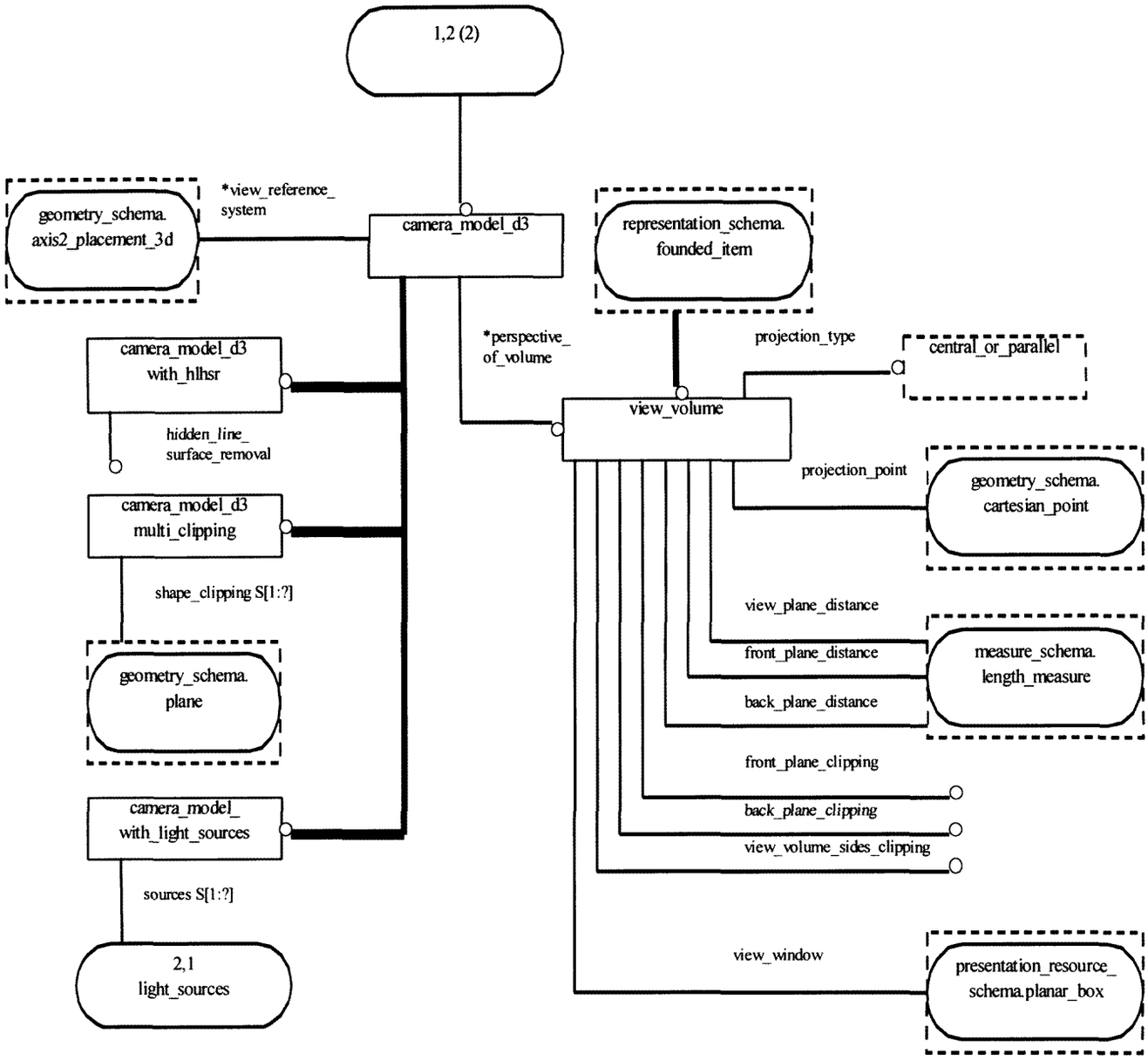


Рисунок Е.4 — EXPRESS-G диаграмма presentation_organization_schema 4 из 7

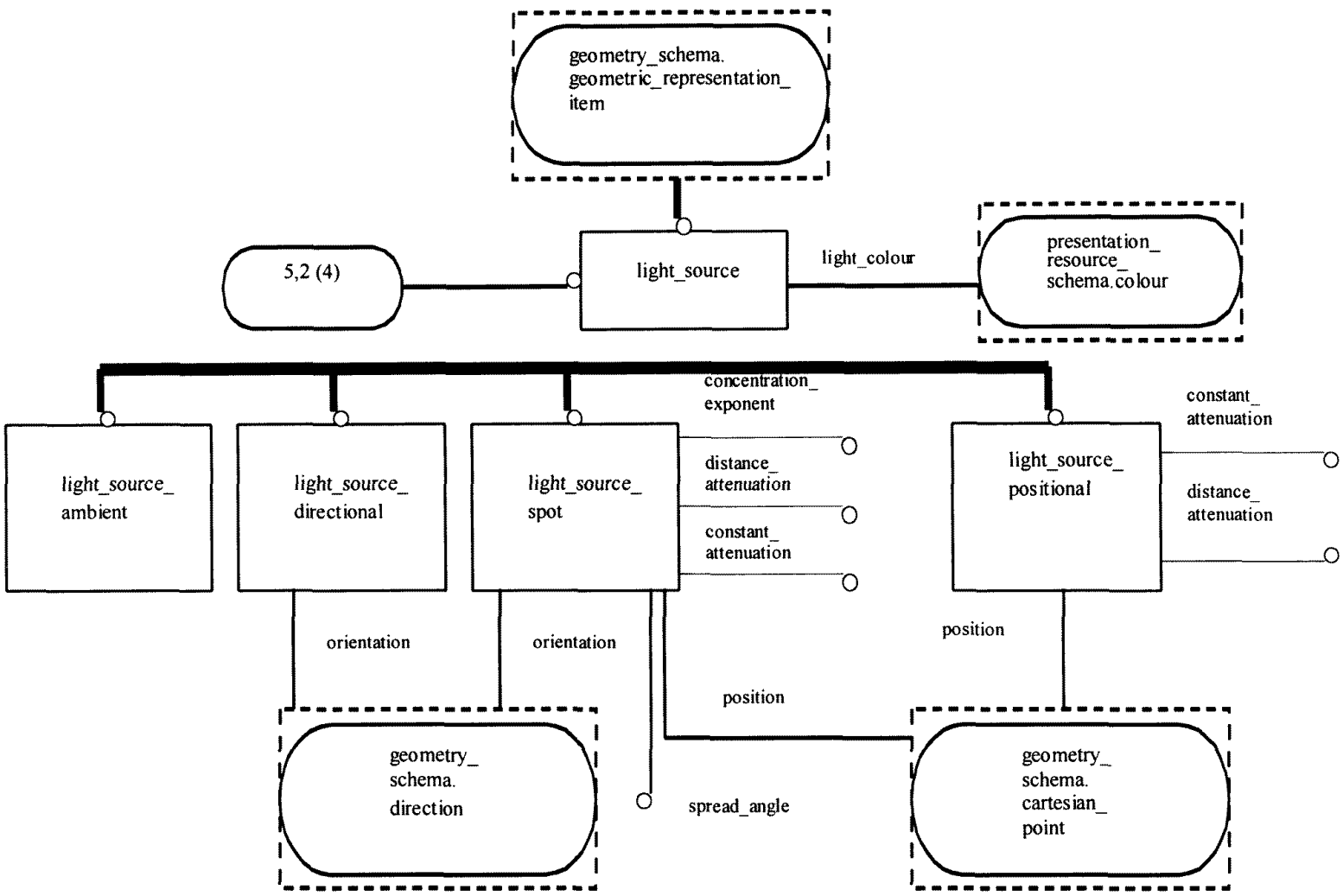


Рисунок E.5 — EXPRESS-G диаграмма presentation_organization_schema 5 из 7

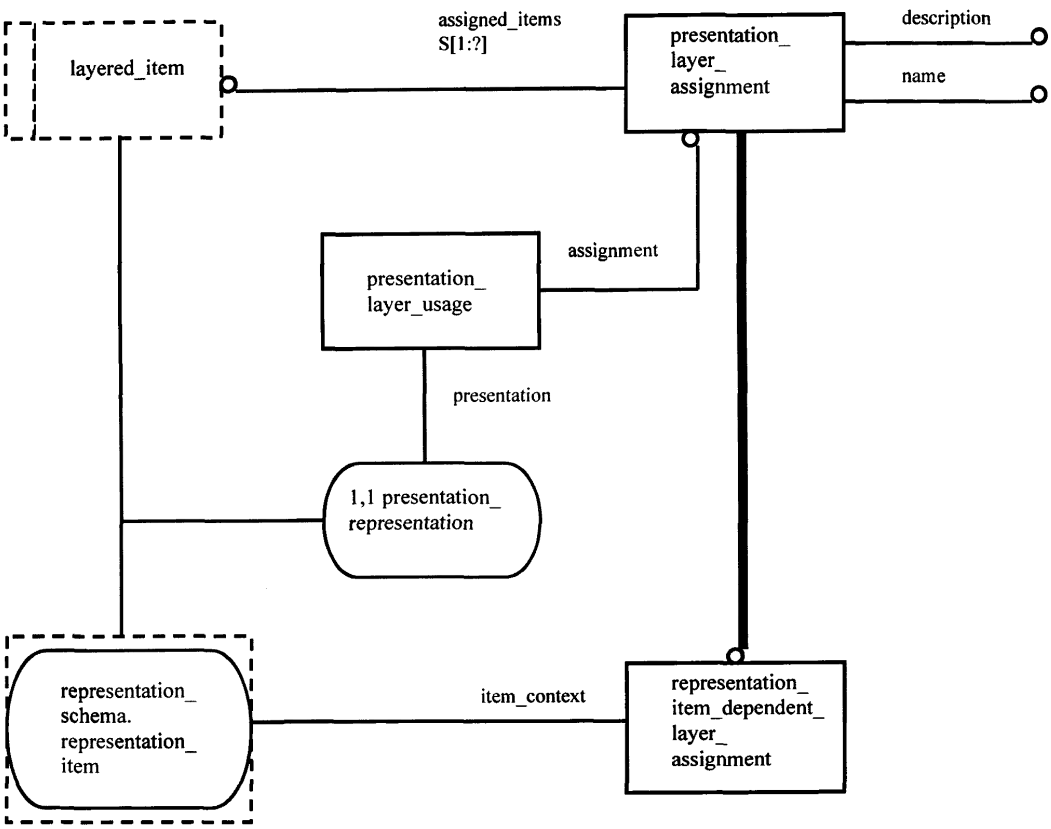


Рисунок Е.6 — EXPRESS-G диаграмма presentation_organization_schema 6 из 7

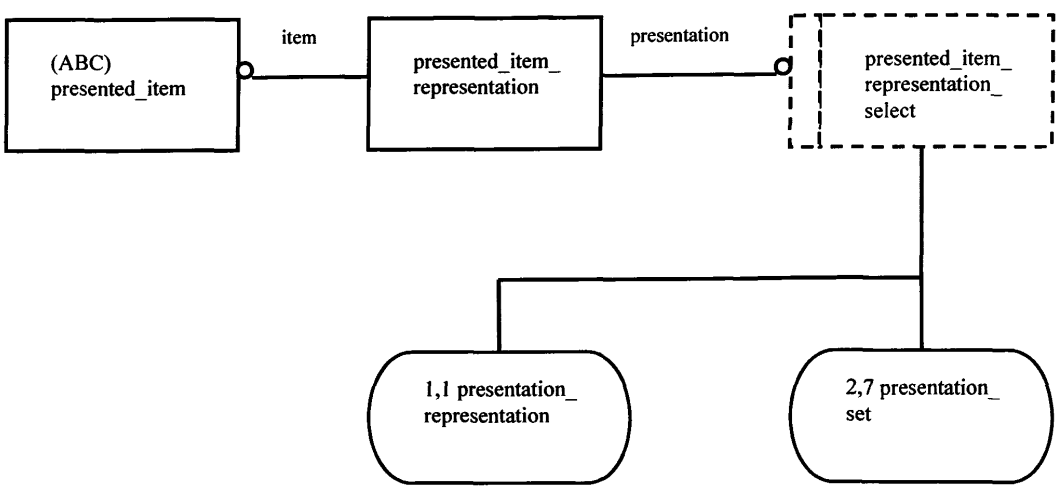


Рисунок Е.7 — EXPRESS-G диаграмма presentation_organization_schema 7 из 7

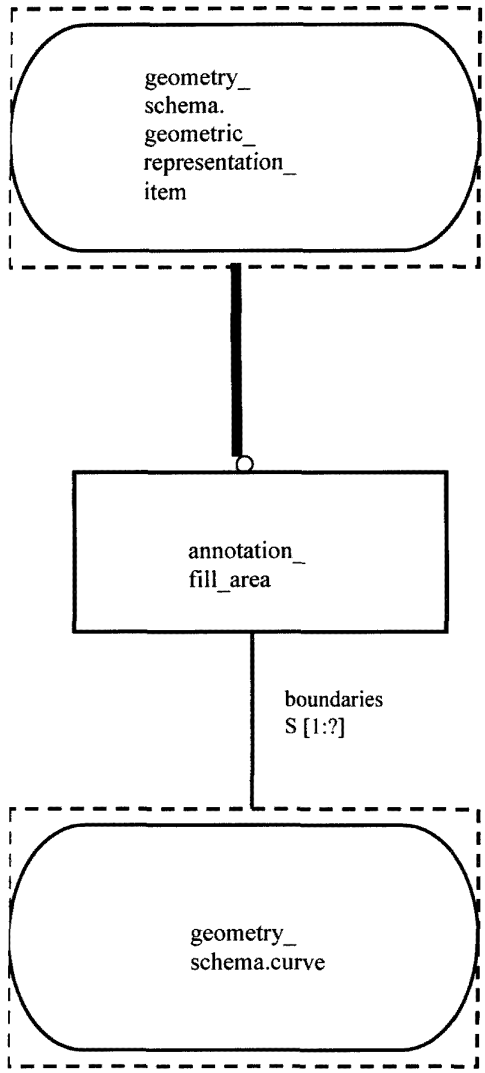


Рисунок Е.8 — EXPRESS-G диаграмма presentation_definition_schema 1 из 9

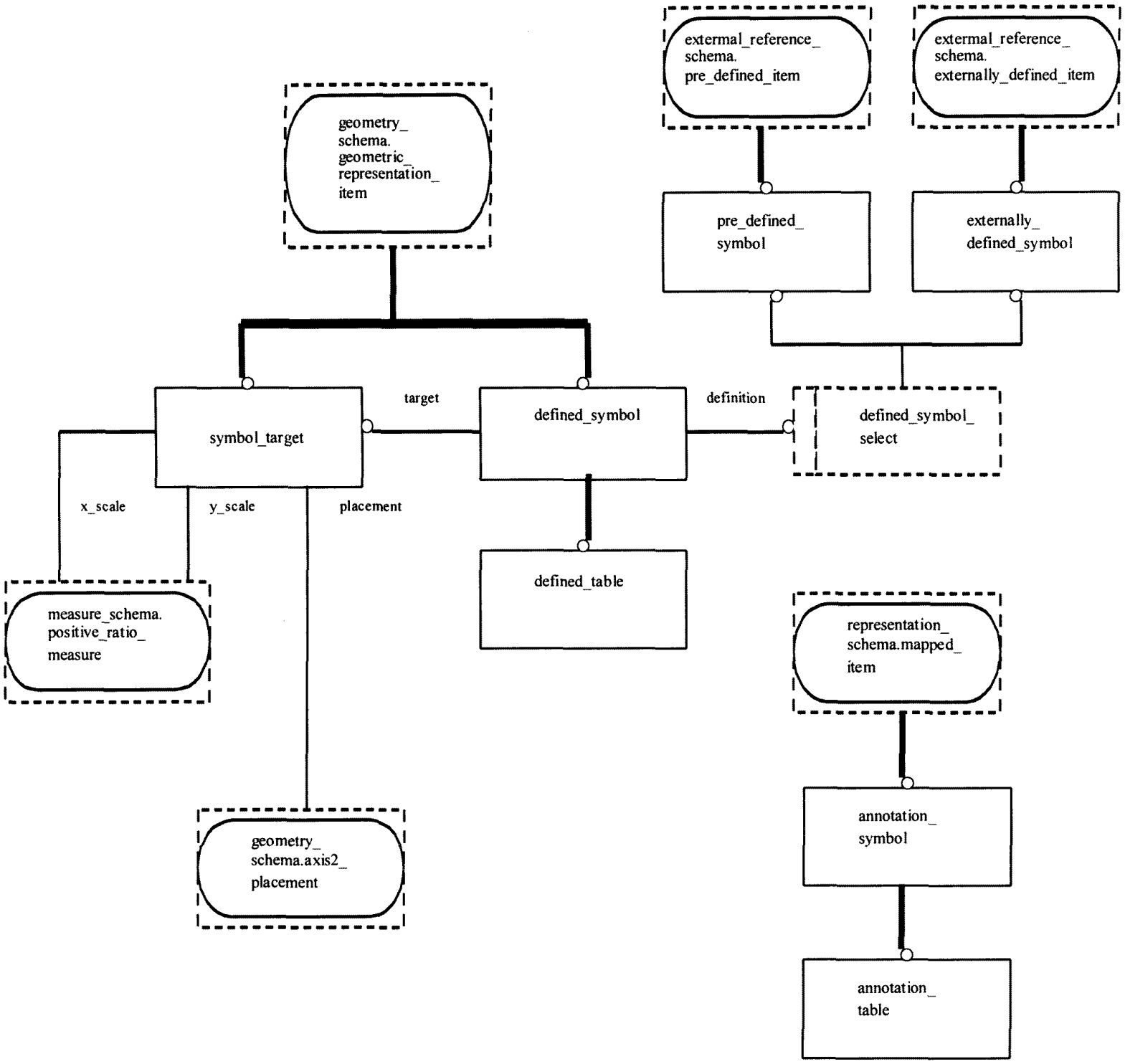


Рисунок E.9 — EXPRESS-G диаграмма presentation_definition_schema 2 из 9

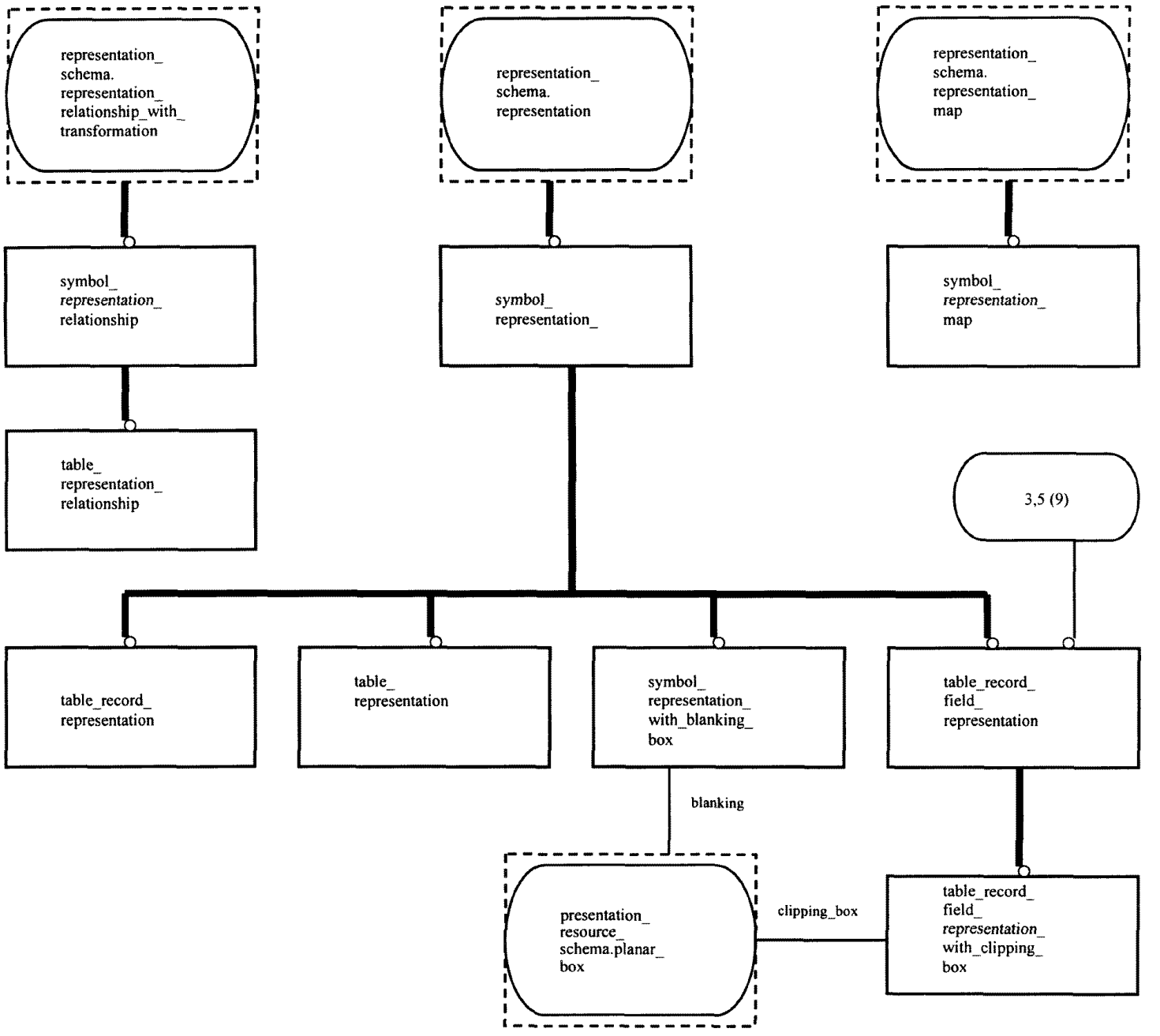


Рисунок Е.10 — EXPRESS-G диаграмма presentation_definition_schema 3 из 9

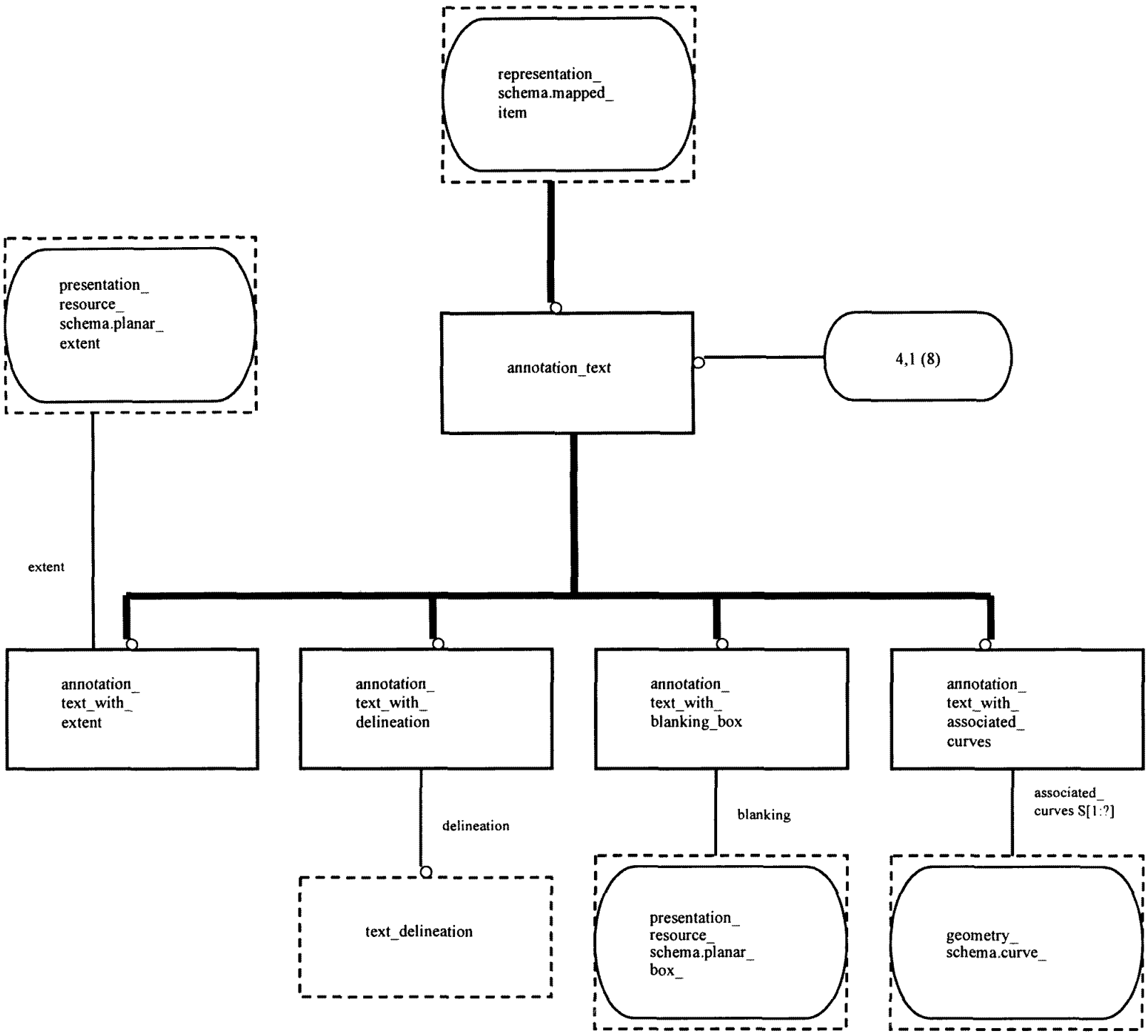


Рисунок Е.11 — EXPRESS-G диаграмма presentation_definition_schema 4 из 9

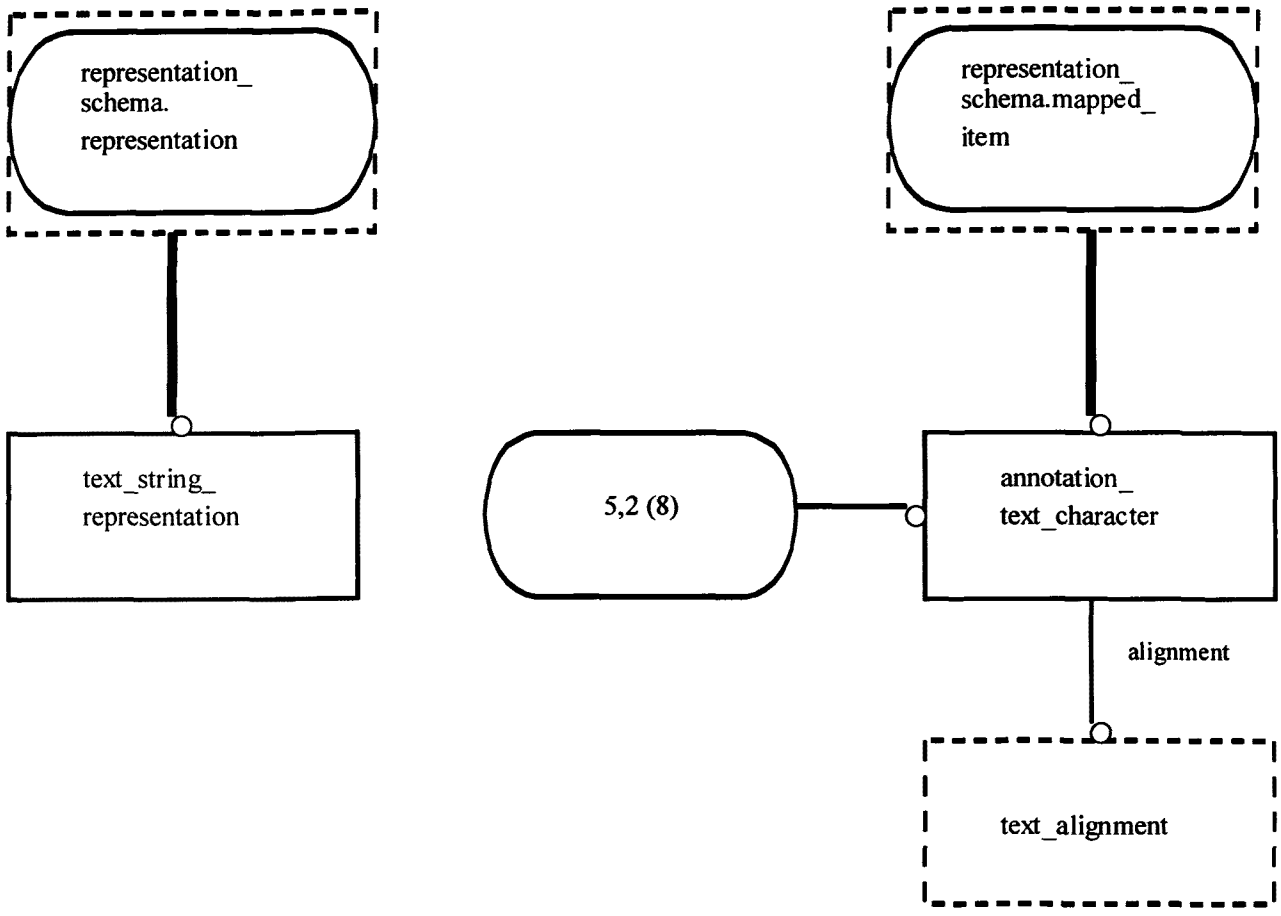


Рисунок Е.12 — EXPRESS-G диаграмма presentation_definition_schema 5 из 9

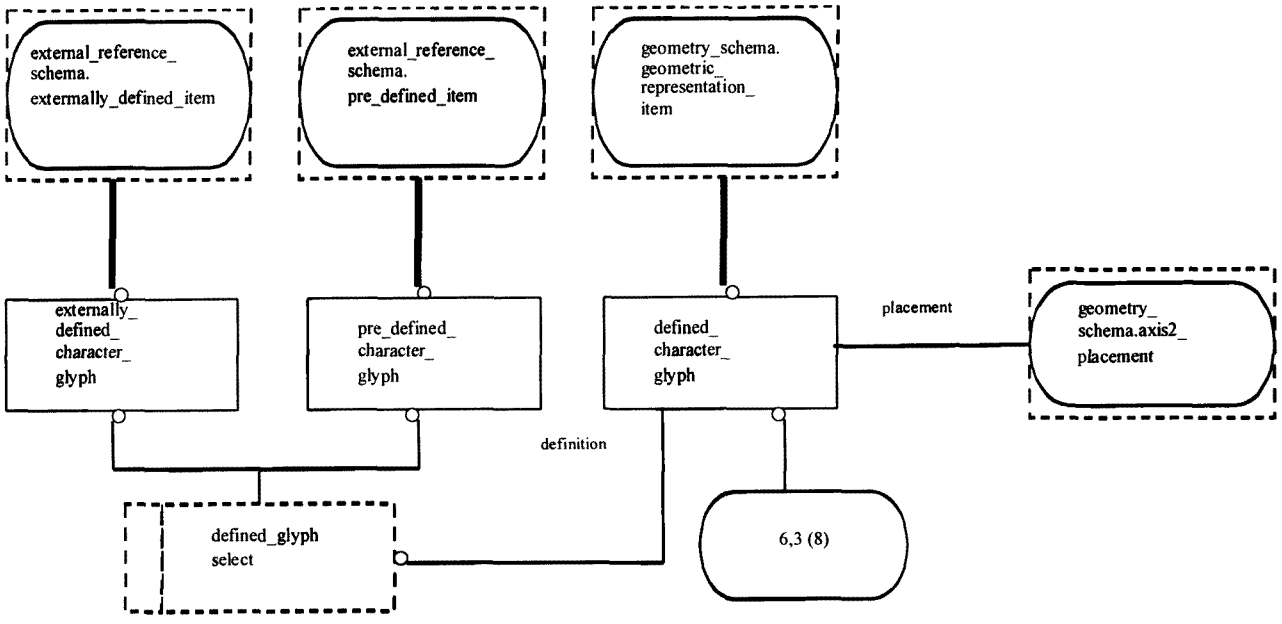


Рисунок Е.13 — EXPRESS-G диаграмма presentation_definition_schema 6 из 9

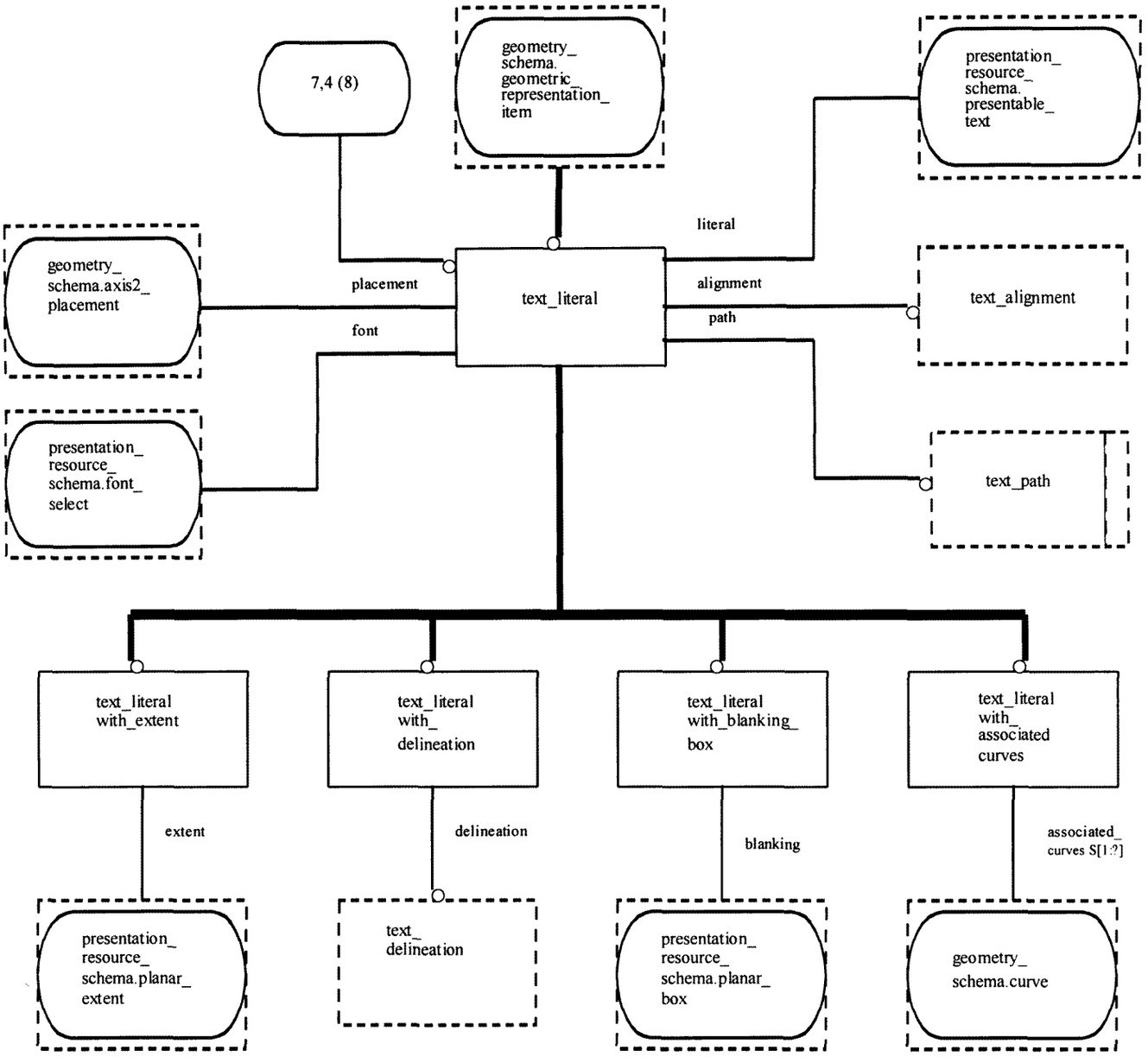


Рисунок E.14 — EXPRESS-G диаграмма presentation_definition_schema 7 из 9

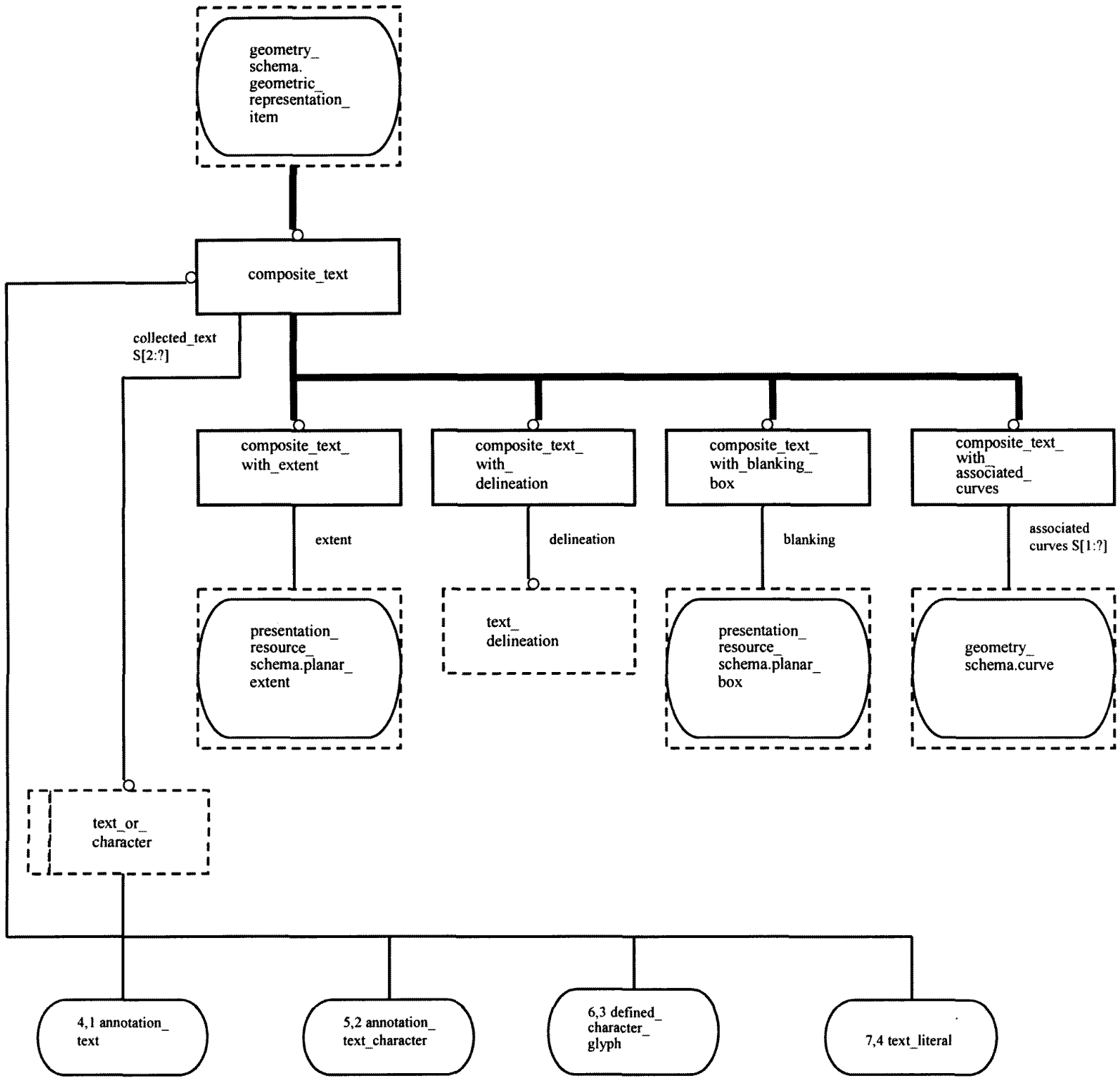


Рисунок Е.15 — EXPRESS-G диаграмма presentation_definition_schema 8 из 9

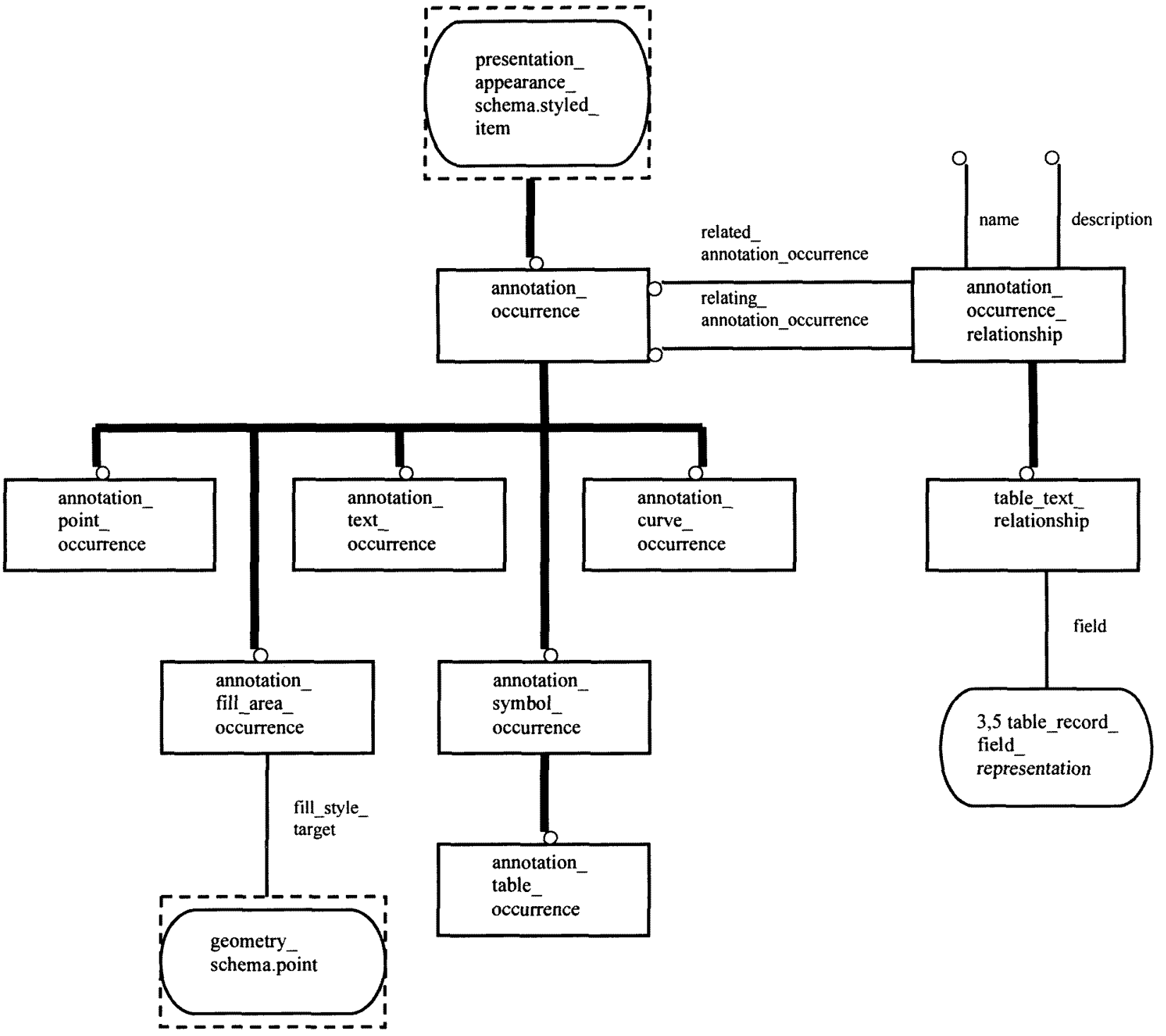


Рисунок Е.16 — EXPRESS-G диаграмма presentation_definition_schema 9 из 9

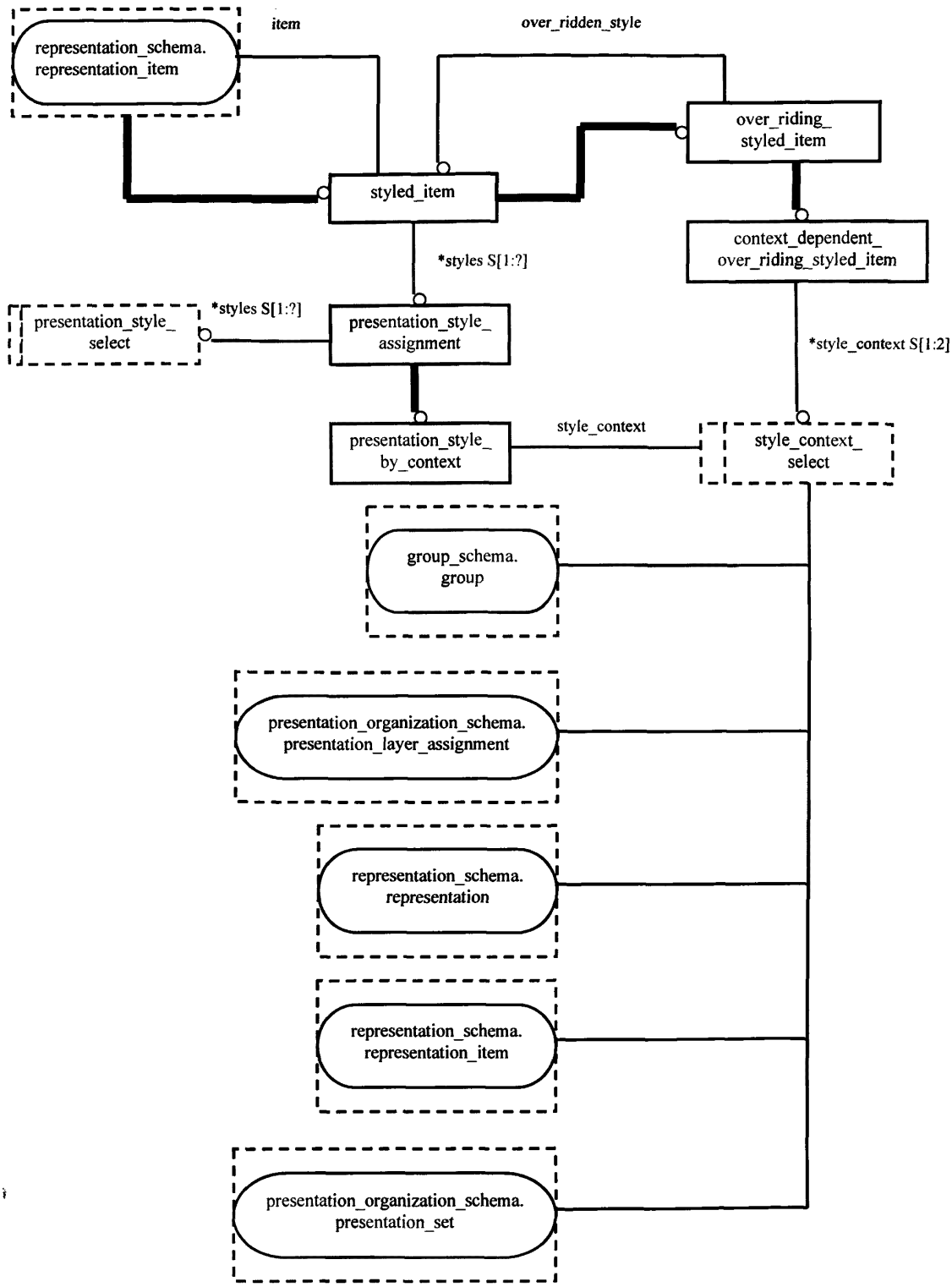


Рисунок Е.17 — EXPRESS-G диаграмма presentation_appearance_schema 1 из 21

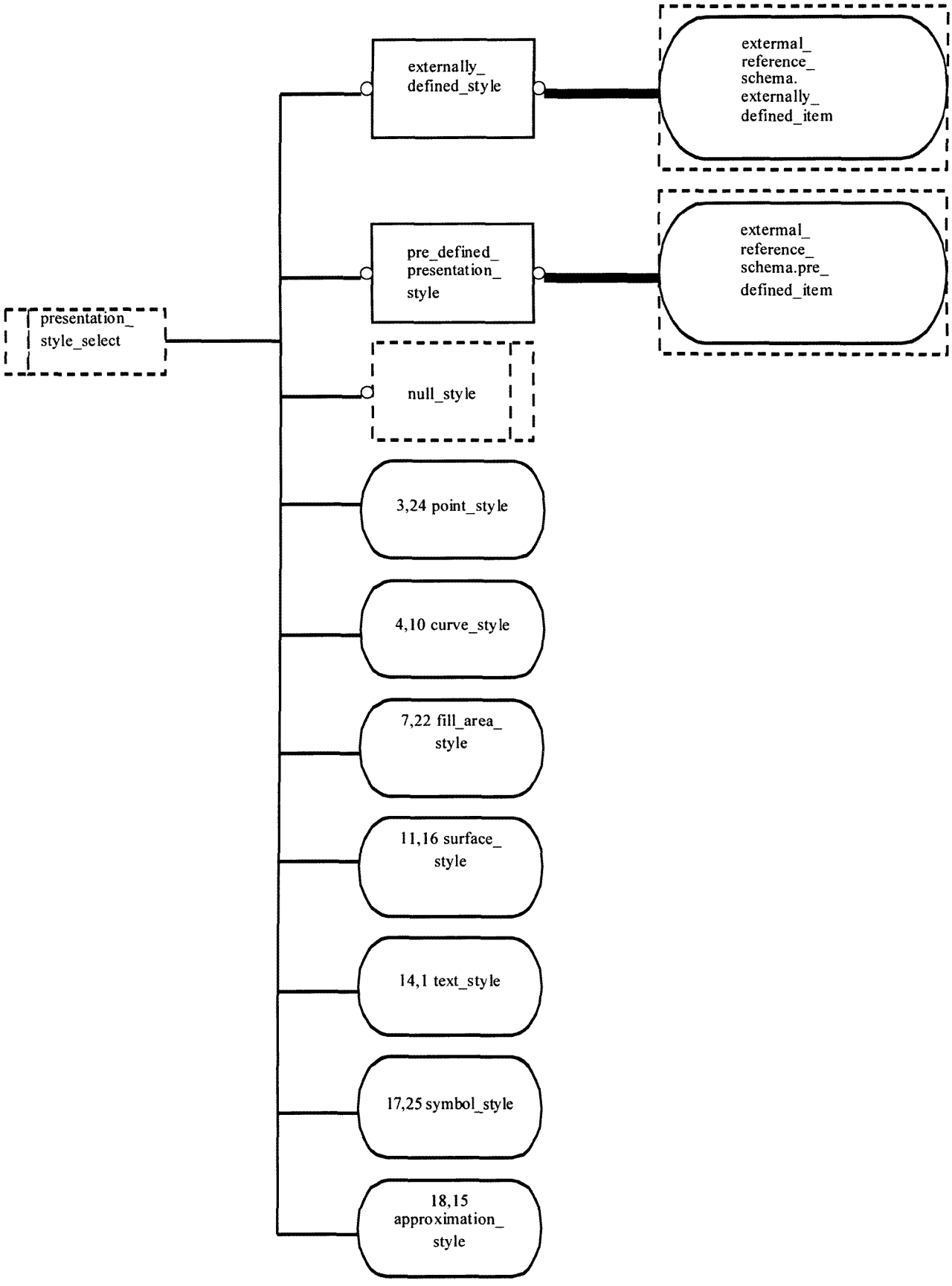


Рисунок Е.18 — EXPRESS-G диаграмма presentation_appearance_schema 2 из 21

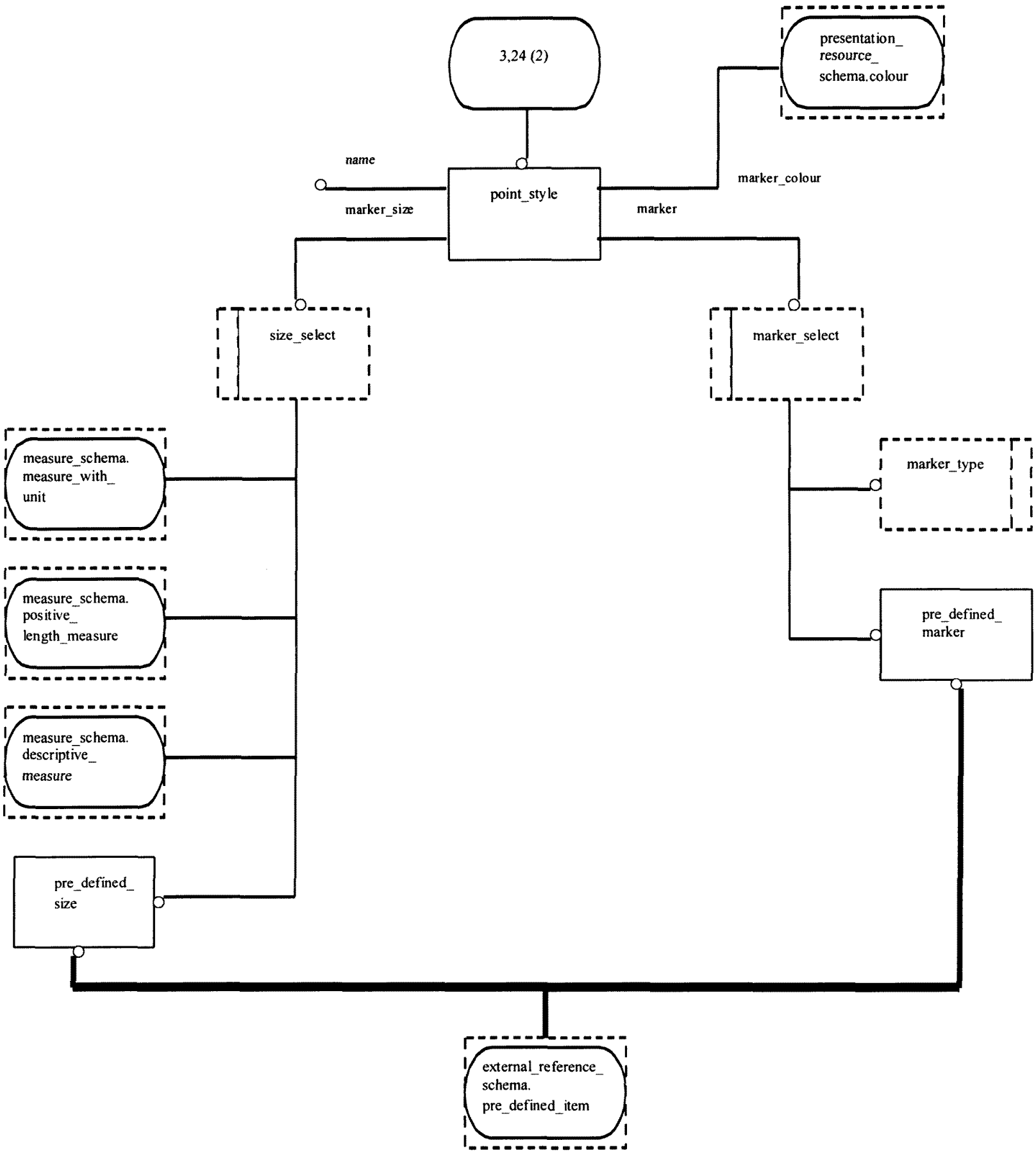


Рисунок Е.19 — EXPRESS-G диаграмма presentation_appearance_schema 3 из 21

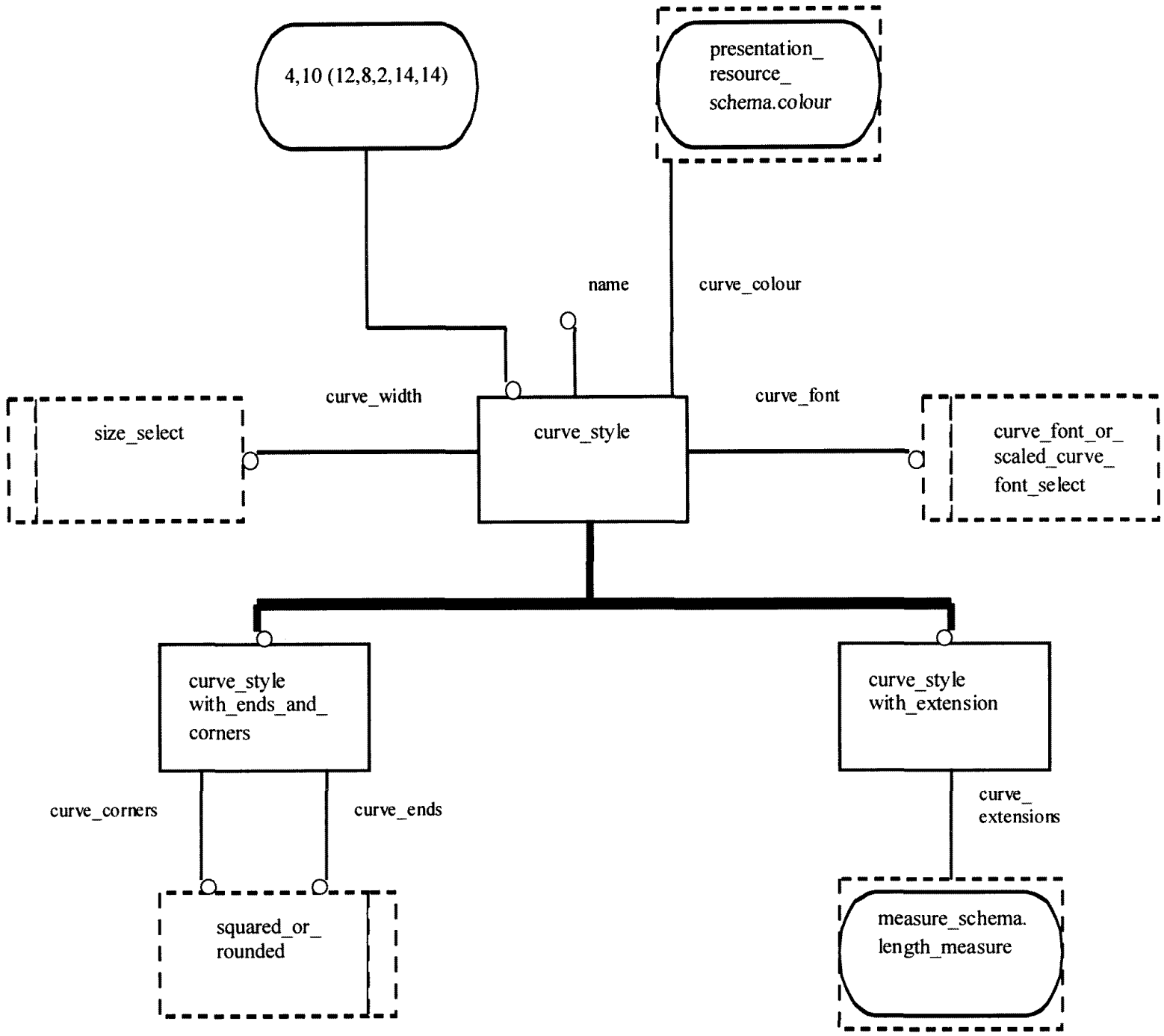


Рисунок E.20 — EXPRESS-G диаграмма presentation_appearance_schema 4 из 21

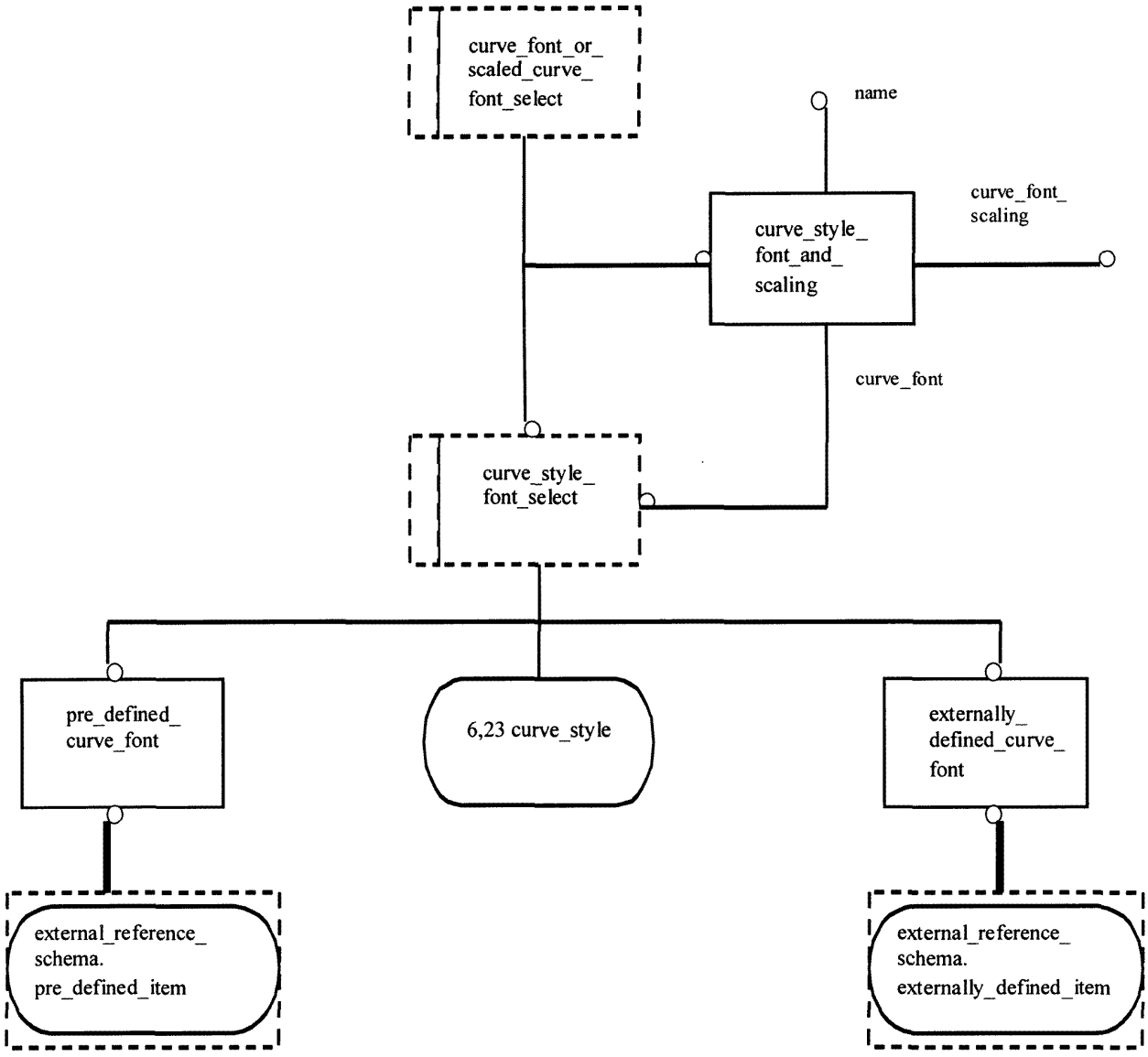


Рисунок Е.21 — EXPRESS-G диаграмма presentation_appearance_schema 5 из 21

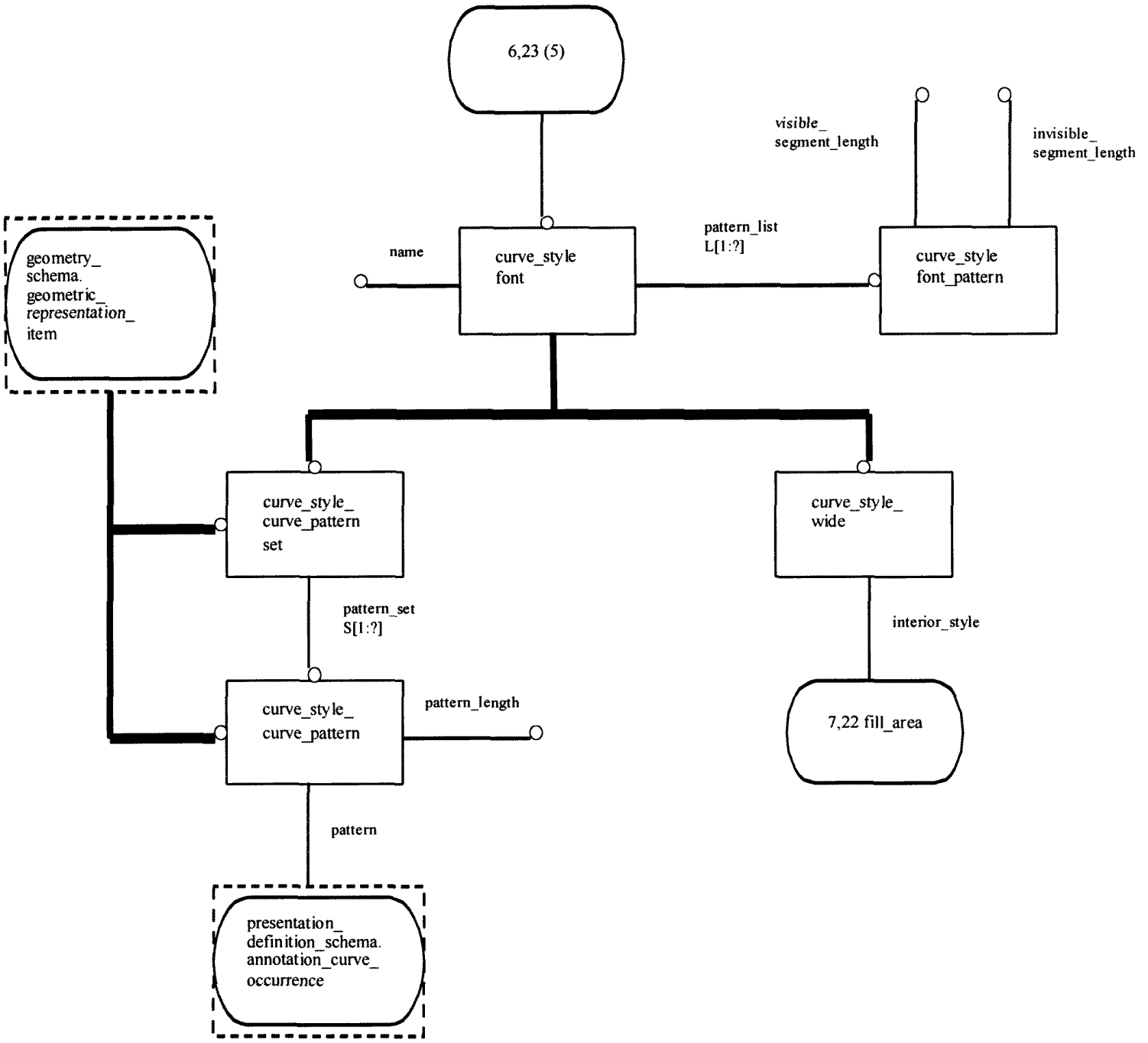


Рисунок Е.22 — EXPRESS-G диаграмма presentation_appearance_schema 6 из 21

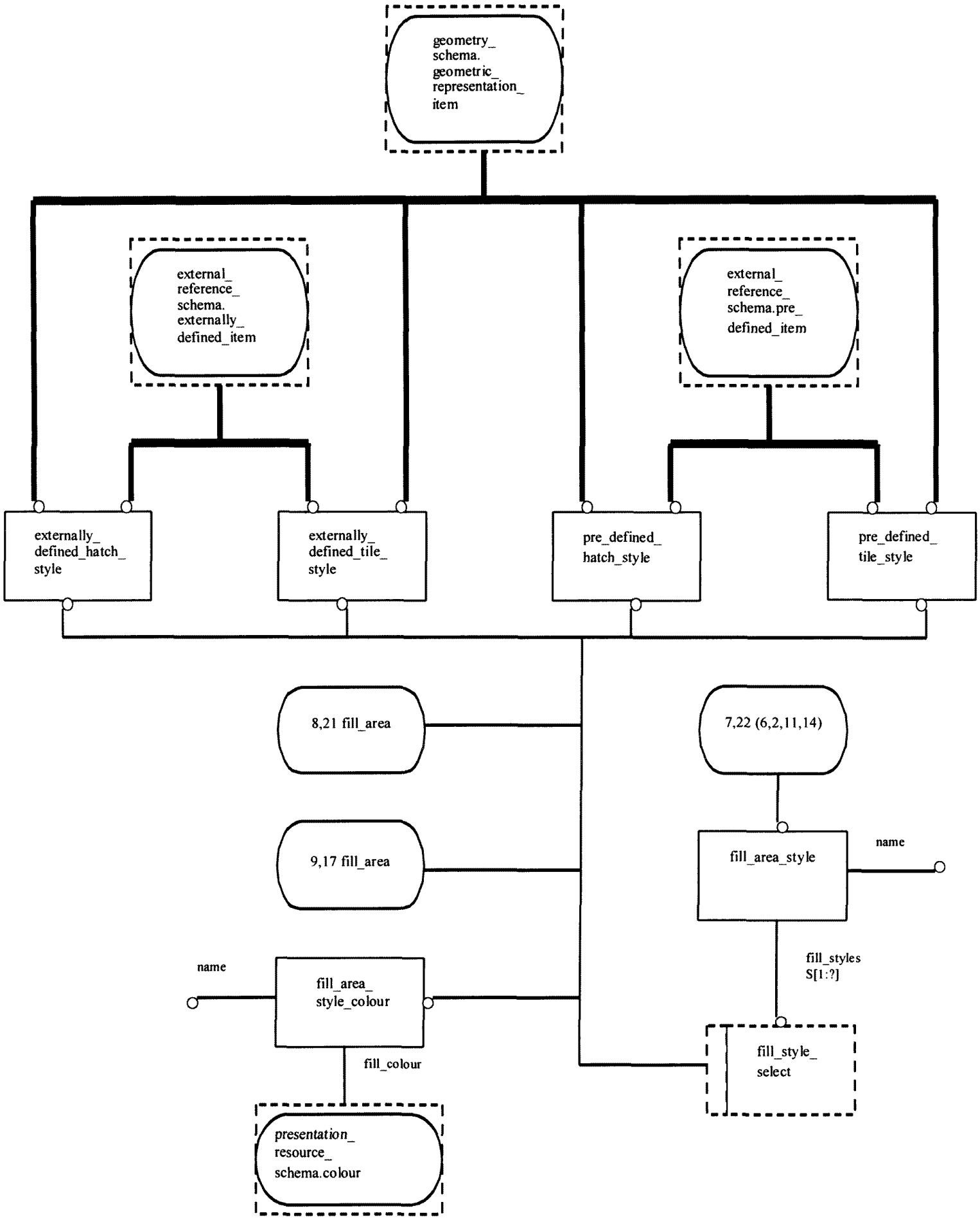


Рисунок E.23 — EXPRESS-G диаграмма presentation_appearance_schema 7 из 21

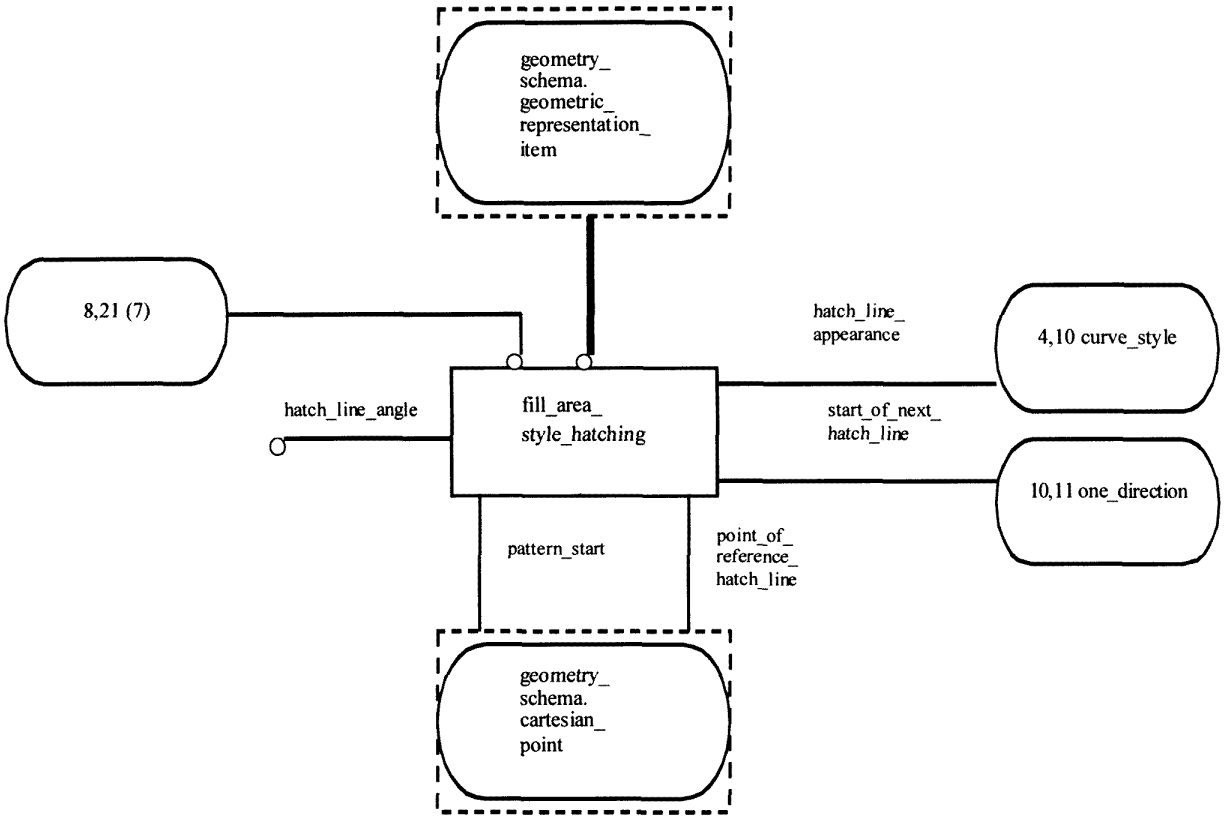


Рисунок Е.24 — EXPRESS-G диаграмма presentation_appearance_schema 8 из 21

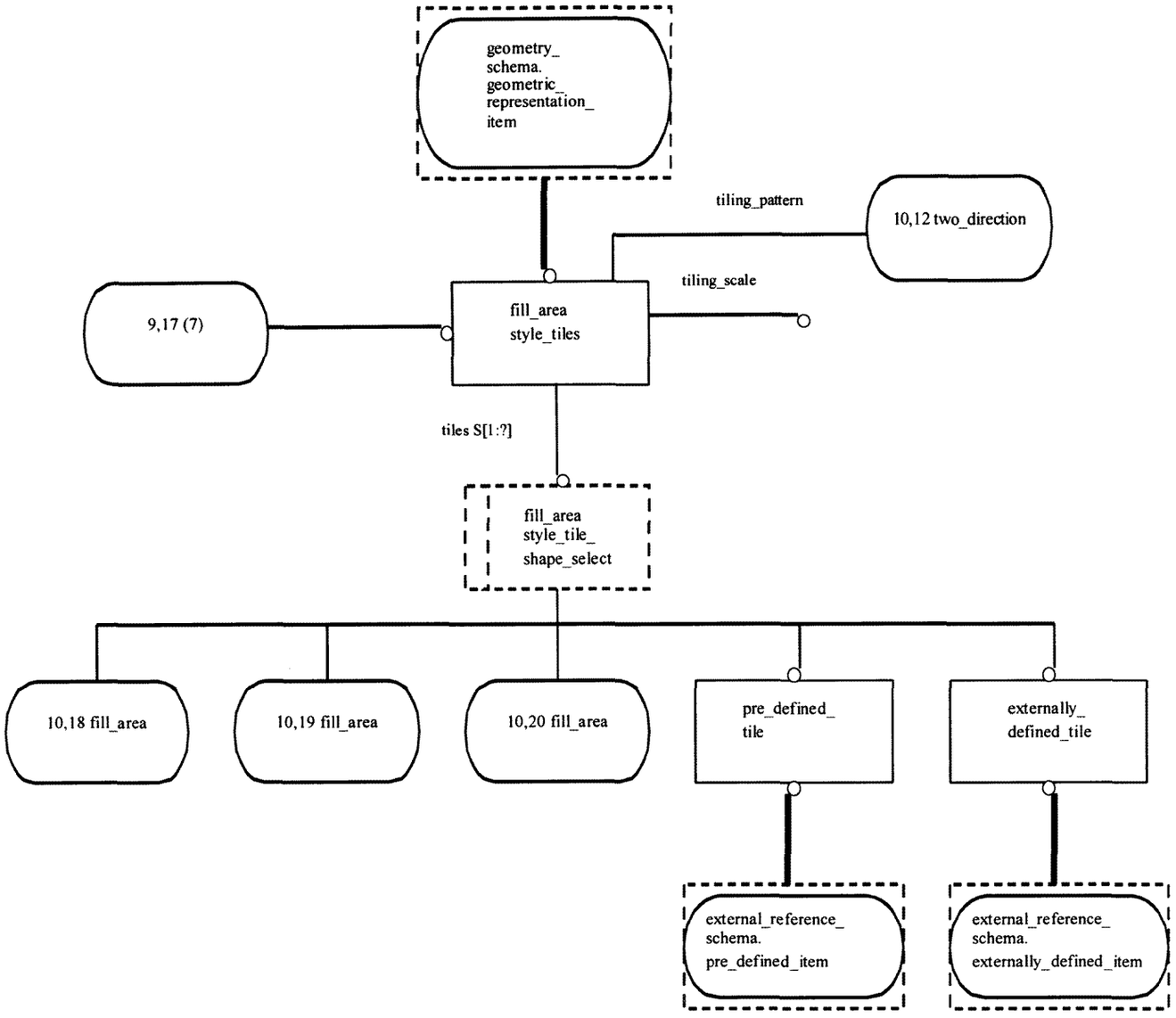
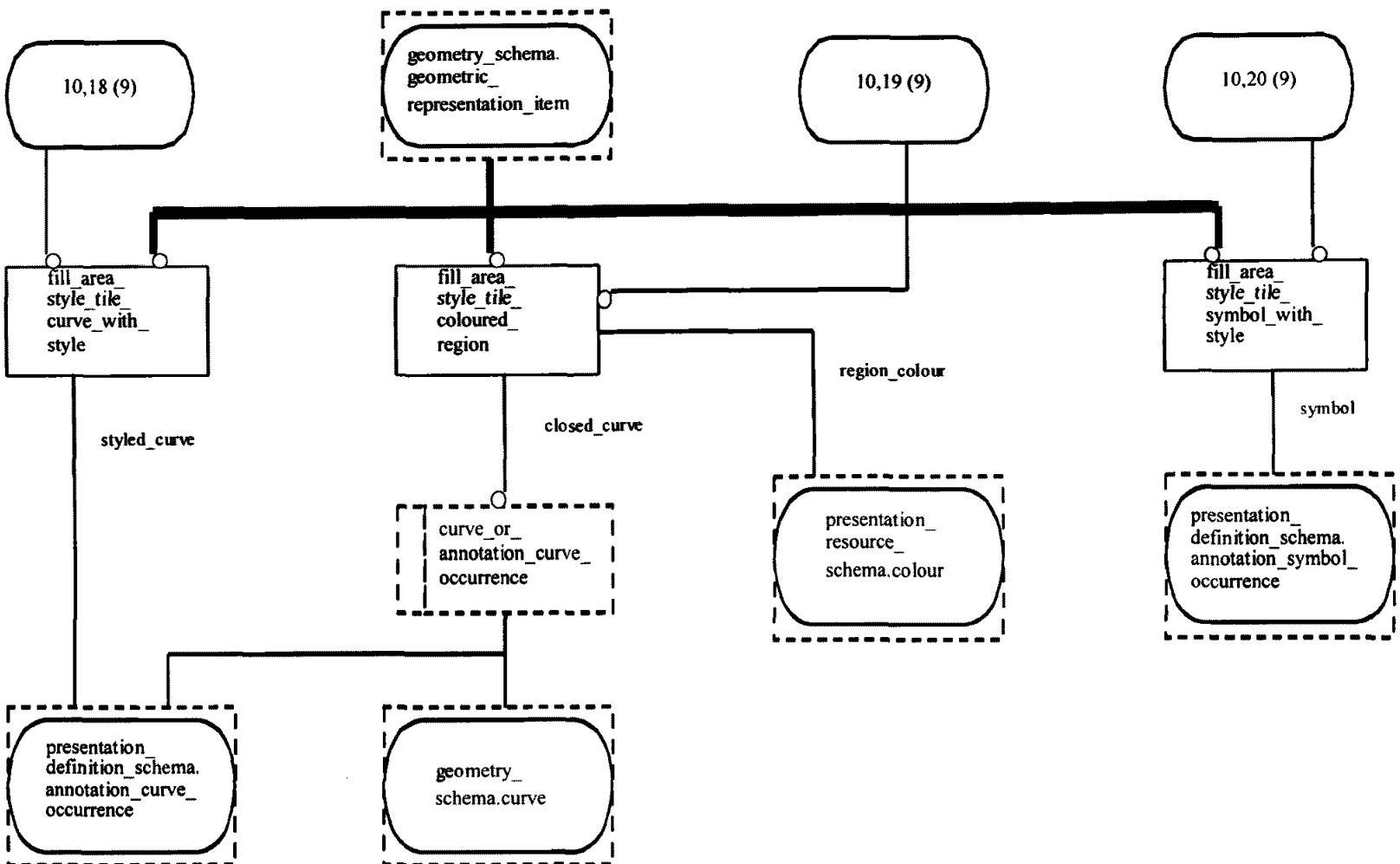


Рисунок E.25 — EXPRESS-G диаграмма presentation_appearance_schema 9 из 21



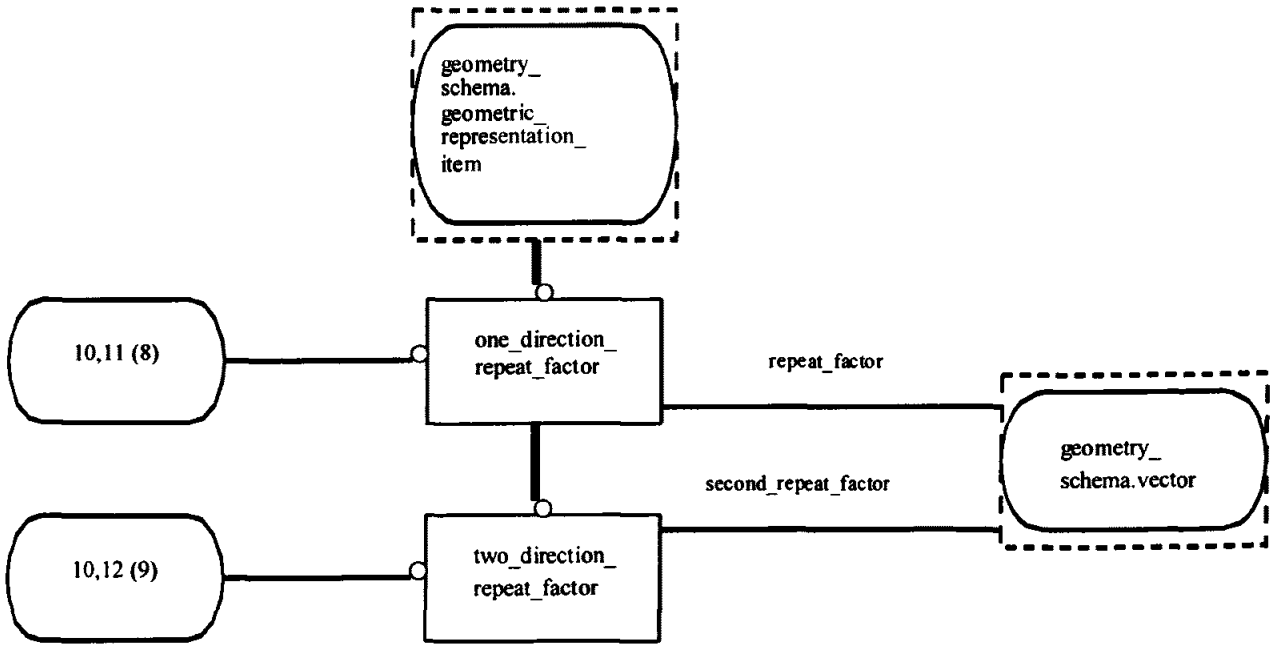


Рисунок E.26 — EXPRESS-G диаграмма presentation_appearance_schema 10 из 21

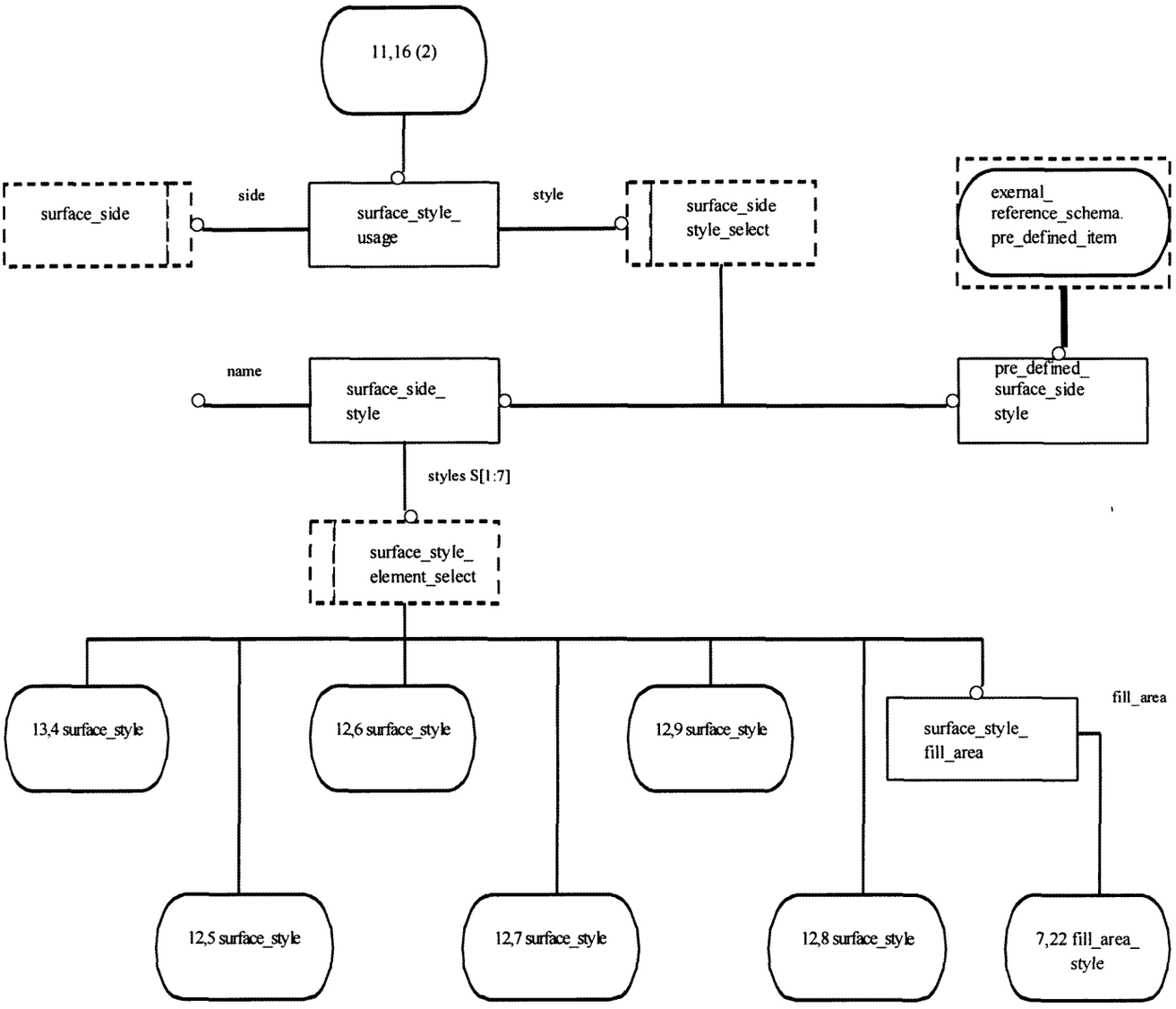


Рисунок E.27 — EXPRESS-G диаграмма presentation_appearance_schema 11 из 21

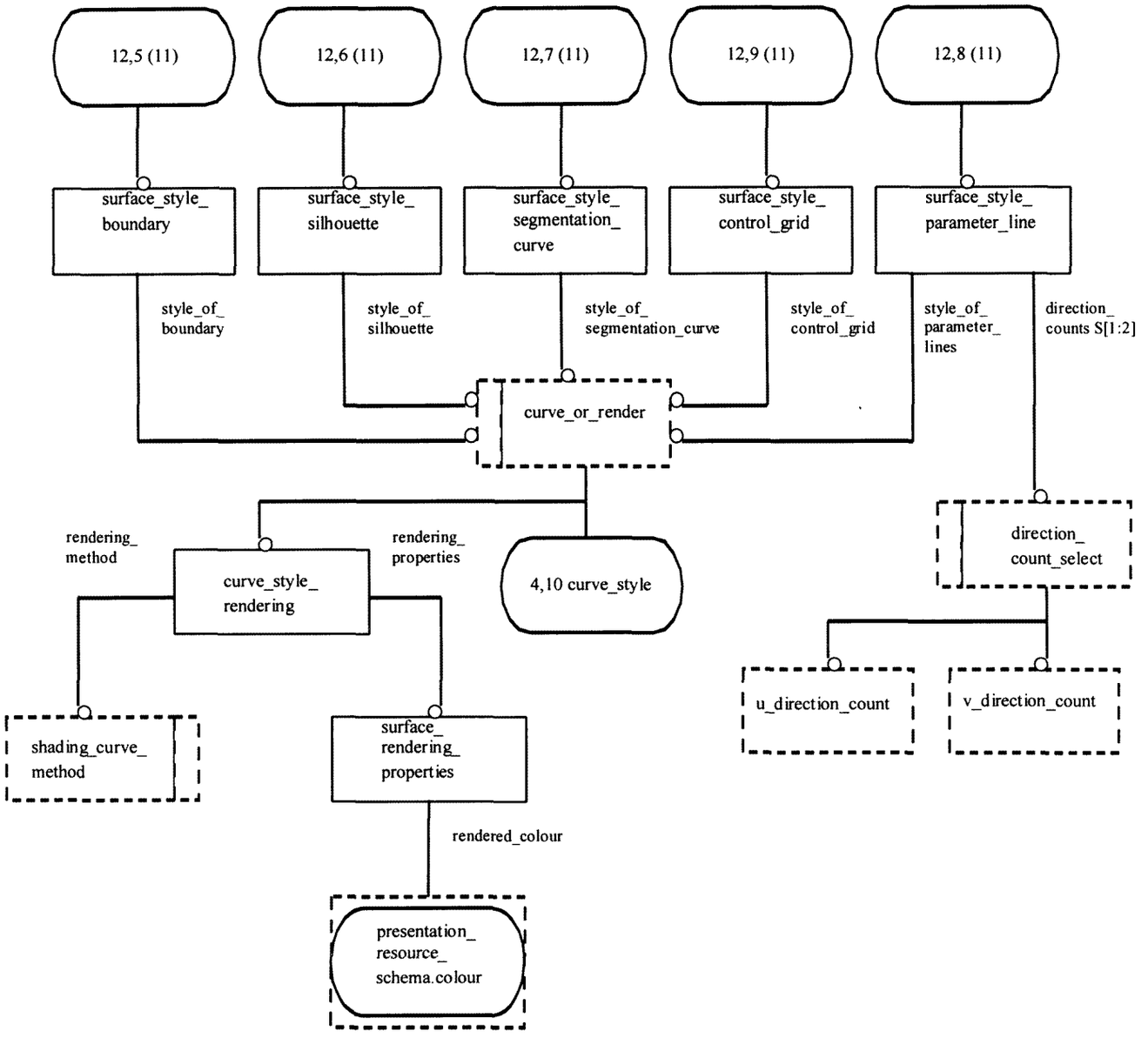


Рисунок E.28 — EXPRESS-G диаграмма presentation_appearance_schema 12 из 21

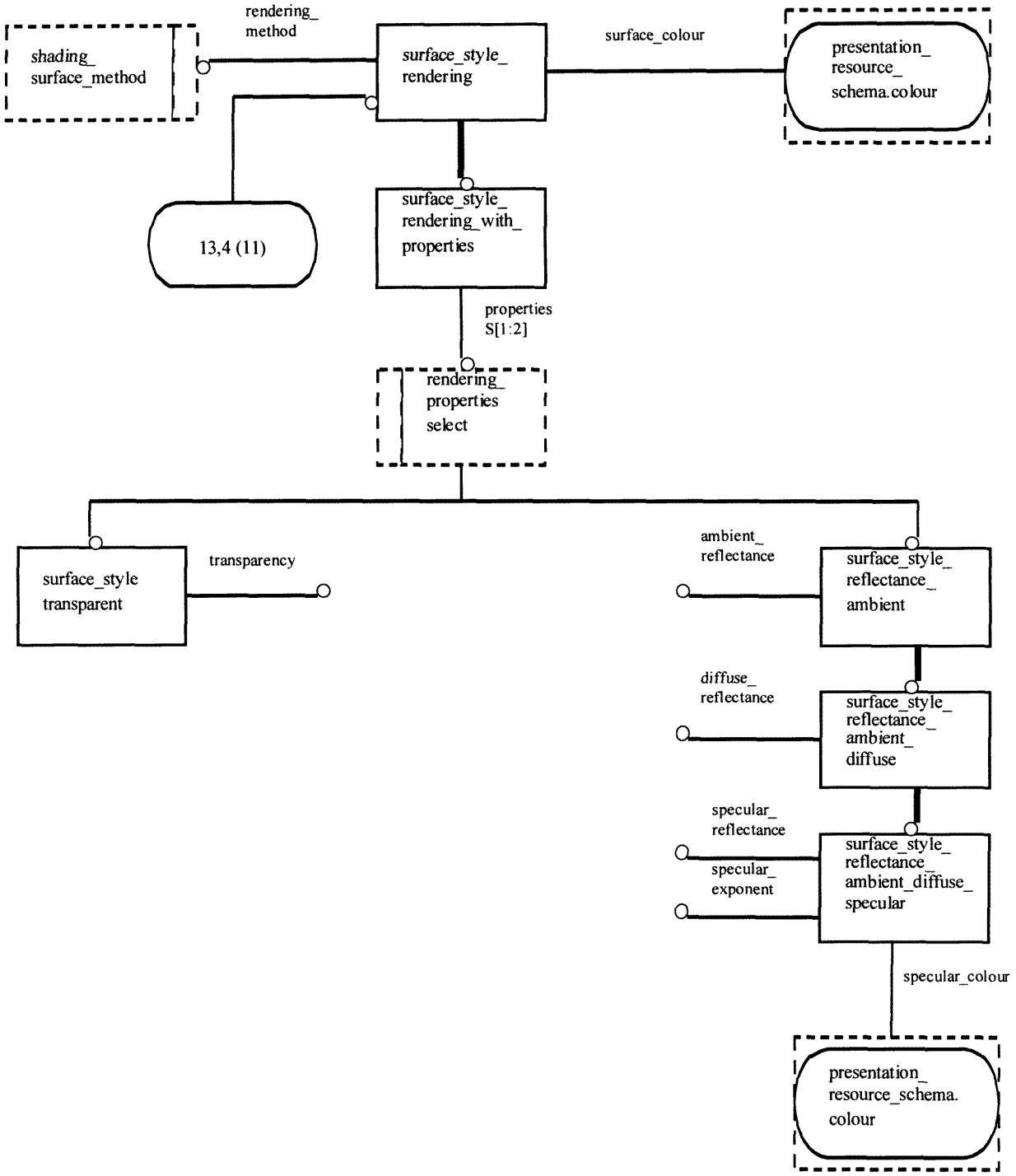


Рисунок E.29 — EXPRESS-G диаграмма presentation_appearance_schema 13 из 21

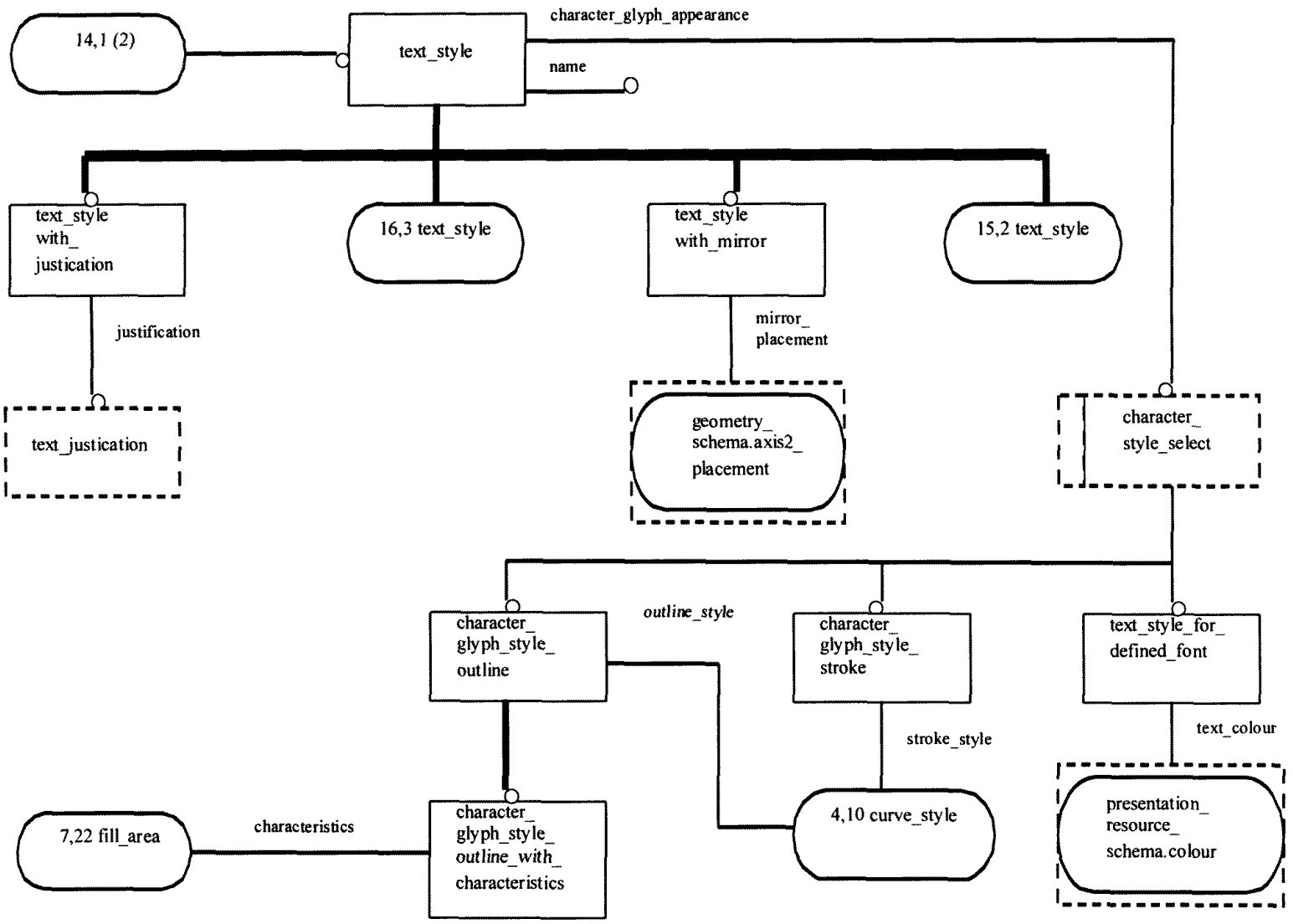


Рисунок E.30 — EXPRESS-G диаграмма presentation_appearance_schema 14 из 21

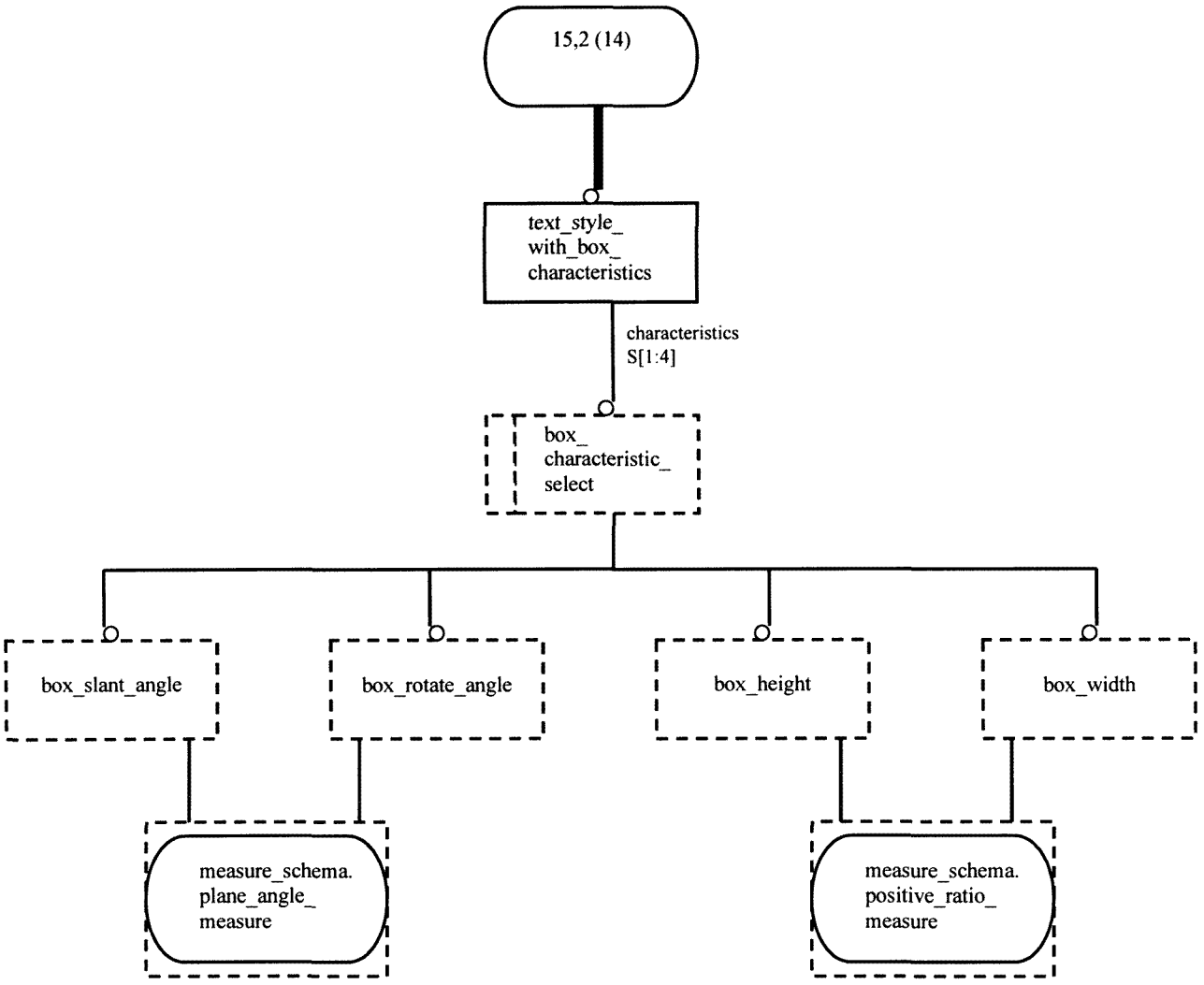


Рисунок E.31 — EXPRESS-G диаграмма presentation_appearance_schema 15 из 21

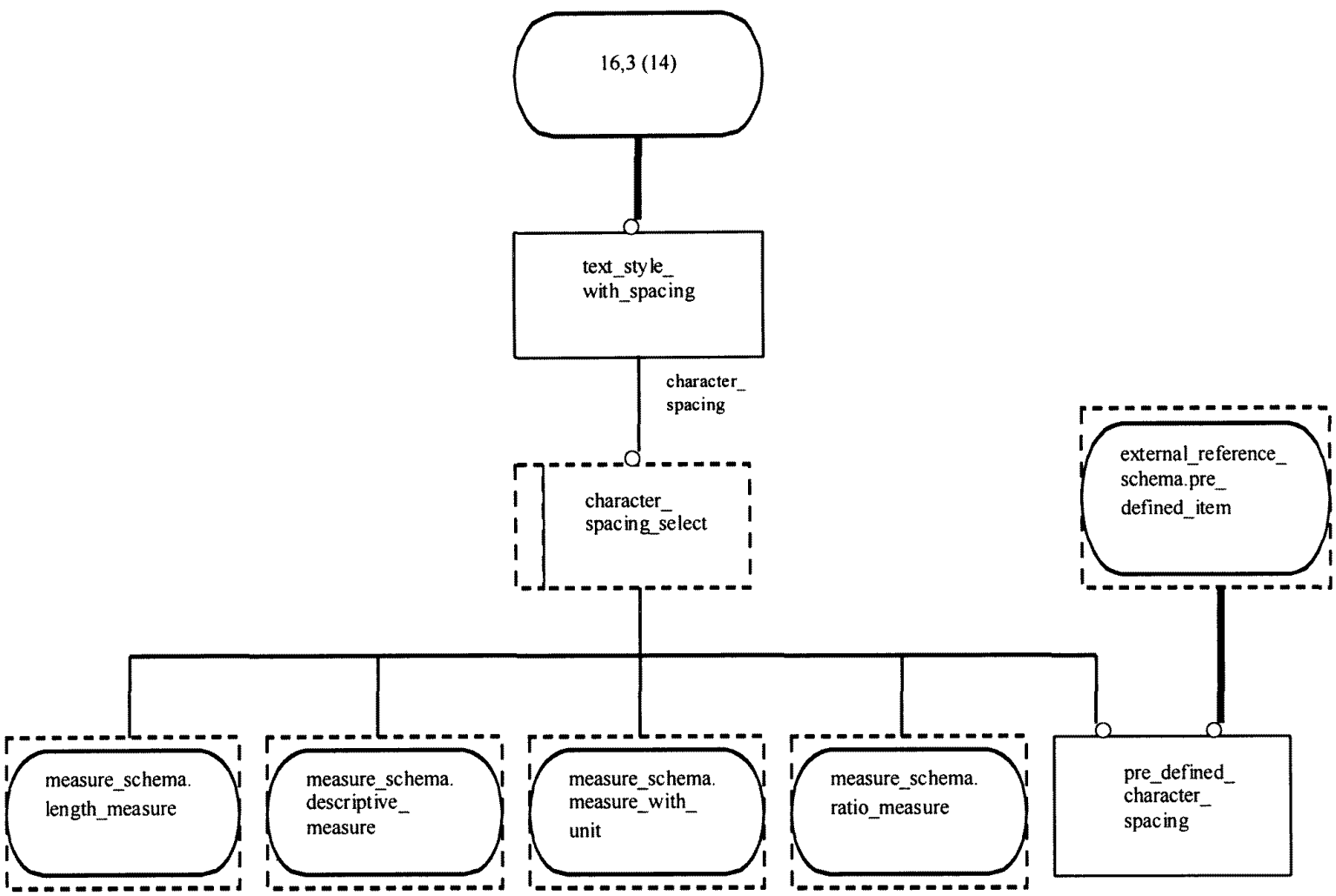


Рисунок E.32 — EXPRESS-G диаграмма presentation_appearance_schema 16 из 21

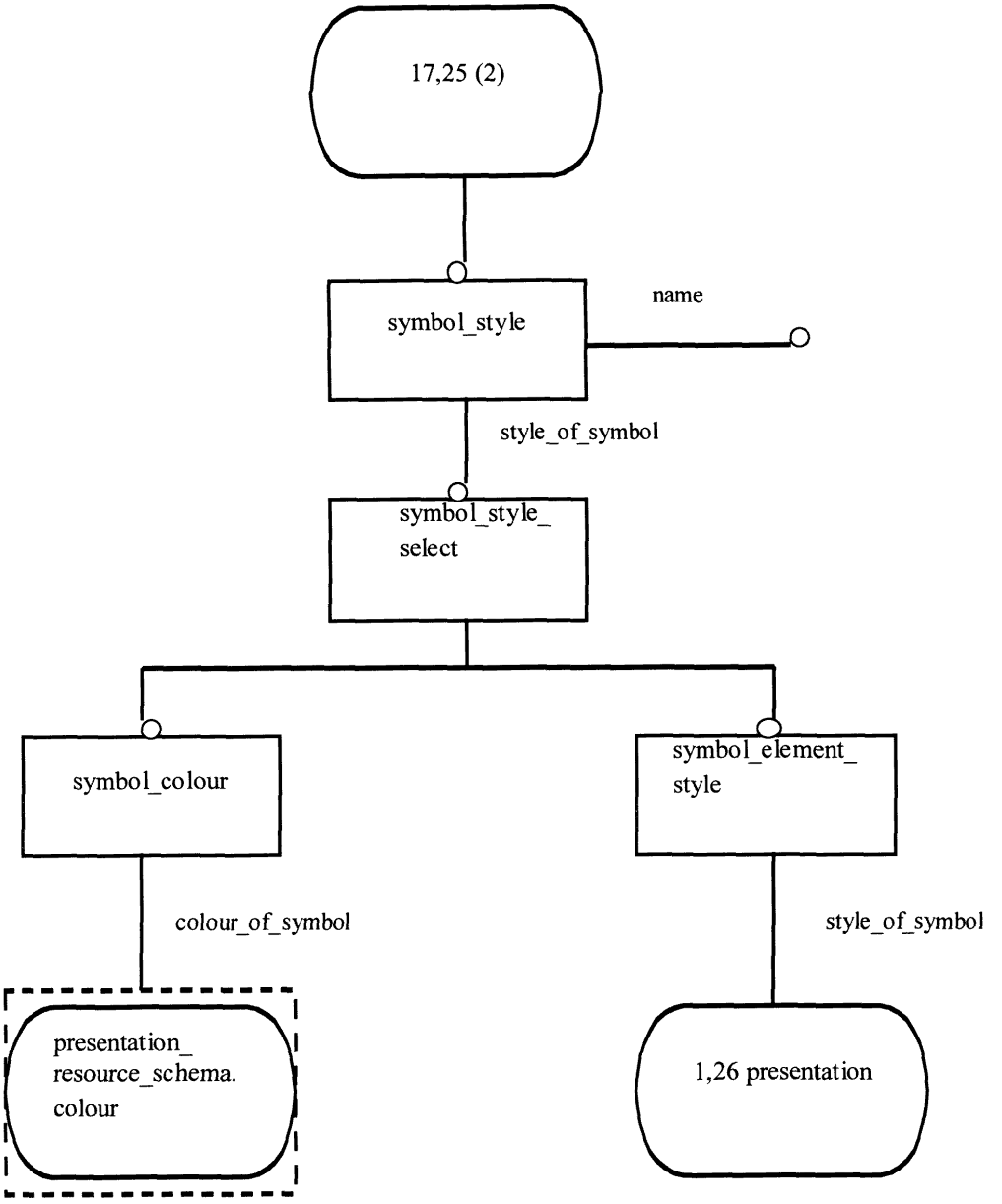


Рисунок Е.33 — EXPRESS-G диаграмма presentation_appearance_schema 17 из 21

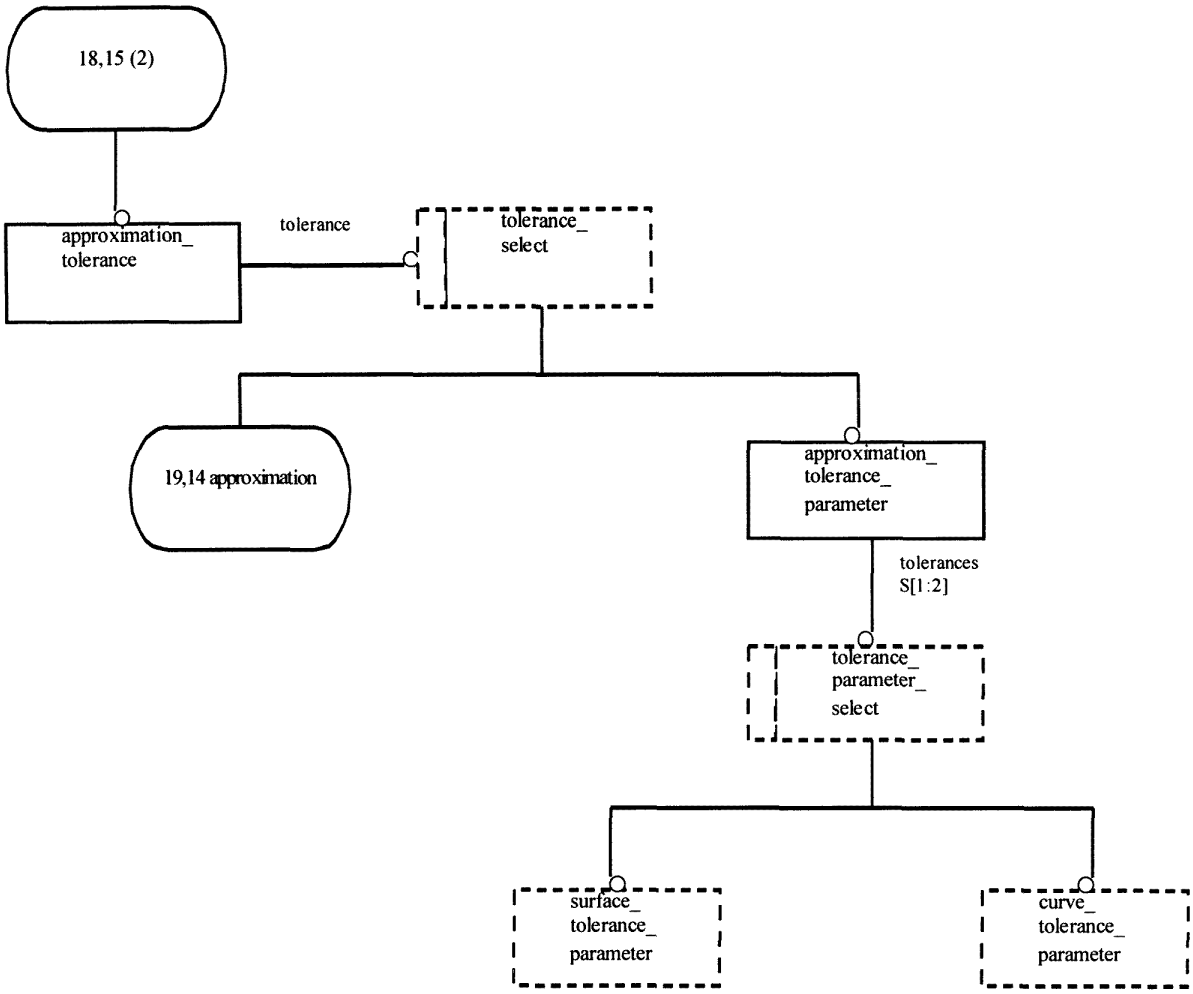


Рисунок E.34 — EXPRESS-G диаграмма presentation_appearance_schema 18 из 21

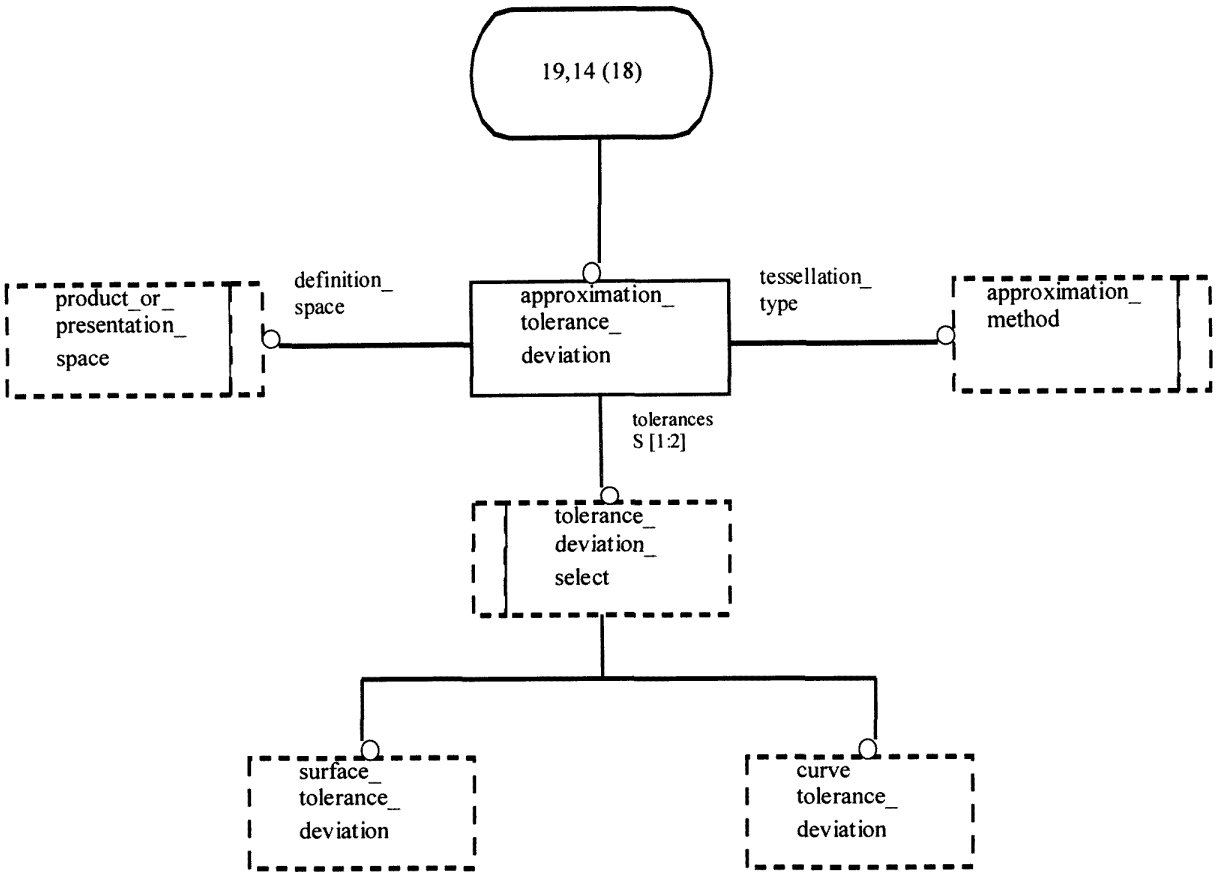


Рисунок E.35 — EXPRESS-G диаграмма presentation_appearance_schema 19 из 21

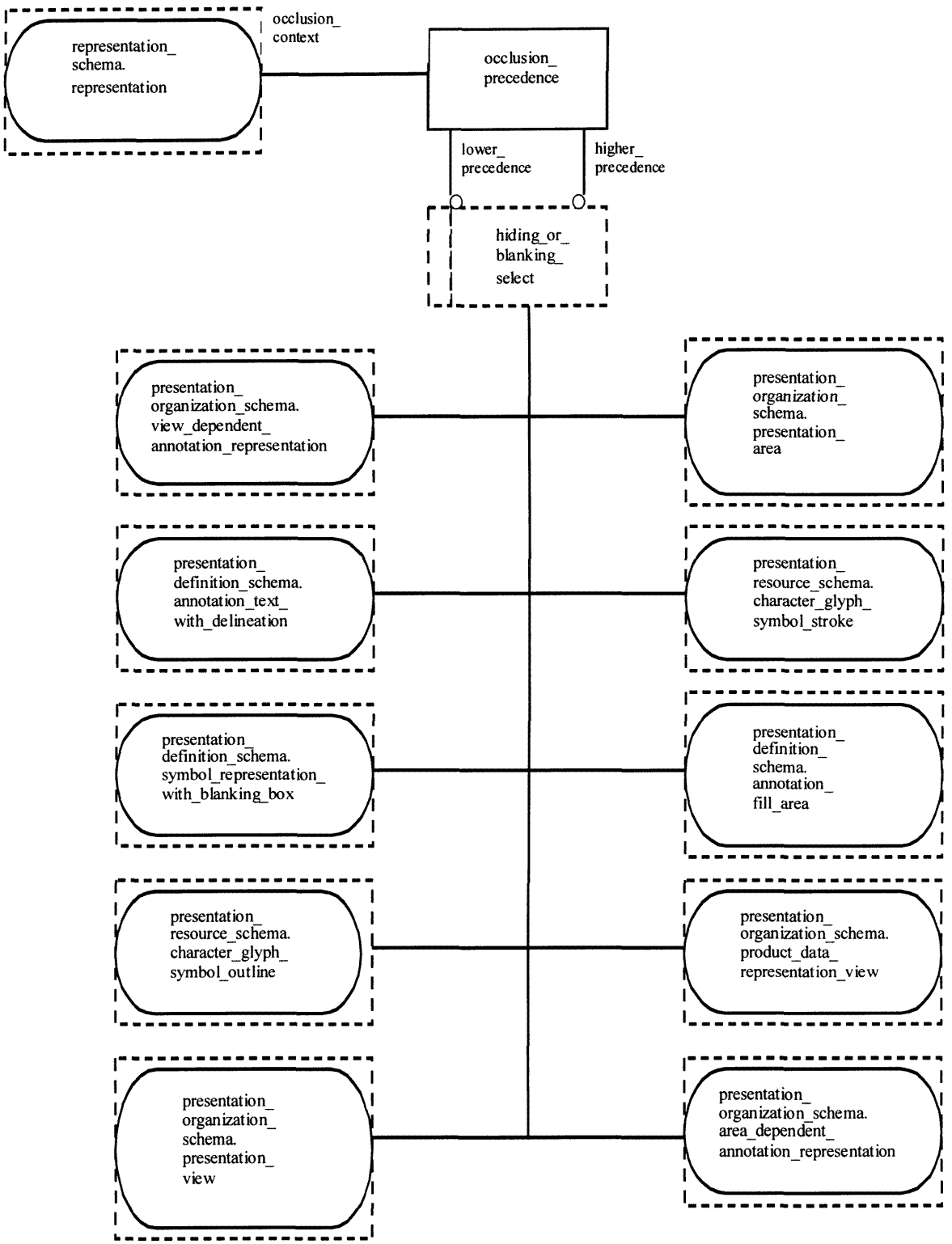


Рисунок Е.36 — EXPRESS-G диаграмма presentation_appearance_schema 20 из 21

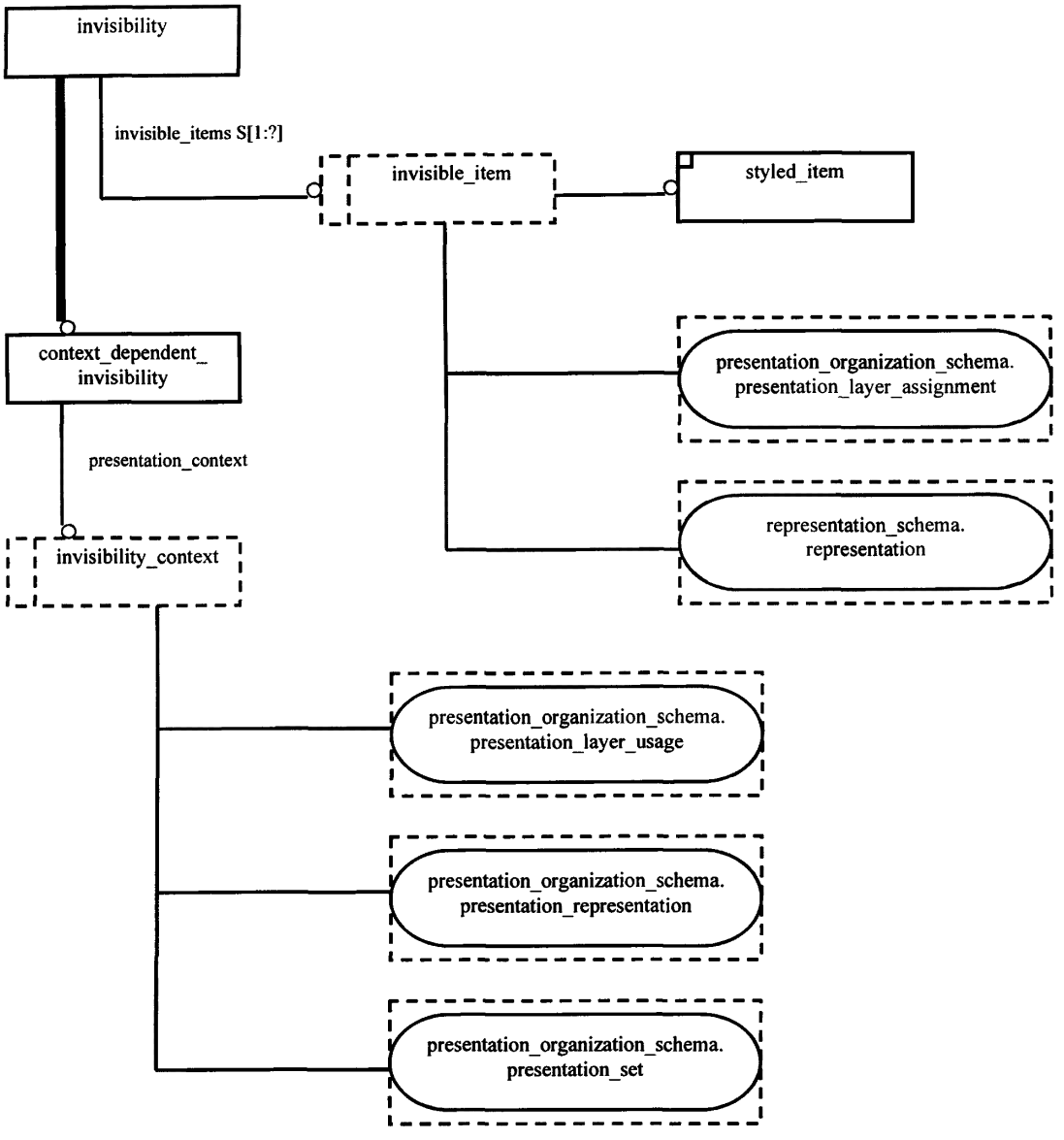


Рисунок Е.37 — EXPRESS-G диаграмма presentation_appearance_schema 21 из 21

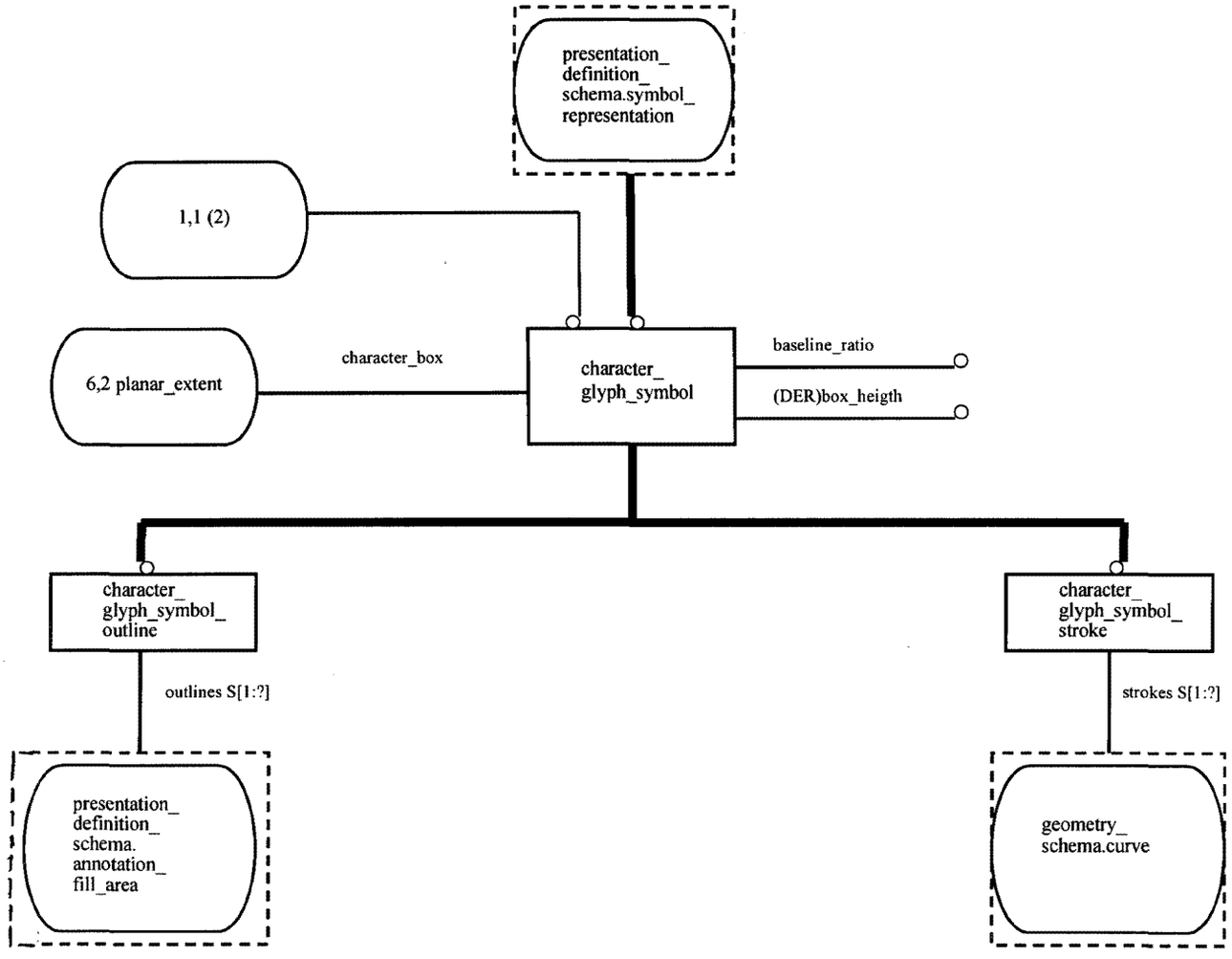


Рисунок Е.38 — EXPRESS-G диаграмма presentation_resource_schema 1 из 5

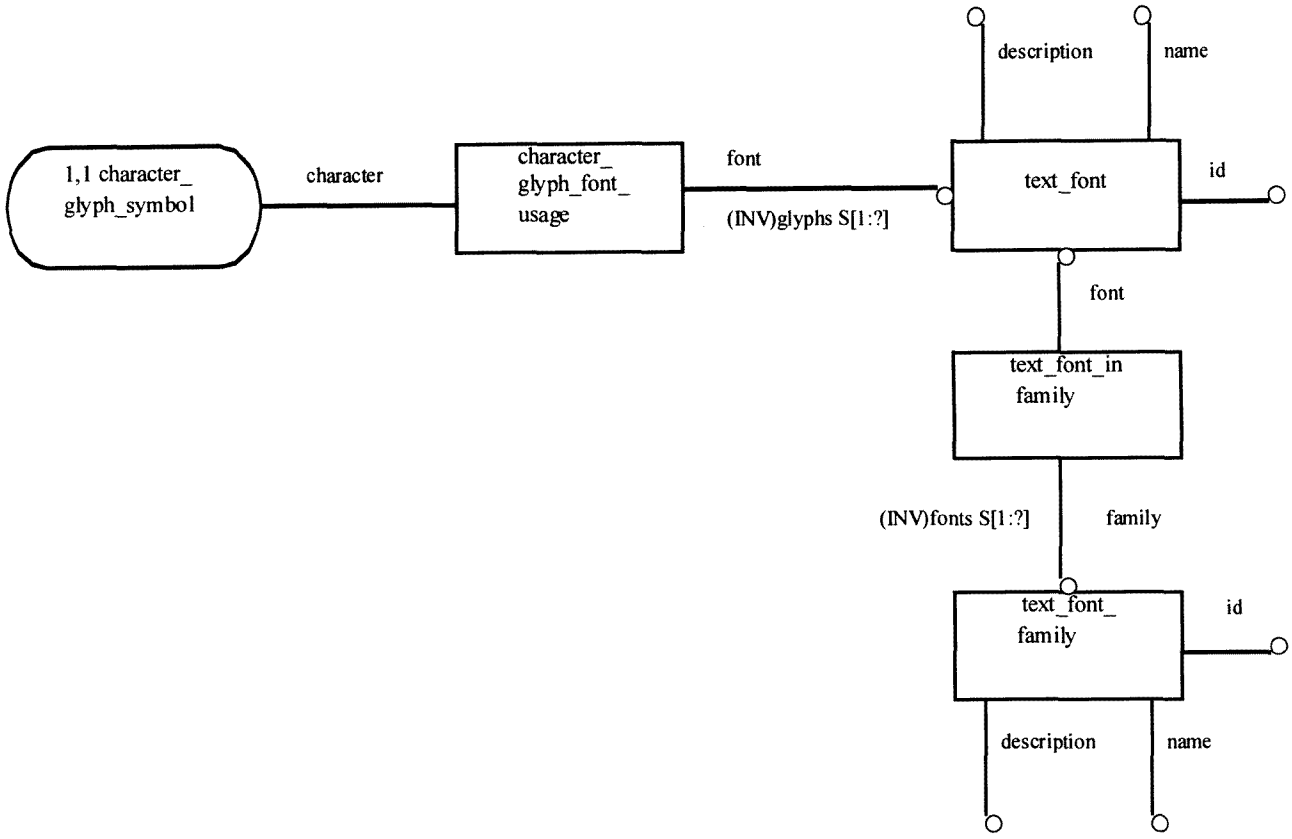


Рисунок E.39 — EXPRESS-G диаграмма presentation_resource_schema 2 из 5

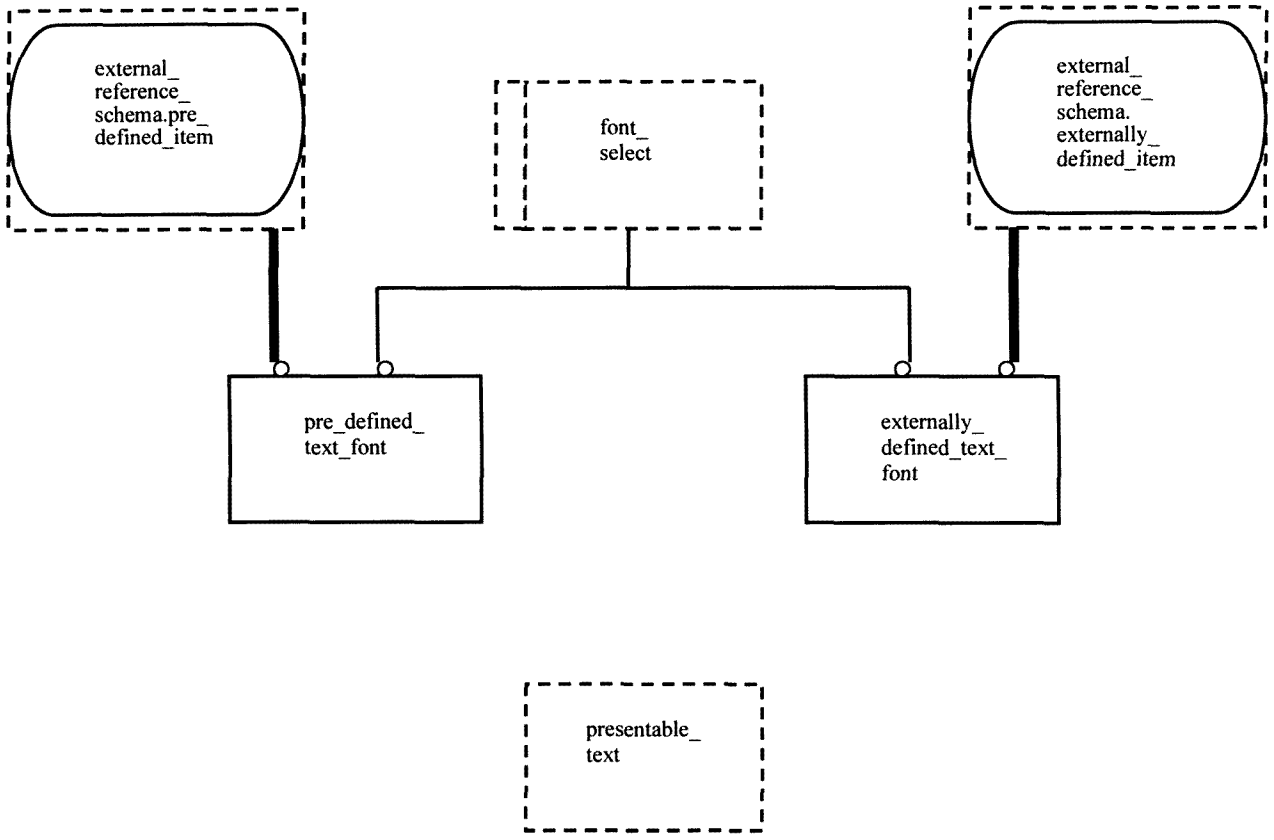


Рисунок E.40 — EXPRESS-G диаграмма presentation_resource_schema 3 из 5

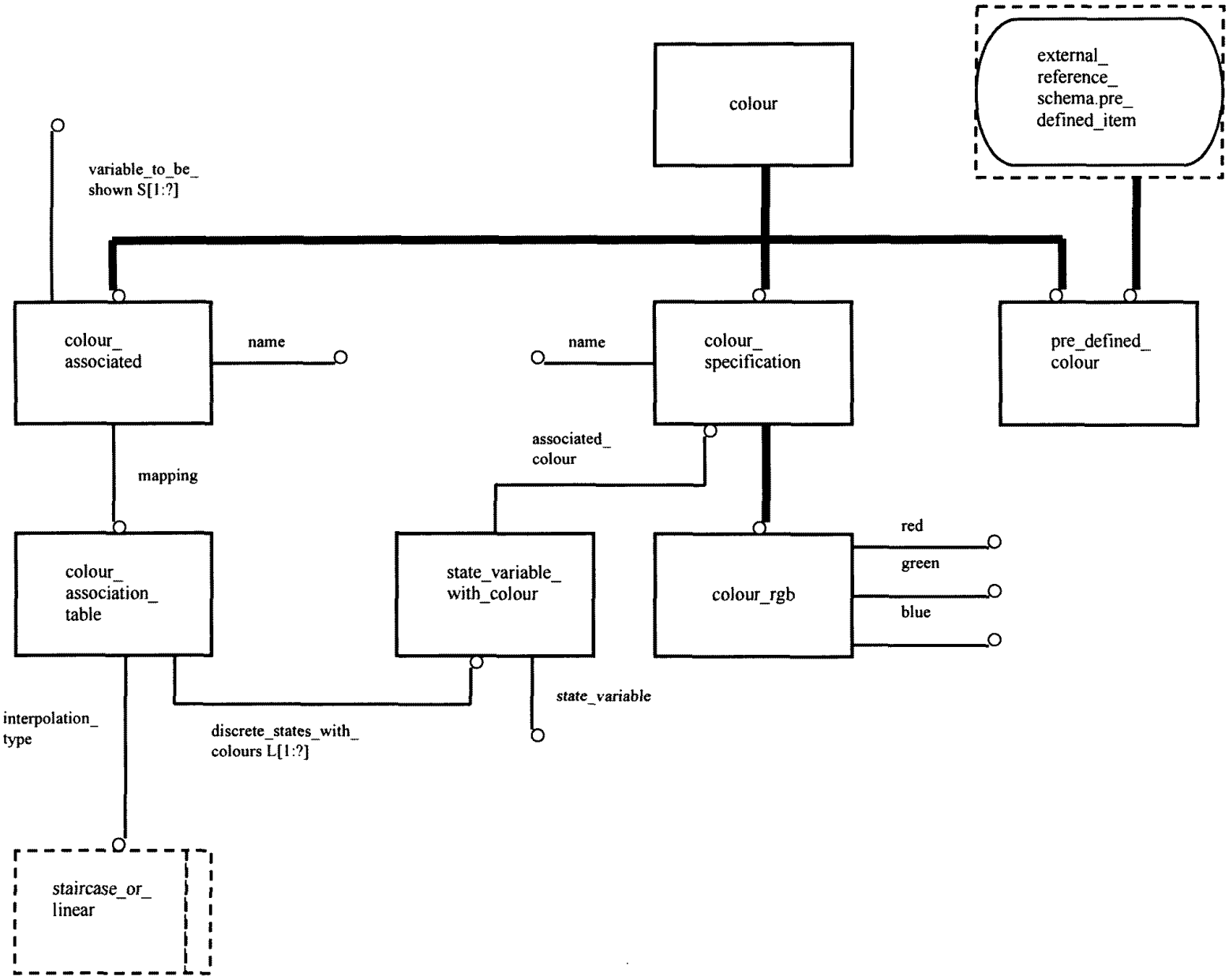


Рисунок E.41 — EXPRESS-G диаграмма presentation_resource_schema 4 из 5

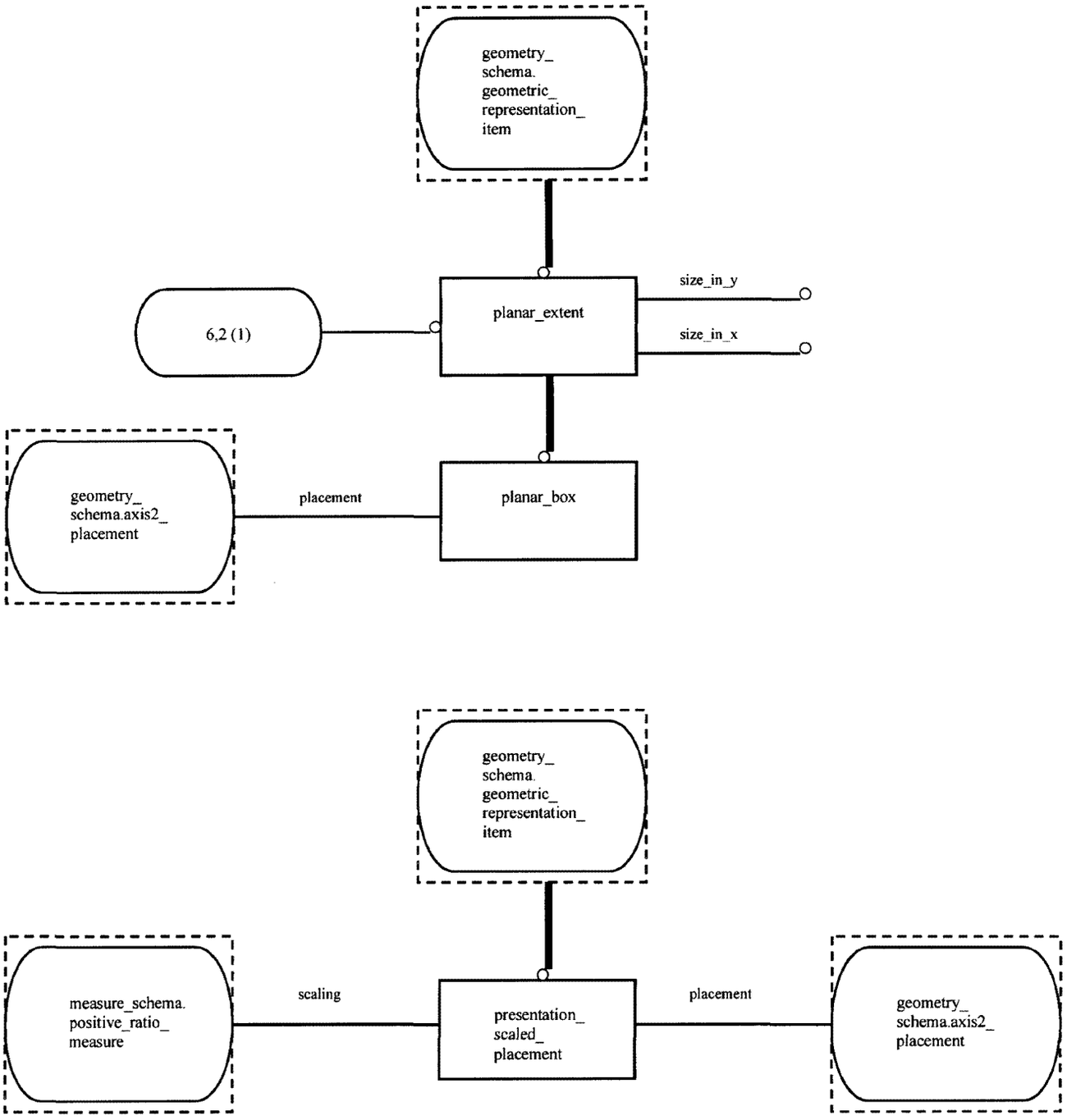


Рисунок Е.42 — EXPRESS-G диаграмма presentation_resource_schema 5 из 5

ПРИЛОЖЕНИЕ F
(справочное)

Библиография

[1] ИСО 8805—85* Системы обработки информации. Машинная графика. Функциональное описание графического ядра трехмерной системы (GKS-3D)
[2] ИСО/МЭК 9592-1—97* Информационная технология. Машинная графика и обработка изображений. Иерархическая интерактивная графическая система программиста (PHIGS). Часть 1. Функциональное описание
[3] ИСО/МЭК 9592-2—97* Информационная технология. Машинная графика и обработка изображений. Иерархическая интерактивная графическая система программиста (PHIGS). Часть 2. Формат архивного файла
[4]ИСО/МЭК 9592-3—97* Информационная технология. Машинная графика и обработка изображений. Иерархическая интерактивная графическая система программиста (PHIGS). Часть 2. Требования к кодированию архивного файла открытым текстом
[5] ИСО/МЭК 9592-4—92* Информационная технология. Машинная графика. Иерархическая интерактивная графическая система программиста (PHIGS). Часть 4. Освещенность и поверхности (PHIGS PLUS)
[6] NEWMAN, WILLIAM M.and SPROULL, ROBERT F., Principles of Interactive Computer Graphics, Mc Graw Hills, 1981, ISBN 0-07-046338-7
[7] FOLEY, JAMES D. and VAN DAM, ANDRIES, Fundamentals of Interactive Computer Graphics, Addison Wesley, ISBN 0-201-14468-9

* Оригиналы международных стандартов ИСО — во ВНИИКИ Госстандарта России.

Тематический указатель

acyclic_composite_text . . . 5.6.1
acyclic_occulusion_precedence . . . 6.13.1
acyclic_presentation_representation_relationship . . . 4.9.1
acyclic_symbol_representation-relationship . . . 5.6.2
annotation_curve_occurrence . . . 5.5.3
annotation_fill_area . . . 5.4.1
annotation_fill_area_occurrence . . . 5.5.4
annotation_occurrence . . . 5.5.1
annotation_occurrence_relationship . . . 5.5.8
annotation_point_occurrence . . . 5.5.2
annotation_symbol . . . 5.4.7
annotation_symbol_occurrence . . . 5.5.6
annotation_table . . . 5.4.8
annotation_table_occurrence . . . 5.5.7
annotation_text . . . 5.4.18
annotation_text_character . . . 5.4.24
annotation_text_occurrence . . . 5.5.5
annotation_text_with_associated_curves . . . 5.4.22
annotation_text_with_blanking_box . . . 5.4.21
annotation_fext_with_delineation . . . 5.4.20
annotation_fext_with_extent . . . 5.4.19
approximation_method. . . . 6.3.33
approximation_tolerance . . . 6.11.1
approximation_tolerance_deviation . . . 6.11.2
approximation_tolerance_parameter. . . . 6.11.3
area_dependent_annotation_representation. . . . 4.4.6
area_in_set. . . . 4.4.4
area_or_view. . . . 4.3.2
background_colour . . . 4.4.10
box_characteristic_select . . . 6.3.25
box_height . . . 6.3.26
box_rotate_angle . . . 6.3.29
box_slant_angle . . . 6.3.28
box_width . . . 6.3.27
camera_image . . . 4.5.14
camera_model . . . 4.5.1
camera_model_d2 . . . 4.5.2
camera_model_d2_shape_clipping . . . 4.5.3
camera_model_d3 . . . 4.5.4
camera_model_d3_multi_clipping . . . 4.5.7
camera_model_d3_with_hlhsr . . . 4.5.6
camera_model_with_light_sources . . . 4.5.8
camera_usage . . . 4.5.15
central_or_parallel . . . 4.3.3
character_glyph_font_usage . . . 7.3.4
character_glyph_style_outline . . . 6.9.3
character_glyph_style_outline_with_characteristics . . . 6.9.4
character_glyph_style_stroke . . . 6.9.2
character_glyph_symbol . . . 7.3.1
character_glyph_symbol_outline . . . 7.3.3
character_glyph_symbol_stroke . . . 7.3.2
character_spacing_select . . . 6.3.30
character_style_select . . . 6.3.23
colour . . . 7.3.10
colour_associated . . . 7.3.13
colour_association_table . . . 7.3.14
colour_rgb . . . 7.3.12
colour_specification . . . 7.3.11
composite_text . . . 5.4.33

composite_text_with_associated_curves . . . 5.4.37
composite_text_with_blanking_box . . . 5.4.36
composite_text_with_delineation . . . 5.4.35
composite_text_with_extent . . . 5.4.34
context_dependent_invisibility . . . 6.12.3
context_dependent_over_riding_styled_item . . . 6.4.3
curve_font_or_scaled_curve_font_select . . . 6.3.7
curve_or_annotation_curve_occurrence . . . 6.3.12
curve_or_render . . . 6.3.16
curve_style . . . 6.6.1
curve_style_curve_pattern . . . 6.6.10
curve_style_curve_pattern_set . . . 6.6.9
curve_style_font . . . 6.6.6
curve_style_font_and_scaling . . . 6.6.11
curve_style_font_pattern . . . 6.6.7
curve_style_font_select . . . 6.3.8
curve_style_rendering . . . 6.8.6
curve_style_wide . . . 6.6.8
curve_style_with_ends_and_corners . . . 6.6.2
curve_style_witn_extension . . . 6.6.3
curve_tolerance_deviation . . . 6.3.35
curve_tolerance_parameter . . . 6.3.39
defined_character_glyph . . . 5.4.25
defined_glyph_select . . . 5.3.5
defined_symbol . . . 5.4.2
defined_symbol_select . . . 5.3.2
defined_table . . . 5.4.3
derection_count_select . . . 6.3.18
externally_defined_character_glyph . . . 5.4.26
externally_defined_curve_font . . . 6.6.5
externally_defined_hatch_style . . . 6.7.4
externally_defined_style . . . 6.4.7
externally_defined_symbol . . . 5.4.6
externally_defined_text_font . . . 7.3.8
externally_defined_tile . . . 6.7.13
externally_defined_tile_style . . . 6.7.7
field_in_table . . . 5.6.3
fill_area_style . . . 6.7.1
fill_area_style_colour . . . 6.7.2
fill_area_style_hatching . . . 6.7.5
fill_area_style_tile_coloured_region . . . 6.7.10
fill_area_style_tile_curve_with_style . . . 6.7.9
fill_area_style_tile_shape_select . . . 6.3.11
fill_area_style_tile_symbol_with_style . . . 6.7.11
fill_area_style_tiles . . . 6.7.8
fill_style_select . . . 6.3.10
font_select . . . 7.2.3
graphical_transformation . . . 4.4.12
hiding_or_blanking_select . . . 6.3.41
invisibility . . . 6.12.2
invisibility_context . . . 6.3.42
invisibile_item . . . 6.3.43
layered_item . . . 4.3.4
light_source . . . 4.5.9
light_source_ambient . . . 4.5.10
light_source_directional . . . 4.5.11
light_source_positional . . . 4.5.12
light_source_spot . . . 4.5.13
marker_select . . . 6.3.4
marker_type . . . 6.3.5
null_style . . . 6.3.3
occlusion_precedence . . . 6.12.1

one_direction_repeat_factor . . . 6.7.14
over_riding_styled_item . . . 6.4.2
planar_box . . . 7.3.18
planar_extent . . . 7.3.17
point_style . . . 6.5.1
pre_defined_character_glyph . . . 5.4.27
pre_defined_character_spacing . . . 6.9.9
pre_defined_colour . . . 7.3.16
pre_defined_curve_font . . . 6.6.4
pre_defined_hatch_style . . . 6.7.3
pre_defined_marker . . . 6.5.2
pre_defined_presentation_style 6.4.6
pre_defined_size . . . 6.5.3
pre_defined_surface_side_style . . . 6.8.2
pre_defined_symbol . . . 5.4.5
pre_defined_text_font . . . 7.3.9
pre_defined_tile . . . 6.7.12
pre_defined_tile_style . . . 6.7.6
presentable_text . . . 7.2.2
presentation_area . . . 4.4.3
presentation_layer_assignment 4.6.1
presentation_layer_usage . . . 4.6.3
presentation_representation . . . 4.4.2
presentation_representation_relationship . . . 4.4.11
presentation_representation_select 4.3.5
presentation_scaled_placement 7.3.19
presentation_set . . . 4.4.1
presentation_size . . . 4.4.9
presentation_size_assignment_select . . . 4.3.1
presentation_style_assignment . . . 6.4.4
presentation_style_by_context . . . 6.4.5
presentation_style_select . . . 6.3.2
presentation_view . . . 4.4.5
presented_item . . . 4.7.2
presented_item_representation . . . 4.7.1
product_data_representation_view . . . 4.4.7
product_or_presentation_space 6.3.37
rendering_properties_select 6.3.22
representation_item_dependent_layer_assignment . . . 4.6.2
shading_curve_method . . . 6.3.17
shading_surface_method . . . 6.3.21
size_select . . . 6.3.6
squared_or_rounded . . . 6.3.9
staircase_or_linear . . . 7.2.1
state_variable_with_colour . . . 7.3.15
style_context_select . . . 6.3.1
styled_item . . . 6.4.1
surface_rendering_properties 6.8.7
surface_side . . . 6.3.13
surface_side_style . . . 6.8.3
surface_side_style_select . . . 6.3.15
surface_style_boundary . . . 6.8.5
surface_style_control_grid . . . 6.8.10
surface_style_element_select . . . 6.3.15
surface_style_fill_area . . . 6.8.4
surface_style_parameter_line . . . 6.8.11
surface_style_reflectance_ambient . . . 6.8.14
surface_style_reflectance_ambient_diffuse . . . 6.8.15
surface_style_reflectance_ambient_diffuse_specular . . . 6.8.16
surface_style_rendering . . . 6.8.12
surface_style_rendering_with_properties . . . 6.8.13
surface_style_segmentation_curve . . . 6.8.9

surface_style_silhouette . . . 6.8.8
surface_style_transparent . . . 6.8.17
surface_style_usage . . . 6.8.1
surface_tolerance_deviation . . . 6.3.36
surface_tolerance_parameter . . . 6.3.40
symbol_colour . . . 6.10.3
symbol_element_style . . . 6.10.2
symbol_representation . . . 5.4.10
symbol_representation_map . . . 5.4.9
symbol_representation_relationship . . . 5.4.16
symbol_representation_rule . . . 4.8.1
symbol_representation_with_blanking_box . . . 5.4.11
symbol_style . . . 6.10.1
symbol_style_select . . . 6.3.31
symbol_target 5.4.4
table_record_field_representation . . . 5.4.14
table_record_field_representation_with_clipping_box . . . 5.4.15
table_record_representation . . . 5.4.13
table_representation . . . 5.4.12
table_representation_relationship . . . 5.4.17
table_text_relationship . . . 5.5.9
text_alignment . . . 5.3.4
text_delineation . . . 5.3.1
text_font . . . 7.3.5
text_font_family . . . 7.3.6
text_font_in_family . . . 7.3.7
text_justification . . . 6.3.24
text_literal . . . 5.4.28
text_literal_with_associated_curves . . . 5.4.32
text_literal_with_blanking_box . . . 5.4.31
text_literal_with_delineation . . . 5.4.30
text_literal_with_extent . . . 5.4.29
text_or_character . . . 5.3.3
text_path . . . 5.3.6
text_string_representation 5.4.23
text_style . . . 6.9.1
text_style_for_defined_font . . . 6.9.5
text_style_with_box_characteristics . . . 6.9.7
text_style_with_justification 6.9.6
text_style_with_mirror . . . 6.9.10
text_style_with_spacing . . . 6.9.8
tolerance_deviation_select . . . 6.3.34
tolerance_parameter_select . . . 6.3.38
tolerance_select . . . 6.3.32
two_direction_repeat_factor . . . 6.7.15
u_direction_count . . . 6.3.19
v_direction_count . . . 6.3.20
view_dependent_annotation_representation 4.4.8
view_volume . . . 4.5.5

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, структуры представлений

Редактор *В. П. Огурцов*
Технический редактор *Л. А. Гусева*
Корректор *Н. И. Гавришук*
Компьютерная верстка *З. И. Мартыновой*

Изд. лиц. № 02354 от 14.07.2000. Сдано в набор 14.01.2003. Подписано в печать 27.03.2003. Усл. печ. л. 18,60.
Уч.-изд. л. 17,70. Тираж 410 экз. С 10218. Зак. 184

ИПК Издательство стандартов, 107076 Москва, Колодезный пер., 14.
http.//gost.e-ecolog.ru e-mail:gost.e-ecolog.ru
Набрано в Калужской типографии стандартов на ПЭВМ.
Калужская типография стандартов, 248021 Калуга, ул. Московская, 256.
ПЛР № 040138

gost.e-ecolog.ru