ГОСТ Р ИСО/МЭК 9072—1—93

ГОСУДАРСТВЕННЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

СИСТЕМЫ ОБРАБОТКИ ИНФОРМАЦИИ

ПЕРЕДАЧА ТЕКСТА. УДАЛЕННЫЕ ОПЕРАЦИИ

ЧАСТЬ 1. МОДЕЛЬ, НОТАЦИЯ И ОПРЕДЕЛЕНИЕ УСЛУГ

Издание официальное

БЗ 11—92/1121

ГОССТАНДАРТ РОССИИ
Москва

gost.e-ecolog.ru

ГОСТ Р ИСО/МЭК 9072—1—93

ГОСУДАРСТВЕННЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

СИСТЕМЫ ОБРАБОТКИ ИНФОРМАЦИИ

ПЕРЕДАЧА ТЕКСТА. УДАЛЕННЫЕ ОПЕРАЦИИ

Часть 1 МОДЕЛЬ, НОТАЦИЯ И ОПРЕДЕЛЕНИЕ УСЛУГ

Издание официальное

Москва
1994

Предисловие

1 РАЗРАБОТАН И ВНЕСЕН Техническим комитетом по стандартизации ТК 22 «Информационная технология»

2 УТВЕРЖДЕН И ВВЕДЕН В ДЕЙСТВИЕ Постановлением Госстандарта России от 29.12.93 № 292

Настоящий стандарт подготовлен на основе применения аутентичного текста международного стандарта ИСО/МЭК 9072—1—89 «Системы обработки информации. Передача текста. Удаленные операции. Часть 1. Модель, нотация и определение услуг»

3 ВВЕДЕН ВПЕРВЫЕ

© Издательство стандартов, 1994

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 2
4 Сокращения 5
5 Соглашения 5
6 Модель удаленных операций 6
7 Общее описание нотаций и услуг 11
8 Взаимоотношение с другими ЛОП и услугами нижерасположенных уровней 13
9 Нотация удаленных операций 14
10 Определение услуг 23
11 Преобразователь нотаций в услуги 33
12 Последовательность выполнения 36
Приложение А Нотация, поддерживающая спецификацию сервисных-элементов-прикладного-уровня и прикладных-контекстов 41
Приложение В Руководство для разработчиков протоколов прикладного уровня по использованию СЭУО 45

ГОСУДАРСТВЕННЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

Системы обработки информации

ПЕРЕДАЧА ТЕКСТА. УДАЛЕННЫЕ ОПЕРАЦИИ

Часть 1. Модель, нотация и определение услуг

Information processing systems. Text communication. Remote Operations
Part 1: Model, notation and service definition

Дата введения 1994—07—01

1 ОБЛАСТЬ ПРИМЕНЕНИЯ

Настоящий стандарт содержит определение нотации удаленных операций (УО) с целью определения услуг, создаваемых для интерактивных применений. В настоящем стандарте определяются также услуги, обеспечиваемые сервисным элементом удаленных операций (СЭУО). Услуги СЭУО обеспечиваются путем использования протокола СЭУО (ГОСТ Р ИСО/МЭК 9072—2) совместно с услугами сервисного элемента управления ассоциацией (СЭУА) (ИСО 8649), протоколом СЭУА (ИСО 8650), факультативно с услугами сервисного элемента надежной передачи (СЭНП) (ГОСТ Р ИСО/МЭК 9066—1) и протоколом СЭНП (ГОСТ Р ИСО/МЭК 9066—2), а также с услугами-уровня-представления (ГОСТ 34.971).

Относительно соответствия настоящему стандарту никаких требований не предъявляется.

Издание официальное

2 НОРМАТИВНЫЕ ССЫЛКИ

В настоящем стандарте использованы ссылки на следующие стандарты

ГОСТ 28906—91 (ИСО 7498—84, ИСО 7498—84 Доп 1—84) Системы обработки информации Взаимосвязь открытых систем Базовая эталонная модель

ГОСТ 34 971—91 (ИСО 8822—88) Системы обработки информации Взаимосвязь открытых систем Определение услуг уровня представления в режиме с установлением соединения

ГОСТ 34 973—91 (ИСО 8824—87) Системы обработки информации Взаимосвязь открытых систем Спецификация абстрактно синтаксической нотации версии один (АСН 1)

ГОСТ 34 974—91 (ИСО 8825—87) Системы обработки информации Взаимосвязь открытых систем Спецификация базовых правил кодирования для абстрактно синтаксической нотации версии один (АСН 1)

ГОСТ Р ИСО/МЭК 9066—1—93 Системы обработки информации Передача текста Надежная передача Часть 1 Модель и определение услуг

ГОСТ Р ИСО/МЭК 9066—2—93 Системы обработки информации Передача текста Надежная передача Часть 2 Спецификация протокола

ГОСТ Р ИСО/МЭК 9072—2—93 Системы обработки информации Передача текста Удаленные операции Часть 2 Спецификация протокола

ИСО/ТО 8509—87* Системы обработки информации Взаимосвязь открытых систем Соглашения по услугам

ИСО 8649—88* Системы обработки информации Взаимосвязь открытых систем Определение услуг для сервисного элемента управления ассоциацией

ИСО 8650—88* Системы обработки информации Взаимосвязь открытых систем Спецификация протокола для сервисного элемента управления ассоциацией

3 ОПРЕДЕЛЕНИЯ

3.1 Определения, относящиеся к эталонной модели

Настоящий стандарт основывается на концепциях, разработан

---

* До прямого применения данного документа в качестве государственного стандарта Российской Федерации он может быть получен во ВНИИКИ Госстандарта России

ных ГОСТ 28906, и использует следующие определенные в нем термины:

а) прикладной уровень;
б) прикладной-процесс;
в) логический-элемент-прикладного-уровня;
г) сервисный-элемент-прикладного-уровня;
д) протокольный-блок-данных-прикладного-уровня;
е) протокольная-управляющая-информация-прикладного-уровня;
ж) уровень представления;
и) услуга-уровня-представления;
к) соединение-уровня-представления;
л) услуга-сеансового-уровня;
м) соединение-сеансового-уровня;
н) синтаксис-передачи;
п) элемент-пользователя.

3.2 Определения, относящиеся к соглашениям по услугам

В настоящем стандарте применены следующие термины, определенные в ИСО/ТО 8509:

а) поставщик-услуг;
б) пользователь-услуг;
в) подтверждаемая услуга;
г) неподтверждаемая услуга;
д) услуга, инициируемая поставщиком;
е) сервисный-примитив; примитив;
ж) запрос (примитив);
и) индикация (примитив);
к) ответ (примитив) и
л) подтверждение (примитив).

3.3 Определения, относящиеся к услугам уровня представления

В настоящем стандарте применены следующие термины, определенные в ГОСТ 34.971:

а) абстрактный синтаксис;
б) имя абстрактного синтаксиса;
в) имя синтаксиса передачи;
г) контекст уровня представления.

3.4 Определения, относящиеся к управлению ассоциацией

В настоящем стандарте применены следующие термины, определенные в ИСО 8649:

а) прикладная-ассоциация, ассоциация;
б) прикладной контекст;

в) сервисный элемент управления ассоциацией.

3.5 Определения, относящиеся к надежной передаче

В настоящем стандарте применен следующий термин, определенный в ГОСТ Р ИСО/МЭК 9066—1:

а) сервисный элемент надежной передачи.

3.6 Определения, относящиеся к СЭУО

Для целей настоящего стандарта применимы следующие определения.

3.6.1 *Прикладной-логический-объект-инициирующий-ассоциацию; инициатор-ассоциации* — логический-объект-прикладного-уровня, инициирующий прикладную-ассоциацию

3.6.2 *Прикладной-логический-объект-отвечающий-на-ассоциацию; ответчик-ассоциации* — логический-объект-прикладного-уровня, отвечающий на прикладную-ассоциацию, инициированную другим ЛОП.

3.6.3 *Привлекающий-логический-объект-прикладного-уровня; заявитель* — логический-объект-прикладного-уровня, привлекающий удаленную операцию.

3.6.4 *Исполняющий-логический-объект-прикладного-уровня; ис*

3.6.4 *Исполняющий-логический-объект-прикладного-уровня; исполнитель* — логический-объект-прикладного-уровня, выполняющий удаленную операцию, привлеченную другим логическим-объектом-прикладного-уровня.

3.6.5 *Запросчик* — та часть логического-объекта-прикладного-уровня, которая выдает примитив запроса конкретной услуги СЭУО.

3.6.6 *Получатель* — та часть логического-объекта-прикладного-уровня, которая получает примитив индикации конкретной услуги СЭУО.

3.6.7 *Взаимосвязанные-операции* — совокупность операций, формируемых одной порождающей-операцией и одной или несколькими порождаемыми-операциями.

3.6.8 *Порождающая-операция* — операция, при выполнении которой исполнитель может привлечь взаимосвязанные порождаемые-операции, которые должны быть выполнены заявителем порождающей-операции.

3.6.9 *Порождаемая-операция* — операция, которая может быть привлечена исполнителем взаимосвязанной порождающей операции при выполнении порождающей-операции и которая выполняется заявителем порождающей-операции.

3.6.10 *Удаленные операции:*

1) концепция и нотации, обеспечивающие спецификацию интерактивного обмена данными между логическими-объектами-прик-

ладного-уровня. Сюда относится сервисный элемент удаленной операции и преобразование нотации в примитивы услуг используемых сервисных-элементов-прикладного-уровня,

2) совокупность операций-связки, операций-развязки и операции.

3.6.11 *Нотация УО* — нотация, используемая для спецификации удаленных операции, определенных в настоящем стандарте.

3.6.12 *Пользователь-СЭУА* — специфичная для прикладного уровня функция, которая выполняет преобразование операций-связки и операции-развязки нотации-УО в СЭУА.

3.6.13 *Сервисный элемент удаленной операции* — сервисный элемент прикладного-уровня, определяемый в настоящем стандарте.

3.6.14 *Поставщик-СЭУО* — поставщик услуг сервисного элемента удаленных операций

3.6.15 *Пользователь-СЭУО* — специфичная прикладная функция, выполняющая преобразование операции и ошибок нотации УО СЭУО

3.6.16 *Пользователь-СЭНП* — специфичная прикладная функция, выполняющая преобразование операции-связки и операции-развязки нотации УО в СЭНП.

3.6.17 *Интерфейс-операции* — интерфейс логического объекта прикладного уровня между элементом пользователя и сервисными элементами прикладного уровня, определенный как совокупность услуг сервисных элементов прикладного уровня (удаленных операции), доступный для элемента пользователя в нотации-УО.

4 СОКРАЩЕНИЯ

ЛОП — логический-объект-прикладного-уровня;
СЭУА — сервисный элемент управления ассоциацией;
СЭП — сервисный-элемент-прикладного-уровня;
ПБДП — протокольный-блок-данных-прикладного-уровня;
ВОС — взаимосвязь открытых систем;
УО — удаленные операции;
СЭУО — сервисный элемент удаленных операций;
НП — надежная передача;
СЭНП — сервисный элемент надежной передачи.

5 СОГЛАШЕНИЯ

Настоящий стандарт определяет услуги СЭУО, следуя описательным соглашениям, принятым в ИСО/ТО 8509. В разделе 10 настоящего стандарта определение каждой услуги СЭУО содержит

таблицу с перечнем параметров ее примитивов. Для заданного примитива наличие параметра описывается одним из следующих значений:

пробел — не допускается;
О — обязательно;
Ф — факультативно по усмотрению пользователя;
У — условно;
П — представляет собой факультативную услугу СЭУО по усмотрению поставщика.

Кроме того, обозначение (=) указывает, что значение параметра семантически равно значению, указанному слева от него в таблице.

6 МОДЕЛЬ УДАЛЕННЫХ ОПЕРАЦИЙ

В функциональной среде ВОС обмен данными между прикладными процессами представляется в виде обмена данными между парой логических объектов прикладного уровня (ЛОП), использующих услуги уровня представления. Обмен данными между несколькими логическими объектами прикладного уровня по своему характеру является интерактивным. Обычно один логический объект запрашивает выполнение конкретной операции, а другой логический объект пытается выполнить эту операцию и выдать отчет о результатах попытки. В данном разделе вводится концепция удаленных операций как средства для поддерживающих интерактивных применений.

Общая структура операции представляет собой элементарное взаимодействие запрос/ответ. Операции выполняются в контексте прикладной-ассоциации.

На рисунке 1 представлена модель такой структуры.

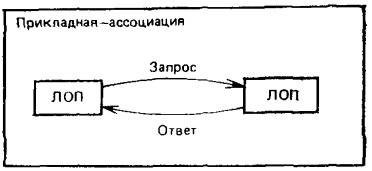


Рисунок 1 — Модель удаленных операций

Операции, привлекаемые одним из ЛОП (заявителем), выполняются другим ЛОП (исполнителем). Операции могут классифицироваться в зависимости от того, ожидается от исполнителя отчет о результатах его действий или нет:

— в случае успешного или безуспешного выполнения операции (в первом случае в ответ сообщается результат, во втором случае сообщается ошибка);
— в случае только безуспешного выполнения операции (ответ не выдается, если операция прошла успешно, и в ответ выдается ошибка, если операция прошла безуспешно);
— в случае только успешного выполнения операции (в ответ выдается результат, если операция прошла успешно, и не выдается никакого ответа, если операция прошла безуспешно);
— не имеет значения (в ответ ничего не выдается независимо от результата выполнения операции)

Операции могут классифицироваться также в зависимости от двух возможных режимов работы: синхронного, при котором заявитель требует ответа от исполнителя до вызова другой операции, и асинхронного, при котором заявитель может продолжать вызов следующих операций, не дожидаясь ответа.

Определены следующие операционные классы:

Операционный класс 1: Синхронные с уведомлением об успешном или безуспешном выполнении операции (результат или ошибка).

Операционный класс 2: Асинхронные с уведомлением об успешном или безуспешном выполнении операции (результат или ошибка).

Операционный класс 3: Асинхронные с уведомлением только о безуспешном выполнении операции (ошибка).

Операционный класс 4: Асинхронные с уведомлением только об успешном выполнении операции (результат).

Операционный класс 5: Асинхронные без уведомления о результате выполнения операции.

Операционный класс каждой операции должен быть согласован между логическими объектами прикладного уровня (например, в стандарте по протоколам прикладного уровня).

В некоторых случаях полезно объединить операции в группу взаимосвязанных операций, которые формируются одной порождающей и одной или несколькими порождаемыми операциями. Исполнитель порождающей операции при ее выполнении может привлечь одну или несколько порождаемых операций или не привлекать их совсем. Заявитель порождающей операции является исполнителем порождаемых операций. Порождаемая операция может рекурсивно перейти в порождающую операцию в другой группе взаимосвязанных операций. На рисунке 2 представлена модель этой концепции.

Прикладная-ассоциация определяет взаимоотношения между парой ЛОП и образуется путем обмена протокольно-управляющей-информацией-прикладного-уровня с использованием услуг-уровня-представления. Объект ЛОП, инициирующий прикладную-ассоциацию, называется инициатором-ассоциации, тогда как ЛОП, отвечающий на прикладную-ассоциацию другого ЛОП, называется ответчиком-ассоциации. Только ЛОП, инициирующий ассоциацию, может разъединить установленную прикладную-ассоциацию.

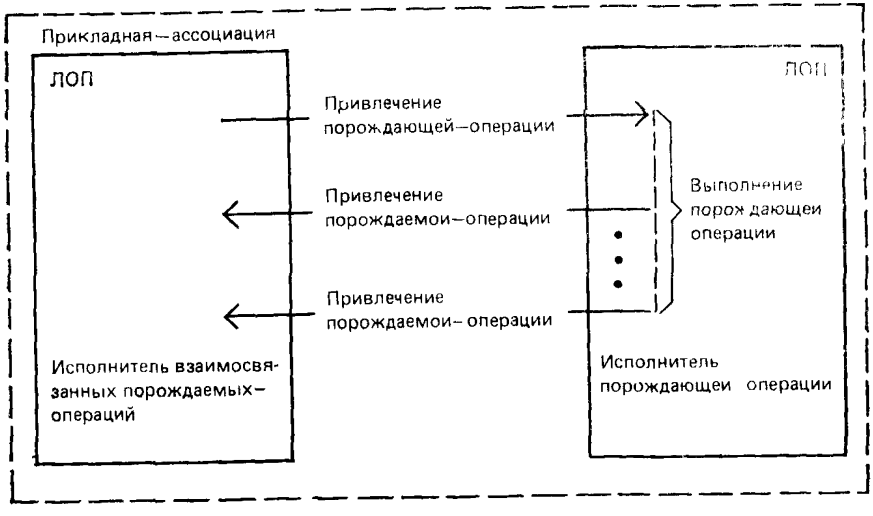


Рисунок 2 — Взаимосвязанные-операции

Прикладные-ассоциации классифицируются в зависимости от логических-объектов-прикладного-уровня, которым разрешено привлекать операции:

Ассоциативный класс 1: Только ЛОП, инициирующий ассоциацию, может привлекать операции.

Ассоциативный класс 2: Только ЛОП, отвечающий на ассоциацию, может привлекать операции.

Ассоциативный класс 3: Оба ЛОП: инициирующий-ассоциацию и отвечающий-на-ассоциацию могут привлекать операции.

Взаимосвязанные-операции требуют ассоциациативного класса 3.

Ассоциациативный класс должен согласовываться между логическими-объектами-прикладного-уровня (например, в стандарте по протоколу прикладного уровня).

Функциональные возможности ЛОП подразделяются на один элемент-пользователя и набор сервисных-элементов-прикладного уровня (СЭП). Каждый СЭП, в свою очередь, может быть разделен на несколько (более простых) СЭП. Взаимодействия между ЛОП описываются с точки зрения использования ими СЭП.

Конкретная комбинация элемента-пользователя и набора СЭП, входящих в ЛОП, определяет прикладной-контекст

На рисунке 3 приведен пример прикладного-контекста, содержащего сервисный элемент удаленных операций (СЭУО). Заметим, что в этом примере не предполагается симметричность применения. Интерактивным применением часто свойственна асимметрия, т. е. любому ЛОП или обоим ЛОП может быть разрешено привлекать операции, и операции, привлекаемые каждым ЛОП, могут быть разными. Руководящие правила, по которым ЛОП может привлекать операции, и тип операций, которые он может привлекать, определены с использованием УО-нотации в стандарте по протоколу прикладного уровня, который определяет прикладной-контекст.

Набор СЭП, доступных для элемента пользователя ЛОП на операционном интерфейсе, определяется с использованием нотации удаленных операций (УО-нотации). УО-нотация основана на макроконцепции, определенной в ГОСТ 34.973. Сложность конкретного набора СЭП зависит от потребностей применения и не ограничивается концепцией удаленных операций.

Важной характеристикой удаленных операций является то, что они обеспечивают применения независимо от услуг обмена данными ВОС. Поскольку их нотация основана на установленных объектно-ориентированных принципах программирования, то возможна разработка автоматических средств объединения удаленных операций в исполнительную функциональную среду конкретных применений.

Элементы СЭП, доступные для элемента-пользователя, нуждаются в обмене данными по прикладной-ассоциации. Управление этой прикладной-ассоциацией (установлением, разъединением, прерыванием) осуществляется либо со стороны сервисного элемента управления ассоциацией (СЭУА), определенного в ИСО 8649, либо со стороны сервисного элемента надежной передачи (СЭНП), определенного в ГОСТ Р ИСО/МЭК 9066—1, и сервисного элемента управления ассоциацией (СЭУА). Обмен данными по прикладной-ассоциации выполняется сервисным элементом удаленных операций (СЭУО), определяемым в настоящем стандарте.

Специфичная для применения функция выполняет преобразование операций, доступных элементу пользователя, либо в ус-

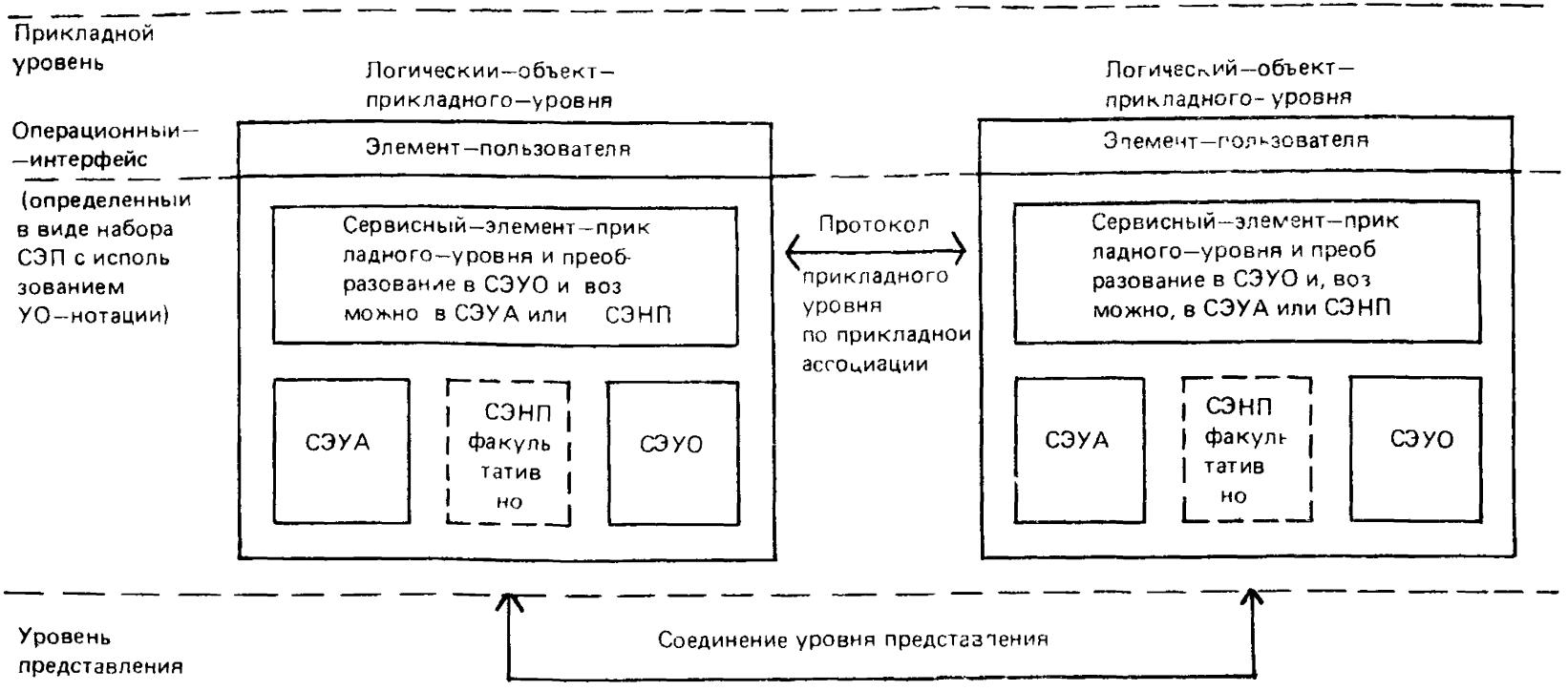


Рисунок 3 — Модель прикладного контекста, включая службу удаленных операций

луги СЭУА либо в услуги СЭНП, а также в услуги СЭУО. Метод преобразования определяется в настоящем стандарте. Можно сказать, что функция, выполняющая преобразование операций в услуги СЭУО или СЭНП, а также в услуги СЭУО, является пользователем СЭУА, СЭНП и СЭУО или пользователем-СЭУА, пользователем-СЭНП и пользователем-СЭУО.

Если СЭНП введен в состав прикладного-контекста, то функцией преобразования является пользователь-СЭНП и пользователь-СЭУО, сам СЭУО является пользователем-СЭНП, СЭНП является пользователем-СЭУА и пользователем-услуг уровня представления, а СЭУА является также пользователем-услуг уровня представления.

Если СЭНП не входит в прикладной-контекст, то функция преобразования является пользователем-СЭУА и пользователем-СЭУО, а СЭУО и СЭУА являются пользователями-услуг уровня представления.

7 ОБЩЕЕ ОПИСАНИЕ НОТАЦИЙ И УСЛУГ

7.1 Общее описание нотаций

Настоящий стандарт определяет УО-нотацию для спецификации прикладного-контекста и соответствующий абстрактно-синтаксический компонент контекста уровня представления.

Функциональные возможности прикладного-контекста предоставляются элементу-пользователя посредством удаленных операций и ошибок, образующих операционный интерфейс.

Следующие типы удаленных-операций образуют операционный интерфейс:

— операция-связки для установления прикладной-ассоциации;
— набор операций и для каждой операции перечень ошибочных ситуаций (отрицательный ответ);
— операция-развязки для разъединения прикладной-ассоциации.

Приведенная в ГОСТ 34.973 нотация абстрактного синтаксиса используется для определения следующих макрокоманд:

а) BIND (СВЯЗКА);
б) UNDIND (РАЗВЯЗКА);
в) OPERATION (ОПЕРАЦИЯ);
г) ERROR (ОШИБКА).

Эти макрокоманды обеспечивают оба вида нотаций и значений нотаций для удаленных операций и ошибок.

Тип нотации макрокоманды BIND обеспечивает возможность спецификации типа операций-связки и типов значений данных

пользователя (если они имеются), подлежащих обмену в фазе установления прикладной ассоциации. Значение нотации макрокоманды BIND обеспечивает возможность спецификации значений данных пользователя, подлежащих обмену в фазе установления прикладной-ассоциации.

Тип нотации макрокоманды UNBIND обеспечивает возможность спецификации типа операций-развязки и типов данных пользователя (если они имеются), подлежащих обмену в фазе разъединения прикладной-ассоциации. Значение нотации макрокоманды UNBIND обеспечивает возможность спецификации значений данных пользователя (если они имеются), подлежащих обмену в фазе разъединения прикладной-ассоциации.

Тип нотации макрокоманды OPERATION обеспечивает возможность спецификации операций и типов данных пользователя, подлежащих обмену по запросу с выдачей положительного ответа. Кроме того, этот тип нотации обеспечивает возможность спецификации списка действительных ситуаций отрицательного ответа. Если данная операция является порождающей, то этот тип нотации дает возможность спецификации перечня порождаемых-операций. Значение нотации макрокоманды OPERATION обеспечивает возможность спецификации идентификатора операции.

Тип нотации макрокоманды ERROR обеспечивает возможность спецификации типов данных пользователя, подлежащих обмену в ситуации отрицательного ответа. Значение нотации макрокоманды ERROR обеспечивает возможность спецификации идентификации ошибки.

Дополнительные макрокоманды, поддерживающие нотацию для спецификации сервисных-элементов-прикладного-уровня и прикладного контекста, определены в приложении А.

7.2 Общее описание услуг

В настоящем стандарте определяются следующие услуги СЭУО:

а) УО-ПРИВЛЕЧЕНИЕ;

б) УО-РЕЗУЛЬТАТ;

в) УО-ОШИБКА;

г) УО-Пл-ОТКЛОНЕНИЕ;

д) УО-Пс-ОТКЛОНЕНИЕ.

Услуга УО-ПРИВЛЕЧЕНИЕ позволяет привлекать ЛОП для запроса операций, подлежащих выполнению ЛОП-исполнителем.

Услуга УО-РЕЗУЛЬТАТ позволяет ЛОП-исполнителю выдавать привлекающему ЛОП положительный ответ об успешном выполнении операции.

Услуга УО-ОШИБКА позволяет исполняющему ЛОП выдавать привлекающему ЛОП отрицательный ответ о безуспешном выполнении операции.

Услуга УО-Пл-ОТКЛОНЕНИЕ позволяет одному из ЛОП отклонять запрос или ответ другого ЛОП, если пользователь СЭУО столкнулся с проблемой.

Услуга УО-Пс-ОТКЛОНЕНИЕ позволяет информировать пользователя-СЭУО о проблемах, обнаруженных поставщиком-СЭУО.

7.3 Преобразование нотаций в услуги

Функцию, выполняющую преобразование макрокоманд OPERATION и ERROR УО-нотации в услуги СЭУО, называют пользователем-СЭУО. В то же время функцию, выполняющую преобразование макрокоманд BIND и UNBIND УО-нотации в услуги СЭУО или СЭНП, называют пользователем-СЭУА или пользователем-СЭНП соответственно.

Спецификация преобразования УО-нотации в используемые услуги СЭУА, СЭНП и СЭУО приведена в разделе 11. Поэтому для стандартов, использующих УО-нотацию для спецификации протокола, нет необходимости определять преобразования в эти используемые услуги.

8 ВЗАИМООТНОШЕНИЕ С ДРУГИМИ ЛОП И УСЛУГАМИ НИЖЕРАСПОЛОЖЕННЫХ УРОВНЕЙ

8.1 Другие сервисные элементы прикладного уровня

Элемент СЭУО используют с другими ЛОП для поддержки задач обработки специфичной интерактивной информации. Поэтому ожидается, что СЭУО войдет в большое число спецификаций прикладного-контекста.

Для взаимно скоординированного использования возможностей услуг-уровня-представления необходима совокупность СЭУО и других ЛОП, входящих в прикладной-контекст.

СЭУО требует наличия прикладной-ассоциации, управляемой со стороны СЭУА.

В спецификации некоторых прикладных-контекстов включен СЭНП.

Протокольная спецификация пользователя-СЭУО использует УО-нотацию. Она определяет один или несколько абстрактных синтаксисов и обеспечивает уникальные абстрактно-синтаксические имена идентификаторов типа объекта для каждого абстрактного синтаксиса.

Если поименованный абстрактный синтаксис определяет операции и ошибки, то ПБДП СЭУО, определенный в ГОСТ Р ИСО/МЭК 9072—2, входит в состав этого поименованного абстрактного синтаксиса. Если определена группа поименованных абстракт-

ных синтаксисов для операций и ошибок, то ПБДП СЭУО входит в состав каждого поименованного абстрактного синтаксиса.

Если поименованный абстрактный синтаксис определяет операцию-связки, то ПБДП, определяемый значением нотации макрокоманды BIND, входит в состав этого поименованного абстрактного синтаксиса Если СЭНП входит в состав прикладного контекста, то блоки ПБДП операций-связки коллективно используют отдельный поименованный абстрактный синтаксис с ПДБП СЭНП, определенными в ГОСТ Р ИСО/МЭК 9066—2

Если поименованный абстрактный синтаксис определяет операцию-развязки, то ПБДП, определенные значением нотации макрокоманды UNDOIND, входят в состав этого поименованного абстрактного синтаксиса

Блоки ПБДП, образованные при спецификации операций-связки, операций-развязки, операций и ошибок, а также ПБДП СЭНП могут коллективно использовать отдельный поименованный абстрактный синтаксис.

8.2 Услуги-уровня-представления

Если прикладной контекст, содержащий СЭНП и СЭУО, определен, то услуги СЭУО не используют услуг-уровня-представления.

Если прикладной контекст, содержащий СЭУО, но не содержащий СЭНП, определен, то услуги СЭУО нуждаются в доступе к услугам П-ДАННЫЕ и в использовании дуплексных функциональных модулей услуг-уровня-представления. Услуги СЭУО не используют никаких других услуг уровня представления и не налагают никаких ограничений на их использование.

Поименованный абстрактный синтаксис, логически связанный с совместимым синтаксисом передачи (согласованным на уровне представления), образует контекст уровня представления.

Значение объективного идентификатора [совместный-исо-мккттт базовый код(1) асн1(1)], определенный в ГОСТ 34.974, может быть использован в качестве имени абстрактного синтаксиса. В этом случае протокольная спецификация пользователя-СЭУО не нуждается ни в имени, ни в спецификации синтаксиса передачи.

9 НОТАЦИЯ УДАЛЕННЫХ ОПЕРАЦИЙ

9.1 Общие положения

Нотация, используемая в настоящем стандарте, определена следующим образом:

— нотация синтаксиса данных и нотация макрокоманд определены в ГОСТ 34.973;

— макрокоманды удаленных операций определены в пункте 9.2 настоящего стандарта.

Операция-связки определяет, где начинается связующий объект (установление прикладной-ассоциации). Если такая связь установлена, операция может привлекаться. Операция-развязки определяет, где разъединяется связующий объект.

Интерактивный протокол определяется с использованием удаленной операции и типов данных с ошибками. В данном разделе определяются такие типы. В нем поясняются также нотационные определения конкретных удаленных операций и конкретных ошибок, о которых может быть выдано уведомление. Эта нотация определяется посредством макросредств, определенных в ГОСТ 34.973. Это макроопределение обеспечивает возможность обобщенной спецификации преобразования в различные исполнительные среды.

Макрокоманды, обеспечивающие возможность спецификаций операций-связки, операций-развязки и ошибок, перечислены на рисунке 4.

```
Нотация-удаленных-операций { совместная-ИСО-МККТТ нота-
ция(0) удаленных операций(4) }
DEFINITIONS ::=
BEGIN
EXPORTS BIND, UNBIND, OPERATION, ERROR;
--макроопределения операций-связки
BIND MACRO ::=
BEGIN
TYPE  NOTATION ::= Argument Result Error
VALUE NOTATION ::= Argument-value | Result-value | Error-
                   value
Argument          ::= empty |   «ARGUMENT» Name type
                                (Argument-type)
                                --Ожидает поступления лю-
                                --бого типа АСН.1 и присва-
                                --ивает его переменной «тип-
                                --аргумента»

END
--Продолжение нотации удаленных операций следует
```

Рисунок 4 (Лист 1) — Формализованное определение типов данных удаленных операций

```
-- Продолжение нотации удаленных операций
Result              ::= empty |   «RESULT» Name type (Result-
                                  type)
                                  -- Ожидает поступления лю-
                                  -- бого типа ACH 1 и присваи-
                                  -- вает его переменной «тип-
                                  -- результата»
Error               ::= empty |   «BIND-ERROR» Name type
                                  (Error-type)
                                  -- Ожидает поступления любо-
                                  -- го типа ACH.1 и присваива-
                                  -- ет его переменной «тип-
                                  -- ошибки»
Name                ::= empty |   identifier
Argument-value      ::= empty |   «ARGUMENT» value (Arg-va-
                                  lue Argument-type)
                                  -- Ожидает поступления зна-
                                  -- чения типа в переменной
                                  -- «тип-аргумент» и присваи-
                                  -- вает его переменной «зна-
                                  -- чение-аргумента»
                                  <VALUE [16] EXPLICIT
                                  Argument-type ::= Arg-
                                  value>
                                  -- Передает окончательное
                                  -- значение в виде явно теги-
                                  -- рованного типа
Result-value        ::= empty     «RESULT» value (Res-value
                                  Result-type)
                                  -- Ожидает поступления зна-
                                  -- чения типа в переменной
                                  -- «тип-результата и присваи-
                                  -- вает его переменной «зна-
                                  -- чение-результата»
                                  <VALUE [17] EXPLICIT
                                  Result-type ::=Res-value>
                                  -- Передает окончательное
                                  -- значение в виде явно теги-
                                  -- рованного типа
END
-- Продолжение нотации удаленных операций следует
```

Рисунок 4 (Лист 2) — Формализованное определение типов данных удаленных операций

Продолжение нотации удаленных операций

Error value = empty | «ERROR» value (ERR value Error type)
Ожидает поступления значения типа в переменной «тип ошибки» и присваивает его переменной «значение ошибки»
<VALUE [18] EXPLICIT Error type = Err value>
Передает окончательно значение в виде явно тегированного типа

Макроопределения операций развязки

```
UNBIND MACRO =
BEGIN
TYPE NOTATION = Argument Result Errors
VALUE NOTATION = Argument value | Result value |
                 Error value
```

Argument = empty | «ARGUMENT» Name type (Argument type)
Ожидает поступления любого типа ACH 1 и присваивает его переменной «тип аргумента»

Result = empty | «RESULT» Name type (Result type)
Ожидает поступления любого типа ACH 1 и присваивает его переменной «тип результата»

Error = empty | «UNBIND ERROR» Name type (Error type)
Ожидает поступления любого типа ACH 1 и присваивает его переменной «тип ошибки»

```
END
```

Продолжение нотации удаленных операций следует

Рисунок 1 (Лист 3) — Формализованное определение типов данных удаленных операций

```
- - Продолжение нотации удаленных операций
Name                : : = empty |  identifier
Argument-value      : : = empty |  «ARGUMENT» value (Arg-
                                   value Argument-type)
                                   - - Ожидает поступления зна-
                                   - - чения типа в переменной
                                   - - «тип-аргумент» и присваи-
                                   - - вает его переменной «зна-
                                   - - чение-аргумента»
                                   <VALUE [19] EXPLICIT
                                   Argument-type : : = Arg-
                                   value>
                                   - - Передает окончательное
                                   - - значение в виде явно теги-
                                   - - рованного типа
Result-value        : : = empty |  «RESULT» value (Res-value
                                   Result-type)
                                   - - Ожидает поступления зна-
                                   - - чения типа в переменной
                                   - - «тип-результата» и присва-
                                   - - ивает его переменной «зна-
                                   - - чение-результата»
                                   <VALUE [20] EXPLICIT
                                   Result-type : : = Res-value>
                                   - - Передает окончательное
                                   - - значение в виде явно теги-
                                   - - рованного типа
Error-value         : : = empty |  «ERROR» value (ERR-value
                                   Error-type)
                                   - - Ожидает поступления зна-
                                   - - чения типа в переменной
                                   - - «тип-ошибки» и присваива-
                                   - - ет его переменной «значе-
                                   - - ние-ошибки»
                                   <VALUE [21] EXPLICIT
                                   Error-type : : = Err-value>
                                   - - Передает окончательное
                                   - - значение в виде явно теги-
                                   - - рованного типа
END
- - Продолжение нотации удаленных операций следует
```

Рисунок 4 (Лист 4) — Формализованное определение типов данных удаленных операций

```
-- Продолжение нотации удаленных операций
-- Макроопределения операций

OPERATION MACRO ::=
BEGIN
TYPE   NOTATION          ::= Argument Result Errors
                             LinkedOperations
VALUE NOTATION           ::= value (VALUE CHOICE {
                                 localValue INTEGER,
                                 globalValue OBJECT
                                 IDENTIFIER})
Argument                 ::= «ARGUMENT» NamedType |
                             empty
Result                   ::= «RESULT» ResultType | empty
ResultType               ::= NamedType | empty
Errors                   ::= «ERRORS» «{» ErrorNames «}»
                             | empty
LinkedOperations         ::= «LINKED» «{» LinkedOpera-
                             tionNames «}» | empty
ErrorNames               ::= ErrorList | empty
ErrorList                ::= Error | ErrorList «,» Error
Error                    ::= value   (ERROR)  -- должен
                             -- указать значение ошибки
                           | type -- должен  указать  тип
                                 -- ошибки,  если  ни  одно
                                 -- значение  ошибки  не
                                 -- определено
LinkedOperationNames     ::= OperationList | empty
OperationList            ::= Operation | OperationList «,»
                             Operation
Operation                ::= value  (OPERATION)  -- дол-
                             -- жен указать значение оши-
                             -- бки
                           | type -- должен  указать  тип
                                 -- операции, если ни од-
                                 -- но значение операции
                                 -- не определено

END
-- Продолжение нотации удаленных операций следует
```

Рисунок 4 (Лист 5) — Формализованное определение типов данных удаленных операций

```
- - Продолжение нотации удаленных операций
NamedType                    : : = identifier type | type
END
- - макроопределения ошибок операций
ERROR MACRO : : =
BEGIN
TYPE   NOTATION              . = Parameter
VALUE NOTATION               · · = value (VALUE CHOICE{
                                     localValue INTEGER,
                                     globalValue OBJECT
                                     IDENTIFIER})
Parameter                    :    «PARAMETER» NamedType |
                                  empty
NamedType                    : : = identifier type | type
END
END - - конец нотации удаленных операций
```

Рисунок 4 (Лист 6) — Формализованное определение типов данных удаленных операций

9 2 Спецификация операций-связки

Отдельные значения данных — аргумент операции-связки может сопровождать запрос на установление прикладной-ассоциации. О выполнении некоторых операций-связки (успешном или безуспешном) выдается уведомление. Для других операций-связки сообщается в случае их безуспешного выполнения, о результате выполнения некоторых операций вообще ничего не сообщается. Отдельное значение данных — результат операции-связки может сопровождать положительный ответ Другое отдельное значение данных — ошибка-связки операции связки может сопровождать отрицательный ответ.

Ключевым словом нотации операции-связки является BIND, за которым факультативно может следовать ключевое слово ARGUMENT, тип аргумента операции-связки, присвоенное ему справочное имя и вид отчетности о результате выполнения операции (если он должен выдаваться). Если сообщается об успешном выполнении операции-связки, то ключевое слово RESULT, тип результата и факультативно назначенное ему справочное имя определены Если же сообщается о безуспешном выполнении операции-связки, то будут определены ключевое слово BIND-ERROR, тип сообщаемой информации об ошибке и факультативно присвоенное ему справочное имя.

Значением нотации операции-связки является либо значение аргумента, либо значение результата, либо значение ошибки. Значением нотации для значения аргумента (если оно используется) является ключевое слово ARGUMENT, за которым следует значение типа аргумента. Значением нотации для значения результата (если оно используется) является ключевое слово RESULT, за которым следует значение типа результата. Значением нотации для значения ошибки (если оно используется) является ключевое слово ERROR, за которым следует значение типа ошибки.

9.3 Спецификация операций-развязки

Отдельное значение данных — аргумент операции-развязки может сопровождать запрос на установление прикладной-ассоциации. О выполнении некоторых операций-развязки (успешном или безуспешном) выдается уведомление. Для других операций-развязки результат сообщается в случае их безуспешного выполнения, о результате выполнения некоторых операций вообще ничего не сообщается. Отдельное значение данных — результат операции-развязки или отдельное значение данных — ошибка-развязки или операция-развязки могут сопровождать ответ.

Ключевым словом нотации операции развязки является UNBIND, за которым факультативно может следовать ключевое слово ARGUMENT, тип аргумента операции-развязки, присвоенное ему справочное имя и вид отчетности о результате выполнения операции-развязки (если он должен выдаваться). Если сообщается об успешном выполнении операции-развязки, то ключевое слово RESULT, тип результата и факультативно назначенное ему справочное имя определены. Если же сообщается о безуспешном выполнении операции-развязки, то будут определены ключевое слово ERROR, тип сообщаемой информации об ошибке и факультативно присваиваемое ему справочное имя.

Значением нотации операции-развязки является либо значение аргумента, либо значение результата, либо значение ошибки. Значением нотации для значения аргумента (если оно используется) является ключевое слово ARGUMENT, за которым следует значение типа аргумента. Значением нотации для значения результата (если оно используется) является ключевое слово RESULT, за которым следует значение типа результата. Значением нотации для значения ошибки (если оно используется) является ключевое слово ERROR, за которым следует значение типа ошибки.

9.4 Спецификация операций

Значение данных типа операции представляет собой идентификатор операции, которую может запросить пользователь-СЭУО в одной открытой системе для ее выполнения равноправным

пользователем-СЭУО в другой открытой системе. Отдельное значение данных — аргумент операции может сопровождать этот запрос. О результате выполнения некоторых операций (успешном или безуспешном) сообщается. Для других операций сообщается только о безуспешном их выполнении. О выполнении некоторых операций ничего не сообщается независимо от результата. Отдельное значение данных — результат операции сопровождает отчет об успешном выполнении операции; отчет о безуспешном выполнении операции идентифицирует возникшие особые условия.

Нотацией типа операции является ключевое слово OPERATION, за которым факультативно может следовать ключевое слово ARGUMENT, тип аргумента операции, факультативно присвоенное ему справочное имя и вид отчетности о результате операции (если она используется). Если сообщается об успешном выполнении операций, то ключевое слово RESULT, факультативно тип результата и факультативно назначенное ему справочное имя определены. Если сообщается о безуспешном выполнении операций, то определены ключевое слово ERROR и справочные имена значений ошибок или типов ошибок, о которых сообщается. Если операция является порождающей из набора операций-связки, то ключевое слово LINKED-OPERATION, справочные имена связующих порождаемых операций или типов порождаемых-операций определены. Ссылка на значения ошибок или значения порождаемых операций предпочтительнее, однако ссылки на типы должны быть использованы только в том случае, если эти значения везде определены (см. 9.6).

Нотацией значения операции является идентификатор операции. Если локальный уникальный идентификатор (локальное значение) достаточен, то идентификатор относится к типу INTEGER (ЦЕЛОЕ). Если глобальный идентификатор (глобальное значение) необходим для того, чтобы обеспечить уникальную идентификацию операций, используемых в нескольких абстрактных синтаксисах, идентификатор относится к типу OBJECT IDENTIFIER.

Порождаемые операции и ошибки, на которые ссылаются конкретные операции, должны коллективно использовать отдельный поименованный абстрактный синтаксис (см. 8.1) с этой операцией, если порождаемые операции или ошибки идентифицированы локальными значениями. Использование глобальных значений не ограничено.

9.5 Спецификация ошибок

Значение данных типа ошибок представляет собой идентификатор особых случаев, о которых пользователь-СЭУО одной открытой системы может уведомить равноправного пользователя-

СЭУО другой открытой системы, где об особых случаях сообщается результатом обработки особых случаев ранее запрошенной операции. Уведомление может сопровождаться отдельным значением и параметром ошибки.

Нотацией типа ошибок является ключевое слово ERROR, за которым факультативно может следовать ключевое слово PARAMETER и тип параметра ошибки, а также факультативно присваиваемое ему справочное имя.

Нотацией значения ошибки является идентификатор ошибки. Если локальный уникальный идентификатор (локальное значение) достаточен, то идентификатор относится к типу INTEGER. Если глобальный уникальный идентификатор (глобальное значение) необходим для того, чтобы обеспечить уникальную идентификацию ошибок, используемых в нескольких абстрактных синтаксисах, то идентификатор относится к типу OBJECT IDENTIFIER.

9.6 Экспорт и импорт операций и ошибок

Значения операций и значения ошибок должны быть уникальными в пределах поименованного абстрактного синтаксиса. Если операции и ошибки определены в нескольких модулях ACH.1 и введены в модуль, определяющий конкретный поименованный абстрактный синтаксис, то применимо одно из следующих правил:

а) Если локальные значения использованы и экспортированы, то обеспечение их уникальности возлагается на разработчика импортирующего модуля.

б) Модуль может определять и экспортировать типы операций и типы ошибок. Значения операций и значения ошибок присваиваются в модуле, импортирующем эти типы. Каждому типу операции и типу ошибки должно быть присвоено отдельное значение.

в) Если глобальные значения присвоены и экспортированы, то уникальность обеспечена.

Однако для конфликтующих локальных значений могут быть использованы различные поименованные абстрактные синтаксисы.

10 ОПРЕДЕЛЕНИЕ УСЛУГ

Услуги СЭУО перечислены в таблице 1.

Идентификация поименованного абстрактного синтаксиса предполагается во всех услугах СЭУО, однако этот вопрос является частным и не входит в предмет рассмотрения настоящего стандарта.

Таблица 1 — **Услуги СЭУО**

Услуга	Тип
УО-ПРИВЛЕЧЕНИЕ	Неподтверждаемая
УО-РЕЗУЛЬТАТ	Неподтверждаемая
УО ОШИБКА	Неподтверждаемая
УО Пл-ОТКЛОНЕНИЕ	Неподтверждаемая
УО-Пс ОТКЛОНЕНИЕ	Инициированная поставщиком

10.1 Услуга УО-ПРИВЛЕЧЕНИЕ

Услуга УО-ПРИВЛЕЧЕНИЕ используется одним из пользователей-СЭУО (заявителем), чтобы побудить выполнение ОПЕРАЦИИ другим пользователем-СЭУО (исполнителем). Эта услуга относится к неподтверждаемым.

Соответствующая структура услуги содержит два сервисных примитива, как это показано на рисунке 5.

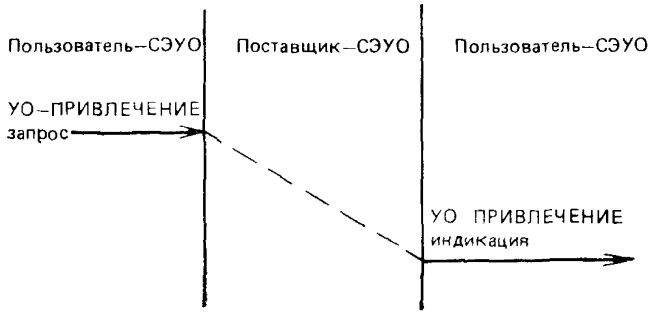


Рисунок 5 — Примитивы услуги УО-ПРИВЛЕЧЕНИЕ

10.1.1 *Параметры услуги УО-ПРИВЛЕЧЕНИЕ*

Параметры услуги УО-ПРИВЛЕЧЕНИЕ перечислены в таблице 2.

Таблица 2 — **Параметры услуги УО-ПРИВЛЕЧЕНИЕ**

Имя параметра	*Запрос*	*Индикация*
Значение операции	О	О(=)
Операционный класс	Ф	
Аргумент	Ф	У(=)
Ид-привлечения	О	О(=)
Ид-взаимосвязи	Ф	У(=)
Приоритет	Ф	

10.1.1.1 Значение операции

Этот параметр является идентификатором привлекаемой операции. Его значение должно быть согласовано между пользователями-СЭУО. Он должен быть обеспечен запросчиком услуги.

10.1.1.2 Операционный класс

Этот параметр определяет, какой ответ ожидается: синхронный или асинхронный, а также характер ожидаемого ответа, т. е. результат, ошибка, или отсутствие ответа (см. раздел 6). Он должен обеспечиваться запросчиком услуги. Он используется только для оптимизации управления полномочиями (см. 8.1.1 ГОСТ Р ИСО/МЭК 9072—2).

10.1.1.3 Аргумент

Этот параметр является аргументом привлекаемой операции. Ее тип должен быть согласован между пользователями-СЭУО. Он должен обеспечиваться запросчиком услуги.

10.1.1.4 Ид-привлечения

Этот параметр идентифицирует запрос услуги УО-привлечение и используется для увязки этого запроса с соответствующим ответом (услугами УО-РЕЗУЛЬТАТ, УО-ОШИБКА, УО-Пл-ОТКЛОНЕНИЕ и УО-Пс-ОТКЛОНЕНИЕ) или с привлечением связанной порождаемой операции (УО-ПРИВЛЕЧЕНИЕ). Он должен обеспечиваться запросчиком услуги.

Указанный параметр разграничивает различные запросы услуги, которые запросчик может активизировать (асинхронные операции). Запросчик может начать повторное использование значений Ид-привлечения при каждом их выборе с тем ограничением, что он не может повторно использовать значение Ид-привлечения, присвоенное ранее запросу ожидаемой им, но еще не полученной услуги, ответу или привлечению связанной порождаемой операции.

Пользователь СЭУО, которому выдан примитив УО-ПРИВЛЕЧЕНИЕ индикация, предполагает, что значение Ид-привлечения, которое нарушает вышеуказанное правило, продублировано, и поэтому не выполняет привлеченную операцию, а вместо этого отклоняет дублированное привлечение.

При использовании операций классов 3, 4 или 5 запросчик этой услуги может повторно использовать значение Ид-привлечения либо по истечении достаточно большого промежутка времени, либо если ответ передан — другими способами (например, в результате завершения операции).

В некоторых прикладных контекстах равноправные пользователи-СЭУО могут обмениваться значениями Ид-привлечения. Для этого тип параметра Ид-привлечения экспортируется модулем,

определяющим абстрактный синтаксис удаленных операций в разделе 9 ГОСТ Р ИСО/МЭК 9072—2.

10.1.1.5 Ид-взаимосвязи

При наличии этого параметра привлеченная операция является порождаемой операцией и этот параметр идентифицирует привлечение взаимосвязанной порождаемой операции. Он должен обеспечиваться запросчиком услуги. Его значением служит значение параметра «Ид-привлечения» примитива УО-ПРИВЛЕЧЕНИЕ индикация порождающей операции.

10.1.1.6 Приоритет

Этот параметр определяет приоритет, назначенный передаче соответствующего ПБДП относительно других ПБДП, подлежащих обмену между ЛОП. Чем меньше значение, тем выше приоритет. Если в очереди на передачу ожидают несколько ПБДП одного и того же приоритета, то они передаются в порядке «первым пришел — первым вышел».

Примечания

1 В ассоциации с двунаправленной поочередной передачей параметр «приоритет» имеет тот смысл, что упорядочивает по приоритетам передаваемые ПБДП и может использоваться для того, чтобы определить, когда запросить «полномочие» на передачу ПБДП. Параметр «приоритет» может иметь также локальную значимость в ассоциации с двунаправленной одновременной передачей.

2 Приоритет ответа (УО-РЕЗУЛЬТАТ, УО-ОШИБКА и УО-ПЛ ОТКЛОНЕНИЕ) обычно должен быть более высоким (меньшим по номеру), чем приоритет соответствующего привлечения.

10.2 Услуга УО-РЕЗУЛЬТАТ

Услуга УО-РЕЗУЛЬТАТ используется пользователем СЭУО для ответа на предыдущий примитив УО-ПРИВЛЕЧЕНИЕ индикация в случае успешного выполнения операции. Эта услуга является неподтверждаемой.

Соответствующая структура услуги содержит два сервисных примитива, как показано на рисунке 6.

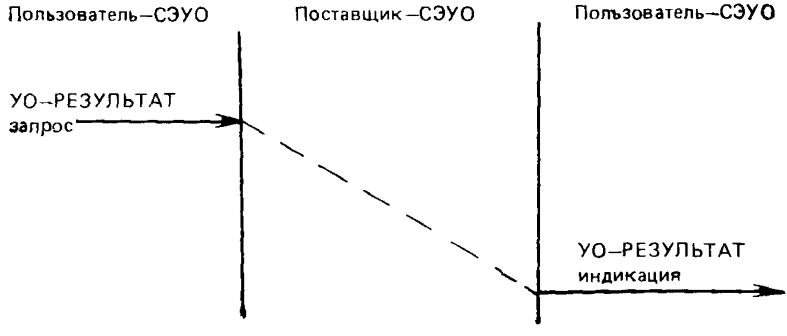


Рисунок 6 — Примитивы услуги УО-РЕЗУЛЬТАТ

10.2.1 *Параметры услуги УО-РЕЗУЛЬТАТ*

Параметры услуги УО-РЕЗУЛЬТАТ перечислены в таблице 3.

Таблица 3 — **Параметры услуги УО-РЕЗУЛЬТАТ**

Имя параметра	Запрос	Индикация
Значение-операции	Ф	У(=)
Результат	Ф	У(=)
Ид привлечения	О	О(—)
Приоритет	Ф	

10.2.1.1 Значение-операции

Этот параметр является идентификатором привлеченной и успешно выполненной операции. Он должен обеспечиваться запросчиком услуги. Он имеет то же значение, что и в соответствующем примитиве УО-ПРИВЛЕЧЕНИЕ индикация. Этот параметр должен присутствовать только в том случае, если присутствует параметр «результат».

10.2.1.2 Результат

Этот параметр представляет собой результат привлеченной и успешно выполненной операции. Его тип должен быть согласован между пользователями-СЭУО. Этот параметр должен обеспечиваться запросчиком услуги.

10.2.1.3 Ид-привлечения

Этот параметр идентифицирует соответствующее привлечение (см. 10.1.1.4) и должен обеспечиваться запросчиком услуги. Он имеет то же значение, что и в соответствующем примитиве УО-ПРИВЛЕЧЕНИЕ индикация.

10.2.1.4 Приоритет

Этот параметр определяет приоритет, назначенный для передачи соответствующего ПБДП (см. 10.1.1.6).

10.3 Услуга УО-ОШИБКА

Услуга УО-ОШИБКА используется пользователем-СЭУО для ответа на предыдущий примитив УО-ПРИВЛЕЧЕНИЕ индикация в случае безуспешного выполнения операции. Эта услуга является неподтверждаемой.

Соответствующая структура услуги содержит два сервисных примитива, как показано на рисунке 7.

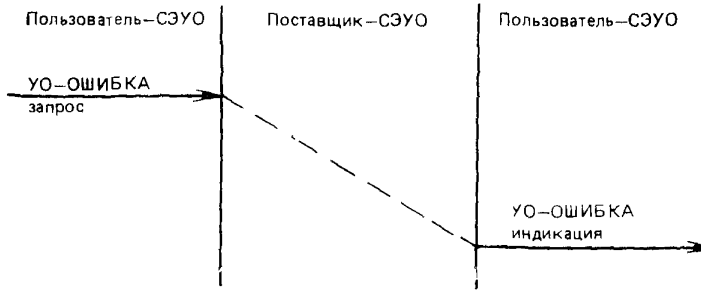


Рисунок 7 — Примитивы услуги УО-ОШИБКА

10.3.1 *Параметры услуги УО-ОШИБКА*

Параметры услуги УО-ОШИБКА перечислены в таблице 4

Таблица 4 — **Параметры услуги УО-ОШИБКА**

Имя параметра	Запрос	Индикация
Значение-ошибки	О	О(=)
Параметр-ошибки	Ф	Ф(=)
Ид привлечения	О	О(=)
Приоритет	Ф	

10.3.1.1 Значение-ошибки

Этот параметр идентифицирует ошибку, возникающую при выполнении операции. Ее значение должно быть согласовано между пользователями-СЭУО. Этот параметр должен обеспечиваться запросчиком услуги.

10.3.1.2 Параметр-ошибки

Этот параметр обеспечивает дополнительную информацию относительно ошибки. Тип ошибки (если он используется) должен быть согласован между пользователями-СЭУО Этот параметр должен обеспечиваться запросчиком услуги.

10.3.1.3 Ид-привлечения

Этот параметр идентифицирует соответствующее привлечение (см 10.1.1.4) и должен обеспечиваться запросчиком услуги. Он имеет то же значение, что и в соответствующем примитиве УО-ПРИВЛЕЧЕНИЕ индикация.

10.3.1.4 Приоритет

Этот параметр определяет приоритет, назначенный для передачи соответствующего ПБДП (см. 10.1.1.6)

10 4 Услуга УО-Пл-ОТКЛОНЕНИЕ

Услуга УО-Пл-ОТКЛОНЕНИЕ используется пользователем-СЭУО для отклонения запроса (УО-ПРИВЛЕЧЕНИЕ индикация) другого пользователя-СЭУО в случае обнаружения им какой-либо проблемы Услуга УО-Пл-ОТКЛОНЕНИЕ может использоваться также пользователем-СЭУО для отклонения ответа (УО-РЕЗУЛЬТАТ индикация, УО-ОШИБКА индикация) другого пользователя-СЭУО Однако, чтобы не допустить нарушения правил последовательности других СЭП в некоторых прикладных контекстах, для отклонения ответа пользователь-СЭУО может выбрать вариант неиспользования услуги УО Пл-ОТКЛОНЕНИЕ. Эта услуга является неподтверждаемой.

Соответствующая структура услуги содержит два сервисных примитива, как показано на рисунке 8.

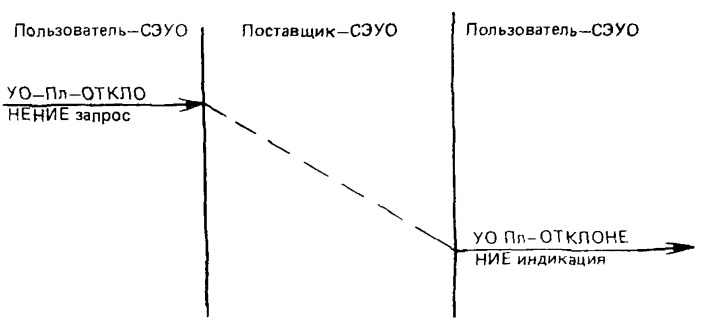


Рисунок 8 — Примитивы услуги УО-Пл-ОТКЛОНЕНИЕ

10 4.1 *Параметры услуги УО-Пл-ОТКЛОНЕНИЕ*

Параметры услуги УО-Пл-ОТКЛОНЕНИЕ перечислены в таблице 5.

Таблица 5 — **Параметры услуги УО-Пл-ОТКЛОНЕНИЕ**

Имя параметра	Запрос	Индикация
Причина отклонения	О	О(=)
Идентификатор привлечения	О	О(=)
Приоритет	Ф	

10.4.1.1 Причина-отклонения

Этот параметр определяет причину отклонения следующим образом:

а) Проблема-привлечения: отклонение пользователем примитива УО-ПРИВЛЕЧЕНИЕ индикация со значениями:

— дублированное привлечение: означает, что параметр Ид-привлечения нарушает правила назначения, изложенные в 10.1.1.4;

— нераспознаваемая операция: означает, что данная операция не входит в число операций, согласованных между пользователями-СЭУО;

— аргумент неправильного типа: означает, что тип обеспечиваемого аргумента операции не согласован между пользователями-СЭУО;

— ограничение ресурсов: исполняющий пользователь-СЭУО не способен выполнить привлеченную операцию ввиду ограничения ресурсов;

— разъединение-от-инициатора: инициатор ассоциации не желает выполнять привлеченную операцию, поскольку он собирается разъединить ассоциацию-прикладного-уровня;

— нераспознаваемый-Ид-связи: означает, что ни одна из выполняемых операций не имеет параметра Ид-привлечения, равного определенному параметру Ид-связи;

— неожидаемый-ответ-связи: означает, что привлеченная операция, указанная параметром Ид-связи, не является порождающей операцией;

— неожидаемая-порождаемая-операция: означает, что привлеченная порождаемая операция не входит в число тех, которые привлеченная порождающая-операция разрешила указанием взаимосвязи.

б) Проблема-возврата-результата: пользователь отклоняет примитив УО-РЕЗУЛЬТАТ индикация со значениями:

— нераспознаваемое-привлечение: означает, что среди выполняемых операций нет операции с определенным параметром Ид-привлечения;

— неожидаемый-результат-ответа: означает, что привлеченная операция не сообщила результат;

— результат-ошибочного-типа: означает, что тип обеспеченного параметра «результат» не относится к согласованным между пользователями-СЭУО.

в) Проблема-возврата-ошибки: пользователь отклоняет примитив УО-ОШИБКА индикация со значениями:

— нераспознаваемое-привлечение: означает, что ни одна из выполняемых операций не имеет заданного параметра Ид-привлечения;

— неожидаемый-ответ-об-ошибке: означает, что привлеченная операция не уведомила о безуспешности ее выполнения;

— нераспознанная-ошибка: означает, что переданная ошибка не относится к ошибкам, согласованным между пользователями-СЭУО,

— неожидаемая-ошибка: означает, что переданная ошибка не относится к тем, о которых может сообщить привлеченная операция;

— параметр-ошибочного-типа: означает, что обеспеченный тип параметра «ошибка» не относится к согласованным между пользователями-СЭУО.

Этот параметр должен обеспечиваться запросчиком услуги.

10.4.1.2 Ид-привлечения

Этот параметр идентифицирует соответствующее привлечение (см. 10.1.1.4) и должен обеспечиваться запросчиком услуги. Его значением является значение отклоненного примитива УО-ПРИВЛЕЧЕНИЕ индикация, УО-РЕЗУЛЬТАТ индикация или УО-ОШИБКА индикация.

10.4.1.3 Приоритет

Этот параметр определяет приоритет, назначенный для передачи соответствующего ПБДП (см. 10.1.1.6).

10.5 Услуга УО-Пс-ОТКЛОНЕНИЕ

Услуга УО-Пс-ОТКЛОНЕНИЕ используется пользователем-СЭУО для того, чтобы известить пользователя-СЭУО о проблеме, обнаруженной поставщиком-СЭУО. Эта услуга относится к услугам, инициируемым поставщиком.

Соответствующая структура услуги содержит один сервисный примитив, как показано на рисунке 9.

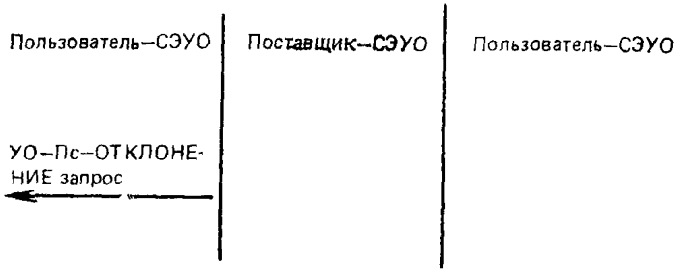


Рисунок 9 — Примитив услуги УО-Пс-ОТКЛОНЕНИЕ

10.5.1 Параметры услуги УО Пс-ОТКЛОНЕНИЕ

Параметры услуги УО-Пс-ОТКЛОНЕНИЕ перечислены в таблице 6.

Таблица 6 — **Параметры услуги УО-Пс-ОТКЛОНЕНИЕ**

Имя параметра	Индикация
Ид-привлечения	Ф
Возвращенные-параметры	Ф
Причина отклонения	Ф

10.5.1 1 Ид-привлечения

Этот параметр идентифицирует соответствующее привлечение (см. 10.1.1.4). Он обеспечивается поставщиком-СЭУО. Его значение равно значению отклоненных примитивов УО-ПРИВЛЕЧЕНИЕ запрос, УО-РЕЗУЛЬТАТ запрос, УО-ОШИБКА запрос или УО-Пл-ОТКЛОНЕНИЕ запрос. Этот параметр может быть опущен, если он недоступен.

10.5.1 2 Возвращенные-параметры

Этот параметр содержит параметры примитивов УО-ПРИВЛЕЧЕНИЕ запрос, УО-ОШИБКА запрос или УО-ОТКЛОНЕНИЕ-Пл запрос, если соответствующий ПБДП не может быть передан поставщиком-СЭУО. Этот параметр и параметр «причина-отклонения» взаимно исключают друг друга.

10 5.1.3 Причина-отклонения

Этот параметр определяет причину отклонения следующим образом:

а) Общая-проблема: отклонение поставщиком ПБДП со значениями:

— нераспознаваемый-ПБДП: означает, что тип ПБДП так, как он видится его идентификатором типа, не входит в число четырех, определенных ГОСТ Р ИСО/МЭК 9072—2;

— ПБДП-ошибочного типа: означает, что структура ПБДП не соответствует ГОСТ Р ИСО/МЭК 9072—2;

— неправильно-построенный-ПБДП: означает, что структура ПБДП не соответствует стандартной нотации и кодированию, определенным ГОСТ 34.973 и ГОСТ 34.974.

Этот параметр обеспечивается поставщиком-СЭУО. Данный параметр и параметр «возвращенные-параметры» взаимно исключают друг друга.

11 ПРЕОБРАЗОВАНИЕ НОТАЦИЙ В УСЛУГИ

11.1 Прикладной контекст и операции

В данном разделе описывается, какие образом прикладной контекст определяется с помощью нотаций, обеспечиваемых макрокомандами в разделе 9.

Такая спецификация прикладного контекста содержит:

а) операцию-связки, определенную посредством макрокоманды BIND, и

б) операцию-развязки, определенную посредством макрокоманды UNBIND, и

в) набор операций, определенных посредством макрокоманды OPERATION, и

г) набор ошибок, относящихся к операциям и определенных посредством макрокоманды ERROR.

Удаленные операции (т. е. операция-связки, операция-развязки и операции) могут быть привлечены элементом-пользователя.

Инициирующий-ассоциацию элемент-пользователя устанавливает прикладную-ассоциацию путем привлечения операции-связки. В установленной прикладной-ассоциации операции могут привлекаться элементом-пользователя. Когда инициирующий-ассоциацию элемент-пользователя желает разъединить прикладную-ассоциацию, он привлекает операцию-развязки.

11.2 Преобразование удаленных операций в услуги СЭУА, услуги СЭНП и услуги СЭУО

Операция-связки и операция-развязки преобразуются либо в услуги СЭУА, либо в услуги СЭНП.

Операции преобразуются в услуги СЭУО.

11.2.1 *Преобразование услуг СЭУА*

Операция-связки преобразуется в услугу П-АССОЦИАЦИЯ, а операция-развязки — в услугу П-РАЗЪЕДИНЕНИЕ.

11.2.1.1 Преобразование операции-связки

Операция-связки преобразуется в услугу П-АССОЦИАЦИЯ.

11.2.1.1.1 Привлечение операции-связки

Привлечение операции-связки преобразуется в сервисные примитивы П-АССОЦИАЦИЯ запрос и П-АССОЦИАЦИЯ индикация.

Значение аргумента операции-связки преобразуется в параметр «информация пользователя» указанных сервисных примитивов

11.2.1.1.2 Ответ операции-связки

Ответ операции-связки преобразуется в сервисные примитивы П-АССОЦИАЦИЯ ответ и П-АССОЦИАЦИЯ подтверждение.

При успешном выполнении операции-связки параметр «результат» указанных сервисных примитивов является «воспринятым» и

значение результата операции-связки преобразуется в параметр «информация пользователя» этих сервисных примитивов.

При безуспешном выполнении операции-связки значением параметра «результат» этих сервисных примитивов является «отклонено (устойчивое условие)», а значение ошибки операции-связки преобразуется в параметр «информация пользователя» этих же сервисных примитивов.

11.2.1 2 Преобразование операции-развязки

Операция-развязки преобразуется в услугу П-РАЗЪЕДИНЕНИЕ.

11 2.1 2.1 Привлечение операции-развязки

Привлечение операции-развязки преобразуется в сервисные примитивы П-РАЗЪЕДИНЕНИЕ запрос и П-РАЗЪЕДИНЕНИЕ индикация

Значение аргумента операции-развязки преобразуется в параметр «информация пользователя» указанных сервисных примитивов Параметр «причина» этих сервисных примитивов имеет значение «норма».

11 2 1 2 2 Ответ операции-развязки

Ответ операции-развязки преобразуется в сервисные примитивы П-РАЗЪЕДИНЕНИЕ ответ и П-РАЗЪЕДИНЕНИЕ подтверждение

При успешном выполнении операции-развязки параметр «причина» указанных сервисных примитивов имеет значение «норма», значение результата операции-развязки преобразуется в параметр «информация пользователя» этих сервисных примитивов, а параметр «результат» имеет значение «положительный».

При безуспешном выполнении операции-развязки параметр «результат» указанных сервисных примитивов имеет значение «не закончено», значение ошибки операции-развязки преобразуется в параметр «информация пользователя» этих сервисных примитивов, а параметр «результат» имеет значение «положительный».

11.2.2 *Преобразование в услуги СЭНП*

Операция-связки преобразуется в услугу НП-ОТКРЫТИЕ, а операция-развязки — в услугу НП-ЗАКРЫТИЕ.

11 2.2.1 Преобразование операции-связки

Операция-связки преобразуется в услугу НП-ОТКРЫТИЕ.

11 2 2.1.1 Привлечение операции-связки

Привлечение операции-связки преобразуется в сервисные примитивы НП-ОТКРЫТИЕ запрос и НП-ОТКРЫТИЕ индикация.

Значение аргумента операции-связки преобразуется в параметр «данные пользователя» указанных сервисных примитивов.

Параметр «режим-диалога» имеет значение «двунаправленный поочередный».

11.2.2.1.2 Ответ операции-связки

Ответ операции-связки преобразуется в сервисные примитивы НП-ОТКРЫТИЕ ответ и НП-ОТКРЫТИЕ подтверждение.

При успешном выполнении операции-связки параметр «результат» указанных сервисных примитивов имеет значение «принято», а значение результата операции-связки преобразуется в параметр «данные пользователя» этих сервисных примитивов.

При безуспешном выполнении операции-связки параметр «результат» указанных сервисных примитивов имеет значение «отклонено (устойчивое условие)», а значение ошибки операции-связки преобразуется в параметр «данные пользователя» этих сервисных примитивов.

11.2.2.2 Преобразование операции-развязки

Операция-развязки преобразуется в услугу НП-ЗАКРЫТИЕ.

11.2.2.2.1 Привлечение операции-развязки

Привлечение операции-развязки преобразуется в сервисные примитивы НП-ЗАКРЫТИЕ запрос и НП-ЗАКРЫТИЕ индикация

Значение аргумента операции-развязки преобразуется в параметр «данные пользователя» указанных сервисных примитивов. Параметр «причина» этих сервисных примитивов имеет значение «норма».

11.2.2.2.2 Ответ операции-развязки

Ответ операции-развязки преобразуется в сервисные примитивы НП-ЗАКРЫТИЕ ответ и НП-ЗАКРЫТИЕ подтверждение.

При успешном выполнении операции-развязки параметр «причина» указанных сервисных примитивов имеет значение «норма», а значение результата операции-развязки преобразуется в параметр «данные пользователя» этих сервисных примитивов.

При безуспешном выполнении операции-развязки параметр «результат» указанных сервисных примитивов имеет значение «не закончено», а значение ошибки операции-развязки преобразуется в параметр «данные пользователя» этих сервисных примитивов.

11.2.3 *Преобразование в услуги СЭУО*

Операция преобразуется в услуги СЭУО.

11.2.3.1 Привлечение операции

Привлечение операции преобразуется в услугу УО-ПРИВЛЕЧЕНИЕ.

Значение, присвоенное операции, преобразуется в параметр «значение-операции» этой услуги. Значение поименованного-типа в разделе ARGUMENT макрокоманды OPERATION преобразуется в параметр «аргумент» этой услуги.

11.2.3.2 Ответ операции

При успешном выполнении операции ответ преобразуется в услугу УО-РЕЗУЛЬТАТ.

Значение поименованного типа в разделе RESULT макрокоманды OPERATION преобразуется в параметр «результат» этой услуги.

При безуспешном выполнении операции ответ преобразуется в услугу УО-ОШИБКА.

В этом случае может быть применена одна из ошибок из списка идентификаторов наименований ошибок раздела ERROR макрокоманды OPERATION. Значение, присвоенное применимой ошибке, преобразуется в параметр «ошибка» этой услуги. Значение поименованного типа в разделе PARAMETER макрокоманды ERROR применимой ошибки преобразуется в параметр «параметр-ошибки» этой услуги.

12 ПОСЛЕДОВАТЕЛЬНОСТЬ ВЫПОЛНЕНИЯ

В настоящем разделе определяется взаимодействие между удаленными операциями, а также взаимодействие между услугой СЭУА и услугами СЭУО.

12.1 Информация о последовательности выполнения удаленных операций

12.1.1 *Операция-связки*

12.1.1.1 Ограничения использования

Операция-связки не используется при установлении прикладной-ассоциации. Успешное выполнение операции-связки приводит к установлению прикладной-ассоциации.

12.1.1.2 Разрушаемые удаленные операции

Операция-связки не разрушает никаких удаленных операций.

12.1.1.3 Разрушающие удаленные операции

Разрушающих удаленных операций нет.

12.1.1.4 Конфликты

Конфликты операции-связки возникают, когда элементы-пользователя в обоих ЛОП одновременно привлекают друг у друга операцию-связки. В этом случае устанавливаются две независимые прикладные-ассоциации.

12.1.2 *Операция-развязки*

12.1.2.1 Ограничения использования

Операция-развязки используется только при установлении прикладной-ассоциации. Она используется только тем элементом пользователя, который привлекает операцию-связки, и только в том случае, если нет невыданных ответов от операционного класса 1 или 2.

Установление прикладной-ассоциации прекращается независимо от успешности выполнения операции-развязки.

12.1.2.2 Разрушаемые удаленные операции

Операция-развязки не разрушает удаленных операций в случае ассоциативного класса 2 и операционного класса 1 или 2.

Во всех остальных случаях операция-развязки может разрушить операции. Однако, если в конкретном прикладном-контексте используется операционный класс 3, 4 или 5 и/или ассоциативный класс 2 или 3, то предполагается, что либо разрушение приемлемо, либо прикладной-контекст обеспечивает операции, предотвращающие разрушения.

12.1.2 3 Разрушающие удаленные операции

Разрушающих удаленных операций нет.

12.1.2.4 Конфликты

Ввиду того, что только инициатор-ассоциации может разъединить прикладную-ассоциацию, конфликты отсутствуют.

12.1.3 *Операции*

12 1.3.1 Ограничения использования

Операции используются только при установлении прикладной-ассоциации.

12.1.3.2 Разрушаемые удаленные операции

Операции не разрушают никаких удаленных операций.

12.1.3.3 Разрушающие удаленные операции

Операции могут разрушаться операцией-развязки (см. 12.1.2.2).

12.1.3.4 Конфликты

При выполнении операций конфликты отсутствуют.

12.1.4 *Дополнительная информация о последовательности выполнения*

Разрушающие услуги не наблюдаемы на операционном-интерфейсе. Однако операции могут быть разрушены услугами (см. 12.2).

Операции-связки и операции-развязки разрушаются услугами П-ПРЕРЫВАНИЕ, П-Пс-ПРЕРЫВАНИЕ, НП-Пл-ПРЕРЫВАНИЕ и НП-Пс-ПРЕРЫВАНИЕ.

Операции разрушаются услугами П-ПРЕРЫВАНИЕ, П-Пс-ПРЕРЫВАНИЕ, НП-Пл-ПРЕРЫВАНИЕ, НП-Пс-ПРЕРЫВАНИЕ, УО-Пл-ОТКЛОНЕНИЕ и УО-Пс-ОТКЛОНЕНИЕ.

Кроме того, операции могут быть разрушены услугой П-РАЗЪЕДИНЕНИЕ. Но это отражает только случай разрушения операции операцией-развязки. Разрушаемые операции не рассматриваются в разрушаемых услугах (12.2).

Поскольку все удаленные операции преобразуются в услуги, а разрушающие удаленные операции (операции-развязки) представ-

лены услугами, то никакие разрушающие операции не рассматриваются в разделах по разрушающим услугам (12.2).

12.2 Последовательность выполнения услуг

12.2.1 *Услуги СЭУА*

Последовательность выполнения услуг СЭУА описана в ИСО 8649. Дополнительная информация представлена в данном разделе.

12.2.1.1 Разрушаемые услуги СЭУО

Помимо разрушаемых услуг, определенных в ИСО 8649, все услуги СЭУО, за исключением услуги УО-Пс-ОТКЛОНЕНИЕ, разрушаются услугами П-ПРЕРЫВАНИЕ и П-Пс-ПРЕРЫВАНИЕ и могут быть разрушены услугой П-РАЗЪЕДИНЕНИЕ (см. 12.2 3.6).

12.2.1.2 Разрушающие услуги СЭУО

Разрушающих услуг СЭУО нет.

12.2.2 Услуги СЭНП

Информация о последовательности выполнения услуг СЭНП содержится в ГОСТ Р ИСО/МЭК 9066—1. Дополнительная информация представлена в данном разделе.

12.2.2.1 Разрушаемые услуги СЭУО

Помимо разрушаемых услуг, определенных в ГОСТ Р ИСО/МЭК 9066—1, все услуги СЭУО, за исключением услуги УО-Пс-ОТКЛОНЕНИЕ, разрушаются услугами НП-Пл-ПРЕРЫВАНИЕ, НП-Пс-ПРЕРЫВАНИЕ и услугами отрицательного подтверждения НП-ПЕРЕДАЧА.

12.2.2.2 Разрушающие услуги СЭУО

Разрушающих услуг СЭУО нет.

12.2.3 *Услуги СЭУО*

В данном разделе описывается взаимодействие между услугами СЭУО. Взаимодействие с услугами СЭУА описано в 12.2.1, а взаимодействие с услугами СЭНП — в 12.2.2.

12.2.3.1 Услуга УО-ПРИВЛЕЧЕНИЕ

12.2.3.1.1 Тип услуги

Услуга УО-ПРИВЛЕЧЕНИЕ является неподтверждаемой услугой.

12.2.3.1.2 Ограничения использования

Услуга УО-ПРИВЛЕЧЕНИЕ используется только при установлении прикладной-ассоциации.

12.2.3.1.3 Разрушаемые услуги

Услуга УО-ПРИВЛЕЧЕНИЕ не разрушает никаких других услуг.

12.2.3.1.4 Разрушающие услуги

Услуга УО-ПРИВЛЕЧЕНИЕ разрушается услугой УО-Пс-ОТКЛОНЕНИЕ.

12.2.3.1.5 Конфликты

Услуга УО-ПРИВЛЕЧЕНИЕ не создает конфликтов.

12.2.3.2 Услуга УО-РЕЗУЛЬТАТ

12.2.3.2.1 Тип услуги

Услуга УО-РЕЗУЛЬТАТ является неподтверждаемой услугой.

12 2.3.2.2 Ограничения использования

Услуга УО-РЕЗУЛЬТАТ используется только при установлении прикладной-ассоциации и в ответ на услугу УО-ПРИВЛЕЧЕНИЕ.

12.2.3.2.3 Разрушаемые услуги

Услуга УО-РЕЗУЛЬТАТ не разрушает никаких других услуг.

12.2.3.2.4 Разрушающие услуги

Услуга УО-РЕЗУЛЬТАТ разрушается услугой УО-Пс-ОТКЛОНЕНИЕ.

12.2.3.2.5 Конфликты

Услуга УО-РЕЗУЛЬТАТ не создает конфликтов.

12.2.3.3 Услуга УО-ОШИБКА

12.2.3.3.1 Тип услуги

Услуга УО-ОШИБКА является неподтверждаемой услугой.

12.2.3.3.2 Ограничения использования

Услуга УО-ОШИБКА используется только при установлении прикладной-ассоциации и в ответ на услугу УО-ПРИВЛЕЧЕНИЕ.

12.2.3.3.3 Разрушаемые услуги

Услуга УО-ОШИБКА не разрушает никаких других услуг.

12.2.3.3.4 Разрушающие услуги

Услуга УО-ОШИБКА разрушается услугой УО-Пс-ОТКЛОНЕНИЕ.

12.2.3.3.5 Конфликты

Услуга УО-ОШИБКА не создает конфликтов.

12.2.3.4 Услуга УО-Пл-ОТКЛОНЕНИЕ

12.2.3.4.1 Тип услуги

Услуга УО-Пл-ОТКЛОНЕНИЕ является неподтверждаемой услугой.

12.2.3.4.2 Ограничения использования

Услуга УО-Пл-ОТКЛОНЕНИЕ используется только при установлении прикладной-ассоциации и в ответ на услуги УО-ПРИВЛЕЧЕНИЕ, УО-РЕЗУЛЬТАТ и УО-ОШИБКА.

12.2.3.4 3 Разрушаемые услуги

Услуга УО-Пл-ОТКЛОНЕНИЕ не разрушает никаких других услуг.

12 2 3 4 4 Разрушающие услуги

Услуга УО Пл ОТКЛОНЕНИЕ разрушается услугой УО-Пс-ОТКЛОНЕНИЕ

12 2 3 4 5 Конфликты

Услуга УО Пл-ОТКЛОНЕНИЕ не создает конфликтов

12 2 3 5 Услуга УО-Пс-ОТКЛОНЕНИЕ

12 2 3 5 1 Тип услуги

Услуга УО Пс ОТКЛОНЕНИЕ относится к услугам, инициируемым поставщиком

12 2 3 5 2 Ограничения использования

Не налагаются

12 2 3 5 3 Разрушаемые услуги

Услуга УО-Пс-ОТКЛОНЕНИЕ разрушает все остальные услуги СЭУО

12 2 3 5 4 Разрушающие услуги

Услуга УО-Пс-ОТКЛОНЕНИЕ не разрушается никакими другими услугами

12 2 3 5 5 Конфликты

Если услуга СЭУА или услуга СЭНП вызывает прерывание или разъединение прикладной-ассоциации, то вопрос информирования пользователя-услуг об отсутствующих услугах УО-Пс ОТКЛОНЕНИЕ для возвращаемых параметров является частным вопросом

12 2 3 6 Дополнительная информация о последовательности выполнения

Ограничения использования операций развязки (12 1 2 1) и преобразование операций развязки в услугу П-РАЗЪЕДИНЕНИЕ предотвращают разрушение услуг СЭУО, если только используются ассоциативный класс 1 и операционные классы 1 и 2.

При использовании ассоциативного класса 2 или 3, либо операционного класса 3, 4 или 5 услуга П-РАЗЪЕДИНЕНИЕ может разрушить услуги СЭУО В этом случае разработчик прикладного-контекста решает, принять ли разрушение или обеспечить средства (например, операции) для подготовки разъединения прикладной ассоциации

ПРИЛОЖЕНИЕ А
(обязательное)

НОТАЦИЯ, ПОДДЕРЖИВАЮЩАЯ СПЕЦИФИКАЦИЮ СЕРВИСНЫХ-ЭЛЕМЕНТОВ-ПРИКЛАДНОГО-УРОВНЯ И ПРИКЛАДНЫХ-КОНТЕКСТОВ

Данное приложение определяет нотацию, поддерживающую спецификацию сервисных-элементов-прикладного-уровня и прикладных-контекстов, определенных посредством УО-нотации. УО-нотация может быть использована для определения операции-связки и операции-развязки прикладного-контекста. Кроме того, УО-нотация может быть использована для спецификации типов операции и типов ошибок различных сервисных-элементов-прикладного-уровня (элементов СЭП пользователя-СЭУО). Если УО-нотация используется в сочетании с другими операциями, могут быть использованы другие определенные в различных источниках инструменты спецификаций.

Настоящее приложение определяет две макрокоманды, обеспечивающие спецификацию сервисных-элементов-прикладного-уровня и прикладных-контекстов. Формализованное описание этих макрокоманд показано на рисунке А.1.

А.1 Сервисные-элементы-прикладного-уровня

Излагаемая нотация обеспечивает однозначную идентификацию СЭП.

Если в качестве СЭП выступает пользователь СЭУО, то данная нотация обеспечивает также спецификацию характеристик СЭП. Операционный интерфейс и протокол СЭП (пользователь СЭУО) определяется набором типов операций и набором типов ошибок.

Протокол, определенный для конкретного СЭП, по своему характеру может быть:

а) симметричным или

б) асимметричным.

В случае симметричного протокола оба пользователя-СЭП могут привлечь один и тот же набор типов операций.

В случае асимметричного протокола один из пользователей-СЭП (называемый поставщиком в контексте данного приложения) обеспечивает некоторые функциональные возможности по обработке информации, используемые равноправным пользователем-СЭП (называемым потребителем в контексте данного приложения). В этом случае тип конкретной операции может быть:

а) привлечен потребителем и/или

б) привлечен поставщиком

функциональных возможностей обработки информации.

Примечание — Конкретный смысл терминов «поставщик» и «потребитель» часто носит интуитивный характер. Кто-то может вполне естественно рассматривать, например, файловую систему как поставщика, а ее пользователя — как потребителя. Однако, строго говоря, смысл этих двух терминов носит произвольный характер.

Если СЭП использует концепцию взаимосвязанных-операций, то тип конкретной операции может быть привлечен в виде порождаемой-операции. Заявитель порождаемой операции является исполнителем взаимосвязанной порождающей-операции. Те типы операций, которые привлекаются исключительно как порождаемые операции, не должны включаться в спецификацию СЭП, они перечислены в спецификации типов порождающих их операций.

Те ошибочные типы операций, о которых могут поступать уведомления, не входят в спецификацию СЭП; они перечислены в спецификации типов операций

Все остальные операции (которые привлекаются не только как порождаемые операции) и те, кому разрешено привлекать эти операции, могут быть выражены в виде формальной нотации, специфицирующей пользователя-СЭУО СЭП

А11 Спецификация сервисного-элемента-прикладного-уровня

СЭП может быть специфицирован с помощью формальной нотации, обеспечиваемой макрокомандой APPLICATION-SERVICE-ELEMENT (см рисунок А 1)

Нотация типов макрокоманды APPLICATION-SERVICE-ELEMENT обеспечивает возможность спецификации СЭП. Нотация значений макрокоманды APPLICATION-SERVICE-ELEMENT обеспечивает возможность спецификации уникального идентификатора СЭП

Нотация типа СЭП является ключевым словом APPLICATION-SERVICE-ELEMENT, за которым факультативно следует спецификация операций.

Если протокол, определенный для СЭП, является симметричным, то ключевое слово OPERATION и справочные имена операций являются специфицированными.

Если протокол, определенный для СЭП, является асимметричным, то специфицированы будут ключевые слова CONSUMER INVOKES (ПРИВЛЕКАЕТ ПОТРЕБИТЕЛЬ), справочные имена тех операций, которые может привлечь потребитель, и/или ключевые слова SUPPLIER INVOKES (ПРИВЛЕКАЕТ ПОСТАВЩИК) и справочные имена тех операций, которые может привлечь поставщик

```
Remote Operations Notation-extension { joint-iso-ceitt remote-operations(4)
notatio-extension(2) }
DEFINITIONS
BEGIN
EXPORTS APPLICATION-SERVICE-ELEMENT, APPLICATION-CONTEXT,
aCSE,
IMPORTS OPERATION, BIND, UNBIND FROM Remote-Operation-Notation
                                    { joint-iso-ceitt remote-operations(4)
                                      notation(0) },
-- макроопределения СЭП
APPLICATION-SERVICE ELEMENT MACRO ::=
BEGIN
TYPE NOTATION                 ::= SymmetricAse | ConsumerInvoke SupplierInvokes | empty
VALUE NOTATION                ::= value (VALUE OBJECT IDENTIFIER)
SymmetricAse                  ::= «OPERATIONS» {«OperationList»}»
ConsumerInvokes               ::= «CONSUMER INVOKES» «{«OperationList»}
                                                                  | empty
SupplierInvokes               ::= «SUPPLIER INVOKES» «{«OperationList»}»
                                                                  | empty
OperationList                   Operation | OperationList «,» Operation
Operation                     ::= value (OPERATION)
END
aCSE APPLICATION-SERVICE-ELEMENT ::= { joint-iso-ceitt remote
                                    operations(4) aseID-ACSE(4) }
-- продолжение расширения нотации удаленных операций следует
```

Рисунок А.1 (Лист 1) — Формализованное описание СЭП и типов данных прикладного-контекста

```
   продолжение расширения нотации удаленных операций
   макроопределения прикладного контекста
APPLICATION CONTEXT MACRO    =
BEGIN
TYPE NOTATION                = NonROelements Binding ROelements
                                                  AbstractSyntaxes
VALUE NOTATION               = value (VALUE OBJECT IDENTIFIER)
NonROelements                = «APPLICATION SERVICE ELEMENTS»
                                                  «}» AseList «}»
Binding                      = «BIND» type    должен указать тип опера
                                                  ции связи
                               «UNBIND» type  должен указать тип опе
                                                  рации развязки
ROelements                   = «REMOTE OPERATIONS» «}» AseID «}»
                               определение СЭУО
                               SymmetricAses AsymmetricAses | empty
SymmetricAses                = «OPERATIONS OF» «}» AseList «}» | empty
AsymmetricAses               = InitiatorConsumerOf ResponderConsumerOf
InitiatorConsumerOf          = «INITIATOR CONSUMER OF» «{» AseList
                                                  «}» | empty
ResponderConsumerOf          = «RESPONDER CONSUMER OF» «{» AseList
                                                  «}» | empty
AbstractSyntaxes             = «ABSTRACT SYNTAXES» «}» Abstract
                                                  SyntaxList «}»
AseList                      = AseID | AseList « » AseID
AseID                        = value (APPLICATION SERVICE-ELEMENT)
AbstractSyntaxList           = AbstractSyntax | AbstractSyntaxList « »
                                                  AbstractSyntax
AbstractSyntax               = value (OBJECT IDENTIFIER)   идентифи
                                                  кация абстрактного синтаксиса
END
END                          конец расширения нотации удаленных операции
```

Рисунок А 1 (Лист 2) — Формализованное описание СЭП и типов данных прикладного контекста

А.2 Прикладные контексты

В контексте данного приложения прикладной контекст в явном виде идентифицирует

а) операцию связки,

б) операцию-развязки и

в) набор сервисных-элементов-прикладного-уровня

и требует наличия единого или нескольких абстрактных синтаксисов

В набор сервисных-элементов-прикладного-уровня входят

а) элементы СЭП, не использующие УО-нотацию, т е СЭУА, факультативно СЭНП и другие,

б) факультативно СЭУО и СЭП — пользователи-СЭУО

Если прикладной контекст содержит СЭП — пользователей-СЭУО с асимметричным протоколом, то данная нотация обеспечивает спецификацию независимо от того, кто является потребителем такого СЭП инициатор-ассоциации или ответчик ассоциации

Идентификация операций связки, операций развязки, сервисных элементов-прикладного уровня и абстрактных синтаксисов может быть выражена формальной нотацией, специфицирующей прикладной-контекст

А.2.1 Спецификация прикладного контекста

Прикладной контекст может быть определен формальной нотацией, обеспечиваемой макрокомандой APPLICATION-CONTEXT (см рисунок А 1)

Нотация типов макрокоманды APPLICATION-CONTEXT обеспечивает возможность спецификации прикладного контекста. Нотация значений макрокоманды APPLICATION-CONTEXT обеспечивает возможность однозначной идентификации прикладного контекста.

Нотацией типа прикладного контекста является ключевое слово APPLICATION-CONTEXT с последующими ключевыми словами APPLICATION SERVICE ELEMENT и справочными именами СЭП, не использующими УО-нотацию, далее ключевым словам BIND и справочным именем типа операции связки, ключевым словом UNBIND и справочным именем типа операции-развязки, факультативной спецификацией операций, использующих СЭП, ключевыми словами ABSTRACT SYNTAXES и справочными именами абстрактных синтаксисов.

Если прикладной контекст содержит СЭП — пользователей-СЭУО, то используются ключевые слова REMOTE OPERATION и справочное имя СЭУО с последующей спецификацией СЭП с симметричным протоколом и/или спецификацией СЭП с асимметричным протоколом Спецификацией СЭП с симметричным протоколом являются ключевые слова OPERATIONS OF и справочные имена СЭП. Спецификацией СЭП с асимметричным протоколом являются ключевые слова INITIATOR CONSUMER OF (ИНИЦИАТОР ПОТРЕБИТЕЛЬ) и справочные имена СЭП, потребителем которых является инициатор-ассоциации, и/или ключевые слова RESPONDER CONSUMER (ИНИЦИАТОР ОТВЕТЧИК) и справочные имена СЭП, потребителем которых является ответчик-ассоциации.

А.2.2 Преобразование нотаций в услуги

Идентификатор прикладного контекста и перечень имен абстрактного синтаксиса, специфицированные посредством макрокоманды APPLICATION CONTEXT, преобразуются либо в услуги СЭУО НП-ОТКРЫТИЕ, если СЭНП входит в состав прикладного контекста, либо также и в услуги СЭУА П-АССОЦИАЦИЯ.

Значение прикладного контекста преобразуется в параметр «имя прикладного контекста» услуги НП-ОТКРЫТИЕ или П-АССОЦИАЦИЯ

Имена абстрактного синтаксиса преобразуются в параметры «список определения контекста уровня представления» и «список результатов определений контекста уровня представления» услуги НП-ОТКРЫТИЕ или П-АССОЦИАЦИЯ.

ПРИЛОЖЕНИЕ В
(справочное)

РУКОВОДСТВО ДЛЯ РАЗРАБОТЧИКОВ ПРОТОКОЛОВ ПРИКЛАДНОГО УРОВНЯ ПО ИСПОЛЬЗОВАНИЮ СЭУО

Данное приложение содержит примеры и руководящие материалы для разработчиков протоколов прикладного уровня по использованию СЭУО.

В.1 Примеры операций и ошибок

В этом разделе приведены примеры определения операций и ошибок

В 1 1 Операционные классы

В данном подразделе содержатся примеры определений операций операционных классов 1—5

Операция operationExample12 операционного класса 1 или 2 (см пример ниже) уведомляет об успешности (результат типа ArgumentType12) или безуспешности (ошибки errorExample1 или errorExample2) ее выполнения Аргумент операции operationExample12 относится к типу ArgumentType12 Значение операции operationExample12 равно 1.

```
operationExample12 OPERATION
                     ARGUMENT ArgumentType12
                     RESULT ResultType12
                     ERROR { errorExample1, errorExample2 }
                       = 1
```

Операция operationExample3 операционного класса 3 (см пример ниже) уведомляет только о безуспешности (ошибка errorExample1) ее выполнения. Аргумент операции operationExample3 относится к типу ArgumentType3 Значение операции operationExample3 равно 2

```
operationExample3  OPERATION
                     ARGUMENT ArgumentType3
                     ERROR { errorExample1 }
                       = 2
```

Операция operationExample4 операционного класса 4 (см пример ниже) уведомляет только об успешности (результат типа ResultType4) ее выполнения Аргумент операции operationExample4 относится к типу ArgumentType4. Значение операции operationExample4 равно 3.

```
operationExample4  OPERATION
                     ARGUMENT ArgumentType4
                     RESULT ResultType4
                       = 3
```

Операции operationExample51 и operationExample52 операционного класса 5 (см пример ниже) не сообщает о результате Аргумент операции operationExample51 относится к типу ArgumentType4, операция operationExample52 не имеет аргумента Значение операции operationExample51 равно 4, а значение операции operationExample52 равно 5.

```
operationExample51 OPERATION
                     ARGUMENT ArgumentType4
                       = 4
operationExample52 OPERATION
                     . = 5
```

В.1.2 Взаимосвязанные операции

Приводимый ниже пример показывает определение набора взаимосвязанных операций, состоящих из порождающей-операции parent-op12 и порождаемых операций operationExample51 и operationExample52.

```
parent-op12 OPERATION
            ARGUMENT ArgumentType12
            RESULT ResultType12
            ERROR { errorExample1, errorExample2 }
            LINKED { operationExample51, operationExample52 }
            : : = 6
```

В.1.3 Ошибки

Ошибки (см. ниже errorExample1 и errorExample2) уведомляют о неудачном результате. Параметр ошибки errorExample1 относится к типу ParameterType1, ошибка errorExample2 не имеет параметров. Значение ошибки errorExample1 равно 1, а значение ошибки errorExample2 равно 2.

```
errorExample1 ERROR
              PARAMETER ParameterType1
              : : = 1
errorExample2 ERROR
              : : = 2
```

В.2 Примеры операций-связки и операций-развязки

В этом подразделе содержатся примеры определения операций-связки и операций-развязки.

В.2.1 Операции-связки

Операции-связки используются для установления прикладной-ассоциации.

Запрос операции связки BindExample1 на установление прикладной-ассоциации сопровождается аргументом типа BindArgumentType1. Положительный ответ на установление прикладной-ассоциации сопровождается результатом типа BindArgumentType1. Отрицательный ответ на установление прикладной-ассоциации сопровождается ошибкой-связки типа BindErrorType1.

```
BindExample : : = BIND
                  ARGUMENT BindArgumentType1
                  RESULT BindResultType1
                  BIND-ERROR BindErrorType1
```

Запрос операции-связки BindExample2 на установление прикладной-ассоциации сопровождается аргументом типа BindArgumentType1. Положительный ответ на установление прикладной-ассоциации не сопровождается никакими данными пользователя. Отрицательный ответ на установление прикладной-ассоциации сопровождается ошибкой-связки типа BindErrorType1.

```
BindExample2 : : = BIND
                   ARGUMENT BindArgumentType1
                   BIND-ERROR BindErrorType1
```

Заметим, что аргумент, результат и ошибка-связки операции-связки факультативны. Как запрос операции-связки BindExample3 на установление прикладной-ассоциации, так и ответ на установление прикладной-ассоциации не сопровождаются никакими данными пользователя.

```
BindExample3 : : = BIND
```

В.2.2 Операции-развязки

Операции-развязки используются для разъединения прикладной-ассоциации.

Запрос операции-развязки UnbindExample1 на разъединение прикладной-ассоциации сопровождается аргументом типа UnbindArgumentType1. Ответ на

разъединение прикладной-ассоциации сопровождается либо результатом типа UnbindResultType1, либо ошибкой-развязки типаа UnbindErrorType1.

```
UnbindExample1 : : = UNBIND
                     ARGUMENT UnbindArgumentType1
                     RESULT UnbindResultType1
                     UNBIND-ERROR UnbindErrorType1
```

Запрос операции-развязки UnbindExample2 на разъединение прикладной-ассоциации сопровождается аргументом типа UnbindArgumentType1. Ответ на разъединение прикладной-ассоциации может факультативно сопровождаться ошибкой-развязки типа UnbindErrorType1.

```
UnbindExample2 : : = UNBIND
                     ARGUMENT UnbindArgumentType1
                     UNBIND-ERROR UnbindErrorType1
```

Заметим, что аргумент, результат и ошибка-развязки факультативны. Как запрос операции-развязки UnbindExample3 на разъединение прикладной ассоциации, так и ответ на разъединение прикладной-ассоциации не сопровождаются никакими данными пользователя.

```
UnbindExample3 : : = UNBIND
```

B.3 Экспорт и импорт операций и ошибок

В этом подразделе приведены примеры способа экспортирования и импортирования операций и ошибок.

Приводимый ниже пример показывает способ экспортирования операций и ошибок. Operation10 и error10 имеют локальное значение. OperationTypeA и ErrorTypeA являются типами, а конкретные значения должны быть присвоены им в импортирующих модулях. Operation11 и error11 имеют глобальные уникальные значения.

```
ExportingModule { objectidentifier1 } DEFINITIONS : : =
BEGIN
    EXPORTS operation10, OperationTypeA, operation11,
            error10, ErrorTypeA, error11;
    IMPORTS OPERATION, ERROR, BIND, UNBIND
            FROM Remote-Operation-Notation
                 { joint-iso-ccitt remoteoperation(4) notation(0) };
    operation10 OPERATION
                  ARGUMENT ArgumentType10
                  RESULT ResultType10
                  ERROR { error10 }
                  : : = 10
    OperationTypeA : : =
                     OPERATION
                     ARGUMENT ArgumentTypeA
                     RESULT ResultTypeA
    operation11 OPERATION
                  ARGUMENT ArgumentType11
                  RESULT ResultType11
                  ERROR { error11 }
                  : : = { objectidentifier2component11 }
    error 10 ERROR
             PARAMETER ParameterType10
             : : = 10
```

```
    errorTypeA : : =
                    ERROR
                    PARAMETER ParameterTypeA
    error11 ERROR
            PARAMETER ParameterType11
              = { objectidentifier2component12 }
END
```

Приводимый ниже пример показывает способ импортирования операций и ошибок. Operation10 и error10 имеют локальные значения, определенные в экспортирующем модуле. Разработчик импортирующего модуля несет ответственность за уникальность этих значений в пределах абстрактного синтаксиса. Операция *operation13 относится к типу OperationTypeA и имеет присвоенное значение* 13. Ошибка error13 относится к типу ErrorTypeA и имеет присвоенное значение 13. Operation11 и error11 имеют глобальные уникальные значения, определенные в экспортирующем модуле.

```
ImportingModule { objectidentifier3 } DEFINITIONS
BEGIN
    IMPORTS operation10 OperationTypeA, operation11,
            error10, ErrorTypeA, error11
            FROM ExportingModule {
                objectidentifier },
    operation13 OperationTypeA : = 13
    error13 ErrorTypeA : = 13
END
```

В.4 Определение элементов-услуг-прикладного-уровня

В этом разделе приведены примеры способа определения сервисных элементов прикладного-уровня. Эти примеры ссылаются на операции и ошибки, определенные в примерах раздела Б.1.

Сервисный элемент прикладного-уровня element1 содержит операции operationExample12 и operationExample3, а также ошибки errorExample1 и errorExample2. Заметим, что *ошибки включены непосредственно путем определения* операции. Сервисный элемент-прикладного-уровня element1 является симметричным, т. е. оба элемента пользователя могут привлечь операции operationExample12 и operationExample3.

```
element1APPLICATION-SERVICE-ELEMENT
        OPERATIONS { operationExample12,
        operationExample3 }
          = { objectidentifierOfElement }
```

Сервисный элемент прикладного уровня element2 охватывает операции operationExample3 и operationExample4 и ошибку errorExample1. Сервисный-элемент-прикладного уровня element2 является асимметричным, т. е. только один элемент-пользователя (который находится в роли потребителя) может привлекать операции operationExample3 и operationExample4.

```
element2APPLICATION-SERVICE-ELEMENT
        CONSUMER INVOKES { operationExample3,
        operationExample4 }
          = { objectidentifierOfElement2 }
```

Сервисный-элемент-прикладного-уровня element3 охватывает операции operationExample12 и operationExample51 и ошибку errorExample2. Сервисный-элемент-прикладного-уровня element3 является асимметричным, т. е. только один элемент-пользователя (который находится в роли поставщика) может привлекать *операции operationExample12 и operationExample51*.

element3APPLICATION-SERVICE-ELEMENT
SUPPLIER INVOKES { operationExample12,
operationExample51 }
::= { *objectidentifierOfElement3* }

Сервисный-элемент-прикладного-уровня element4 охватывает операции parent-op12, operationExample51 и operationExample52, а также ошибки errorExample1 и errorExample2. Заметим, что порождаемые-операции operationExample51 и operationExample52 включены косвенно путем определения порождающей-операции parent-op12. Сервисный-элемент-прикладного-уровня element4 является асимметричным, т. е. только один элемент-пользователя (который находится в роли поставщика) может привлекать порождаемые операции operationExample51 и operationExample52 при появлении порождающей операции parent-op12.

element4APPLICATION-SERVICE-ELEMENT
CONSUMER INVOKES { parent-op12 }
::= { objectidentifierOfElement4 }

Сервисный-элемент-прикладного-уровня element5 содержит те же операции и ошибки, что и сервисный-элемент-прикладного-уровня element4. Единственное отличие состоит в том, что элемент-пользователя в роли поставщика может привлекать операцию operationExample52 либо как порождаемую операцию, либо вне набора взаимосвязанных-операций (как непорождаемую операцию)

element5APPLICATION-SERVICE-ELEMENT
CONSUMER INVOKES { parent-op12 }
SUPPLIER INVOKES { operationExample52 }
::= { objectidentifierOfElement3 }

В.5 Определение прикладных контекстов

В этом разделе приводятся примеры способов определения прикладных контекстов для различных классов ассоциаций. Эти примеры ссылаются на сервисный-элемент-прикладного-уровня, определенный в примерах раздела Б 4, а также на операции-связки и операции-развязки, определенные в примерах раздела Б 2.

Прикладной контекст context1 содержит сервисные-элементы-прикладного-уровня СЭУА, СЭНП, СЭУО и element2; операцию-связки BindExample1 и операцию-развязки UnbindExample3. Инициатор-ассоциации может привлечь операции operationExample3 и operationExample4. Ответчику-ассоциации не разрешается привлекать никаких операций (ассоциативный класс 1)

context1 APPLICATION-CONTEXT
APPLICATION-SERVICE-ELEMENT { aCSE, rTSE }
BIND BindExample1
UNBIND UnbindExample3
REMOTE OPERATIONS { rOSE }
INITIATOR CONSUMER OF { element2 }
ABSTRACT SYNTAXES
{{ joint-iso-ccitt association-control(2)
abstract-Syntax(1) apdus(0) version1(1) },
objectidentifierOfAbstractSyntax1 }
::= { objectidentifierOfContext1 }

Прикладной контекст context2 содержит сервисные-элементы-прикладного-уровня СЭУА, СЭУО, element2 и element3; операцию-связки BindExample1 и операцию-развязки UnbindExample3 Ответчик-ассоциации может привлечь операции operationExample3, operationExample4, operationExample12 и operationExample51 Инициатору-ассоциации не разрешается привлекать никаких операций (ассоциативный класс 2)

```
context2 APPLICATION-CONTEXT
    APPLICATION SERVICE ELEMENTS { aCSE }
    BIND BindExample1
    UNBIND UnbindExample3
    REMOTE OPERATIONS { rOSE }
    INITIATOR CONSUMER OF { element3 }
    RESPONDER CONSUMER OF { element2 }
    ABSTRACT SYNTAXES
      {{ joint iso ccitt association-control(2)
      abstract Syntax(1) apdus(0) version1(1) },
      objectIdentifierOfAbstractSyntax2 }
      = { objectIdentifierOfContext2 }
```

Прикладной контекст context3 содержит сервисные элементы прикладного уровня СЭУА, СЭНП, СЭУО, и element2, операцию связки BindExample1 и операцию развязки UnbindExample3 Ответчик ассоциации может привлечь операции operationExample3, operationExample4 Инициатору-ассоциации не разрешается *привлекать никаких операции (ассоциативный класс 2)*

```
context3 APPLICATION-CONTEXT
    APPLICATION SERVICE ELEMENT { aCSE, rTSE }
    BIND BindExample1
    UNBIND UnbindExample3
    REMOTE OPERATIONS { rOSE }
    RESPONDER CONSUMER OF { element2 }
    ABSTRACT SYNTAXES
      { objectIdentifierOfAbstractSyntax3 }
       = { objectIdentifierOfContext3 }
```

Прикладной контекст context4 содержит сервисные-элементы-прикладного-уровня СЭУА, СЭУО, и element2, операцию-связки BindExample3 и операцию-развязки UnbindExample3 Прикладной контекст context3 является симметричным, т е как инициатор-ассоциации, так и ответчик-ассоциации может привлечь операции operationExample2 и operationExample3 (ассоциативный класс 3)

```
context4 APPLICATION-CONTEXT
    APPLICATION-SERVICE-ELEMENTS { aCSE}
    BIND BindExample3
    UNBIND UnbindExample3
    REMOTE OPERATIONS { rOSE }
    OPERATIONS OF { element1 }
    ABSTRACT SYNTAXES
      { objectIdentifierOfAbstractSyntax4 }
       = { objectIdentifierOfContext4 }
```

Два последних номера исходят из одного поименованного абстрактного синтаксиса

B.6 Упорядоченное разъединение прикладной-ассоциации

B 6 1 Введение

Настоящий стандарт определяет пять операционных классов в зависимости от сообщаемых операциями результатов и три класса ассоциации в зависимости от того, кто может привлекать операции инициатор-ассоциации, ответчик-ассоциации либо тот и другой В данном разделе определены правила, обеспечивающие упорядоченное разъединение прикладных ассоциаций и привлекаемые для этого операционные классы

B 6 2 Цели

Излагаемые правила ставят перед собой одну из двух следующих целей в зависимости от ситуации

а) Цель «в точности один раз» В идеальном случае логический объект прикладного уровня должен быть способен вести учет привлечениям операции, добиваясь того, чтобы операция выполнялась только один раз, а не несколько раз или ни одного раза

б) Цель «максимум один раз» В некоторых ситуациях цель «в точности один раз» не может быть достигнута В меньшей степени, но все же полезной целью является такое привлечение операции, при котором эта операция будет выполнена максимум один раз, т е, возможно, ни одного раза, но никогда дважды

В.6.3 Определение правил

Перечисленные ниже общие правила применимы в любых обстоятельствах

С1. Исполнитель должен сообщить результат или ошибку для каждой подтверждаемой операции по той же прикладной ассоциации, по которой была привлечена эта операция

С2. Инициатор не должен разъединять прикладную ассоциацию до тех пор, пока не будут подтверждены все привлеченные им операции

Следующие специфические правила применимы в некоторых обстоятельствах

S1. При каждом использовании услуги УО-ПРИВЛЕЧЕНИЕ заявитель должен обеспечить различные Ид-привлечения (если только это не повторная попытка привлечения) даже через последовательность прикладных-ассоциаций Это дает возможность исполнителю достичь цели «максимум один раз» путем подавления дублирований

S2. Если исполнитель сталкивается с дубликатом Ид-привлечения в услуге УО-ПРИВЛЕЧЕНИЕ, он должен использовать услугу УО-Пл-ПРИВЛЕЧЕНИЕ с дублированным привлечением в качестве параметра «причина-отказа» Это будет способствовать достижению цели «в точности один раз»

S3. Инициатор-ассоциации должен отклонить любые привлечения, которые он не выполнил до разъединения прикладной ассоциации

S4. Инициатор-ассоциации должен ответить на любые привлечения, если он выполнил их до разъединения прикладной-ассоциации

В.6.4 Применение правил

Общие правила применимы всегда Специфические правила управляют прикладными-ассоциациями конкретных ассоциативных классов и операциями (привлекаемыми посредством таких ассоциаций) конкретных операционных классов следующим образом

а) Прикладные-ассоциации ассоциативного класса 1. К операциям операционных классов 1 и 2 не применимы никакие специфические правила. К операциям операционных классов 3, 4 и 5 применимы специфические правила S1 и S2

б) Прикладные-ассоциации ассоциативных классов 2 и 3· К операциям операционных классов 1 и 2 применимы специфические правила S3 и S4. (Любое привлечение, выданное ответчиком-ассоциации после того, как инициатор-ассоциации выдал разъединение, теряется Отказы ответчика также могут быть потеряны). К операциям операционных классов 3, 4 и 5 применимы специфические правила S1, S2, S3 и S4

Для тех протоколов, которые содержат только операции операционных классов 1 или 2, единственным ограничением при значениях Ид-привлечения, обеспечиваемых для услуги УО-ПРИВЛЕЧЕНИЕ, является то, что они изменяются в течение времени существования прикладной-ассоциации

Логические-объекты-прикладного-уровня обеспечивают для привлекающего уникальность идентификаторов Ид-привлечения и через последовательные прикладные-ассоциации путем обмена адресами-уровня-представления за установленное прикладной ассоциацией время и для каждого адреса-уровня-представления

обеспечивают целостность идентификаторов Ид-привлечение, что повышает монотонность в течение некоторого разумного периода времени.

Чтобы обеспечить в точности одноразовое выполнение операций операционных классов 3, 4 и 5, логический-объект-прикладного-уровня должен выявлять причину-отказа дублированных привлечений путем двухкратного (или многократного) привлечения с одним и тем же Ид-привлечения. В противном случае самое большее, что может быть достигнуто, это цель «максимум один раз», исходя из того, что в общем случае заявитель не следит за выполнением неподтверждаемых операций.

В.6.5 Заключительные замечания

Каждая операция операционных классов 1 или 2 выполняется в точности один раз

Полезность неподтверждаемых операций или операций, подтверждаемых в зависимости от условий, может зависеть от конкретного применения. Разработчикам протоколов следует рекомендовать определять только операции операционных классов 1 или 2, если только не предъявляются некоторые очень специальные требования

УДК 681.3 : 621.39 : 006.354 П85

Ключевые слова: обработка данных, обмен информацией, взаимосвязь сети, передача текста, передача данных, пересылка данных, телеобработка, модели, определения

ОКСТУ 4002

Редактор *Т. С. Шеко*
Технический редактор *В. Н. Прусакова*
Корректор *М. С. Кабашова*

Сдано в набор 25.02 94 Подп. в печ 13.04 94. Усл. печ. л. 3,49. Усл кр.-отт. 3,49. Уч.-изд. л 3,57. Тир. 413 экз. С 1202.

Ордена «Знак Почета» Издательство стандартов, 107076, Москва, Колодезный пер., 14.
Калужская типография стандартов, ул. Московская, 256. Зак. 505

gost.e-ecolog.ru