ФЕДЕРАЛЬНОЕ АГЕНТСТВО

ПО ТЕХНИЧЕСКОМУ РЕГУЛИРОВАНИЮ И МЕТРОЛОГИИ

НАЦИОНАЛЬНЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

ГОСТ Р МЭК 61512-2— 2016

УПРАВЛЕНИЕ СЕРИЙНЫМ ПРОИЗВОДСТВОМ

Часть 2

Структуры данных и руководство по языку

(IEC 61512-2:2001, IDT)

Издание официальное

Москва
Стандартинформ
2016

gost.e-ecolog.ru

Предисловие

1 ПОДГОТОВЛЕН ООО «НИИ экономики связи и информатики «Интерэкомс» (ООО «НИИ «Интерэкомс») на основе собственного перевода на русский язык англоязычной версии стандарта, указанного в пункте 4

2 ВНЕСЕН Техническим комитетом по стандартизации ТК 100 «Стратегический и инновационный менеджмент»

3 УТВЕРЖДЕН И ВВЕДЕН В ДЕЙСТВИЕ Приказом Федерального агентства по техническому регулированию и метрологии от 10 октября 2016 г. № 1336-ст

4 Настоящий стандарт идентичен международному стандарту МЭК 61512-2:2001 «Управление серийным производством. Часть 2. Структуры данных и руководство по языку» (IEC 61512-2:2001 «Batch control — Part 2: Data structures and guidelines for languages», IDT).

Международный стандарт разработан Техническим комитетом 65 Подкомитетом 65А Международной электротехнической комиссии (МЭК).

При применении настоящего стандарта рекомендуется использовать вместо ссылочных международных стандартов соответствующие им национальные стандарты Российской Федерации, сведения о которых приведены в дополнительном приложении ДА

5 ВВЕДЕН ВПЕРВЫЕ

*Правила применения настоящего стандарта установлены в статье 26 Федерального закона от 29 июня 2015 г. № 162-ФЗ «О стандартизации в Российской Федерации». Информация об изменениях к настоящему стандарту публикуется в ежегодном (по состоянию на 1 января текущего года) информационном указателе «Национальные стандарты», а текст изменений и поправок — в ежемесячном информационном указателе «Национальные стандарты». В случае пересмотра (замены) или отмены настоящего стандарта соответствующее уведомление будет опубликовано в ежемесячном информационном указателе «Национальные стандарты». Соответствующая информация, уведомление и тексты размещаются также в информационной системе общего пользования — на официальном сайте Федерального агентства по техническому регулированию и метрологии в сети Интернет (*gost.e-ecolog.ru

© Стандартинформ, 2016

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

Содержание

1 Область применения ... 1
2 Нормативные ссылки ... 1
3 Термины и определения ... 1
4 Модель данных ... 2
4.1 Общие положения ... 2
4.2 Обзорная модель ... 2
4.3 Модель рецептуры ... 4
4.4 Модель оборудования ... 14
4.5 Разработка производственного и календарного планов ... 16
4.6 Управление производственной информацией ... 18
5 Реляционные таблицы обмена информацией ... 21
5.1 Общие положения ... 21
5.2 Информация о технологической рецептуре ... 32
5.3 Обмен моделями оборудования технологического цеха ... 44
5.4 Обмен информацией календарного планирования ... 49
5.5 Обмен производственной информацией ... 53
5.6 Применение таблиц обмена ... 56
6 Процедурные функциональные диаграммы ... 57
6.1 Нотация процедурных функциональных диаграмм ... 58
6.2 Отображение рецептуры управления ... 69
6.3 Работа в исключительных ситуациях ... 69
Приложение А (обязательное) Методика моделирования данных ... 71
Приложение В (обязательное) Листинг определений Языка структурированных запросов SQL ... 73
Приложение С (справочное) Сокращения ... 83
Приложение D (справочное) Руководство по языку ... 84
Приложение Е (справочное) Пример обработки процедурных функциональных диаграмм ... 86
Приложение ДА (справочное) Сведения о соответствии ссылочных международных стандартов национальным стандартам ... 88
Библиография ... 89

Введение

В МЭК 61512-1 определяются модели и терминология, применяемые в области управления серийным производством, в МЭК 61512-2 рассматриваются соответствующие структура данных и руководство по языку. Структура данных рассматривается на основе модели данных, определенной в разделе 4, которая более точно идентифицирует объекты и отношения, рассмотренные с помощью специальных моделей и понятий МЭК 61512-1. Структуры данных также рассматриваются с помощью реляционных таблиц обмена информацией, определенных в разделе 5. Языки рассматриваются в совокупности с методологией отображения рецептур, определенной в разделе 6.

Назначением модели данных, определенной в настоящем стандарте, является создание точки отсчета для разработки спецификаций интерфейса компонентов программного обеспечения, использующих разделы МЭК 61512-1. Модели данных, установленные в настоящем стандарте, используют разделы МЭК 61512-1 в качестве интегральной модели объекта. При этом заранее не предполагается и не исключается возможность использования специальной системной архитектуры или архитектуры для обмена информацией. Данная модель не предполагает какого-либо специального разделения функциональности между системами.

В разделе 5 определен специальный метод обмена выбранными данными. Реляционные таблицы реализуют указанный метод обмена информацией, так как в рамках рассматриваемой области они:

- широко используют имеющиеся технологии;
- могут ассоциироваться и интегрироваться с другими технологиями;
- легко воспринимаются;
- согласуются с другими разделами настоящего стандарта.

Ряд методов передачи информации в настоящем стандарте не рассмотрен. Не рассмотрен также вопрос идентификации обмениваемой информации. В последующих редакциях могут быть определены дополнительные методы, обеспечивающие альтернативные пути обмена данных.

В разделе 6 определяются условные обозначения и правила графического языка, которые могут быть использованы для описания рецептур. Рецептуры являются центральной отличительной особенностью системы управления серийным производством. Они могут существенно различаться по степени сложности. Однако нет ни одного отображения, которое было бы идеальным во всех обстоятельствах. Простая таблица, например, — это наиболее пригодная форма отображения рецептуры для простых случаев. Настоящий стандарт устанавливает метод описания технологических рецептур и рецептурных процедур управления, применяемых в широком диапазоне задач.

Несмотря на то, что настоящий стандарт и предназначен, прежде всего, для описания процессов серийного производства, он может использоваться также и для описания процессов другого типа.

НАЦИОНАЛЬНЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

УПРАВЛЕНИЕ СЕРИЙНЫМ ПРОИЗВОДСТВОМ

Часть 2

Структуры данных и руководство по языку

Batch control. Part 2. Data structures and guidelines for languages

Дата введения — 2017—06—01

1 Область применения

В настоящем стандарте устанавливаются модели данных, относящиеся к управлению серийным производством и применяемые в соответствующих отраслях промышленности, структуры данных, обеспечивающие внутренние и внешние связи между различными имплементациями системы управления серийным производством, руководство по языку, используемому для представления рецептур. Приложение А содержит нотацию универсального языка моделирования (UML), приложение В содержит сводный анализ всех определений языка структурированных запросов (SQL), приведенных в разделе 5.

2 Нормативные ссылки

В настоящем стандарте использованы нормативные ссылки на следующие стандарты, которые необходимо учитывать при его применении. При ссылках на документы, у которых указана дата утверждения, необходимо пользоваться только указанной редакцией, если эта дата не приведена, — последней редакцией ссылочных документов, включая любые поправки и изменения к ним.

IEC 60848:2013, GRAFCET specification language for sequential function charts (Язык спецификаций GRAFCET для последовательных функциональных схем)

IEC 60050-351:2013, International Electrotechnical Vocabulary — Part 351: Control technology (Международный электротехнический словарь. Часть 351. Технология управления)

IEC 61131-3:2013, Programmable controllers — Part 3: Programming languages (Контроллеры программируемые. Часть 3. Языки программирования)

IEC 61512-1:1997, Batch control — Part 1: Models and terminology (Управление серийным производством. Часть 1. Модели и терминология)

ISO/IEC 9075:1992 (all parts), Information technology — Database languages — SQL (Информационные технологии. Языки базы данных. Язык структурированных запросов (SQL) (все части ISO/IEC 9075))

3 Термины и определения

В настоящем стандарте применены термины по МЭК 61512-1, МЭК 60050-351, а также следующие термины с соответствующими определениями.

3.1 **обозначение выделения ресурса** (allocation symbol): Графическое обозначение, используемое для представления (инкапсуляции) процедуры оформления правил выделения (высвобождения) ресурса для процедурного элемента рецептуры.

3.2 **структурный элемент** (building block): Рецептурная сущность, представленная в библиотеке.

Издание официальное

3.3 **множество элементов перечисления** (enumeration set): Список предварительно определенных строк и соответствующих им ассоциированных численных значений.

3.4 **таблица обмена** (exchange table): Таблица базы данных, используемая для обмена информации, связанной с производством партии изделий, между системами.

3.5 **соединительное звено, связь** (link): Объект, задающий порядок соединения между двумя различными объектами (например, порядок соединения между отдельными рецептурными сущностями или между рецептурными сущностями и переходами).

3.6 **процедурная функциональная диаграмма** (procedure function chart): Графическое представление рецептурной процедуры, задающей порядок обработки процедурных элементов рецептуры.

3.7 **рецептурный элемент** (recipe element): Структурная сущность, используемая для представления рецептурных сущностей и рецептурных обозначений, за исключением переходов и направленных связей (соединительных звеньев), используемых в процедурных функциональных диаграммах.

3.8 **рецептурная сущность** (recipe entity): Комбинация процедурного элемента и ассоциированной рецептурной информации (например, заголовка, формулы, требований к оборудованию, прочей информации).

Примечание — Общая рецептура, рецептура, связанная с местом производства, технологическая рецептура и рецептура управления также являются рецептурными сущностями.

4 Модель данных

4.1 Общие положения

Настоящий раздел содержит модели данных, описывающие множество объектов, атрибутов и их базовые отношения, распространяющиеся на понятия МЭК 61512-1 на высоком уровне абстракции. Данная модель применима к интерфейсам систем управления серийным производством для любой используемой технологии. Указанные модели не предназначены для организации внутренней архитектуры систем управления серийным производством.

Данные модели можно рассматривать как точку отсчета для процесса разработки спецификаций интерфейса компонент программного обеспечения для любого подмножества МЭК 61512-1.

Данную модель можно рассматривать в качестве интегральной модели объектов МЭК 61512-1 без необходимости учета особых предпочтений или исключений в части специальной архитектуры системы или архитектуры для обмена информацией. Рассматриваемые модели не предполагают какого-либо специального разделения функциональности между системами.

Если объекты и отношения, определенные в настоящем разделе, представлены посредством интерфейса, то данный интерфейс должен использовать имена объектов, имена атрибутов и отношений данного раздела, соразмерных с выбранной технологией интерфейса и с доступными возможностями. Примером такого интерфейса является интерфейс реляционных таблиц языка SQL, определенный в разделе 5.

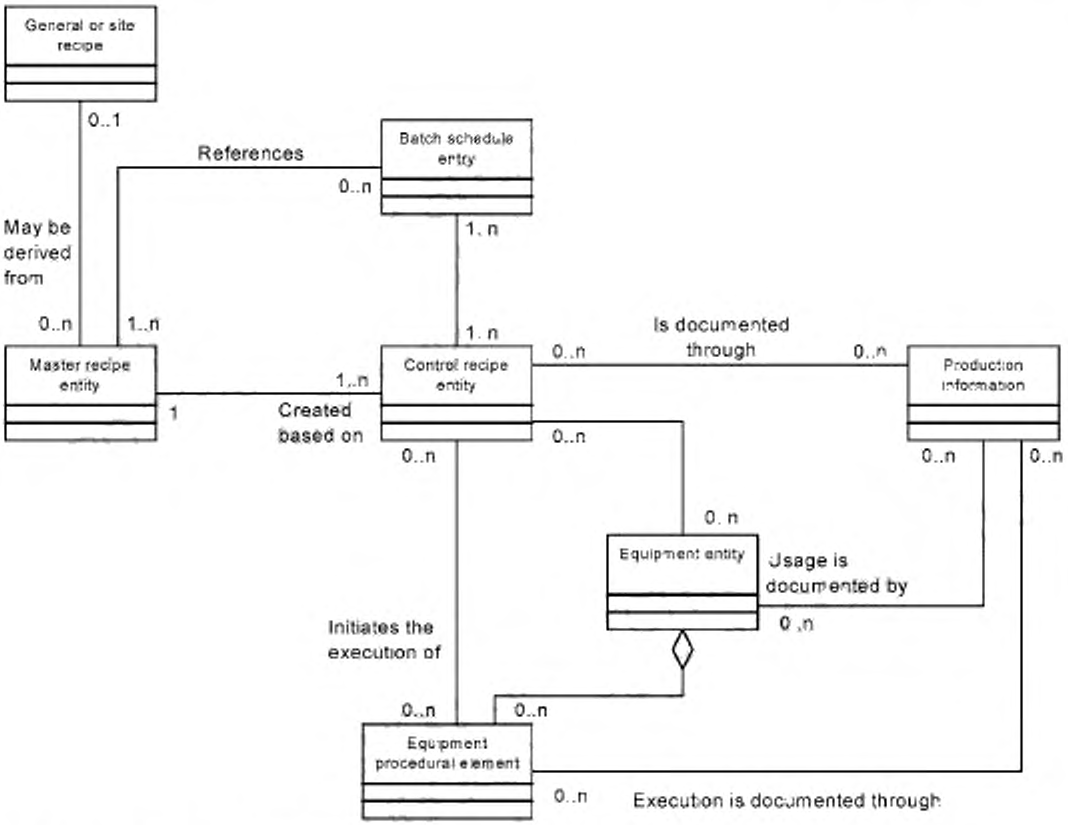
Формат обмена или спецификации интерфейса обеспечивают практическую реализацию только некоторых объектов или частей объектов (например, когда определены не все свойства). Формат обмена или спецификация интерфейса могут также обеспечивать дополнительные объекты или их свойства (например, информацию о продолжительности фазы), включая расширение любого атрибута модели данных на несколько атрибутов. Каждая подобная практическая реализация должна соответствовать представленной в настоящем стандарте модели данных и понятиям, определенным в МЭК 61512-1.

Модели, описанные далее, построены на основе языка UML (см. раздел А.1).

Рассматриваемые таблицы описывают только атрибуты класса объектов. Отношения между объектами приведены на рисунках.

4.2 Обзорная модель

Данная модель (см. рисунок 1) обеспечивает высокий уровень рассмотрения определенных здесь основных классов, а также взаимосвязи между данными классами для области серийного производства, описанных моделью управляющих действий в МЭК 61512-1. Классы индивидуальных объектов более детально описаны специальными моделями в данном подразделе.



General or site recipe — общая рецептура или технологическая рецептура; references — ссылается; batch schedule entry — календарная запись процесса изготовления партии; may be derived from — может быть выведена из; master recipe entity — сущность технологической рецептуры; created based on — создана на основе ...; control recipe entity — сущность рецептуры управления; is documented through — задокументировано с помощью ..., production information — производственная информация; equipment entity — сущность оборудования; usage is documented by — использование задокументировано ...; initiates the execution of — инициирует выполнение ..., equipment procedural element — процедурный элемент оборудования; execution is documented through — выполнение задокументировано ...

Рисунок 1 — Обзорная модель

Общая рецептура или рецептура, связанная с местом производства, представляют собой иерархию сущностей общей рецептуры, соответствующих процедурным сущностям (стадиям производства, технологическим операциям, технологическим действиям).

Технологическая рецептура может быть выведена из общей рецептуры или из рецептуры, связанной с местом производства. Саму технологическую рецептуру можно рассматривать как сущность технологической рецептуры верхнего уровня. Технологическая рецептура представляет собой иерархию сущностей технологической рецептуры, соответствующих процедурным сущностям (например, собственно процедурам, процедурам технологической установки, операциям, фазам).

Запись в календарном плане производства партии изделий характеризует конкретную партию изделий посредством выполнения соответствующей рецептуры. Календарный план производства партии изделий — это список, определяющий процесс производства партии изделий. Он также включает информацию о сроках. Необходимая информация о конкретном продукте выводится из соответствующей сущности технологической рецептуры.

Основанная на записи в календарном плане, рецептура управления изначально формируется как копия конкретной версии технологической рецептуры. Затем она модифицируется в реальную рецептуру.

по которой осуществляется производство партии изделий. Рецептура управления включает в себя информацию, необходимую для управления оборудованием.

Сущности рецептуры управления разрабатываются на основе сущностей технологической рецептуры. Рецептура управления может быть усилена дополнительной информацией (например, о масштабировании, о назначении оборудования). Она может быть модифицирована (включая создание или удаление сущностей рецептуры управления).

Сущности оборудования выбираются и выделяются для всех сущностей рецептуры управления.

Сущность рецептуры управления может быть соединительным звеном для процедурной сущности оборудования внутри сущности оборудования (как правило, это технологическая установка). Процедурная сущность оборудования может быть инициирована, ее параметрами могут быть назначенные рецептурные значения.

Производственная информация генерируется в ходе производства партии изделий. Данная информация может быть взаимосвязана с рецептурными сущностями, сущностями оборудования и/или с процедурными элементами оборудования.

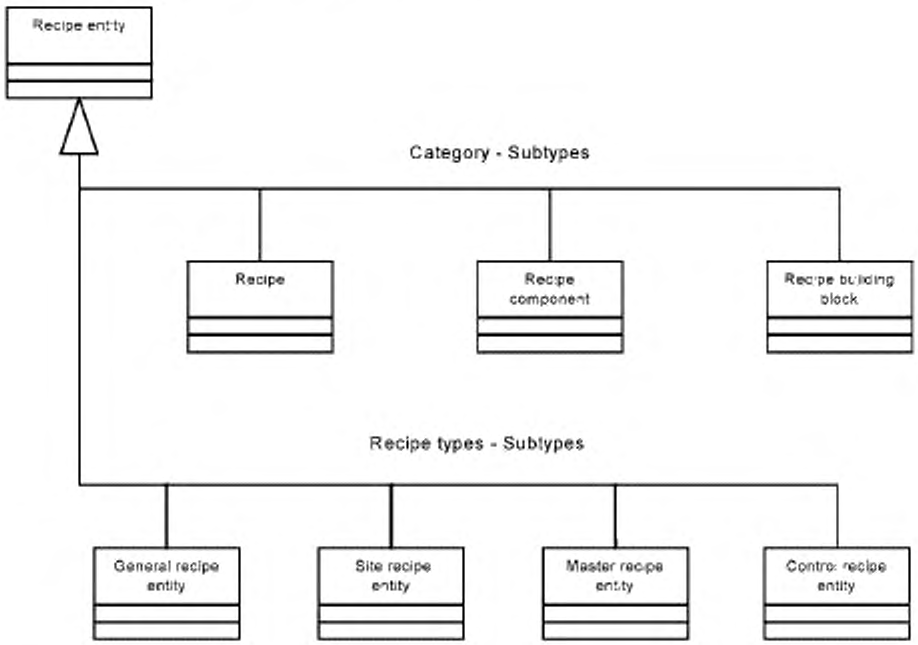
4.3 Модель рецептуры

4.3.1 Рецептурная сущность

Рецептуры организованы иерархически с различными категориями информации на каждом уровне. Рецептурная сущность — это компонент структуры, используемый для представления сопряжения данных на рассматриваемом уровне.

Рецептурная сущность — это базовая структура всех видов рецептур (см. рисунок 2). Рецептурная сущность структурно задействована в процедурном элементе рецептуры в соответствии с МЭК 61512-1. Она может включать любой или все компоненты рецептуры: процедурные определения, параметры со своими значениями, требования к оборудованию и прочую информацию.

Спецификации классов приведены в таблице 1.



Recipe entity — рецептурная сущность; category — subtypes — категория – подтипы; recipe — рецептура; recipe component — компоненты рецептуры; recipe building block — структурный элемент рецептуры; recipe types – subtypes — типы рецептуры — подтипы, general recipe entity — сущность общей рецептуры; site recipe entity — сущность рецептуры, связанной с местом производства; master recipe entity — сущность технологической рецептуры; control recipe entity — сущность рецептуры управления

Рисунок 2 — Рецептурные сущности

Таблица 1 — Рецептурные сущности

Имя	Рецептурная сущность
FunctionalDescription	Рецептурная сущность – это комбинация процедурного элемента и ассоциированной рецептурной информации (например, заголовка, формулы, требований к оборудованию и прочей информации). Общая рецептура, рецептура, связанная с местом производства, технологическая рецептура и рецептура управления – это рецептурные сущности. Примером является рецептура технологической установки (см. МЭК 61512-1, пункт 5.3.2). «Рецептура технологической установки – часть рецептуры управления, уникально определяющая сопутствующие производственные требования к технологической установке» (см. МЭК 61512-1, пункт 3.62)
Атрибуты Идентификатор рецептурной сущности	Обеспечивает уникальную идентификацию

Рецептура является рецептурной сущностью (категория: рецептура; recipe). Рецептура строится из рецептурных сущностей нижнего уровня (например, рецептуры технологической установки) (категория: компонент; component). Если строится особая рецептура, то ее компоненты могут быть взяты из библиотеки элементов (категория: структурный элемент; building block).

Понятие рецептурной сущности применяется ко всем типам рецептур: общая рецептура, рецептура, связанная с местом производства, технологическая рецептура и рецептура управления. Если рецептура выполнена, то представления выполненной рецептурной сущности в истории производства партии изделий имеют похожую структуру и, следовательно, показаны как подкласс. Обзор подклассов приведен в таблице 2. Категории подклассов приведены в таблицах 3–5. Типы подклассов приведен в таблицах 6–9. Общие рецептуры и рецептуры, связанные с местом производства, больше в данном подразделе не обсуждаются.

Таблица 2 — Подклассы — обзор

	СУЩНОСТЬ ОБЩЕЙ РЕЦЕПТУРЫ	РЕЦЕПТУРНАЯ СУЩНОСТЬ, СВЯЗАННАЯ С МЕСТОМ ПРОИЗВОДСТВА	СУЩНОСТЬ ТЕХНОЛОГИЧЕСКОЙ РЕЦЕПТУРЫ	СУЩНОСТЬ РЕЦЕПТУРЫ УПРАВЛЕНИЯ
РЕЦЕПТУРА	Законченная и самодостаточная общая рецептура	Законченная и самодостаточная рецептура, связанная с местом производства	Законченная и самодостаточная технологическая рецептура	Законченная и самодостаточная рецептура управления
СТРУКТУРНЫЙ ЭЛЕМЕНТ РЕЦЕПТУРЫ	Характерный тип сущности общей рецептуры, который может быть инстанцирован в конкретной рецептуре или в другом структурном элементе	Структурные элементы рецептур, связанных с местом производства, могут не существовать. Рецептуры, связанные с местом производства, нормально модифицируются с помощью структурных элементов общей рецептуры	Характерный тип сущности технологической рецептуры, который может быть инстанцирован в конкретной рецептуре или в другом структурном элементе	Структурные элементы рецептур управления не существуют. Рецептуры управления модифицируют с помощью структурных элементов технологической рецептуры
КОМПОНЕНТ РЕЦЕПТУРЫ	Компонентом общей рецептуры или библиотечным элементом может быть инстанциация структурного элемента	Компонентом рецептуры, связанной с местом производства, или библиотечным элементом может быть инстанциация структурного элемента общей рецептуры	Компонентом технологической рецептуры или библиотечным элементом может быть инстанциация структурного элемента	Компонентом рецептуры управления может быть инстанциация структурного элемента технологической рецептуры

Таблица 3 — Рецептура

ИМЯ	РЕЦЕПТУРА
FunctionalDescription	Рецептурная сущность верхнего уровня
АТРИБУТЫ RecipeID	Идентифицирует рецептуру. При объединении с атрибутом «RecipeVersion» определяет уникальную реализацию рецептуры
RecipeVersion	Идентифицирует версию рецептуры. При объединении с атрибутом «RecipeID» определяет уникальную реализацию рецептуры (например, атрибут «Красный Дуб» см. раздел А10.3)
VersionDate*	Идентифицирует дату и время создания или модификации данной версии рецептуры
ApprovalDate*	Идентифицирует дату и время утверждения данной версии рецептуры
EffectiveDate*	Идентифицирует самую раннюю дату и время, после чего данная версия рецептуры может быть использована
ExpirationDate*	Идентифицирует дату и время, после чего данная версия рецептуры утрачивает силу
ProductID*	Идентифицирует продукт или семейство продуктов, которые могут быть созданы путем выполнения данной версии рецептуры (например, пиво «Premium Beer»)
Author*	Идентифицирует физическое лицо или систему, являющуюся автором данной версии рецептуры (например, J. Smith)
ApprovedBy*	Идентифицирует физическое лицо или систему, утвердившее данную версию рецептуры
Decription	Содержит описание данной версии рецептуры и/или продукта (например, пиво «Finest Premium Beer»)
Status*	Определяет статус информации (например, «Утверждено для производства», «Утверждено для испытаний», «Не утверждено», «В резерв», «Устарело»)
*Не требуется для рецептуры управления (доступно путем ссылки на технологическую рецептуру)	

Таблица 4 — Компоненты рецептуры

ИМЯ	КОМПОНЕНТ РЕЦЕПТУРЫ
FunctionalDescription	Рецептурная сущность, являющаяся частью рецептуры или структурного элемента (например, реализация структурного элемента в заданной рецептуре или наличие рецептурной сущности структурного элемента)
АТРИБУТЫ Level	Указывает процедурный уровень иерархии (например, стадию производства, технологическую операцию или технологическое действие для общих рецептур, рецептур, связанных с местом производства, процедур технологической установки, операций или фаз для технологических рецептур и рецептур управления)
RE_Use	Определяет, является ли компонент рецептуры копией структурного элемента или только ссылается на него

Таблица 5 — Структурный элемент рецептуры

ИМЯ	СТРУКТУРНЫЙ ЭЛЕМЕНТ РЕЦЕПТУРЫ
FunctionalDescription	Рецептурная сущность, существующая в библиотеке. Структурный элемент может быть параметризован и использован при построении рецептуры
АТРИБУТЫ RecipeVersion	Идентифицирует версию рецептурной сущности
VersionDate	Идентифицирует дату и время создания или модификации данной версии рецептуры
ApprovalDate	Идентифицирует дату и время утверждения данной версии рецептуры
Author	Идентифицирует физическое лицо или систему, которая является автором данной версии рецептуры (например, J. Smith)
ApprovedBy	Идентифицирует физическое лицо или систему, утвердившую данную версию рецептуры
Description	Содержит описание функции, полученной посредством выполнения данной версии рецептурной сущности

Окончание таблицы 5

ИМЯ	СТРУКТУРНЫЙ ЭЛЕМЕНТ РЕЦЕПТУРЫ
Level	Указывает на уровень рецептурной сущности
UsageConstraint	Определяет прочие правила, задающие порядок использования сущности (например, «всегда предшествуется ..», «никогда не происходит параллельно ....»)
Status	Определяет статус рецептурной сущности (например, «Утвержден для производства», «Утвержден для испытаний», «Не утвержден», «Направить в резерв», «Устарело»)
Function	Определяет порядок выполнения рецептурной сущности (например, путем ссылки на процедурный элемент оборудования, посредством следования встроенной логике)

Таблица 6 — Сущность общей рецептуры

ИМЯ	СУЩНОСТЬ ОБЩЕЙ РЕЦЕПТУРЫ
FunctionalDescription	Все общие рецептуры или рецептуры, связанные с местом производства, компоненты общей рецептуры или рецептуры, связанной с местом производства, структурные элементы, необходимые для создания общей рецептуры или рецептуры, связанной с местом производства
АТРИБУТЫ	
ScaleReference	Определяет измерительную шкалу (шкалу отсчета) для значений параметра

Таблица 7 — Рецептурная сущность, связанная с местом производства

ИМЯ	РЕЦЕПТУРНАЯ СУЩНОСТЬ, СВЯЗАННАЯ С МЕСТОМ ПРОИЗВОДСТВА
FunctionalDescription	Все рецептуры, связанные с местом производства, все компоненты рецептур, связанных с местом производства, все структурные элементы, необходимые для создания рецептур, связанных с местом производства
АТРИБУТЫ	
ScaleReference	Определяет измерительную шкалу (шкалу отсчета) для значений параметра

Таблица 8 — Сущность технологической рецептуры

ИМЯ	СУЩНОСТЬ ТЕХНОЛОГИЧЕСКОЙ РЕЦЕПТУРЫ
FunctionalDescription	Все технологические рецептуры, компоненты технологических рецептур, структурные элементы, необходимые для создания технологических рецептур
АТРИБУТЫ	
ScaleReference	Определяет измерительную шкалу (шкалу отсчета) для значений параметра
ProcessCellID	Идентифицирует категорию оборудования, для которой данная рецептурная сущность определена (например, технологический цех (технологические цеха), для которого данная технологическая рецептура определена)

Таблица 9 — Сущность рецептуры управления

ИМЯ	СУЩНОСТЬ РЕЦЕПТУРЫ УПРАВЛЕНИЯ
FunctionalDescription	Рецептурная сущность, представляющая собой всю рецептуру управления или ее часть
АТРИБУТЫ	
BatchID	Задает фактический идентификатор партии
BatchSize	Определяет запрошенный размер или объем партии изделий, основанный на масштабном факторе партии, определенном в технологической рецептуре
Status	Определяет статус состояния (например, «еще не активировано», «активировано» или «работа завершена»)

4.3.2 Части рецептурной сущности

Модель, представленная на рисунке 3 и в таблицах 10–13, определяет категории информации о рецептуре в соответствии с МЭК 61512-1. Модель предполагает, что данные компоненты могут суще-

ствовать на любом уровне декомпозиции рецептуры на составные части (например, рецептура технологической установки может содержать свои собственные требования к оборудованию).

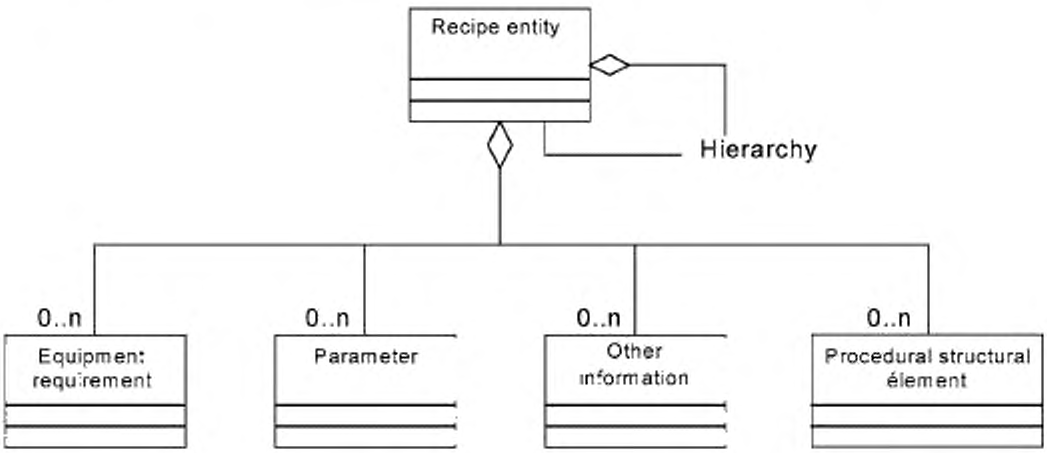
Категория информации заголовка сама содержится в атрибутах рецептурной сущности, вместо того, чтобы быть отдельным классом объектов данной модели.

Категория формулы в соответствии с МЭК 61512-1 моделируется как множество объектов параметров. Все уровни разложения рецептуры на составные части могут иметь параметры, включая саму рецептуру. См. 4.3.6.

Моделирование требований к оборудованию обсуждается в 4.3.5.

В соответствии с МЭК 61512-1, категория прочей информации представлена как отдельный класс объектов, даже если прочая информация может иметь несколько элементов и различную структуру.

В соответствии с МЭК 61512-1, категория процедуры моделируется как множество процедурных структурных элементов.



Recipe entity — рецептурная сущность; hierarchy — иерархия; equipment requirement — требования к оборудованию; parameter — параметр; other information — прочая информация; procedural structural element — процедурный структурный элемент

Рисунок 3 — Части рецептурных сущностей

Таблица 10 — Параметр

ИМЯ	ПАРАМЕТР
FunctionalDescription	Значения формул или границы изменения значений, которые ставятся в соответствие рецептурной сущности в ходе ее выполнения
АТРИБУТЫ ParameterID	Обеспечивает уникальную идентификацию
ParameterType	Определяет порядок интерпретации значения параметра (например, константа, базисное уравнение)
Description	Содержит описание параметра или порядок его использования
EngineeringUnits	Идентифицирует технические единицы измерения значений (например, кг, фунт)
Value	Содержит значение параметра. Если значение — это отношение, то оно содержит уравнение, форму, отложенное правило или что-то, что связывает параметры вместе. Если значение — это параметр структурного элемента, то данный атрибут содержит значение по умолчанию
Scaled	Задает правило масштабирования. Простейший случай: масштабированный или не масштабированный с базисным размером партии

Окончание таблицы 10

ИМЯ	ПАРАМЕТР
Usage	Задает параметр как входной сигнал технологического процесса, выходной сигнал технологического процесса, параметр процесса

Таблица 11 — Требования к оборудованию

ИМЯ	ТРЕБОВАНИЯ К ОБОРУДОВАНИЮ
FunctionalDescription	Представляет требования к оборудованию в соответствии с рецептурной сущностью
АТРИБУТЫ	

Таблица 12 — Прочая информация

ИМЯ	ПРОЧАЯ ИНФОРМАЦИЯ
FunctionalDescription	Категория рецептурной информации, содержащая вспомогательную информацию для поддержания процессов производства партии изделий и не содержащаяся в других частях рецептуры (например, информация о соответствии нормативным документам, информация о материалах, информация по обеспечению безопасности, технологические диаграммы, упаковочная/маркировочная информация)
АТРИБУТЫ	

Таблица 13 — Процедурные структурные элементы

ИМЯ	ПРОЦЕДУРНЫЕ СТРУКТУРНЫЕ ЭЛЕМЕНТЫ
FunctionalDescription	Процедурные элементы рецептуры и информация о порядке их выполнения
АТРИБУТЫ	

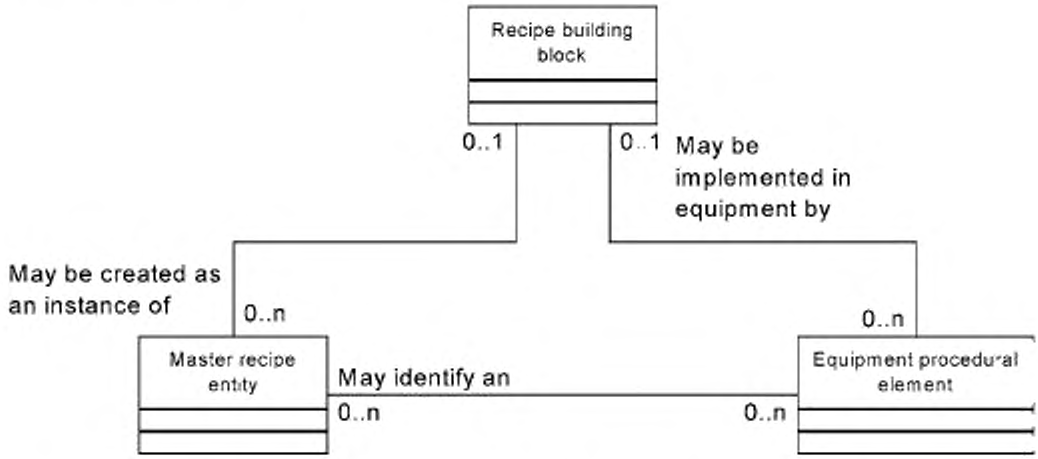
4.3.3 Взаимосвязи рецептурной сущности (процедурная структура)

Рецептурные сущности иерархически раскладываются на составные части по структурам процедурных сущностей в соответствии с МЭК 61512-1 (например, рецептурная процедура содержит процедуры технологической установки, которые содержат операции, содержащие, в свою очередь, фазы). Данная иерархия моделируется с помощью рекурсивного вложения. Объекты высокого уровня могут содержать объекты нижнего уровня.

Процедурные структурные элементы включают процедурные элементы рецептур и связи (например, соединительные звенья, переходы), используемые для их упорядочивания (например, процедурные структурные элементы рецептуры технологической установки — это операции и встроенные процедуры упорядочивания данных операций). Рассматриваемые процедурные структурные элементы могут быть взаимосвязаны с другими процедурными структурными элементами.

4.3.4 Структурные элементы рецептуры

Структурные элементы рецептуры — это важное понятие модели данных (см. рисунок 4). Данный рисунок определяет взаимосвязи на одном отдельном уровне иерархии процедур.



Recipe building block — структурные элементы рецептуры; may be implemented in equipment by — может быть реализовано в оборудовании с помощью; may be created as an instance of — может быть создано как реализация (экземпляр); master recipe entity — сущность технологической рецептуры; may identify an — может идентифицировать; equipment procedural element — процедурный элемент оборудования

Рисунок 4 — Структурные элементы рецептуры

Структурные элементы рецептуры — это структурные элементы, из которых создаются технологические рецептуры. При инстанцировании структурного элемента рецептуры в технологической рецептуре как сущности технологической рецептуры, результат инстанцирования (экземпляр) может содержать параметры, требования к оборудованию и прочую информацию, а также конкретные значения параметров технологических рецептур. Содержимое нижнего уровня структурного элемента рецептуры (например, подчиненные рецептурные сущности) может копироваться в сущности технологической рецептуры. Указанное содержимое нижнего уровня может также быть доступно путем ссылок на соответствующие структурные элементы рецептуры.

Функциональность структурного элемента рецептуры обеспечивается путем имплементации в оборудование процедурных элементов оборудования (см. таблицу 14), необходимых для выполнения рецептурных сущностей самого нижнего уровня (например, рецептурных сущностей, предназначенных для соединения с процедурными элементами оборудования).

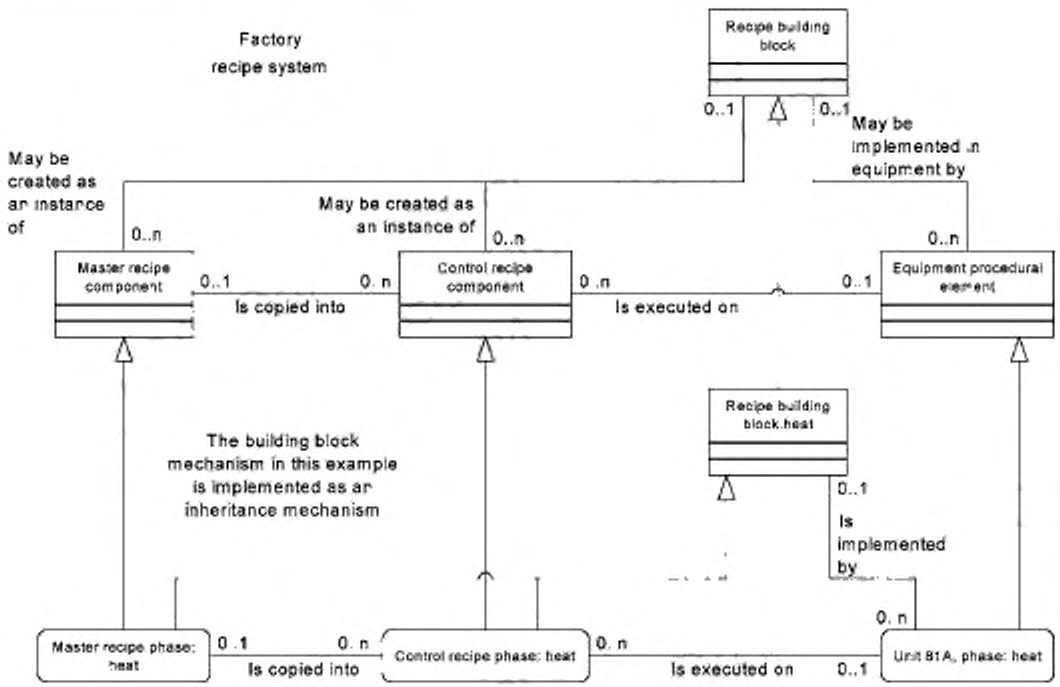
Таблица 14 — Процедурные элементы оборудования

ИМЯ	ПРОЦЕДУРНЫЕ ЭЛЕМЕНТЫ ОБОРУДОВАНИЯ
FunctionalDescription	Процедурные элементы, ассоциированные с конкретным оборудованием (например, фазой оборудования или операцией оборудования)
АТРИБУТЫ EquipmentProcedura lElementID	Обеспечивает уникальную идентификацию
Version	Идентифицирует версию процедурного элемента
VersionDate	Идентифицирует дату и время создания или модификации данной версии
ApprovalDate	Идентифицирует дату и время утверждения данной версии
Author	Идентифицирует физическое лицо или систему, являющуюся автором данной версии (например, J. Smith)
ApprovedBy	Идентифицирует физическое лицо или систему, утвердившую данную версию
Description	Содержит описание функции, полученной путем выполнения рецептурной сущности

Окончание таблицы 14

ИМЯ	ПРОЦЕДУРНЫЕ ЭЛЕМЕНТЫ ОБОРУДОВАНИЯ
Level	Указывает уровень сущности оборудования. Сущность данного оборудования может быть использована только на данном уровне
Mode	Указывает текущий режим процедурного элемента
State	Указывает текущее состояние процедурного элемента

Рассматриваемый механизм иллюстрируется следующим примером (см. рисунок 5). Он представляет собой модель объекта части реального приложения. Характерное понятие структурного элемента инстанцировано как конкретный структурный элемент «количество теплоты». Отношение типа «основан на» между структурными элементами и компонентами заменено на отношение подкласса (это одна возможная практическая реализация, и она указывает, что если структурный элемент «количеств теплоты» изменен, то данное изменение распространяется на все рецептуры, использующие элемент «количество теплоты»). Другая практическая реализация может заключаться в том, что элемент «количество теплоты» просто копируется при инстанцировании.



Factory recipe system — рецептурная система производственного предприятия; recipe building block — структурный элемент рецептуры; may be created as an instance of — может быть создан как реализация (экземпляр); may be implemented in equipment by — может быть реализован в оборудовании с помощью; master recipe component — компонент технологической рецептуры; is copied into — копируется в; control recipe component — компонент рецептуры управления; is executed on — выполняется с помощью; equipment procedural element — процедурный элемент оборудования; the building block mechanism in this example is implemented as an inheritance mechanism — в данном примере механизм структурного элемента реализуется как механизм наследственности; recipe building block: heat — структурный элемент рецептуры: нагрев; is implemented by — реализуется; master recipe phase heat — фаза технологической рецептуры: нагрев; control recipe phase: heat — фаза рецептуры управления: нагрев; unit 81a, phase: heat — технологическая установка № 81а, фаза – нагрев

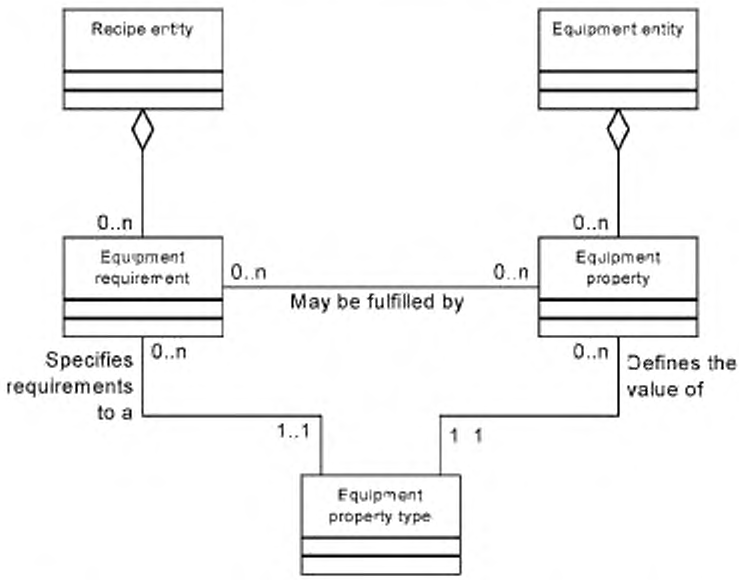
Рисунок 5 — Понятие структурного элемента

4.3.5 Требования к оборудованию

Рецептурные сущности могут содержать требования к оборудованию (см. рисунок 6, таблицы 15–17). Требования к оборудованию ссылаются на конкретные типы свойств оборудования (например, типом свойств оборудования может быть «размер резервуара» или «облицовка резервуара»). В данном случае конкретное требование к оборудованию может описывать минимальное значение размера резервуара.

Данное требование может быть справедливо в отношении единицы оборудования с конкретным свойством, которое ссылается на один и тот же тип свойств оборудования. Например, свойство конкретной технологической установки (например, ТЕХНОЛОГИЧЕСКОЙ УСТАНОВКИ №12) может иметь значение для заданного типа свойств «размер резервуара».

Сущность оборудования — это конкретная единица оборудования. Ее можно заменить классом оборудования. См. 4.4.



Recipe entity — рецептурная сущность; equipment entity — сущность оборудования; equipment requirement — требование к оборудованию; may be fulfilled by — может быть выполнено с помощью; equipment property — свойство оборудования; specifies requirements to a — устанавливает требования к, defines the value of — определяет значение для; equipment property type — тип свойств оборудования

Рисунок 6 — Требования рецептурной сущности к оборудованию

Таблица 15 — Сущность оборудования

ИМЯ	СУЩНОСТЬ ОБОРУДОВАНИЯ
FunctionalDescription	Совокупность физической обработки, управляющего оборудования и управления оборудованием, собранных вместе для реализации одной конкретной функции управления или множества функций управления
АТРИБУТЫ EquipmentEntityID	 Обеспечивает уникальную идентификацию
EquipmentLevel	Задает физический уровень иерархии (например, технологический цех, технологическая установка, блок оборудования, блок управления)
Mode	Указывает текущий рабочий режим сущности оборудования
State	Указывает текущее состояние сущности оборудования

Таблица 16 — Свойство оборудования

ИМЯ	СВОЙСТВО ОБОРУДОВАНИЯ
FunctionalDescription	Идентифицирует свойство, которое обеспечивает рассматриваемая сущность оборудования или класс оборудования. Данные свойства зависят от конкретного приложения (например, тип облицовки, размер, теплопроводность, температура пара)
АТРИБУТЫ PropertyID	 Обеспечивает уникальную идентификацию
Value	Идентифицирует значение свойства (например, «стекло», 50 000, 650)
ValueRange	Определяет пределы или ограничения, относящиеся к указанному значению
EngineeringUnits	Определяет технические единицы измерения свойств
Description	Описывает рассматриваемый тип свойств оборудования

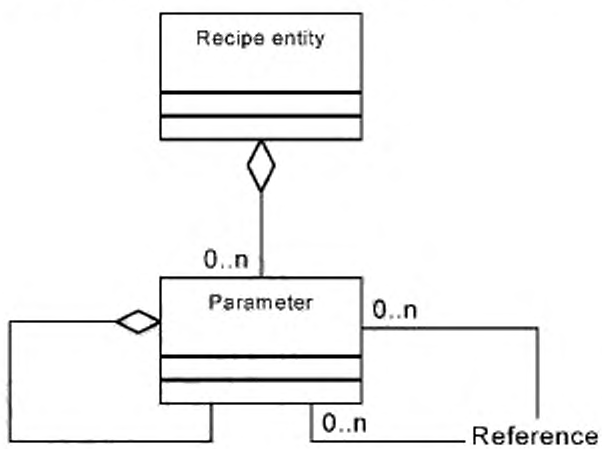
Таблица 17 — Тип свойств оборудования

ИМЯ	ТИП СВОЙСТВ ОБОРУДОВАНИЯ
FunctionalDescription	Общий класс атрибутов оборудования (например, тип облицовки, размер, теплопроводность, температура пара)
АТРИБУТЫ	

Сущности рецептуры управления изначально содержат требования к оборудованию, которые копируются из сущности технологической рецептуры, которая, в свою очередь, выполняется с помощью соответствующего свойства одной или нескольких сущностей оборудования, чтобы выделить конкретное оборудование в соответствии с запросом. Начальное требование к оборудованию может быть заменено выделением конкретного оборудования. Данное выделение также можно моделировать как требование к оборудованию.

4.3.6 Параметры рецептуры

Параметры — это переменные, ассоциированные с рецептурной сущностью. Данные переменные могут быть использованы процедурными элементами оборудования. Они могут быть использованы прочими действиями (например, при разработке календарного плана), или на них можно ссылаться из других частей рецептуры (например, критерий перехода) (см. рисунок 7).



Recipe entity — рецептурная сущность; parameter — параметр; reference — ссылка

Рисунок 7 — Параметрическая модель

Параметры можно категоризировать как входные сигналы технологического процесса, выходные сигналы технологического процесса, параметры процесса.

Одни параметры могут представлять собой совокупности других параметров. Рассматриваемая модель поддерживает понятие структурированного параметра. Следовательно, данная модель допускает возможность включения параметров различных типов (параметры процесса, входные сигналы технологического процесса, выходные сигналы технологического процесса) в ту же самую структуру, также как и определение структур данных отдельного типа.

Атрибуты значений параметров могут быть организованы путем определения типов значений параметров. Типы значений параметров могут включать:

- базовые типы данных в соответствии с МЭК 61131-3;
- информацию о матрице совместимости, используемую для определения требований «чистки по месту» (CIP, clean-in-place) или «стерилизации-по-месту» (SIP, sterilize-in-place);
- наборы данных, определяющие транзакции материала (передача, потребление, генерация материала);
- наборы данных (например, отслеживаемый профиль температур).

Значения параметров могут быть простыми значениями, выражениями или ссылками на параметры, определяемыми на том же самом уровне или на более высоких уровнях в процедурной иерархии. Значения, являющиеся выражениями, могут включать ссылки на другие параметры.

Допустимые формы представления параметров:

- алгебраические или булевы выражения;
- специальные формы записи информации о продукте, включающие один или несколько параметров;
- стандартные рабочие (операционные) процедуры (SOP), которые отображают или используют параметры другим способом (например, динамические значения, значения рецептуры);
- отнесение параметров к различным рецептурным сущностям (на том же самом уровне или на другом уровне);
- внешние приложения, использующие параметры.

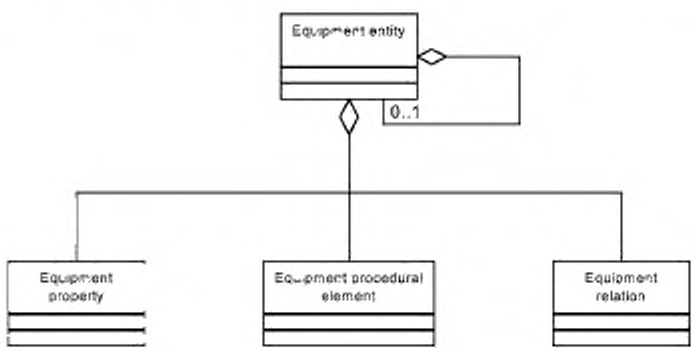
Формулы представляются в модели данных как параметры рецептуры (см. таблицу 10). Формула рецептуры — это совокупность выбранных параметров рецептурной процедуры. Она также может включать параметры, определенные на нижних уровнях процедурной иерархии.

Масштабирование параметров часто зависит от объема партии изделий или на другом ключевом атрибуте. Масштабирование может быть более сложным, чем простое линейное отношение. Более сложные методы масштабирования можно адаптировать к алгоритмам и отношениям, определенным пользователем.

4.4 Модель оборудования

При оценке выбора оборудования в ходе выполнения рецептуры необходимо принимать во внимание физическую структуру установки (см. рисунок 8 и таблицу 18). В частности, для маршрутизации производства партии изделий важно учитывать пропускную способность оборудования или возможность выделения оборудования для общего пользования.

Сущности оборудования определяются в соответствии с имеющейся иерархией (см. МЭК 61512-1). Указанная иерархия моделируется с учетом рекурсивной природы объектов. Рассматриваемый конструктивный элемент допускает расширение и сжатие конфигурации.



Equipment entity — сущность оборудования; equipment property — свойство оборудования; equipment procedural element — процедурный элемент оборудования; equipment relation — взаимосвязи оборудования

Рисунок 8 — Структура оборудования

Таблица 18 — Взаимосвязи оборудования

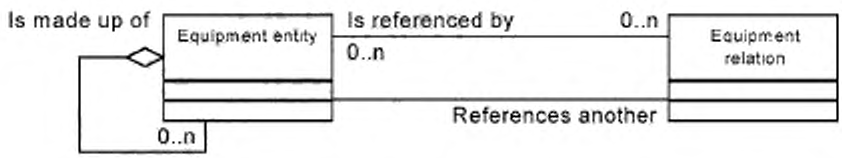
ИМЯ	ВЗАИМОСВЯЗИ ОБОРУДОВАНИЯ
FunctionalDescription	Представление связей между единицами оборудования (например, трубы, конвейеры, гибкие связи). Также может использоваться для представления прочих видов отношений между единицами оборудования (например, взаимосвязи оборудования совместного использования)
АТРИБУТЫ RelationID	Обеспечивает уникальную идентификацию

Оборудование, например, технологического цеха (например, технологические установки, блоки оборудования, блоки управления) связываются друг с другом трубами или соединительными элементами. Данные соединительные элементы могут моделироваться как взаимосвязи оборудования (см. рисунок 9). Направление соединения можно выбирать (например, направление потока). Указанные взаимосвязи (например, трубы) — это часть сущности оборудования более высокого уровня. Данные соединения можно категоризировать в классы отношений, что обеспечивает их правильную оценку. К взаимосвязям оборудования относятся:

- постоянные соединения;
- временные соединения;
- могут использоваться как ресурс;
- всегда используются для одного продукта.

Отметим, что отношения, отличные от указанных, также возможны.

Оборудование может иметь свойства. Для каждой практической реализации характерны свои свойства. Свойства оборудования могут использоваться для проверки характеристик оборудования на предмет их соответствия требованиям рецептуры. См. 4.3.5



Is made up of — составлено из; equipment entity — сущность оборудования; is referenced by — является ссылкой для; references another — ссылается на другую; equipment relation — взаимосвязь оборудования

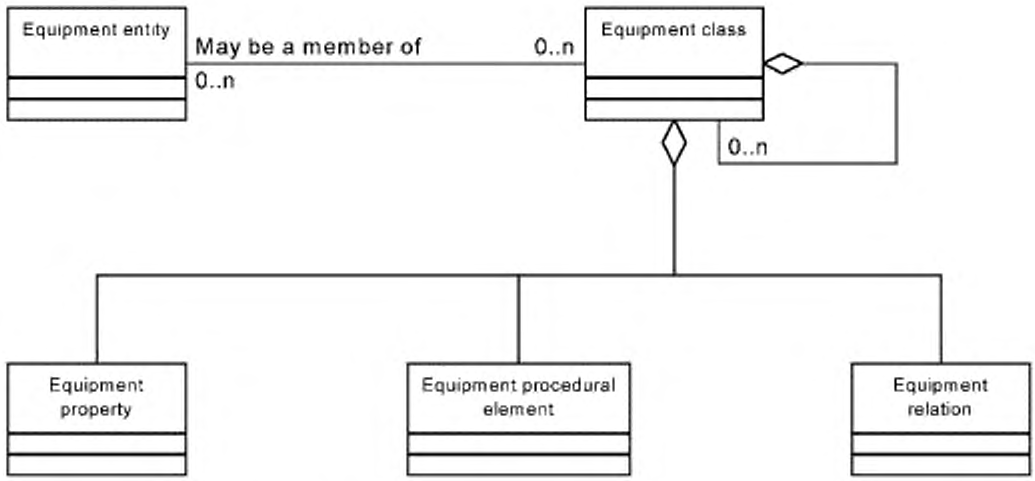
Рисунок 9 — Отношения сущностей оборудования

Классы оборудования (см. рисунок 10 и таблицу 19) обеспечивают средства группировки сущностей оборудования по общим характеристикам. Сущности оборудования могут быть членами одного или нескольких классов оборудования. Они могут не принадлежать ни к какому классу. Классы оборудования могут быть использованы для описания групп технологических установок. Они могут также быть использованы в качестве альтернативы при выборе оборудования. Например, рецептура может потребовать использование реактора для конкретной процедуры технологической установки: ее требования к оборудованию могут описывать один конкретный реактор (например, реактор R-101), несколько реакторов (например, реакторы R-101, R-103) или целый класс реакторов (например, класс "реактор", содержащий реакторы R-101, R-102 и R-103).

Сущности оборудования могут быть элементами класса оборудования, а класс определяет некоторые свойства элементов класса. Определенные свойства оборудования (например, облицовка стеклом) могут являться общими для всего класса.

Сущности оборудования могут не принадлежать никаким или принадлежать нескольким классам оборудования (например, резервуар BV1 может быть как реактором, так и накопительной емкостью).

Класс оборудования может определять некоторые или все свойства оборудования, процедурные элементы оборудования и взаимосвязи оборудования для ссылочных сущностей оборудования.



Equipment entity — сущность оборудования; may be a member of — может быть элементом; equipment class — класс оборудования; equipment property — свойство оборудования; equipment procedural element — процедурный элемент оборудования; equipment relation — взаимосвязь оборудования

Рисунок 10 — Классы оборудования

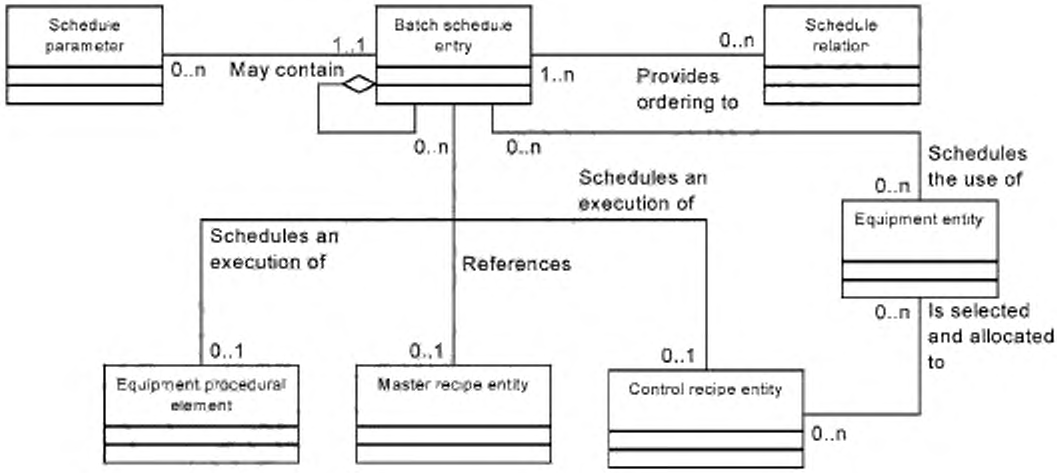
Таблица 19 — Классы оборудования

ИМЯ	КЛАСС ОБОРУДОВАНИЯ
FunctionalDescription	Класс сущностей оборудования
АТРИБУТЫ	
EquipmentClass	Обеспечивает уникальную идентификацию
EquipmentLevel	Задает физический уровень иерархии (например, технологический цех, технологическая установка, блок оборудования, блок управления)

4.5 Разработка производственного и календарного планов

Центральной сущностью календарного плана (см. рисунок 11) является запись в календарном плане производства партии изделий. Данный объект определяет планируемую разработку одной или нескольких рецептур производства партии изделий, рецептур управления или прочих сущностей рецептуры управления (обычно это процедуры технологической установки) (см. таблицу 20). Запись в календарном плане может также быть использована для календарного планирования прочих действий (например, учет простоя оборудования). Запись в календарном плане производства партии изделий может также включать значения формулы/параметра, используемые в рецептуре управления (см. таблицу 21).

Запись в календарном плане производства партии изделий может также быть использована для представления календарных сущностей более высокого уровня (например, для производственной кампании и организации производства).



Schedule parameter — календарный параметр; may contain — может содержать; batch schedule entry — запись в календарном плане производства партии изделий; provides ordering to — обеспечивает упорядочивание; schedule relation — календарное отношение, schedules an execution of — формирует календарный план выполнения; schedules the use of — формирует календарный план использования; references — ссылается, equipment entity — сущность оборудования; is selected and allocated — выбрано и выделено для; equipment procedural element — процедурный элемент оборудования; master recipe entity — сущность технологической рецептуры; control recipe entity — сущность рецептуры управления

Рисунок 11 — Календарный план производства партии изделий

Таблица 20 — Запись в календарном плане производства партии изделий

ИМЯ	ЗАПИСЬ В КАЛЕНДАРНОМ ПЛАНЕ ПРОИЗВОДСТВА ПАРТИИ ИЗДЕЛИЙ
FunctionalDescription	Пункт календарного плана, представляющий процедуру технологической установки при производстве партии изделий, готовую партию изделий или множество партий изделий (например, производственную кампанию)
АТРИБУТЫ ID	Обеспечивает уникальную идентификацию (например, фактической производственной кампании, серии, идентификатора партии, идентификатора процедурной сущности)
Level	Задает уровень иерархии (например, производственной кампании, производства партии изделий, процедуры технологической установки)
BatchSize	Определяет запрошенный размер (масштабный фактор) партии изделий, основанный на масштабном факторе партии, определенном в технологической рецептуре
Schedule	Определяет календарные сроки (начала работ/окончания работ)
ResourceUsage	Определяет порядок использования ресурса для данной календарной записи
Status	Задает календарный статус (например, предложенный для оценки (например, анализ возможных вариантов), плановый, задействованный, начатый, законченный)

Таблица 21 — Календарный параметр

ИМЯ	КАЛЕНДАРНЫЙ ПАРАМЕТР
FunctionalDescription	Значение формулы, которое указывается в календарном плане производства партии изделий (поступает в качестве входных данных)
АТРИБУТЫ ParameterID	Обеспечивает уникальную идентификацию
ParameterType	Определяет порядок интерпретации значения (например, константы, ссылки или уравнения)

Окончание таблицы 21

ИМЯ	КАЛЕНДАРНЫЙ ПАРАМЕТР
Description	Содержит описание параметра или порядок использования параметра
EngineeringUnits	Идентифицирует технические единицы измерения для рассматриваемого значения (например, кг, фунт)
Value	Содержит значение параметра. Если это отношение, то значение содержит выражение, форму, отложенное правило или что-то, что связывает параметры вместе. Если значение – это параметр структурного элемента, то данный атрибут содержит значение по умолчанию
Scaled	Определяет правила масштабирования, например, масштабирован или не масштабирован по отношению к базовому значению размера партии
Usage	Идентифицирует параметр как входной сигнал технологического процесса, выходной сигнал технологического процесса, параметр процесса

Рассматриваемые календарные отношения могут быть использованы для представления подмножества рецептурных отношений, соответствующих календарному плану (например, отношение перемещения партии) (см. таблицу 22). На более высоких уровнях, они могут быть использованы для представления требуемых (желательных) отношений между календарными записями (например, партия ***xx*** должна следовать за партией ***yy***, или конкретная операция очистки должна иметь место между производством двух партий изделий). Отношения конкретного подкласса и характеристик календарной записи в настоящем стандарте не рассматриваются.

Запись в календарном плане производства партии изделий более высокого уровня включает календарные записи и отношения более низкого уровня (например, запланированная производственная кампания может включать календарное производство партий изделий и отношения между данными партиями).

В простейшем случае запись в календарном плане производства партии изделий представляет рассматриваемую партию в перечне (очереди) производства партии изделий, инициируемого в установленном порядке. Сюда добавляются плановое время начала работ и расчетная продолжительность (расчетное время окончания) работ. Более того, запись в календарном плане производства партии изделий может определять назначение оборудования и использование прочих ресурсов. Разработка календарного плана может производиться и на более детальном уровне (например, разработка календарного плана индивидуальных рецептур технологической установки, назначение рецептур для оборудования, потенциально возможная разработка календарного плана индивидуальных операций или фаз, их расчетная продолжительность, расходование ресурсов, включая ресурсы совместного использования или эксклюзивные ресурсы, ограничивающие календарное планирование). Технологические установки и блоки оборудования выделяются (освобождаются) в соответствии с конкретной рецептурой управления.

Совокупность календарных записей можно рассматривать как:

- список партий или карту производства партий изделий в технологическом цехе (или его части), обеспечивающую загрузку производственных мощностей;
- список или карту использования ресурсов в соответствии с календарным планом расходования ресурса или загрузки оборудования.

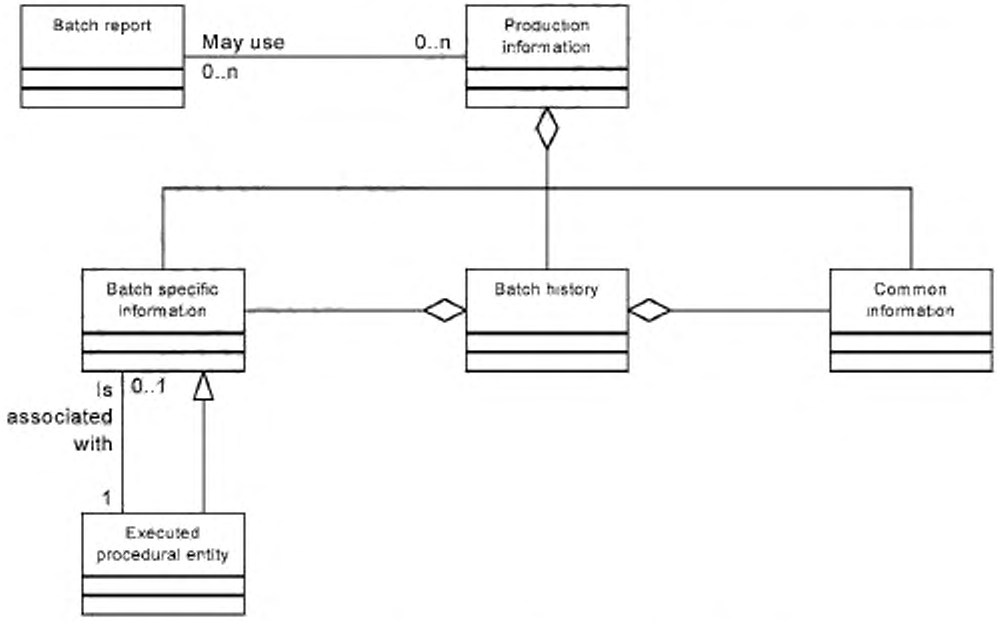
Таблица 22 — Календарное отношение

ИМЯ	КАЛЕНДАРНОЕ ОТНОШЕНИЕ
FunctionalDescription	Представляет отношения между календарными записями (например, требуемые габариты линии, операции очистки между календарными мероприятиями, спецификации последовательных отношений внутри процедур)
АТРИБУТЫ	
ExecutionOrder	Указывает, в какой последовательности несколько календарных записей реализуются по отношению друг к другу

4.6 Управление производственной информацией

Данный подраздел содержит описание моделей, определяющих порядок сбора производственной информации.

Производственная информация, включая своевременно полученную информацию о ходе процесса производства, может включать специальную информацию о производстве партии изделий, информацию, не относящуюся к производству партии изделий, общую информацию (см. рисунок 12 и таблицы 23—28).



Batch report — отчет о производстве партии; may use — может использовать; production information — производственная информация, batch specific information — специальная информация о производстве партии; batch history — история производства партии изделий; common information — общая информация, is associated with — ассоциирована с; executed procedural entity — выполненная процедурная сущность

Рисунок 12 — Производственная информация

Таблица 23 — Производственная информация

ИМЯ	ПРОИЗВОДСТВЕННАЯ ИНФОРМАЦИЯ
FunctionalDescription	Информация, генерируемая в ходе производства партии изделий
АТРИБУТЫ	

Таблица 24 — Специальная информация о производстве партии изделий

ИМЯ	СПЕЦИАЛЬНАЯ ИНФОРМАЦИЯ О ПРОИЗВОДСТВЕ ПАРТИИ ИЗДЕЛИЙ
FunctionalDescription	Данные, относящиеся к одной записи истории производства партии изделий
АТРИБУТЫ	
BatchID	Определяет фактический идентификатор партии
EntryID	Обеспечивает уникальную идентификацию
NewValue	Указывает текущее фактическое значение
EngineeringUnits	Определяет технические единицы измерения (если они есть), соответствующие атрибуту NewValue
EquipmentID	Идентифицирует элемент оборудования, ассоциированный с записью

Окончание таблицы 24

ИМЯ	СПЕЦИАЛЬНАЯ ИНФОРМАЦИЯ О ПРОИЗВОДСТВЕ ПАРТИИ ИЗДЕЛИЙ
UTC	Идентифицирует универсальное координированное время (UTC) и дату внесенной записи
UserID	Идентифицирует пользователя (если он существует), ассоциированного с внесенным изменением

Таблица 25 — История производства партии изделий

ИМЯ	ИСТОРИЯ ПРОИЗВОДСТВА ПАРТИИ ИЗДЕЛИЙ
FunctionalDescription	Элемент информации, документирующий производство партии изделий
АТРИБУТЫ	
BatchID	Определяет фактический идентификатор партии

Таблица 26 — Общая информация

ИМЯ	ОБЩАЯ ИНФОРМАЦИЯ
FunctionalDescription	Данные, относящиеся к более чем одной записи истории производства партии изделий (например, температура охлаждающей воды, атмосферное давление, паропроизводительность)
АТРИБУТЫ	
EntryID	Обеспечивает уникальную идентификацию
NewValue	Указывает текущее фактическое значение
EngineeringUnits	Определяет технические единицы измерения (если они существуют), соответствующие атрибуту NewValue
EquipmentID	Идентифицирует элемент оборудования, ассоциированный с записью
UTC	Идентифицирует универсальное координированное время (UTC) и дату внесения записи
UserID	Идентифицирует пользователя (если он существует), ассоциированного с внесенным изменением

Производственная информация может включать:

- копию рецептуры управления;
- копию технологической рецептуры;
- информацию о материалах, используемых и изготовленных;
- информацию о технологическом процессе в динамике;
- сигналы тревоги и сообщения;
- информацию о взаимодействии оператора с партией (например, внесение записи, подтверждение);
- последние записи, асинхронные записи (например, измерения лабораторных образцов);
- дополнительную информацию (например, выделение оборудования, начало работ/окончание работ).

Выполненная процедурная сущность — это регистрация реализации (экземпляра записи) выполнения рецептурной сущности или процедурной сущности оборудования (см. таблицу 27). Данные ассоциируются с фактом реализации, отражаются в соответствующей записи истории производства партии изделий.

Выполненные процедурные сущности, являющиеся результатом выполнения рецептуры управления, часто имеют ту же самую структуру, что и сущности рецептуры управления. Структура выполненных процедурных элементов может, вместе с тем, в некоторых случаях отличаться от рецептуры управления. Типовые примеры:

- повторные реализации рецептурной сущности, созданные посредством замыкания в процедурной логике;
- реализации, не выполненные вследствие наличия ветвей или условных переходов в процедурной логике;

- рецептурные сущности, вставленные или повторяемые вручную;
- рецептурные сущности, активированные на участке технологической установки и уже ассоциированные (ассоциируемые вручную) с производством партии изделий.

Отношение рецептурной сущности и оставшейся части общей схемы (модели) рецептурной сущности в настоящем стандарте не рассматривается, так как история производства отражает реальные факты, а не плановую структуру.

Таблица 27 — Выполненная процедурная сущность

ИМЯ	ВЫПОЛНЕННАЯ ПРОЦЕДУРНАЯ СУЩНОСТЬ
FunctionalDescription	Представление выполненной рецептурной сущности (например, процедурного элемента оборудования)
АТРИБУТЫ ExecutedProcedural EntityID	Обеспечивает уникальную идентификацию
ProceduralEntity Counter	Уникально идентифицирует повторное выполнение той же самой процедурной сущности

Отчеты о производстве партии изделий (см. таблицу 28) рассматриваются как извлечение данных о производстве партии из всей совокупности производственной информации. Они отображаются на экране (на бумаге) или передаются прочим системам.

Таблица 28 — Отчет о производстве партии изделий

ИМЯ	ОТЧЕТ О ПРОИЗВОДСТВЕ ПАРТИИ ИЗДЕЛИЙ
FunctionalDescription	Компонент отчета о производстве партии изделий
АТРИБУТЫ ReportID	Обеспечивает уникальную идентификацию

5 Реляционные таблицы обмена информацией

5.1 Общие положения

Данный раздел определяет структуру реляционных таблиц Языка структурированных запросов (SQL) для обмена необходимой информацией об управлении производством между системами. Данный раздел устанавливает спецификацию интерфейса (в соответствии с требованиями раздела 4) для обмена информацией о производстве партии изделий для следующих установленных категорий:

- технологическая информация и информация о рецептуре управления;
- информация об оборудовании технологического цеха;
- информация календарного планирования;
- производственная информация.

Таблицы обмена должны включать имя таблицы, имя поля и отношения, определенные в соответствии с настоящим разделом. Не все таблицы могут найти практическую реализацию, но все информационные поля в реализованных таблицах должны присутствовать. Каждая практическая реализация должна согласовываться с представленными табличными определениями и понятиями, установленными в МЭК 61512-1.

Табличный формат обмена определяет только стандартную информацию. Табличные определения могут быть расширены посредством включения дополнительных атрибутов, дополнительных соответствующих таблиц и дополнительной нумерации. Расширенные структуры могут использоваться для обмена информацией между структурно-ориентированными инструментами, но данная информация не является предметом рассмотрения настоящего стандарта.

Примеры дополнительной информации:

- дополнительные иконки, представляющие различные элементы иерархии оборудования, так что рассматриваемое оборудование может быть последовательно представлено его различными инструментами;
- добавление адресов системы управления для процедур элементов оборудования, параметров процедур, элементов данных.

5.1.1 Метод

Структура реляционных таблиц определяется с помощью языка SQL в соответствии с ИСО/МЭК 9075.

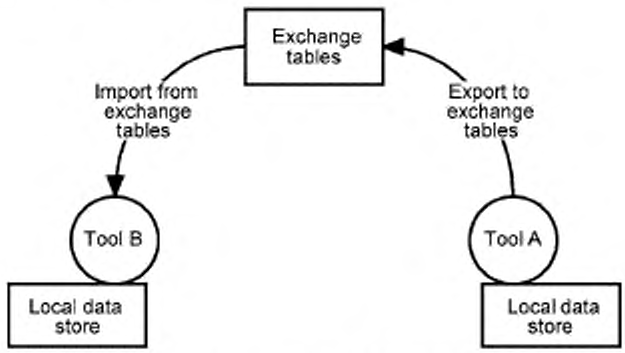
Механизм обмена основан на общей структуре таблиц базы данных. Указанные таблицы определены в качестве схемы базы данных, реализованной на языке SQL. Приложение В содержит табличные определения языка SQL для таблиц обмена.

Рисунок 13 иллюстрирует порядок использования таблиц обмена для реализации информационного обмена между различными инструментами в процессе серийного производства. Каждый инструмент, как правило, имеет свои собственные локальные устройства памяти для хранения информации о производстве партии изделий. Каждый инструмент обеспечивает импортирование информации из (экспортирование информации в) таблицы обмена.

Большинство из оставшихся моделей данного раздела представляются с помощью диаграмм отношения сущностей (ERS) (см. А.3).

Типы инструментов и порядок их использования (с учетом информации о производстве партии изделий) в настоящем стандарте не определены. Данные инструменты включают (и не только):

- системы авторизации (создания) рецептур;
- системы выполнения рецептур;
- системы оформления документации;
- системы конфигурирования;
- системы моделирования;
- системы управления производством партии изделий;
- системы разработки производственного и календарного планов;
- системы управления информацией.



Exchange tables — таблицы обмена; import from exchange tables — импортирование из таблиц обмена; export to exchange tables — экспортирование в таблицы обмена; tool a — инструмент а; tool b — инструмент в; local data store — локальное устройство памяти для хранения данных

Рисунок 13 — Передача данных с помощью таблиц обмена

5.1.2 Таблицы обмена

Настоящий стандарт не описывает инструменты для создания и обслуживания таблиц обмена. Настоящий стандарт только определяет структуру указанных таблиц. Инструменты могут считывать информацию из предварительно заготовленных таблиц обмена, писать информацию в таблицы обмена, считывать и писать информацию, считывать, писать и создавать информацию для таблиц обмена.

Структура таблиц обмена обеспечивает возможность обмена сразу несколькими рецептурами на одном множестве таблиц. Она допускает обмен несколькими версиями одной рецептуры. Структура таблиц обмена также допускает обмен информации о технической оснащенности и о спецификации оборудования, имеющегося в технологическом цехе.

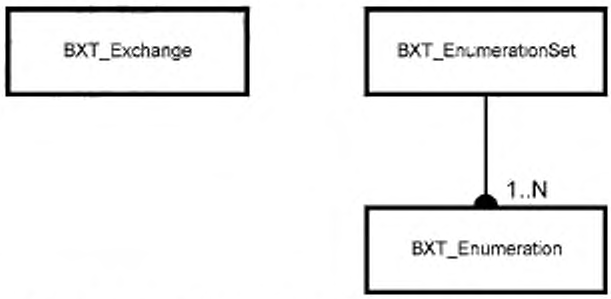
Структура таблиц обмена также допускает передачу как полных, так и неполных подмножеств технологических рецептур, описаний оборудования, календарной информации, истории производства партии изделий.

Синтаксис строк данных таблицы обмена (например, вычисленные значения элементов формул, условия перехода) в настоящем стандарте не определен. Инструменты, записывающие и считывающие таблицы обмена на языке SQL, должны учитывать особенности синтаксиса.

Длины информационных полей, установленных для определения таблиц обмена в языке SQL, должны быть достаточны для представления значений по умолчанию, но не должны выходить за минимальный или максимальный пределы размеров строки. Инструменты, записывающие и считывающие таблицы обмена в языке SQL, должны учитывать длину информационной строки.

5.1.3 Общий обмен информацией

Рассматриваемое множество таблиц содержит информацию, включающую описание формата обмена и используемую, в общем случае, в различных множествах таблиц обмена. На рисунке 14 приведена таблица, используемая для обмена общей информацией.



BXT_Exchange — таблица обмена общей информацией, BXT_EnumerationSet — таблица множества элементов перечисления; BXT_Enumeration — таблица нумерации

Рисунок 14 — Таблица обмена общей информацией

5.1.3.1 Обмен информацией

Таблица обмена общей информацией, BXT_Exchange, содержит всю информацию, используемую только один раз при обмене данными (см. таблицу 29).

Таблица обмена BXT_Exchange содержит (см. таблицу 30) одну запись по каждому пункту (например, признаки SCHEMA и DELIMITER). Она также может содержать прочую информацию для пользователя.

Таблица 29 — BXT_Exchange

Атрибут	Описание
ExchangeID	Идентифицирует элемент обмена
ExchangeValue	Идентифицирует значение для обмениваемой информации

Таблица 30 — Содержание таблицы обмена BXT_Exchange

ExchangeID	Описание
Schema	Версия стандарта, используемого при определении схемы базы данных. Начальное значение версии совпадает с № настоящего стандарта (например, версия МЭК 61512-2:2001)
Delimiter	Признак, используемый для установления различия между именами элементов в иерархии имен рецептурных элементов
ToolID	Идентификация инструмента, создающего таблицы обмена
ToolVersion	Версия инструмента, создавшего таблицы обмена
ToolSchema	Версия адаптированной версии, созданной рассматриваемым инструментом

5.1.3.2 Множества элементов перечисления

Многие таблицы имеют информационные поля, содержащие стандартные или определенные пользователем перенумерованные пункты. Данные перенумерованные пункты проходят в таблице обмена как номера, строки располагаются в таблице множества элементов перечисления. Таблицы множества элементов перечисления (см. таблицу 31) обеспечивают однократное размещение переводов строки на различные языки. Таблица BXT_EnumerationSet определяет множество элементов перечисления. Таблица BXT_Enumeration указывает на соответствующий элемент множества и ассоциированное численное значение.

Таблицы нумерации задают стандартную нумерацию и соответствующие значения. Таблица может быть расширена пользователем. Дополнительная нумерация пользователя для стандартных множеств элементов перечисления может принимать значения от 100 и выше. В настоящем стандарте используются (резервируются) значения нумерации от 0 до 99. Кроме того, в таблице нумерации пользователь может определить множества элементов перечисления и их соответствующие значения. Например, множество элементов перечисления «компаундирование масла» может быть определено для фазовых параметров с элементами «масло первого прессования», «масло с присадками» и «регенерированное масло» со значениями 101, 102 и 103, соответственно.

Таблица 31 — BXT_EnumerationSet

Атрибут	Описание
EnumSet	Идентифицирует стандартное множество элементов перечисления
Description	Поддерживает использование множества элементов перечисления (при транслировании текстовой строки TextString)

Таблица 32 задает стандартное множество элементов перечисления, определенное в настоящем стандарте.

Таблица 32 — Стандартное множество элементов перечисления

EnumSet	Описание
Boolean	Определяет множество булевых значений
DirectionType	Определяет порядок использования параметра
EquipmentLevel	Определяет иерархический уровень оборудования для элементов оборудования
EquipmentType	Определяет тип записи оборудования для элементов оборудования
EvaluationRule	Определяет правила оценки свойств оборудования
FormulaType	Определяет типы рецептурной формулы
FormulaSubType	Задает определения подтипа пользовательской формулы
LinkDepiction	Определяет порядок отображения соединительных звеньев, установленных между рецептурными элементами
LinkToType	Определяет наличие ссылки соединительного звена на шаг или на переход
LinkType	Определяет тип соединительного звена
RE_Type	Определяет рецептурный элемент (RE), уровень рецептурной процедуры, обозначение выделения
RE_Use	Определяет порядок использования рецептурного элемента (RE) в рецептуре
RecipeStatus	Определение возможного статуса рецептуры
RecordSet	Определяет множество элементов перечисления, используемое для классификации записи в категорию информации об истории производства партии изделий
RecordRecordSetControlRecipe	Обеспечивает дальнейшую классификацию исторических записей в категории ControlRecipe
RecordSetMasterRecipe	Обеспечивает дальнейшую классификацию исторических записей в категории MasterRecipe
RecordSetExecutionInfo	Обеспечивает дальнейшую классификацию исторических записей в категории ExecurtionInfo
RecordSetMaterialInfo	Обеспечивает дальнейшую классификацию исторических записей в категории MaterialInfo

Окончание таблицы 32

EnumSet	Описание
RecordSetContinuousData	Обеспечивает дальнейшую классификацию исторических записей в категории ContinuousData
RecordSetEvents	Обеспечивает дальнейшую классификацию исторических записей в категории Events
RecordSetOperatorChange	Обеспечивает дальнейшую классификацию исторических записей в категории OperatorChange
RecordSetOperatorComment	Обеспечивает дальнейшую классификацию исторических записей в категории OperatorComment
RecordSetAnalysisData	Обеспечивает дальнейшую классификацию исторических записей в категории AnalysisData
RecordSetLateRecord	Обеспечивает дальнейшую классификацию исторических записей в категории LateRecord
RecordSetRecipeData	Обеспечивает дальнейшую классификацию исторических записей в категории RecipeData
RecordSetRecipeSpecified	Обеспечивает дальнейшую классификацию исторических записей в категории RecipeSpecified
RecordSetSummaryData	Обеспечивает дальнейшую классификацию исторических записей в категории SummaryData
ScheduleAction	Определяет планируемое действие для календарной записи
ScheduleMode	Определяет режим, в котором начинает выполняться календарная запись
ScheduleStatus	Определяет возможный статус календарного плана
SE_Type	Определяет тип сущности для календарной записи
ValueDataType	Определяет тип данных для ассоциированного значения данных
ValueType	Определяет порядок интерпретации строки значений

Таблица 33 указывает порядок определения нумерации в настоящем стандарте.

Таблица 33 — Таблица BXT_Enumeration

Атрибут	Описание
EnumSet	Идентифицирует имя множества элементов перечисления
EnumValue	Задает численное значение, ассоциированное с элементом нумерации
EnumString	Определяет ассоциированный текст для элемента нумерации
Description	Определяет порядок использования элемента нумерации (способствует переводу текстовой строки TextString)

Таблица 34 содержит список стандартных элементов нумерации, определенных настоящим стандартом.

Таблица 34 — Стандартная нумерация

EnumSet	EnumValue	EnumString	Описание
Boolean	0	FALSE	Определение булевского значения
	1	TRUE	
DirectionType	0	Invalid	Запись некорректна

Продолжение таблицы 34

EnumSet	EnumValue	EnumString	Описание
DirectionType	1	Internal	Идентифицирует порядок пользования параметром. Internal = означает доступность только в рамках рецептурного элемента. Определяется при создании или создается как промежуточное значение
	2	Input	Рецептурный элемент получает значение от внешнего источника
	3	Output	Рецептурный элемент создает значение и делает его доступным для внешнего использования
	4	Input/Output	Рецептурный элемент и внешний элемент обмениваются значением и могут изменить это значение
	5-99		Зарезервированы
	100 +		Определены пользователем
EquipmentLevel	0	Invalid	Запись некорректна
	1	Enterprise	Идентифицирует иерархический уровень оборудования для таблицы BXT_ EquipElement
	2	Site	
	3	Area	
	4	Process Cell	
	5	Unit	
	6	Equipment module	
	7	Control module	
	8-99		Зарезервированы
	100 +		Определены пользователем
EquipmentType	0	Invalid	Запись некорректна
	1	Class	Идентифицирует тип записи для таблицы BXT_EquipElement
	2	Элемент	
	3-99		Зарезервированы
	100 +		Определены пользователем
EvaluationRule	0	Invalid	Запись некорректна
	1	=	Оператор равенства для свойств оборудования
	2	<>	Оператор неравенства для свойств оборудования
	3	<	Оператор «меньше чем» для свойств оборудования
	4	>	Оператор «больше чем» для свойств оборудования
	5	<=	Оператор «меньше или равно» для свойств оборудования
	6	>=	Оператор «больше или равно» для свойств оборудования
	7	Member	Оператор «является элементом» для свойств оборудования

Продолжение таблицы 34

EnumSet	EnumValue	EnumString	Описание
EvaluationRule	8	Not member	Оператор «не является элементом» для свойств оборудования
	9	Not	Оператор «сравнение отсутствует» для свойств оборудования
	10-99		Зарезервированы
	100 +		Определены пользователем
FormulaSubType	0	Invalid	Запись некорректна
	1-99		Зарезервированы
	100 +		Определены пользователем. Допускает дальнейшую классификацию типа формулы FormulaType пользователем
FormulaType	0	Invalid	Запись некорректна
	1	Process input	Тип рецептурной формулы
	2	Process output	
	3	Process parameter	
	4-99		Зарезервированы
	100 +		Определены пользователем
LinkDepiction	0	Invalid	Запись некорректна.
	1	None	Отображение соединительного звена отсутствует.
	2	Line	Соединительное звено обозначено только линией.
	3	ID	Соединительное звено обозначено только идентификатором.
	4	Line & ID	Соединительное звено обозначено и линией, и идентификатором.
	5	Line & Arrow	Соединительное звено указано линией и стрелкой потока материала.
	6	Line, Arrow, & ID	Соединительное звено указано линией, стрелкой потока материала и идентификатором.
	7-99		Зарезервированы.
	100+		Определены пользователем
LinkToType	0	Invalid	Запись некорректна
	1	Recipe Element	Соединительное звено ссылается на запись в таблице BXT_MRecipeElement
	2	Transition	Соединительное звено ссылается на запись в таблице BXT_MRecipeTransition
	3-99		Зарезервированы
	100+		Определены пользователем
LinkType	0	Invalid	Запись некорректна
	1	ControlLink	Определяет соединительное звено между рецептурными элементами, указывающими поток процедурного управления
	2	TransferLink	Определяет соединительное звено между рецептурными элементами, указывающими на передачу материала

Продолжение таблицы 34

EnumSet	EnumValue	EnumString	Описание
LinkType	3	SynchronizationLink	Определяет соединительное звено между рецептурными элементами, если существует некоторая форма синхронизации
	4-99		Зарезервированы
	100+		Определены пользователем
RE_Type	0	Invalid	Запись некорректна
	1	Master Recipe	Задает тип рецептурного элемента
	2	Procedure	
	3	UnitProcedure	
	4	Operation	
	5	Phase	
	6	Allocation	
	7	Begin	
	8	End	
	9	StartParallel	
	10	EndParallel	
	11	StartBranch	
	12	EndBranch	
	13-99		Зарезервированы
	100+		Определены пользователем
RE_Use	0	Invalid	Запись некорректна
	1	Linked	Рецептурный элемент (RE) может иметь несколько ссылочных шагов
	2	Embedded	Рецептурный элемент имеет только один ссылочный RE. Для каждого использования RE определен только один RE
	3	Copied	То же, что Embedded, только рассматриваемый RE отличается от своего оригинального определения
	4-99		Зарезервированы
	100+		Определены пользователем
RecipeStatus	0	Invalid	Запись некорректна
	1	Approved for Production	Рецептура утверждена для производства
	2	Approved for Test	Рецептура утверждена только для испытаний
	3	Not Approved	Рецептура не утверждена для производства или испытаний
	4	Inactive	Рецептура не активна
	5	Obsolete	Рецептура устарела
	6-99		Зарезервированы
	100+		Определены пользователем
RecordSet	0	Invalid	Запись некорректна
	1	RecordSetControlRecipe	Определяет, что информационная запись истории производства партии изделий является частью категории ControlRecipe

Продолжение таблицы 34

EnumSet	EnumValue	EnumString	Описание
RecordSet	2	RecordSetMasterRecipe	Множество записей технологической рецептуры
	3	RecordSetExecutionInfo	Информация о выполнении множества записей
	4	RecordSetMaterialInfo	Информация множества записей о материале
	5	RecordSetContinuousData	Множество записей непрерывных данных
	6	RecordSetEvents	Множество записей о событиях
	7	RecordSetOperatorChange	Множество записей об изменениях, внесенных оператором
	8	RecordSetOperatorComment	Множество комментариев оператора
	9	RecordSetAnalysisData	Множество записей о результатах анализа
	10	RecordSetLateRecord	Множество записей, внесенных задним числом
	11	RecorSetRecipeData	Множество записей о рецептурных данных
	12	RecoredSetRecipeSpecified	Множество записей о рецептурных указаниях
	13	RecorSetSummaryData	Множество записей об итоговых данных
	14-99		Зарезервированы
	100+		Определены пользователем
RecordSet ControlRecipe	0	Invalid	Запись некорректна
	1	Entire Control Recipe	Историческая запись относится ко всей рецептуре управления
	2-99		Зарезервированы
	100+		Определены пользователем
RecordSet MasterRecipe	0	Invalid	Запись некорректна
	1	Entire Master Recipe	Историческая запись относится ко всей технологической рецептуре
	2-99		Зарезервированы
	100+		Определены пользователем
RecordSet ExecutionInfo	0	Invalid	Запись некорректна
	1	Allocation	Выделение ресурса
	2	De-allocation	Освобождение ресурса
	3	State Change	Изменение состояния
	4	State Command	Команда состояния
	5	Mode Change	Изменение режима
	6	Mode Command	Команда режима
	7	Procedural Entity Message	Сообщение процедурной сущности
	8	Procedural Entity Alarm	Аварийный сигнал процедурной сущности
	9	Procedural Entity Version	Версия процедурной сущности
	10	Procedural Entity Prompt	Подсказка процедурной сущности
	11	Procedural Entity Prompt Response	Отклик на подсказку процедурной сущности

Продолжение таблицы 34

EnumSet	EnumValue	EnumString	Описание
RecordSet	12-99		Зарезервированы
ExecutionInfo	100 +		Определены пользователем
RecordSet MaterialInfo	0	Invalid	Запись некорректна
	1	Material Consumption	Расход материала
	2	Material Production	Производство материала
	3	Material Allocation	Выделение материала
	4	Material De-allocation	Отмена выделения материала
	5-99		Зарезервированы
	100+		Определены пользователем
RecordSet ContinuousData	0	Invalid	Запись некорректна
	1	Continuous Data Value	Непрерывное значение данных
	2	Trend Association	Ассоциация тренда
	3	Trend Disassociation	Распад тренда
	4-99		Зарезервированы
	100+		Определены пользователем
RecordSet Events	0	Invalid	Запись некорректна
	1	General Event	Общее событие
	2-99		Зарезервированы
	100 +		Определены пользователем
RecordSet OperatorChange	0	Invalid	Запись некорректна
	1	General Operator Intervention	Общее вмешательство оператора
	2-99		Зарезервированы
	100 +		Определены пользователем
RecordSet OperatorComment	0	Invalid	Запись некорректна
	1	General Operator Comment	Общий комментарий оператора
	2-99		Зарезервированы
	100 +		Определены пользователем
RecordSetAnalysisData	0	Invalid	Запись некорректна
	1	General Analysis Message	Общее сообщение об анализе
	2-99		Зарезервированы
	100+		Определены пользователем
RecordSet LateRecord	0	Invalid	Запись некорректна
	1	General Late Record	Общая запись задним числом
	2-99		Зарезервированы
	100+		Определены пользователем
RecordSetRecipe Data	0	Invalid	Запись некорректна
	1	Generic Recipe Data	Характерные рецептурные данные
	2	Recipe Parameter Value Change	Изменение значения рецептурного параметра
	3	Recipe Result Data	Данные результата рецептуры
	4-99		Зарезервированы
	100+		Определены пользователем

Продолжение таблицы 34

EnumSet	EnumValue	EnumString	Описание
RecordSetrRecipe Specified	0	Invalid	Запись некорректна
	1	Generic Recipe Specified Data	Характерные указанные рецептурные данные
	2-99		Зарезервированы
	100+		Определены пользователем
SummaryData	0	Invalid	Запись некорректна
	1	Generic Summary Data	Характерные итоговые данные
	2	Utilities Consumption	Потребление коммунальных услуг
	3	Equipment Run Time	Время работы оборудования
	4-99		Зарезервированы
	100+		Определены пользователем
ScheduleAction	0	Invalid	Запись некорректна
	1	New	Действие по изменению календарной записи
	2	Update	Обновить
	3	Delete	Стереть
	4-99		Зарезервированы
	100+		Определены пользователем
ScheduleMode	0	Invalid	Запись некорректна
	1	Automatic	Автоматический режим календарной записи
	2	Semi-automatic	Полуавтоматический
	3	Manual	Ручной
	4	Non Specified	Не определен
	5-99		Зарезервированы
	100+		Определены пользователем
ScheduleStatus	0	Invalid	Запись некорректна
	1	Complete	Статус записи о календарном плане производства партии изделий
	2	In-progress	
	3	Scheduled	
ScheduleStatus	4	Schedule Hold	
	5	Not Specified	
	6-99		Зарезервированы
	100+		Определены пользователем
SE_Type	0	Invalid	Запись некорректна
	1	Campaign	Определяет тип календарной записи
	2	Batch	
	3	Unit Procedure	
	4	Operation	
	5	Phase	
	6-99		Зарезервированы
	100+		Определены пользователем

Окончание таблицы 34

EnumSet	EnumValue	EnumString	Описание
ValueDataType	0	Invalid	Запись некорректна
	1	Boolean	Определяет тип данных, рекомендуемых для ассоциированных значений
	2	8-Bit string	
	3	16-Bit string	
	4	32-Bit string	
	5	8-Bit unsigned integer	
	6	16-Bit unsigned integer	
	7	32-Bit unsigned integer	
	8	8-Bit signed integer	
	9	16-Bit signed integer	
	10	32-Bit signed integer	
	11	32-Bit float	
	12	Double float	
	13	Octet string	
	14	DATETIME	
	15-99		Зарезервированы
	100+		Определены пользователем
ValueType	0	Invalid	Запись некорректна
	1	Constant	Определяет порядок интерпретации значения строки. Содержит фиксированное значение как строку
	2	Reference	Определяет порядок интерпретации значения строки. Указывает источник значения
	3	Equation	Определяет, что значение строки — это выражение, которое оценивается для определения данного значения
	4	External	Значение обеспечивается некоторыми внешними средствами. Оно не содержится в рецептуре (например, значение может обеспечиваться оператором или системным разработчиком календарного плана)
	5-99		Зарезервированы
	100+		Определены пользователем

5.2 Информация о технологической рецептуре

В данном подразделе рассмотрены только *технологические рецептуры*. Обмениваемая информация — это информация технологической рецептуры в соответствии с МЭК 61512-1. Данная информация содержит определения процедурного управления, определения значений формулы, рецептурные требования к оборудованию, информацию заголовка, специальную информацию о технологическом цехе, прочую информацию и требования к управлению координацией.

Обмениваемая информация — это обмениваемая информация технологической рецептуры. Она не включает:

- порядок создания информации;
- порядок использования технологической рецептуры в системе.

5.2.1 Определения рецептуры

При создании технологической рецептуры, в соответствии с МЭК 61512-1, может использоваться информация из рецептуры, связанной с местом производства, а также определение производственных возможностей технологического цеха. Технологическая рецептура связывается (линкуется) с производ-

ственными возможностями технологического цеха. Определение производственных возможностей не является частью обмениваемой рецептурной информации.

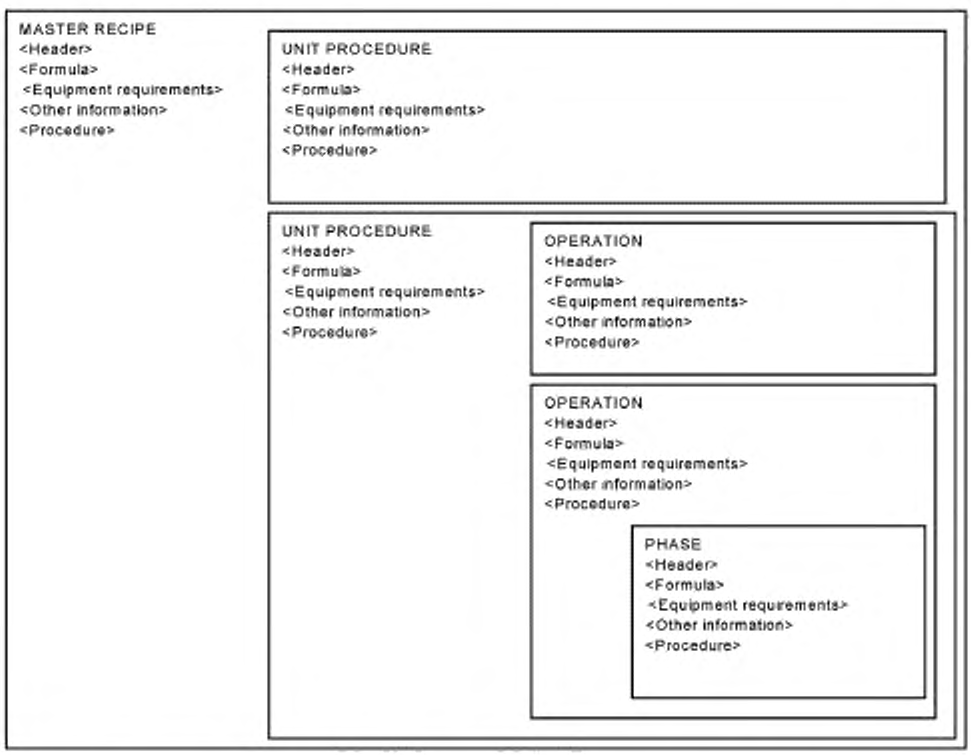
5.2.2 Структура рецептуры

Рецептура включает следующие категории информации: заголовок, формулы, процедуры, требования к оборудованию, прочую информацию.

Схема обмена рецептур имеет рекурсивную природу. Базовая структура данной схемы (см. рисунок 15) построена на пяти указанных категориях информации. Каждый уровень определения содержит все указанные категории информации до тех пор, пока процедурные определения ссылаются на процедурную сущность оборудования.

Термин «рецептурный элемент (RE)» используется для определения некоторых структурных сущностей рецептуры. Рецептурный элемент сам может быть технологической рецептурой, рецептурной процедурой, рецептурной процедурой технологической установки, рецептурной операцией, рецептурной фазой, обозначением выделения ресурса или прочими графическими обозначениями в соответствии с таблицей 34.

В рекурсивной модели схемы, рецептурные элементы содержат как рецептурные элементы нижнего уровня, так и ссылки на процедурные элементы оборудования. Если определение рецептурного элемента содержится в таблице рецептурных элементов, то каждое использование рецептурного элемента называется «шагом».

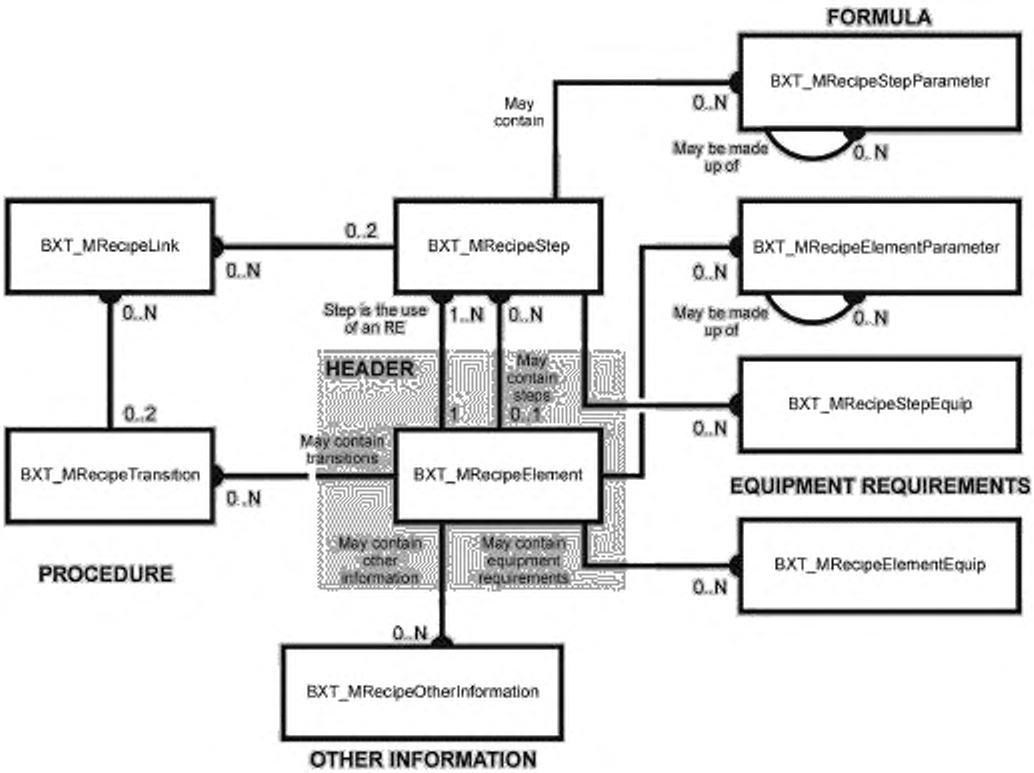


Master recipe — технологическая рецептура; header — <Заголовок>; formula — <Формула>; equipment requirements — <Требования к оборудованию>; other information — <Прочая информация>; procedure — <Процедура>; unit procedure — процедура технологической установки; header — <Заголовок>; formula — <Формула>; equipment requirements — <Требования к оборудованию>; other information — <Прочая информация>; procedure — <Процедура>; operation — операция; header — <Заголовок>, formula — <Формула>; equipment requirements — <Требования к оборудованию>; other information — <Прочая информация>; procedure — <Процедура>; phase — фаза; header — <Заголовок>; formula — <Формула>; equipment requirements — <Требования к оборудованию>; other information — <Прочая информация>; procedure — <Процедура>

Рисунок 15 — Встроенные рецептурные элементы образуют рецептуру

5.2.3 Обзор таблиц и ограничения целостности

На рисунке 16 приведены таблицы, используемые для обмена рецептур и их отношений. Рисунок 16 определяет ассоциированные ограничения целостности для указанных таблиц.



Formula — формула; may contain - может содержать ..., BXT_MRecipeStepParameter — таблица параметров шага рецептуры; BXT_MRecipeLink — таблица рецептурных соединительных звеньев; BXT_MRecipeStep — таблица рецептурных шагов; may be made up of — может состоять из ...; BXT_MRecipeElementParameter — таблица параметров рецептурных элементов, step is the use of an re — шаг – это использование рецептурного элемента; header — заголовок; may contain steps — может содержать шаги ...; BXT_MRecipeStepEquip — таблица рецептурных шагов, связанных с оборудованием; BXT_MRecipeTransition — таблица рецептурных переходов; may contain transitions — может содержать переходы ...; BXT_MRecipeElement — таблица рецептурных элементов; equipment requirements — требования к оборудованию; procedure — процедура; may contain other information — может содержать прочую информацию, may contain equipment requirements — может содержать требования к оборудованию; BXT_MRecipeElementEquip — таблица рецептурных элементов, связанных с оборудованием; BXT_MRecipeOtherInformation — таблица прочей рецептурной информации; other information — прочая информация

Рисунок 16 — Связи между таблицами обмена

Непрямые отношения, такие как отношения между сущностями LINK и RE, здесь не рассматриваются. Вместе с тем, они определяются посредством множества общих ключевых полей таблиц. Записи «NOT NULL» в ассоциированных таблицах языка SQL используются только для того, чтобы усилить ограничения целостности для диаграмм отношения сущностей.

Рекурсивное определение рецептурного элемента (RE) обеспечивается путем использования двух ассоциаций между сущностями BXT_MRecipeStep и BXT_MRecipeElement. Каждое заявленное использование рецептурного элемента регистрируется в таблице BXT_MRecipeStep. Каждое определение рецептурного элемента регистрируется в таблице BXT_MRecipeStep. Одна ассоциация указывает, какие шаги содержит рассматриваемый рецептурный элемент. Другая ассоциация указывает, на какой элемент рассматриваемый шаг ссылается. Каждый рецептурный элемент (RE), на который ссылается рассматриваемый RE, имеет запись в таблице шагов. В свою очередь, таблица шагов ссылается на соответствующее определение RE. Данные таблицы поддерживают отдельные определения RE на каждом шаге. Они также поддерживают несколько шагов, ссылающихся на один RE.

Отношение между шагами и переходами определяется таблицей BXT_MRecipeLink.

Определение формулы технологической рецептуры – это совокупность записей всех параметров технологической рецептуры. Оно содержит описание входного сигнала технологического процесса, выходного сигнала технологического процесса, а также параметров данного процесса в рамках рецептуры. Значения формулы содержатся в таблице BXT_MRecipeStepParameter. Их определения приведены в таблице BXT_MRecipeElementParameter.

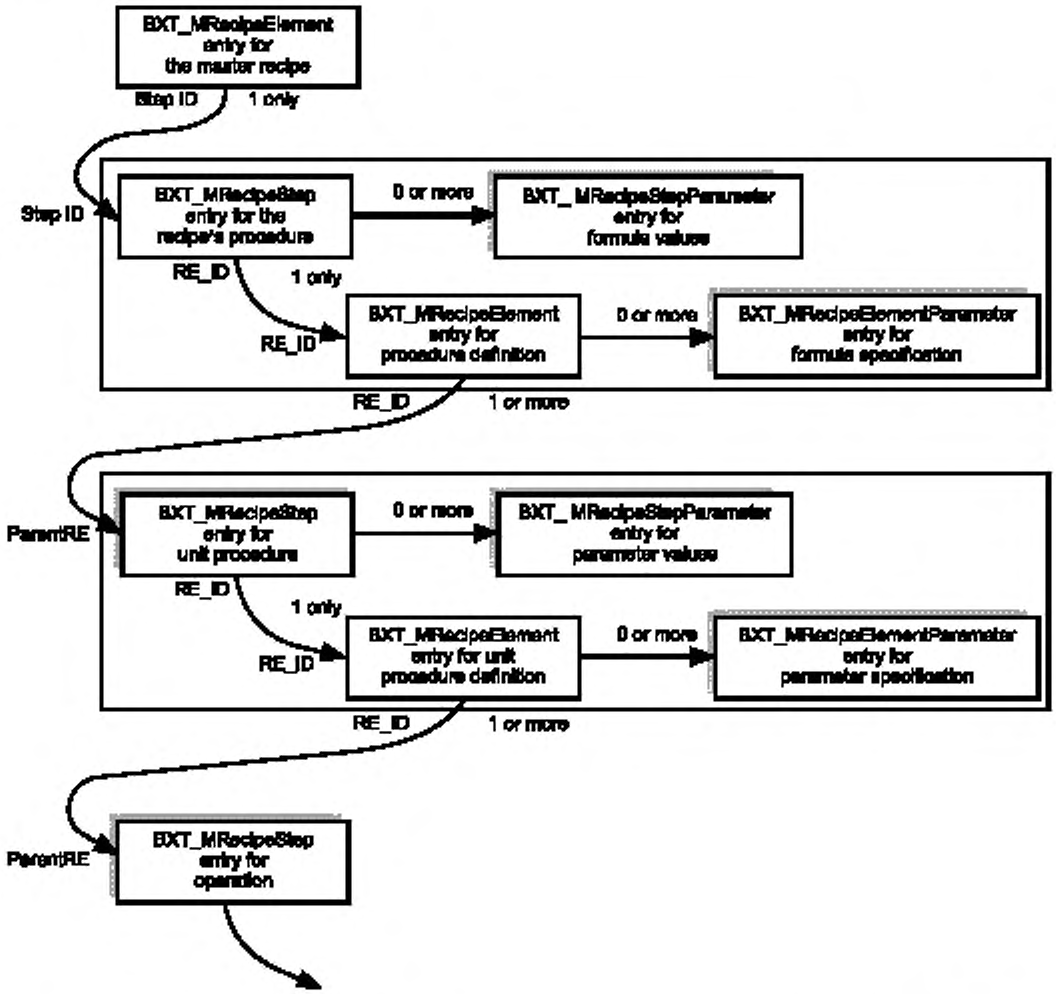
Рецептурные требования к оборудованию содержатся в таблицах BXT_MRecipeStepEquip и BXT_MRecipeElementEquip.

На рисунке 17 приведен порядок внесения записей в каждую таблицу, а также их отношения для таблиц BXT_MRecipeStep, BXT_MRecipeElement, BXT_MRecipeElementParameter и BXT_MRecipeStepParameter. Любая запись таблицы BXT_MRecipeElement существует для каждой версии обмениваемой рецептуры. Имеет место определенное отношение между записью таблицы BXT_MRecipeElement и записью таблицы BXT_MRecipeStep. Данное отношение содержит конкретную рецептурную информацию, включая значения формул, сведенные в таблицу BXT_MRecipeStepParameter.

Таблица BXT_MRecipeStep содержит ключ к отдельным записям таблицы BXT_MRecipeElement. Записи таблицы BXT_MRecipeElement содержат определения рецептурных процедур, включая определения значений формул в таблице BXT_MRecipeElementParameter.

Таблица BXT_MRecipeElement для данной процедуры содержит ключи к нескольким записям таблицы BXT_MRecipeStep. Для каждой процедуры технологической установки используется свой ключ (для простоты, прочие рецептурные элементы процедурного уровня в данном примере не рассматриваются). Записи таблицы BXT_MRecipeStep для процедур технологических установок содержат ключи к записям таблицы BXT_MRecipeElement, определяющим рассматриваемую процедуру технологической установки. Для каждой процедуры технологической установки запись таблицы BXT_MRecipeElement содержит ключ к каждой операции таблицы BXT_MRecipeStep. Рассматриваемая структура далее переходит к определению фаз.

Рассматриваемый табличный формат использует таблицы BXT_MRecipeStep и BXT_MRecipeElement в рамках рассматриваемой процедурной иерархии рецептурной процедуры.



BXT_MRecipeElement entry for the master recipe — запись для технологической рецептуры таблицы BXT_MRecipeElement; StepID — идентификатор шага; 1 only — только один элемент; BXT_MRecipeStep entry for the recipe's procedure — запись для рецептурной процедуры таблицы BXT_MRecipeStep, RE_ID — идентификатор рецептурного элемента; 0 or more — 0 или несколько элементов; BXT_MRecipeStepParameter entry for formula values — запись для значений формулы таблицы BXT_MRecipeStepParameter; BXT_MRecipeElement entry for procedure definition — запись для определения процедуры таблицы BXT_MRecipeElement; BXT_MRecipeStepParameter entry for formula specification — запись для спецификации формулы таблицы BXT_MRecipeStepParameter; ParentRE — родительский рецептурный элемент, BXT_MrecipeStep entry for unit procedure — запись для процедуры технологической установки таблицы BXT_MRecipeStep; BXT_RecipeStepParameter entry for parameter values — запись для значений параметра таблицы BXT_RecipeStepParameter; BXT_MRecipeElement entry for unit procedure definition — запись для определения процедуры технологической установки таблицы BXT_MRecipeElement; BXT_MRecipeElementParameter entry for parameter specification — запись для спецификации параметра таблицы BXT_MRecipeElementParameter; BXT_MRecipeStep entry for operation — запись для операции таблицы BXT_MRecipeStep

Рисунок 17 — Отношения записей в таблицах

5.2.4 Сводный анализ рецептурных таблиц

Таблицы, определенные для рецептурного обмена, указаны в таблице 35.

Таблица 35 — Таблицы рецептурного обмена

Имя таблицы	Описание
BXT_MRecipeStep	Одна запись для каждого использования RE в RE
BXT_MRecipeElement	Одна запись для каждого обмениваемого рецептурного элемента
BXT_MRecipeTransition	Одна запись для каждого перехода, используемого в RE
BXT_MRecipeLink	Одна запись для каждого соединительного звена между шагами и переходами
BXT_MRecipeElementParameter	Одна запись для каждого параметра каждого RE
BXT_MRecipeStepParameter	Одна запись для каждого параметра каждого шага
BXT_MRecipeOtherInformation	Одна запись для каждого элемента прочей информации
BXT_MRecipeElement Equip	Одна запись для каждого требования к свойству RE
BXT_MRecipeStepEquip	Одна запись для каждого значения свойства оборудования, определенного для шага

5.2.5 Определения рецептурных таблиц

5.2.5.1 Информация заголовка

Информация заголовка технологической рецептуры передается в виде поля записи в таблице BXT_MRecipeElement. Таблица BXT_MRecipeElement (см. таблицу 36) содержит один элемент для каждой обмениваемой технологической рецептуры. Комбинация идентификатора рецептурного элемента RE_ID и версии рецептурного элемента REVersion определяет обмениваемую технологическую рецептуру.

Таблица 36 — Таблица BXT_MRecipeElement

Атрибут	Описание
RE_ID	Идентифицирует обмениваемый рецептурный элемент (например, Red Oak (красный дуб)). При объединении идентификатора с «версией», данное поле определяет уникальную реализацию (экземпляр) RE. Если запись представляет технологическую рецептуру, то данное поле содержит идентификатор данной технологической рецептуры
REVersion	Идентифицирует версию RE. При объединении данной версии с «RE_ID», данное поле определяет уникальную реализацию RE (например, V10.3)
VersionDate	Идентифицирует дату и время последней модификации данной версии RE
ApprovalDate	Идентифицирует дату и время утверждения данной версии рецептуры
EffectiveDate	Идентифицирует дату и время действия данной версии рецептуры
ExpirationDate	Идентифицирует дату и время, когда данная версия рецептуры утрачивает силу
Author	Идентифицирует физическое лицо или систему, являющуюся автором данной версии (например, J. Smith)
ApprovedBy	Идентифицирует физическое лицо или систему, утвердившую данную версию рассматриваемой рецептуры
ProcessCellID	Идентифицирует технологический цех (класс технологических цехов), для которых данная версия технологической рецептуры определена
ProductID	Идентифицирует продукт (семейство продуктов), создаваемых путем выполнения данной версии рецептуры (например, Premium Beer)
UsageConstraint	Определяет прочие правила, определяющие использование элемента (например, «должен предшествовать ...», «не должен выполняться параллельно ....» и т. д.)
Description	Содержит описание рецептурного элемента
Status	Определяет статус информации, обмениваемой как элемент перечисления из множества элементов перечисления «статус рецептуры» (recipe status)
RE_Type	Идентифицирует тип RE из множества элементов перечисления «RE_Type»

Окончание таблицы 36

Атрибут	Описание
RE_Function	Содержит необязательную ссылку на таблицы обмена информацией об оборудовании. Формат данной информации в данном Разделе не определен. Примером является ссылка на процедурный элемент оборудования (см. таблицу BXT_EquipInterface). Если значение записи равно NULL, то функция данного RE определена записями таблиц BXT_MRecipeStep и BXT_MRecipeTransition, имеющими родительский элемент RE, соответствующий идентификатору RE_ID
RE_Use	Идентифицирует отношение между элементом RE и таблицей BXT_MRecipeStep, соответствующее элементу перечисления «RE_Use». Утверждение «Linked Specifies» означает, что в таблице BXT_MRecipeStep определение RE может быть использовано несколько раз. Утверждение Linked используется, если таблица BXT_MRecipeElement является библиотечным структурным элементом. Утверждение «Embedded» указывает, что рассматриваемый RE имеет только один ссылочный RE, и для каждого использования RE определен только один RE. В данном случае, RE «is embedded» (встроен) в определение рецептуры отдельного рецептурного шага, и больше он нигде не используется. Утверждение «Copied» аналогично утверждению «Embedded», но при этом рассматриваемый RE модифицирован по отношению к некоторому оригинальному определению. Утверждение «Copied» используется, если элемент RE является библиотечным структурным элементом, полностью репродуцированным в рамках рассматриваемой рецептуры, и его связь с библиотекой удалена
DerivedRE	Идентифицирует рецептурный элемент, из которого выведен рассматриваемый рецептурный элемент
DerivedVersion	Идентифицирует версию рецептурного элемента, из которой выведен рассматриваемый рецептурный элемент

5.2.5.2 Процедурная информация

Процедурные части технологической рецептуры включаются (содержатся) в комбинацию (совокупность) таблиц. Данные записи определяют:

- шаги элемента процедурного управления;
- элементы процедурного управления;
- обозначения выделения и прочие обозначения графических представлений;
- параметризацию процедурных элементов с указанием пределов;
- связи между элементами;
- переходы (определения переходов) между элементами.

Таблицы шагов, таблицы переходов и таблицы соединительных звеньев содержат определения RE, включающие, в свою очередь, RE нижнего уровня. Таблица BXT_MRecipeStep (см. таблицу 37) содержит все реализации (экземпляры) используемого RE. Каждая таблица BXT_MRecipeStep содержит используемые параметры для используемого RE. Если RE используется рецептурой только один раз, то имеется только одна соответствующая запись в таблице BXT_MRecipeElement. Если в таблице BXT_MRecipeStep имеется несколько записей, то RE используется несколько раз. Каждому использованию RE соответствует своя запись.

Таблица 37 — Таблица BXT_MRecipeStep

Атрибут	Описание
ParentRE	Идентифицирует родительский RE или технологическую рецептуру, с которой ассоциирован рассматриваемый шаг
ParentVersion	Идентифицирует родительскую версию RE или технологическую рецептуру, с которой ассоциирован рассматриваемый шаг. При объединении с родительским элементом «ParentRE», данное поле определяет уникальную реализацию RE
StepID	Идентифицирует уникальную реализацию выполнения ссылочного RE, его имя является уникальным в области применения родительского RE (простым примером является № шага в RE)
RE_ID	Идентифицирует имя RE, для которого данный шаг является реализацией. Его имя является уникальным в области применения родительского RE
REVersion	Идентифицирует версию RE, для которой данный шаг является реализацией

Окончание таблицы 37

Атрибут	Описание
VerticalStart	Задает вертикальную позицию начала работ для презентации рассматриваемого элемента в процедурном виде родительского RE. Нормализованные координаты (0,0) – это верхний левый край, координаты (1,1) – это нижний правый край
VerticalStop	Задает вертикальную позицию окончания работ для презентации рассматриваемого элемента в процедурном виде родительского RE. Нормализованные координаты (0,0) – это верхний левый край, координаты (1,1) – это нижний правый край
HorizontalStart	Задает горизонтальную позицию начала работ для презентации рассматриваемого элемента в процедурном виде родительского RE. Нормализованные координаты (0,0) – это верхний левый край, координаты (1,1) – это нижний правый край
HorizontalStop	Задает горизонтальную позицию окончания работ для презентации рассматриваемого элемента в процедурном виде родительского RE. Нормализованные координаты (0,0) – это верхний левый край, координаты (1,1) – это нижний правый край
ScaleReference	Задает ссылочную величину рецептурных элементов; все значения формул основаны на данной ссылочной величине (например, массе 1234,5 кг)
ScaleEngrUnits	Задает технические единицы измерения атрибута ScaleReference
MaximumScale	Задает максимальный масштабный фактор (величину) рецептурного элемента
MinimumScale	Задает минимальный масштабный фактор (величину) рецептурного элемента

5.2.5.2.1 Рецептурный элемент

Таблица BXT_MRecipeElement (см. таблицу 36) содержит одну запись для каждого процедурного элемента, на который производится ссылка из обмениваемой технологической рецептуры. Данная таблица содержит определение самого элемента, но не порядка его использования. В данной таблице одна запись соответствует процедуре, одна – каждой процедуре технологической установки, одна – каждой операции и одна – каждой обмениваемой рецептурной фазы. Таблицы BXT_MRecipeElement и BXT_MRecipeElementParameter содержат спецификации порядка использования RE, номер и типы рассматриваемых параметров, значения параметров по умолчанию.

Элементы RE должны быть уникальными по отношению к родительскому RE. Идентификатор RE_ID – это полное имя RE, соответствующее своему родительскому RE. Таким образом, использование идентификатора RE_ID достаточно для описания всех идентификаторов родительских RE_ID.

5.2.5.2.2 Переходы

Таблица BXT_MRecipeTransition содержит одну запись для каждого перехода, определенного соответствующим RE (см. таблицу 38). Рассматриваемые записи соответствуют переходам, описанным в процедурных функциональных диаграммах (см. раздел 6).

Таблица 38 — Таблица BXT_MRecipeTransition

Атрибут	Описание
RE_ID	Идентифицирует RE, содержащий рассматриваемый переход
REVersion	Идентифицирует версию RE. При объединении с идентификатором «RE_ID», определяет уникальную реализацию технологической рецептуры
TransitionID	Идентифицирует уникальную реализацию выполнения рассматриваемого элемента перехода. Идентификатор ID содержит полную иерархию имен реализаций родительского RE, для которого данный переход является составляющим элементом
Condition	Содержит выражение (условие). Если его значение равно TRUE, то переход разрешен
VerticalStart	Вертикальная позиция начала работ для презентации данного элемента в процедурном виде RE. Нормализованные координаты (0,0) – это верхний левый край, координаты (1,1) – это нижний правый край
VerticalStop	Вертикальная позиция окончания работ для презентации рассматриваемого элемента в процедурном виде RE. Нормализованные координаты (0,0) – это верхний левый край. Координаты (1,1) – это нижний правый край

Окончание таблицы 38

Атрибут	Описание
HorizontalStart	Горизонтальная позиция начала работ для презентации рассматриваемого элемента в процедурном виде RE. Нормализованные координаты (0,0) – это верхний левый край. Координаты (1,1) – это нижний правый край
HorizontalStop	Горизонтальная позиция окончания работ для презентации данного элемента в процедурном виде RE. Нормализованные координаты (0,0) – это верхний левый край. Координаты (1,1) – это нижний правый край.

5.2.5.2.3 Соединительные звенья (связи)

Таблица BXT_MRecipeLink содержит одну запись для каждого соединения, определенного RE и/или соответствующим переходом (см. таблицу 39). Данные записи соответствуют строкам, соединяющим элементы, описанные в процедурных функциональных диаграммах (см. раздел 6).

Таблица 39 — Таблица BXT_MRecipeLink

Атрибут	Описание
RE_ID	Идентифицирует RE, с которым ассоциируется рассматриваемый шаг (переход)
REVersion	Идентифицирует версию RE. При объединении с идентификатором «RE_ID», данное поле определяет уникальную реализацию RE
LinkID	Задает уникальный идентификатор ID соединительного звена, упрощающий доступ к таблице
FromType	Используется как перечисление. Определяет, задает ли атрибут FromElement идентификатор шага StepID или идентификатор перехода TransitionID. Формируется из множества элементов перечисления «LinkToType»
FromElement	Задает имя шага или идентификатор перехода TransitionID. Идентификатор ID содержит полную иерархию имен реализаций родительского элемента RE, в которых содержится данный элемент. Данное обстоятельство соответствует имени шага (перехода) в составе таблицы шагов (переходов)
ToType	Используется как перечисление. Определяет, задает ли атрибут ToElement идентификатор шага StepID или идентификатор перехода TransitionID. Формируется из множества элементов перечисления «LinkToType»
ToElement	Задает имя шага или идентификатор перехода TransitionID. Идентификатор ID содержит полную иерархию имен реализаций родительского элемента RE, содержащих данный элемент. Данное обстоятельство соответствует имени шага (перехода) в таблице шагов (переходов)
LinkType	Определяет, является ли рассматриваемое соединительное звено процедурным потоком управления или ассоциацией передачи материала. Используется как элемент перечисления ControlLink или TransferLink. Формируется из множества элементов перечисления «LinkType»
VerticalStart	Задает вертикальную позицию начала работ для презентации рассматриваемого элемента в процедурном виде родительского RE. Нормализованные координаты (0,0) – это верхний левый край. Координаты (1,1) – это нижний правый край
VerticalStop	Задает вертикальную позицию окончания работ для презентации рассматриваемого элемента в процедурном виде родительского RE. Нормализованные координаты (0,0) – это верхний левый край. Координаты (1,1) – это нижний правый край
HorizontalStart	Задает горизонтальную позицию начала работ для презентации рассматриваемого элемента в процедурном виде родительского RE. Нормализованные координаты (0,0) – это верхний левый край. Координаты (1,1) – это нижний правый край
HorizontalStop	Задает горизонтальную позицию окончания работ для презентации рассматриваемого элемента в процедурном виде родительского RE. Нормализованные координаты (0,0) – это верхний левый край. Координаты (1,1) – это нижний правый край
Depiction	Определяет порядок представления соединительного звена. Формируется из множества элементов перечисления «LinkDepiction»
Evaluation Order	Определяет установленный порядок оценки соединительного звена (при необходимости), обеспечивающий проверку перехода PFC в направлении «слева-направо» (см. раздел 6). Предполагается, что все связи рассматриваемого шага с несколькими переходами оцениваются в порядке, соответствующем порядку информационного поля. Элементы с меньшими номерами оцениваются первыми

5.2.5.2.4 Параметры

Таблица BXT_MRecipeElementParameter содержит одну запись для каждого параметра каждого используемого RE (см. таблицу 40). Например, элемент RE (с именем *CHARGE*) может быть определен двумя параметрами: 1) тип заряжаемого материала, 2) количество заряжаемого материала. Для элемента RE *CHARGE*, одна запись вносится в таблицу BXT_MRecipeElement, две записи вносятся в таблицу BXT_MRecipeElementParameter.

Таблица 40 — Таблица BXT_MRecipeElementParameter

Атрибут	Описание
RE_ID	Идентифицирует RE, с которым ассоциирован рассматриваемый параметр
REVersion	Идентифицирует версию RE. Если данное информационное поле объединить с «RE_ID», то полученное поле будет определять уникальную реализацию технологической рецептуры
ParameterID	Идентифицирует параметр процедурного элемента. Отметим, что если данный параметр является частью множества, то используется информационное поле ParentParamID, а рассматриваемое множество параметров становится частью имени параметра. Например, для множества параметров MINOR_CHARGES и параметра BLUE_DYE, идентификатор ParameterID — это MINOR_CHARGES.BLUE_DYE, где признаком разграничения является период
ParentParamID	Идентифицирует родительское множество параметров, членом которого является рассматриваемый параметр. Данное информационное поле содержит NULL, если множество параметров отсутствует
DataInterpretation	Определяет порядок интерпретации значения параметра по умолчанию: как элемент перечисления (например, константа, ссылка, уравнение), как элемент множества элементов перечисления «ValueType»
DataDirection	Определяет условия работы со значением параметра: как с элементом перечисления (например, input, output, input/output, никак), как с элементом множества элементов перечисления «DirectionType»
DefaultValue	Содержит значение параметра по умолчанию, используемое в том случае, если реализация (экземпляр) выполнения не дает описания значения, которое может быть членом множества пользовательских перечислений
ValueType	Определяет тип данных для значения, принадлежащего множеству элементов перечисления «ValueDataType»
Description	Содержит описание параметра или порядка использования параметра в RE
EngrUnits	Идентифицирует технические единицы измерения значений (например, кг, фунт)
EnumSet	Идентифицирует нумерацию, членом которой является рассматриваемый элемент (не равный NULL)
DefaultScaling	Устанавливает порядок выбора по умолчанию элемента, определенного как реализация в таблице BXT_MRecipeStepParameter. Используется как элемент перечисления из множества элементов перечисления «Boolean». Если имеет значение TRUE, то значение формулы масштабируется, когда масштабируется размер производственной партии изделий. Если имеет значение FALSE, то значение формулы не масштабируется, когда масштабируется размер производственной партии изделий. Нелинейное масштабирование может быть выполнено путем подстановки выражений в значения формулы
ParamType	Указывает порядок выбора по умолчанию, если элемент определен как реализация в таблице BXT_MRecipeStepParameter. Данное информационное поле идентифицирует использование значения формулы как элемента перечисления из множества элементов перечисления «FormulaType» (например, входной сигнал технологического процесса, выходной сигнал технологического процесса, параметр процесса). Множество элементов перечисления может расширяться пользователем
ParamSubType	Указывает порядок выбора по умолчанию, если элемент определен как реализация в таблице BXT_MRecipeStepParameter. Данное информационное поле идентифицирует использование значения формулы как элемента перечисления из множества элементов перечисления FormulaSubType. Все элементы данного множества определяются пользователем

5.2.5.2.5 Стандартные подпараметры

Настоящий стандарт определяет множество подпараметров, которые могут быть поставлены в соответствие значению некоторого параметра, чтобы дополнить его определение и порядок использования. Например, может оказаться полезным определение и передача верхнего и нижнего пределов значений рассматриваемого параметра. Данный тип информации передается с помощью таблиц обмена путем определения подпараметров для рассматриваемого параметра. Подпараметры определяются для данного параметра путем создания новых записей таблицы с задействованным идентификатором ParameterID, используемым как идентификатор ParentParamID. Множество стандартных подпараметров и порядок их использования определены в таблице 41. Прочие подпараметры могут быть определены пользователем. Использование данного множества подпараметров поддерживается таблицами BXT_MRecipeElementParameter, BXT_MRecipeStepParameter (см. 5.2.5.3), BXT_EquipInterfaceParameter (см. 5.3.5.7) и BXT_ScheduleParameter (см. 5.4.3.4).

Таблица 41 — Стандартные подпараметры

ParameterID	Описание
HighValueLimit	Задает наибольшее значение, которое может принимать ассоциированный параметр
LowValueLimit	Задает наименьшее значение, которое может принимать ассоциированный параметр
HighTolerance	Задает наибольшее допустимое отклонение вверх от значения ассоциированного параметра
LowTolerance	Задает наибольшее допустимое отклонение вниз от значения ассоциированного параметра

5.2.5.3 Формулы

Значения компонентов, используемых в производстве (модификации) партии изделий, рассматриваются как параметры шага и их пределы. Таблица BXT_MRecipeStepParameter содержит одну запись для каждого параметра, используемого на данном шаге (см. таблицу 42). Так как элементы RE могут оценивать значения параметров по умолчанию, то не все параметры элементов RE нужно определять на данном шаге.

Таблица 42 — Таблица BXT_MRecipeStepParameter

Атрибут	Описание
ParentRE	Часть ключа к таблице BXT_MRecipeStep для данного параметра
ParentVersion	Часть ключа к таблице BXT_MRecipeStep для данного параметра
StepID	Идентифицирует уникальную реализацию выполнения RE. Его имя является уникальным в области применения его родительского RE
ParameterID	Идентифицирует параметр RE
ParentParameterID	Идентифицирует идентификатор родительского параметра ID, если таковой имеется. (Допускает использование для множества параметров)
ParameterValue	Содержит значение параметра, его №№ передаются как представления ASCII для номера. Если тип параметра — это множество пользовательских перечислений, то данный параметр может быть членом данного множества
DataInterpretation	Определяет порядок интерпретации значения параметра: как элемента перечисления (константа, ссылка, внешний элемент, уравнение), как элемента множества «ValueType»
Scaled	Используется как перечисление из множества элементов перечисления «Boolean». Если имеет значение TRUE, то значение параметра масштабируется при масштабировании размера производства партии изделий. Если имеет значение FALSE, то значение формулы не масштабируется при масштабировании размера производства партии изделий

5.2.5.4 Прочая информация

Обычно прочая информация передается вместе с рецептурой (см. таблицу 43). Смысл данной информации не указывается в определении обмена, но обязательно согласовывается отправителем и получателем. Прочая информация обычно представляет собой экстраординарную информацию (описательную информацию), необходимую для обмена корректной технологической рецептуры. Но это не та информация, которая нужна для выполнения указанной рецептуры. Примеры прочей информации

могут включать документацию соответствия, диаграммы молекулярной структуры, чертежи ожидаемого продукта. Указанная информация содержится в таблицах BXT_MRecipeOtherInformation прочей информации об элементе RE.

Так как структура и форма представления данных часто меняются, язык обмена рецептур содержит необходимые средства для ссылок (обозначений) данной информации. Реальная информация обменивается в информационном поле DataValue, если она является текстовой. Она может ссылаться на другие файлы, если их имя прописано в информационном поле DataValue. Порядок обмена прочих файлов лежит вне области применения настоящего стандарта.

Таблица 43 — Таблица BXT_MRecipeOtherInformation

Атрибут	Описание
RE_ID	Идентифицирует элемент RE, с которым ассоциированы «прочие» данные
REVersion	Идентифицирует версию RE. Если она объединяется с идентификатором «RE_ID», то данное информационное поле определяет уникальную реализацию RE
StepID	Идентифицирует уникальную реализацию выполнения RE, если прочая информация ассоциируется с рассматриваемым шагом. Если поле имеет значение NULL, то прочая информация ассоциируется с элементом RE
DataID	Определяет порядок идентификации рассматриваемого элемента данных с помощью прочей информации
DataType	Идентифицирует тип значений данных с помощью множества элементов перечисления ValueDataType
DataValue	Задает значения данных прочей информации. Данное информационное поле может быть именем файла, содержащего фактические данные. Инструменты импортирования должны иметь доступ к данному файлу
Description	Содержит описание рассматриваемого типа элементов данных

5.2.5.5 Требования к оборудованию

Требования к оборудованию элемента RE содержат ограничения на оборудование и рецептурные условия. Они определены с помощью аналогичных таблиц и отношений, как те, что используются для определения возможностей оборудования в разделе 5.3.

Ограничения (условия) оборудования могут быть ассоциированы с любым рецептурным элементом в имеющейся иерархии рецептурных элементов. Требования к оборудованию RE задают ограничения на оборудование и рецептурные условия, соответствующие рассматриваемому уровню (например, уровню процедуры технологической установки, уровню производственной операции).

Требования к оборудованию содержатся в таблицах BXT_MRecipeElementEquip и BXT_MRecipeStepEquip. Таблица BXT_MRecipeElementEquip содержит определение и значение (по умолчанию) для свойства. Таблица BXT_MRecipeStepEquip содержит конкретное значение свойства, соответствующее рассматриваемому шагу.

5.2.5.5.1 Требования к оборудованию элемента RE

Элемент RE может заявлять требования к оборудованию (например, «имя реактора», «облицовка реактора»). Эти требования должны быть выполнены. Конкретные допустимые значения свойств для конкретных шагов определены в таблице BXT_MRecipeStepEquip.

Требования к оборудованию могут определять элементы данных, обеспечиваемые технологическим цехом в соответствии с рецептурой. Данные элементы могут быть использованы в рецептурных условиях перехода и в других выражениях. Примеры элементов данных: «VesselPressure» (сосуд высокого давления) технологической установки, «SteamPressure» (давление пара) блока оборудования.

Таблица 44 BXT_MRecipeElementEquip содержит указанное множество требований к оборудованию.

Таблица 44 — Таблица BXT_MRecipeElementEquip

Атрибут	Описание
RE_ID	Идентифицирует RE, для которого устанавливаются требования к оборудованию. При объединении данного информационного поля с полем «Version», оно определяет уникальную реализацию элемента RE

Окончание таблицы 44

Атрибут	Описание
REVersion	Идентифицирует версию RE. При объединении данного информационного поля с полем «RE_ID», оно определяет уникальную реализацию RE
PropertyID	Определяет свойство оборудования, необходимое для выполнения RE (например, «класс реактора», «отметка давления в сосуде»)
DefaultValue	Определяет значение по умолчанию для свойства, если оно не определено для рассматриваемого шага (например, «экзотермический реактор», «давление в сосуде»)
DataInterpretation	Задает порядок интерпретации значения, определенного как элемент перечисления (например, константа, ссылка, внешний элемент, уравнение). Формируется из множества элементов перечисления «ValueType»
EvaluationRule	Задает порядок сравнения данного значения со значением свойства оборудования, определенного как элемент перечисления. Формируется из множества элементов перечисления «EvaluationRule»
EngrUnits	Идентифицирует технические единицы измерения рассматриваемого значения (например, кг, фунт)
Description	Содержит описание свойства, поясняет его необходимость для выполнения рецептуры

5.2.5.5.2 Требования к оборудованию для шага RE

Рецептурный шаг BXT_MRecipeStep может задавать конкретные значения свойств оборудования (например, «класс реактора», «экзотермичность», «облицовка реактора = стекло»). Таблица 45 BXT_MRecipeStepEquip определяет указанное множество соответствующих требований к оборудованию.

Таблица 45 — Таблица BXT_MRecipeStepEquip

Атрибут	Описание
ParentRE	Часть ключа к таблице BXT_MRecipeStep для рассматриваемого параметра
ParentVersion	Часть ключа к таблице BXT_MRecipeStep для рассматриваемого параметра
StepID	Идентифицирует уникальную реализацию выполнения RE. Ее имя является уникальным в области применения родительского элемента RE
PropertyID	Идентифицирует свойство RE
PropertyValue	Содержит значение свойства, номера передаются как представления ASCII для указанного номера. Если типом параметра является множество пользовательских перечислений, то данный параметр может быть членом указанного множества

«Смысл» накладываемых ограничений не важен для определений языка. Вместе с тем, названия наборов оборудования или сущностей оборудования должны согласовываться.

5.3 Обмен моделями оборудования технологического цеха

Здесь определено множество таблиц, дающих описание возможностей оборудования в технологическом цехе. Данная информация отражает фактические возможности технологического цеха. Обмен возможностей технологического цеха может быть полезным в контексте обмена информацией даже в отрыве от рассмотрения рецептуры.

5.3.1 Описание оборудования

Таблицы возможностей технологического цеха составлены в контексте иерархии описаний оборудования. Элементы данной иерархии соответствуют элементам иерархии оборудования МЭК 61512-1.

Таблицы обмена информацией об оборудовании включают:

a) элемент оборудования: определяются конкретный элемент оборудования, класс оборудования;

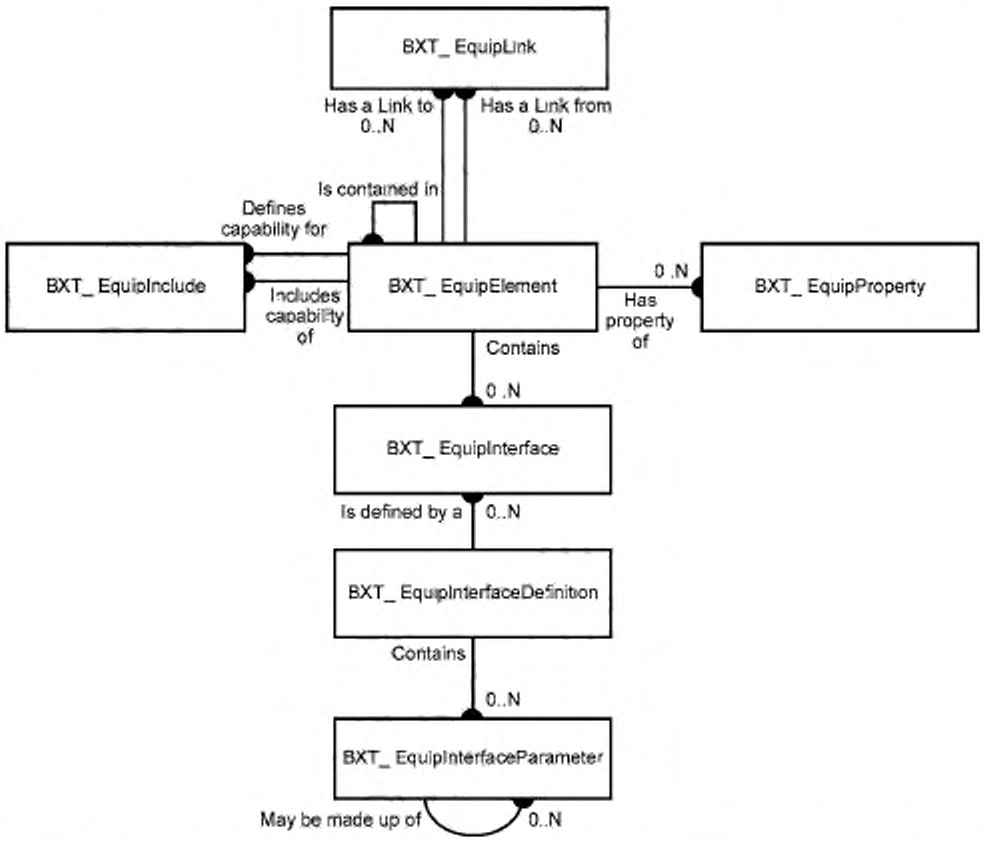
b) интерфейс процедурного элемента оборудования: определяется интерфейс процедурного элемента, доступного в рассматриваемом элементе оборудования;

c) свойство элемента оборудования: определяются свойства оборудования, спецификация свойств отражает спецификацию свойств оборудования RE;

d) соединительные звенья (связи) элементов оборудования: определяются связи оборудования, спецификация свойств отражает связи оборудования элемента RE технологической рецептуры.

5.3.2 Обзор таблиц и ограничения целостности

На рисунке 18 приведена структура таблиц, содержащих информацию об оборудовании.



BXT_EquipLink — таблица связей оборудования; has a Link to 0…N — имеет связь с …; has a Link from 0…N — имеет связь от …; Is contained in — содержится в …; defines capability for — определяет возможность для …; BXT_EquipInclude — таблица состава оборудования; Includes capability of — включает возможность …; BXT_EquipElement — таблица элементов оборудования; contains — содержит …; has property of — имеет свойство …; BXT_EquipProperty — таблица свойств оборудования; BXT_EquipInterface — таблица интерфейса оборудования; Is defined by a — определен с помощью …; BXT_EquipInterfaceDefinition — таблица определений интерфейса оборудования; BXT_EquipInterfaceParameter — таблица параметров интерфейса оборудования; may be made up of — может быть составлена из …

Рисунок 18 — Таблицы обмена информацией об оборудовании

5.3.3 Обзор таблицы описания оборудования

5.3.3.1 Иерархия оборудования

Таблица информации об оборудовании содержит иерархическое определение оборудования, составленное с помощью атрибута «ContainedIn» таблицы BXT_EquipElement. Данная таблица указывает оборудование, содержащееся внутри другой единицы оборудования. Например, рассматриваемый элемент оборудования технологического цеха может содержать другие элементы оборудования, составленные из технологических установок и/или блоков оборудования.

Каждый уровень может иметь ассоциированные спецификации свойств, спецификации данных и спецификации соединительных звеньев. Каждый уровень может также иметь ассоциированные процедурные элементы, чтобы обеспечивать поддержку полной модели, определенной в Части 1 настоящего Стандарта. Данные элементы обычно ассоциируются с технологическими установками или блоками оборудования.

5.3.3.2 Классы оборудования

Таблицы информации об оборудовании обеспечивают спецификацию классов оборудования с помощью ассоциации типа «включение возможностей» между рассматриваемыми элементами. Элементы оборудования могут также включать дополнительную возможность, определенную в одном или нескольких элементах оборудования.

Например, некоторая технологическая установка может быть элементом класса технологических установок. Это обеспечивает реализацию множества процедур, определенных данным классом. Технологическая установка содержит множество свойств класса, а также любую комбинацию рассматриваемой информации. В таблице BXT_EquipElement имеется одна запись для каждого определения реализации оборудования. В таблице BXT_EquipElement имеется одна запись для каждого определения класса.

5.3.4 Сводный анализ таблицы информации об оборудовании

Таблицы, используемые для обмена информацией об оборудовании, сведены в таблицу 46.

Таблица 46 — Таблица обмена информацией об оборудовании

Сокращенное имя таблицы	Полное имя таблицы	Описание
BXT_EquipElement	Элемент оборудования	Одна запись для каждого элемента оборудования (класса оборудования)
BXT_EquipLink	Спецификация связи (соединительного звена) элемента оборудования	Одна запись для каждого соединительного звена (связи) между элементами оборудования
BXT_EquipInclude	Элемент оборудования включает	Одна запись для каждого элемента оборудования, принадлежащего классу элементов оборудования
BXT_EquipProperty	Спецификация свойства элемента оборудования	Одна запись для каждой спецификации свойства, значение свойства для элемента оборудования
BXT_EquipInterface	Интерфейс процедурного элемента оборудования	Одна запись для каждого процедурного элемента оборудования, определенного внутри элемента оборудования
BXT_EquipInterfaceDefinition	Определение интерфейса процедурного элемента оборудования	Одна запись для каждого класса интерфейса процедурного элемента оборудования, определенного внутри элемента оборудования
BXT_EquipInterfaceParameter	Параметр интерфейса процедурного элемента оборудования	Одна запись для каждого входа элемента данных, каждого выхода элемента данных, каждого процедурного элемента оборудования, определенного внутри элемента оборудования

5.3.5 Определения таблицы оборудования

Нижеследующий подраздел содержит подробную спецификацию таблиц информации об оборудовании.

5.3.5.1 Элемент оборудования

Таблица BXT_EquipElement содержит одну запись для каждой сущности оборудования (например, технологический цех, технологическая установка, блок оборудования, блок управления) (см. таблицу 47). Таблица BXT_EquipElement содержит определения сущностей и классов сущностей (например, «Реактор101», «Фильтр20», «Реактор», «Фильтр»). Таблица BXT_EquipInclude, определенная ниже, содержит отношения класса.

Таблица 47 — Таблица BXT_EquipElement

Атрибут	Описание
EquipmentID	Идентифицирует элемент оборудования (класс оборудования)
EE_Type	Идентифицирует тип записи как элемент перечисления (например, определение класса, определение элемента). Используется элемент перечисления типа «EquipmentType»

Окончание таблицы 47

Атрибут	Описание
EE_Level	Идентифицирует уровень оборудования как элемент перечисления (например, производственная площадь, технологическая установка, блок оборудования, блок управления). Используется элемент перечисления типа «EquipmentLevel»
ContainedIn	Идентифицирует элемент оборудования, включающий в себя рассматриваемое оборудование (например, технологический цех, в котором имеется рассматриваемая технологическая установка). Данное информационное поле может иметь значение NULL, если рассматриваемое оборудование не входит в состав какого-либо другого оборудования, или если это другое оборудование не определено
Description	Содержит описание элемента оборудования

5.3.5.2 Соединительные звенья элементов оборудования

Элемент оборудования может иметь спецификацию соединительных звеньев элементов оборудования (например, рассматриваемое оборудование может снабжать топливом другое оборудование). Указанные соединительные звенья обычно устанавливают порядок передачи, материала, принятый в технологическом цехе (технологической установке). Например, рецептура может устанавливать соединительные звенья МИКСЕРА и РЕАКТОРА. Таблица BXT_EquipLink описывает соединительные звенья элементов оборудования (см. таблицу 48).

Таблица 48 — Таблица BXT_EquipLink

Атрибут	Описание
EquipmentID	Идентифицирует элемент оборудования (класс оборудования)
ToEquipmentID	Идентифицирует элемент оборудования (класс оборудования), к которому присоединяется оборудование с указанным идентификатором «EquipmentID»
Description	Содержит описание типов соединительных звеньев элементов оборудования

5.3.5.3 Элемент оборудования включает возможности

Элемент оборудования может включать в себя возможности одного или нескольких классов элементов оборудования. Например, может существовать элемент оборудования, представляющий собой целый класс реакторов с конкретным множеством процедурных элементов, в свою очередь представляющих производственные возможности имеющегося реактора. Особые технологические установки могут включать возможности целого класса реакторов, предоставлять дополнительные возможности (спецификации). Таблица BXT_EquipInclude представляет отношения между классом оборудования и реализацией оборудования (см. таблицу 49).

Таблица 49 — Таблица BXT_EquipInclude

Атрибут	Описание
EquipmentID	Идентифицирует элемент оборудования (класс оборудования)
ClassEquipmentID	Идентифицирует класс оборудования, включающий оборудование с идентификатором EquipmentID
Description	Содержит описание ассоциации

5.3.5.4 Свойства элементов оборудования

Свойства элементов оборудования определяют возможности, доступные для каждого элемента оборудования. Возможности обеспечиваются путем указания доступного оборудования в соответствии с требованиями рецептуры с помощью определения спецификации и задания значения спецификации.

Спецификации могут ассоциироваться с любым элементом оборудования в рамках иерархии оборудования. Спецификации должны применяться в соответствии с уровнем рассмотрения (например, на уровне технологического цеха, на уровне технологической установки, на уровне блока оборудования).

Таблица BXT_EquipProperty (см. таблицу 50) содержит одну запись для каждого свойства, присущего элементу оборудования (например, свойство «тип облицовки»", свойство «стеклянная облицовка технологической установки», свойство «размер», свойство «50000 галлонов» и т. д.). Спецификации

могут также включать элементы данных оборудования, присущие данному оборудованию (например, «температура пара» для теплового блока оборудования, «VesselPressure» (давление в сосуде) для технологической установки). Указанные элементы данных могут быть доступными для использования в рецептурных переходах и выражениях.

Таблица 50 — Таблица BXT_EquipProperty

Атрибут	Описание
EquipmentID	Идентифицирует элемент оборудования (класс оборудования)
PropertyID	Идентифицирует свойство, обеспечиваемое данным оборудованием (например, «тип облицовки», «размер», «теплопроводность», «температура пара»)
PropertyValue	Идентифицирует значение свойства (например, «стекло», «50000», «650»)
EngrUnits	Задает технические единицы измерения свойств (например, «gallons» (галлонов), «BTU/hr» (британских тепловых единиц в час))
Description	Содержит описание типа свойства элемента оборудования

5.3.5.5 Интерфейс процедурного элемента оборудования

Таблица BXT_EquipInterface (см. таблицу 51) содержит одну запись для каждой процедуры элемента оборудования, определенной внутри элемента оборудования. Каждая запись таблицы BXT_EquipInterface дает отображение на определение интерфейса для ассоциированной процедуры элемента оборудования. Так как несколько таблиц BXT_EquipInterface (например, для фаз оборудования) могут иметь одно внешнее определение интерфейса, все они могут ссылаться на одно определение интерфейса типа BXT_EquipInterface. Данная структура допускает определение функционально эквивалентных интерфейсов BXT_EquipInterface рецептур с учетом класса оборудования.

Таблица 51 — Таблица BXT_EquipInterface

Атрибут	Описание
EquipmentID	Идентифицирует элемент оборудования (класс оборудования)
EPI_ID	Идентифицирует интерфейс процедурного элемента оборудования
EPI_Definition	Идентифицирует определение параметров интерфейса BXT_EquipInterface
Description	Содержит описание типа процедурного элемента оборудования

5.3.5.6 Определение интерфейса процедурного элемента оборудования

Таблица BXT_EquipInterfaceDefinition (см. таблицу 52) содержит одну запись для каждого определенного интерфейса процедурного элемента оборудования. Таблица BXT_EquipInterfaceDefinition содержит определения входных и выходных параметров таблицы BXT_EquipInterface.

Таблица 52 — Таблица BXT_EquipInterfaceDefinition

Атрибут	Описание
EPI_Definition	Идентифицирует определение интерфейса процедурного элемента оборудования
Description	Содержит описание ожидаемого поведения интерфейса BXT_EquipInterface

5.3.5.7 Параметры интерфейса процедурного элемента оборудования

Таблица BXT_EquipInterfaceParameter (см. таблицу 53) содержит одну запись для каждого элемента данных, требуемого, генерируемого или модифицируемого путем выполнения заданного процедурного элемента оборудования, определенного внутри некоторого элемента оборудования. Таблица BXT_EquipInterfaceParameter содержит определение типа и единиц измерения рассматриваемого элемента данных, а также необязательную ссылку на некоторое множество элементов перечисления.

Таблица BXT_EquipInterfaceParameter также указывает, является ли значение входа масштабируемым. Она может задавать значение по умолчанию, используемое, если фактические значения параметров на процедурный элемент оборудования не поступают.

Таблица 53 — Таблица BXT_EquipInterfaceParameter

Атрибут	Описание
EPI_Definition	Идентифицирует класс интерфейса процедурного элемента оборудования
ParameterID	Идентифицирует имя параметра, используемого процедурным элементом оборудования
ParentParamID	Идентифицирует родительское множество параметров, членом которого является рассматриваемый параметр. Данное информационное поле равно NULL, если указанное множество параметров отсутствует
Type	Идентифицирует тип данных, используемый процедурным элементом оборудования. Формируется из множества элементов перечисления типа ValueDataType
EngrUnits	Идентифицирует технические единицы измерения данных, используемые рассматриваемым процедурным элементом оборудования
EnumSet	Идентифицирует множество элементов перечисления, членом которого является рассматриваемый элемент (например, если информационное поле не содержит NULL)
Scaled	Указывает, можно ли масштабировать параметр перед тем, как он поступает на процедурный элемент оборудования. Формируется из множества элементов перечисления типа «Boolean». Если информационное поле равно TRUE, то параметр можно масштабировать
DefaultValue	Идентифицирует значение по умолчанию, используемое, если фактические значения параметров на процедурный элемент оборудования не поступают
Description	Содержит описание параметра процедурного элемента оборудования

5.4 Обмен информацией календарного планирования

Календарные таблицы определяют один или несколько календарных планов производства партии изделий в данном технологическом цехе. Каждый календарный план производства партии изделий содержит дополнительную специальную информацию о производстве партии изделий, используемую вместе с информацией о технологической рецептуре для создания рецептуры управления. Каждый рассматриваемый календарный план производства партии изделий может задавать множество значений параметров, необходимых для разработки рецептуры, а также множество требований к оборудованию. Однако не вся информация, необходимая для выполнения рецептуры управления, должна отражаться в таблицах обмена календарной информацией. Дополнительная информация поступает от системы управления или от оператора.

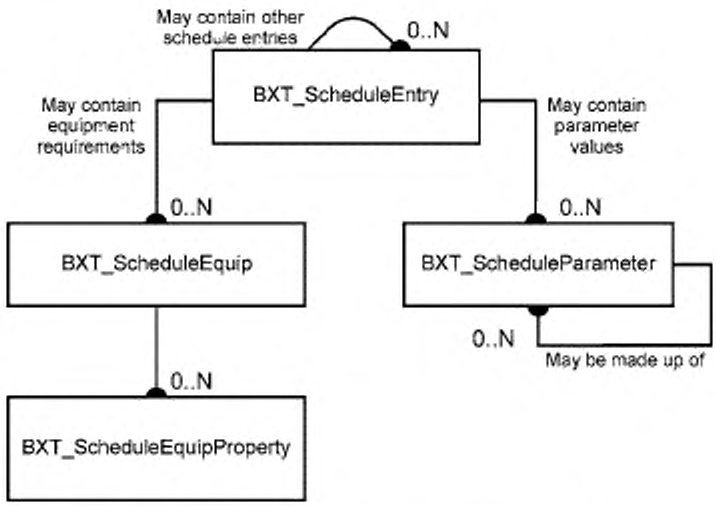
Разработка мероприятий календарного плана может потребовать новой информации, дополнительно к той, что обеспечивается данной таблицей. Разработка календарного плана может потребовать использования других средств для получения новых данных (например, статус оборудования, запасы материала, коммунальные условия). Для обмена данной информацией требуются особые методы.

5.4.1 Обзор календарных таблиц

Таблицы обмена календарной информацией обеспечивают включение информации (по нескольким календарным планам производства партии изделий) в отдельное множество таблиц.

Таблицы обмена календарной информацией не указывают порядок создания информации или порядок ее использования. Инструменты, использующие указанную информацию, могут включать пакеты разработки календарного плана, пакеты автоматизации серийного производства, пакеты отображения операций, пакеты отслеживания порядка выполнения производственных операций. Инструменты импортирования и экспортирования определяют порядок корректного использования календарной информации в рассматриваемых таблицах.

На рисунке 19 приведены пять вспомогательных таблиц, с помощью которых формируется таблица обмена календарной информацией.



May contain other schedule entries — может содержать другие календарные записи; BXT_ScheduleEntry — таблица внесения календарных записей; may contain equipment requirements — может содержать требования к оборудованию; may contain parameter values — может содержать значения параметров; BXT_ScheduleEquip — таблица календарного планирования оборудования; BXT_ScheduleParameter — таблица календарных параметров; may be made up of — может содержать ...; BXT_ScheduleEquipProperty — таблица календарных свойств оборудования

Рисунок 19 — Календарная структура

5.4.2 Сводный анализ календарных таблиц

Таблицы, используемые для обмена календарной информацией, описаны в таблице 54.

Таблица 54 — Таблицы обмена календарной информацией

Имя таблицы	Описание
BXT_ScheduleEntry	Одна запись для каждого календарного пункта (например, производство партии изделий, рецептура технологической установки, мероприятие очистки)
BXT_ScheduleEquip	Одна запись для каждого требования к выбору оборудования. Может определять допустимый выбор оборудования
BXT_ScheduleProperty	Одна запись для каждой спецификации свойства для каждого требования к оборудованию
BXT_ScheduleParameter	Одна запись для каждого пункта параметра календарной записи

5.4.3 Определения календарных таблиц

5.4.3.1 Календарная запись

Таблица BXT_ScheduleEntry (см. таблицу 55) содержит один элемент для каждого календарного мероприятия. Календарные записи могут представлять производство партии изделий и прочие технологические действия (например, рецептуру технологической установки). Если календарное мероприятие — это производство партии изделий, то таблица содержит идентификацию партии изделий, ассоциируемую с календарным планом производства данной партии (например, назначенной на конкретное время) в информационном поле ScheduleEntryID. Технологическая рецептура, ассоциируемая с календарным планом изготовления партии, идентифицируется в информационном поле RE_ID. Информационное поле ScheduleEntryString используется для уникальной идентификации календарных записей, если их фактический идентификатор ID не был назначен.

Таблица 55 — Таблица BXT_ScheduleEntry

Атрибут	Описание
ScheduleEntryID	Указывает уникальный идентификатор ID календарной записи (внутри данной таблицы). Это может быть идентификатор производственной кампании, идентификатор партии, идентификатор процедуры технологической установки, идентификатор уникальной строки без внешнего смысла
ParentSchedID	Идентифицирует родительскую запись календарного пункта, к которой относится данная запись, путем использования родительской строки ScheduleEntryString
ExternalID	Определяет идентификатор, используемый коммерческой структурой для идентификации данной календарной записи
RE_ID	Идентифицирует рецептурный элемент (например, Red Oak (красный дуб)). При объединении данного информационного поля с полем «version», оно уже определяет уникальную реализацию элемента RE. Если рассматриваемая запись представляет технологическую рецептуру, то данное информационное поле содержит идентификатор технологической рецептуры. (Идентифицирует рецептурный элемент, на который ссылается рассматриваемая календарная запись)
REVersion	Идентифицирует версию технологической рецептуры
SE_Type	Определяет тип сущности, представляемой данной календарной записью. Формируется из множества элементов перечисления типа «SE_Type». Данное определение позволяет: в записи партии изделий указывать более подробную календарную информацию о нижних уровнях иерархии процедур (например, свойства и требования к оборудованию могут идентифицироваться для каждой процедуры технологической установки в рамках рецептуры), обеспечивать разработку календарных планов производственных кампаний и производства групп партий изделий
BatchID	Указывает идентификатор партии изделий, включающий рассматриваемый календарный пункт
LotID	Указывает идентификатор партии, включающий данный календарный пункт
CampaignID	Указывает идентификатор производственной кампании, включающий данный календарный пункт
ProductID	Идентифицирует изготавливаемый продукт
OrderID	Идентифицирует порядок изготовления или требование заказчика, которому данная календарная запись поставлена в соответствие
E_Action	Характеризует ожидаемое действие с помощью получающего инструмента как элемента перечисления (например: новое, обновленное, стертое, определенное пользователем). Формируется из множества элементов перечисления типа «ScheduleAction»
SchedStatus	Определяет статус календарной записи (например, выполненная, редактируемая, календарная, вносимая с учетом календарного плана). Формируется из множества элементов перечисления типа «ScheduleStatus»
StartCondition	Задает ожидаемое условие начала работ в календарной записи, если таковое существует (например, «начать работы до ...», «начать работы после ...»)
InitialMode	Определяет режим, в котором начинается выполнение календарной записи как элемента перечисления (например, автоматический, полуавтоматический, ручной). Формируется из множества элементов перечисления «ScheduleMode»
SchedStartTime	Задает ожидаемое время начала работ в календарной записи, если таковое существует
SchedEndTime	Задает ожидаемое время окончания работ в календарной записи, если таковое существует
BatchPriority	Задает приоритет, указанный в календарной записи, если таковой существует. Нижние №№ имеют более высокий приоритет (например, приоритет №1 выше, чем приоритет №7)
BatchSize	Определяет запрошенный размер (масштабный фактор) партии изделий. Основан на масштабном факторе производства партии изделий в соответствии с технологической рецептурой
EngrUnits	Идентифицирует (по выбору) технические единицы измерения размера партии BatchSize
SENote	Содержит информацию (инструкции) по выполнению операций
Description	Содержит описание календарного пункта и/или продукта (например, Premium Beer)

5.4.3.2 Требования к оборудованию в календарной записи

Таблица BXT_ScheduleEquip (см. таблицу 56) содержит один элемент для каждого требования к оборудованию в календарной записи. Соответствующая таблица BXT_ScheduleEquipProperty содержит определения конкретных свойств, которыми должно обладать используемое оборудование.

Типовым требованием для производства партии изделий является корректный выбор оборудования. Требования к оборудованию в календарной записи и свойства оборудования обычно соответствуют требованиям к оборудованию и свойствам оборудования в технологической рецептуре. Например, пакет разработки календарного плана может описывать конкретную технологическую установку, используемую на производстве. Таблица BXT_ScheduleEquip устанавливает идентичность оборудования для элемента RE. Таблица BXT_ScheduleProperty задает имя выбранной технологической установки.

Таблица 56 — Таблица BXT_ScheduleEquip

Атрибут	Описание
ScheduleEntryID	Идентифицирует обмениваемый календарный элемент
RequirementID	Указывает уникальное имя для требования к оборудованию в календарном пункте. Данное имя может относиться к индивидуальной единице оборудования, к классу оборудования, к списку допустимого оборудования, к прочим группам оборудования (например, к агрегату, состоящему из последовательных элементов)
Description	Содержит описание требования (например, первая реакторная установка для производства партии изделий)

5.4.3.3 Требование к свойствам оборудования в календарной записи

Таблица BXT_ScheduleProperty (см. таблицу 57) содержит один элемент для каждой спецификации требования к свойствам оборудования в календарной записи.

Так как отдельно взятое требование к оборудованию в данном календарном пункте может иметь несколько критериев свойств (например, тип конструкционного материала, объем материала), для формирования списка указанных требований используется отдельная таблица.

Таблица 57 — Таблица BXT_ScheduleProperty

Атрибут	Описание
ScheduleEntryID	Идентифицирует обмениваемый календарный пункт
RequirementID	Идентифицирует множество требований, ассоциированных с производством партии изделий (обычно это класс оборудования или класс материалов)
PropertyName	Идентифицирует имя свойства для производства партии изделий в соответствии с календарным планом
PropertyValue	Задает значение свойства для производства партии изделий в соответствии с календарным планом
EngrUnits	Идентифицирует (по выбору) технические единицы измерения для рассматриваемого значения свойства PropertyValue
Description	Содержит описание свойства (например, использование устройства №345 как первой реакторной установки в производстве партии изделий)

5.4.3.4 Календарный параметр

Таблица BXT_ScheduleParameter (см. таблицу 58) содержит один элемент для каждого параметра календарного пункта. Параметр календарного пункта – это обычно параметр технологической рецептуры. Данные параметры могут также содержать информацию для оператора или для прочих пользователей календарной информации.

Пределы пунктов календарных параметров могут быть определены так же, как ниже подпараметры определяют параметры рецептур.

Таблица 58 — Таблица BXT_ScheduleParameter

Атрибут	Описание
ScheduleEntryID	Идентифицирует обмениваемый календарный пункт
ParameterID	Идентифицирует параметр календарного плана производства партии изделий
ParentParameterID	Идентифицирует рассматриваемое родительское множество параметров, в которое входит данный параметр (если такого множества нет, то в информационном поле стоит NULL)
ParameterValue	Задает значение параметра календарного плана производства партии изделий
EngrUnits	Идентифицирует (по выбору) технические единицы измерения значения параметра типа ParameterValue
ItemLocation	Определяет порядок использования параметра в структуре рецептуры. Данная запись используется, если идентификатор ParameterID условный. Идентификатор ParameterID не содержит достаточной информации для идентификации места применения параметра
EnumSet	Идентифицирует множество элементов перечисления, куда входит данный параметр (если такого множества нет, то в информационном поле стоит NULL)
Description	Содержит описание параметра для соответствующего календарного пункта и/или продукта (например, Premium Beer)

5.5 Обмен производственной информацией

Обмен производственной информацией формирует особую структуру, обеспечивающую обмен всей информацией о серийном производстве.

Данную информацию могут использовать многие инструменты (например, автоматические системы серийного производства, лабораторные системы управления информацией, системы оформления отчета о производстве партии изделий, системы анализа производства партии изделий, системы разработки календарного плана, системы моделирования).

Структура таблиц обмена позволяет использовать эти таблицы для обмена данных по нескольким партиям изделий.

Производственная информация содержится в трех местах:

a) в рецептуре управления;

b) в настройках оборудования в соответствии с выполняемой рецептурой;

c) в журнале событий, имеющих место в ходе выполнения рецептуры.

5.5.1 Информация о рецептуре управления

Информацией о рецептуре управления обмениваются с помощью таблиц MR. Идентификатор продукта ProductID – это представление идентификатора партии изделий.

Первоначально рецептура управления представляет собой копию технологической рецептуры. При этом таблицы MR могут содержать технологическую рецептуру производства. Таблица событий может содержать любые изменения рецептуры управления. Таблица MR может содержать модифицированную рецептуру управления. В обоих случаях информация об идентификаторе рецептуры RecipeID и версии элемента REVersion помогает идентифицировать конкретную технологическую рецептуру, использованную для создания рецептуры управления.

5.5.2 Информация об оборудовании

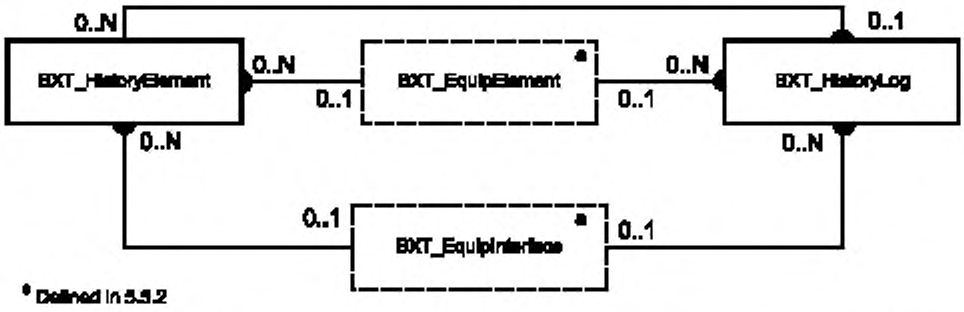
Оборудование, используемое для производства партии изделий, может самостоятельно обмениваться с помощью таблицы BXT_EquipElement (сущностей оборудования). Данная таблица содержит определения оборудования всего технологического цеха, используемого в производстве. Она может содержать только подмножество единиц оборудования, фактически используемых для производства конкретной партии изделий.

5.5.3 История производства партии изделий

История производства партии изделий содержится в двух таблицах: BXT_HistoryElement и BXT_HistoryLog.

Таблица BXT_HistoryElement регистрирует выполнение процедурного элемента рецептуры и/или эквивалентного процедурного элемента оборудования. Данная таблица содержит один элемент для каждой реализации выполнения элементов RE или EPE. На рисунке 20 показаны элементы истории

производства партии изделий и отношения для предварительно определенных таблиц элементов оборудования. Таблица BXT_HistoryElement (см. таблицу 59) содержит записи, используемые для ссылок на ассоциированное оборудование и на процедурные элементы оборудования.



BXT_HistoryElement — таблица элементов истории производства; BXT_EquipElement — таблица элементов оборудования; BXT_HistoryLog — журнал исторических событий; BXT_EquipInterface — таблица интерфейсов оборудования

Рисунок 20 — История производства партии изделий

Таблицы BXT_HistoryLog и BXT_HistoryElement содержат список календарных записей, внесенных в ходе производства партии изделий, — одна запись на каждое событие, зарегистрированное при производстве партии изделий. Таблица BXT_HistoryElement регистрирует каждую реализацию процедурного элемента рецептуры или процедурного элемента оборудования. Таблица BXT_HistoryLog содержит одну запись на каждое событие, которое имело место при реализации процедурного элемента (например, начало выполнения процедурного элемента, изменение режима, изменение состояния, зарегистрированное значение параметра элемента).

Таблицы истории производства партии изделий формируются по нескольким причинам:

- информация, дублируемая несколькими записями, передается в таблицу BXT_HistoryElement. Это позволяет существенно сократить размер журнала BXT_HistoryLog;
- информация, описываемая в таблицах оборудования, отыскивается по ключевому значению. Записи таблицы BXT_HistoryElement: идентифицируют элементы оборудования на множестве ассоциированного оборудования, идентифицируют ассоциированные процедурные элементы оборудования;
- таблица BXT_HistoryLog включает ссылку на элементы оборудования и процедурные элементы оборудования (для упрощения порядка использования журнала BXT_HistoryLog) несмотря на то, что данная информация дублируется таблицей BXT_HistoryElement.

5.5.3.1 Элементы истории производства

Таблица BXT_HistoryElement содержит одну запись на каждый рецептурный элемент (см. таблицу 59).

Таблица 59 — Таблица BXT_HistoryElement

Атрибут	Описание
HistoryElementID	Обеспечивает генерацию идентификатора, необходимого для обеспечения реляционной целостности
BatchID	Задает уникальную идентификацию партии изделий, ассоциируемую с записью таблицы BXT_HistoryElement. Данная информация дублирует информацию, содержащуюся в журнале BXT_HistoryLog
MasterRecipeID	Идентифицирует технологическую рецептуру, ассоциируемую с производством рассматриваемой партии изделий
MasterRecipeVersion	Идентифицирует версию рассматриваемой технологической рецептуры
ControlRecipeID	Идентифицирует рецептуру управления. В некоторых случаях данный идентификатор может отличаться от идентификатора партии
ReferenceEquipProcedure	Указывает, определена ли иерархия процедурного управления, чтобы ссылаться на рецептуру (оборудование) как на булевский элемент перечисления. Если информационное поле содержит TRUE, то производится ссылка на оборудование

Окончание таблицы 59

Атрибут	Описание
RecipeProcedure	Идентифицирует процедуру, ассоциированную с записью таблицы BXT_ HistoryElement
UnitProcedure	Идентифицирует процедуру технологической установки, ассоциированную с записью таблицы BXT_HistoryElement
UnitProcedureCounter	Описывает реализации счетчика исполнения, равные количеству раз выполнения процедуры технологической установки. Данный счетчик необходим, так как рассматриваемая рецептура может выполнять одну и ту же процедуру технологической установки несколько раз вследствие вмешательства оператора или зацикливания
Operation	Идентифицирует операцию, ассоциированную с записью таблицы BXT_ HistoryElement
OperationCounter	Описывает реализации счетчика исполнения, равные количеству раз выполнения процедуры технологической установки. Данный счетчик необходим, так как рассматриваемая рецептура может выполнять одну и ту же процедуру технологической установки несколько раз вследствие вмешательства оператора или зацикливания
Phase	Идентифицирует рецептурную фазу, ассоциированную с записью таблицы BXT_ HistoryElement
PhaseCounter	Описывает реализации счетчика исполнения, равные количеству раз выполнения процедуры технологической установки. Данный счетчик необходим, так как рассматриваемая рецептура может выполнять одну и ту же процедуру технологической установки несколько раз вследствие вмешательства оператора или зацикливания
EquipmentID	Идентифицирует элемент оборудования, ассоциированный с записью таблицы BXT_HistoryElement. Данная информация дублируется в журнале BXT_HistoryLog
EPI_ID	Идентифицирует процедурный элемент оборудования, ассоциированный с рассматриваемой записью. Данная информация дублируется в журнале BXT_HistoryLog.

5.5.3.2 Журнал истории производства

Таблица BXT_HistoryLog (см. таблицу 60) содержит пять массивов информации о событиях, регистрируемых в журнале:

- время события;
- информация о производстве партии изделий и рецептуре, ассоциированной с данным событием;
- оборудование, ассоциированное с событием;
- оператор, ассоциированный с событием;
- информация о событии.

Таблица BXT_HistoryLog содержит информацию о регистрируемых производственных событиях. Информация о регистрируемых производственных событиях категоризируется с помощью элементов перечисления в информационных полях RecordSet и RecordSubSet для облегчения процесса обработки информации (например, для фильтрации и сортировки).

Таблица 60 — Таблица BXT_HistoryLog

Атрибут	Описание
RecordID	Указывает сгенерированный идентификатор, необходимый для обеспечения реляционной целостности
UTC	Идентифицирует универсальное координированное время (UTC) и дату записи
LocalTime	Идентифицирует местное время и дату записи
BatchID	Обеспечивает уникальную идентификацию партии изделий, ассоциированную с рассматриваемой записью
HistoryElementID	Обеспечивает уникальную идентификацию реализации выполнения ассоциированного рецептурного элемента (процедурного элемента оборудования), ассоциированного с записью. Данное информационное поле – это ключ к таблице BXT_HistoryElement
EquipmentID	Идентифицирует элемент оборудования, ассоциированный с записью

Окончание таблицы 60

Атрибут	Описание
EPI_ID	Идентифицирует процедурный элемент оборудования, ассоциированный с записью
UserID	Задает имя пользователя (если оно существует), ассоциированное с записью
RecordSet	Задает тип записи из множества элементов перечисления записи RecordSet
RecordSubSet	Задает подтип записи из множества элементов перечисления, определяемого записью RecordSet
RecordAlias	Независимая спецификация оборудования (например, «максимальная температура сосуда»)
NewValue	Задает значение данных, ассоциированное с типом (подтипом) записи
OldValue	Определяет информационное поле, содержащее предшествующее значение данных
EngrUnits	Задает технические единицы измерения (если они существуют), соответствующие новым и старым значениям параметров NewValue и OldValue

5.6 Применение таблиц обмена

В ключевых информационных полях множества таблиц может содержаться одна и та же информация. Многие информационные поля таблиц имеют одинаковые области применения (например, конкретные типы данных и диапазоны значений). ИСО/МЭК 9075 в части языка SQL не определяет возможные области применения. Таблица 61 содержит определения областей применения для выбранных атрибутов таблиц обмена.

Таблица 61 — Области применения таблиц обмена

Имя области применения	Тип	Описание
BXT_MRecipeStep-Author	CHAR (32)	Имя (идентификатор) автора
BXT_MRecipeStep-StepID BXT_MRecipeStepParameter-StepID BXT_MRecipeStepEquip-StepID BXT_ScheduleEntry-StepID	CHAR (128)	Идентифицирует шаг внутри процедурного элемента рецептуры
BXT_MRecipeStepParameter-EngrUnits BXT_MRecipeStep-ScaleEngrUnits BXT_EquipProperty-EngrUnits BXT_EquipInterfaceParameter-EngrUnits BXT_ScheduleEntry-EngrUnits BXT_ScheduleProperty-EngrUnits BXT_ScheduleParameter-EngrUnits BXT_HistoryLog-EngrUnits	CHAR (32)	Спецификация технических единиц измерения
BXT_MRecipeStep-Status BXT_ScheduleEntry-SchedStatus	INTEGER	Статус рецептуры (процедурного элемента) как элемента перечисления
BXT_MRecipeElement-RE_ID BXT_MRecipeElementParameter-RE_ID BXT_MRecipeStep-ParentRE BXT_MRecipeStep-RE_ID BXT_MRecipeStepParameter-ParentRE BXT_MRecipeTransition-RE_ID BXT_MRecipeLink-RE_ID BXT_MRecipeElementEquip-RE_ID BXT_MRecipeOtherInformation-RE_ID BXT_MRecipeStepEquip-ParentRE	CHAR (128)	Идентифицирует процедурный элемент рецептуры

Окончание таблицы 61

Имя области применения	Тип	Описание
BXT_MRecipeElement-REVersion BXT_MRecipeStep Parameter-REVersion BXT_MRecipeStep-ParentVersion BXT_MRecipeStep-REVersion BXT_MRecipeStepParameter-PrentVersion BXT_MRecipeTransition-REVersion BXT_MRecipeLink-REVersion BXT_MRecipeElement Equip- REVersion BXT_MRecipeStepEquip-ParentVersion BXT_MRecipeOtherInformation-REVersion BXT_ScheduleEntry-Version BXT_HistoryElement-MasterRecipeVersion	CHAR (16)	Идентификатор версии для всех элементов, имеющих версию
BXT_MRecipeStepParameter-ParameterID BXT_MRecipeStepParameter-Parent Param ID BXT_MRecipeStepParameter-ParameterID BXT_MRecipeStepParameter-ParentParamID BXT_EquipInterfaceParameter-ParameterID BXT_EquipInterface-Prent ParamID BXT_ScheduleParameter-ParameterID	CHAR (32)	Идентифицирует параметр элемента RE, устанавливает порядок его использования
BXT_MRecipeStep-ProcessCell ID BXT_EquipElement-EquipmentID BXT_EquipLink-EquipmentID BXT_EquipLink-ToEquipmentID BXT_EquipInclude-EquipmentID BXT_EquipInclude-ClassEquipmentID BXT_EquipProperty-EquipmentID BXT_EquipInterface-EquipmentID BXT_EquipInterfaceParameter-EPI_Definition BXT_ScheduleProperty-RequirementID BXT_ScheduleEquip-RequirementID BXT_HistoryLog-EquipmentID BXT_HistoryElement-EquipmentID BXT_ScheduleEquip-RequirementID BXT_HistoryLog-EquipmentID BXT_HistoryElement-EquipmentID		Идентифицирует технологический цех или прочие сущности оборудования

6 Процедурные функциональные диаграммы

Данный раздел определяет метод графического представления технологических рецептур и рецептур управления. Порядок представления процедуры называют процедурной функциональной диаграммой (PFC). Данный раздел также устанавливает требования для представления формул, требования к оборудованию, требования к заголовкам и прочую информацию. Язык диаграммы PFC, в соответствии с настоящим стандартом, поддерживает рецептуры со сложными процедурами (например, параллельные шаги, процедуры выбора), которые могут изменяться от одного продукта к другому.

Процедурные функциональные диаграммы строятся на функциональных диаграммах в соответствии с МЭК 60848. В процедурных функциональных диаграммах предполагается, что шаги следуют за переходами, а переходы следуют за шагами в соответствии с МЭК 60848. Вместе с тем, имеется значительное отличие между указанным стандартом МЭК 60848 и настоящим стандартом. Оно обусловлено необходимостью удовлетворения требований процедурного управления в отличие от требований документации. При управлении серийным производством процедурные элементы оборудования содержат процедурную логику, устанавливающую порядок их завершения. Кроме того, многие фазы оборудования (при управлении производством партии изделий) сначала завершают работу, а затем отключаются. Использование PFC-диаграмм обеспечивает отделение процедурных элементов рецептуры от процедурных элементов оборудования. При этом предполагается, что если процедурный элемент оборудования запущен, то он выполняется независимо. Другим отличием, учтенным в процедурных функциональных диаграммах, является многоуровневая структура процедурных элементов рецептуры. Необходимо различать обозначения рецептурных фаз, обозначения рецептурных операций, обозначения рецептурных процедур технологической установки и обозначения рецептурной процедуры в целом.

6.1 Нотация процедурных функциональных диаграмм

Процедурные функциональные диаграммы содержат описание процедурной логики с помощью набора обозначений, взаимосвязанных направленными связями, определяющими порядок выполнения последовательности процедурных элементов. Выполнение процедурных элементов может происходить последовательно или параллельно. Выполнение их может зависеть от условной логики. Описанные действия включают планируемое выполнение процедур технологической установки, операций, фаз, а также оценки корректности переходов. Направления выполнения операций – «сверху-вниз» и «слева-направо». Процедурные функциональные диаграммы используются для описания процедурной логики на всех уровнях рецептуры: рецептурные процедуры, рецептурные процедуры технологической установки, рецептурные операции.

6.1.1 Обозначения

Процедурные функциональные диаграммы определяются множеством обозначений:

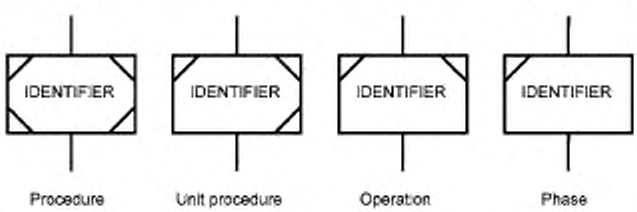
- элементов (например, процедурных элементов рецептур);
- точек начала и окончания процесса;
- выделения ресурса;
- синхронизации элементов;
- рецептурных переходов;
- базовых структур (например, направленных связей, выбора корректной последовательности, параллельных последовательностей).

Учитывается возможность глобального представления обозначений. При этом размеры и технические детали (например, толщина линий, шрифт) зависят от используемого приложения.

6.1.1.1 Элементы

Указанные обозначения используются для представления рецептурной фазы, рецептурной операции, рецептурной процедуры технологической установки, рецептурной процедуры. Графическая индикация внутри обозначения используется для идентификации обозначения, представляющего рецептурную фазу, рецептурную операцию, рецептурную процедуру технологической установки, рецептурную процедуру. Пример идентификации процедурного элемента приведен на рисунке 21.

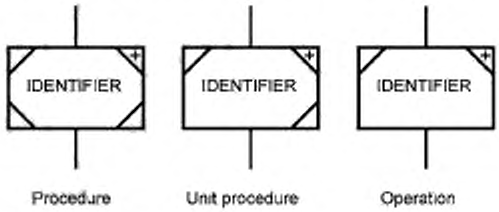
В МЭК 61512-1 определены четыре уровня процедурной иерархии. Только эти четыре уровня рассматриваются в настоящем стандарте. Вместе с тем, возможны и дополнительные уровни, вводимые для различных целей. Тогда как настоящий стандарт рассматривает только четыре указанных уровня процедурных элементов, определенных в МЭК 61512-1, любые (определенные отдельно) дополнительные уровни могут быть идентифицированы как графически (в соответствии с установленными здесь общими принципами), так и текстовой строкой, размещаемой в верхнем левом углу прямоугольника блок-схемы.



Identifier — идентификатор; procedure — процедура; unit procedure — процедура технологической установки; operation — операция; phase — фаза

Рисунок 21 — Обозначения процедурных элементов рецептуры

Процедурные элементы, расположенные выше уровня фазы, могут представлять собой пакеты прочих процедурных элементов, расположенных на последующих нижних уровнях иерархии процедурного управления. Процедурные элементы, представляющие собой инкапсуляцию, в которых процедурные элементы рецептуры нижнего уровня не показаны, идентифицируются знаком «плюс» (+) в верхнем правом углу прямоугольника блок-схемы, представляющего рассматриваемый процедурный элемент (см. рисунок 22). Обозначения процедурных элементов, представляющие собой инкапсуляцию без вложений идентифицируются знаком «минус» (-) (см. рисунок 38). Обозначения процедурных элементов, ссылающихся на процедурные элементы оборудования, не имеют индикации.



Identifier — идентификатор; procedure — процедура; unit procedure — процедура технологической установки; operation — операция

Рисунок 22 — Процедурные элементы, включающие пакеты процедурных элементов рецептуры нижнего уровня

Если процедурный элемент представляет собой пакет подчиненных процедурных элементов, то для его определения может быть использована отдельная процедурная функциональная диаграмма нижнего уровня, задающая эти подчиненные процедурные элементы и ассоциированные упорядочивающие обозначения. Иконка, представляющая инкапсулирующий процедурный элемент рецептуры, также может быть расширена, чтобы обеспечить описание процедурной функциональной диаграммы нижнего уровня внутри границ данной инкапсулирующей иконки. Процесс расширения отдельных обозначений можно продолжать до попадания на подчиненный уровень. Процедуру оборудования можно также показать как расширение процедурного элемента рецептуры, который на данную процедуру ссылается.

6.1.1.2 Точки начала и окончания работ

Процедурные функциональные диаграммы имеют, по крайней мере, одну точку начала работ и, по крайней мере, одну точку окончания работ, в отличие от последовательной функциональной диаграммы, которая может быть непрерывно зациклена.

6.1.1.2.1 Начало работ

Для обозначения начала работ по каждой процедурной функциональной диаграмме (подчиненной процедурной функциональной диаграмме) используется, по крайней мере, одно обозначение начала работ (см. рисунок 23).



Рисунок 23 — Обозначение начала работ

6.1.1.2.2 Окончание работ

По крайней мере, одно обозначение (см. рисунок 24) используется для окончания работ по каждой процедурной функциональной диаграмме (подчиненной процедурной функциональной диаграмме).



Рисунок 24 — Обозначение окончания работ

6.1.1.3 Выделение ресурса

Некоторые ресурсы выделяются (для производства партии изделий) до того как необходимость в них возникает. Управление таймингом данного выделения может оказаться важным для рецептуры или для разработки календарного плана. Необходимо установить правила выделения ресурса и условия начала работ. Выделение (на блок-схеме) указывается овальной иконкой (см. рисунок 25). Оно представляет собой инкапсуляцию требований к выделению ресурса для реализации рецептурной сущности. Обозначение выделения — это процедурный элемент рецептуры, соответствующий уровню процедурной функциональной диаграммы. Он может быть использован на любом уровне процедурной иерархии.

Обозначение выделения содержит данные и/или логику, определяющую: 1) что выделяется для производства партии изделий (например, конкретная технологическая установка, критерий выбора технологической установки, блока оборудования, материала, физического лица), 2) время выделения ресурса для производства партии изделий (например, через два часа после начала работ по другой процедуре технологической установки). Обозначение выделения ресурса может содержать логику, которой нужно следовать в процессе управления или исполнения процедурных элементов оборудования (не обязательно ассоциированных с имеющимся физическим оборудованием). Явный переход, следующий за обозначением выделения ресурса, может быть использован для описания условия начала работ для нижеследующей рецептурной сущности.

Обозначение выделения ресурса также может быть использовано для описания явного освобождения ресурса. В данном случае, соответствующая текстовая аннотация используется для указания освобождения ресурса.

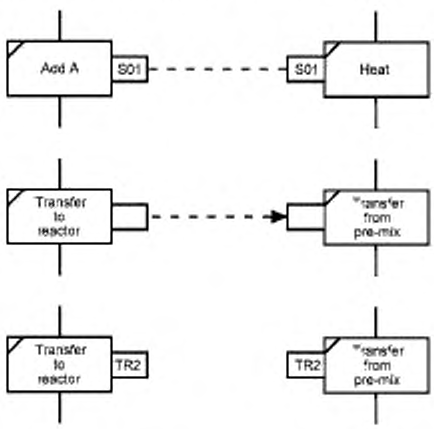


Identifier — идентификатор

Рисунок 25 — Обозначение выделения ресурса

6.1.1.4 Синхронизация элементов

Может оказаться необходимым указание наличия синхронизации рецептурных элементов (см. рисунок 26). Синхронизация (если она есть) описывается прямоугольником блок-схемы с расширением в сторону обозначения любого рецептурного элемента, используемого для синхронизации. Соответствующее обозначение синхронизации может использовать штриховые и прочие линии, отличающиеся от обозначения направленной связи, чтобы восприниматься без путаницы. Если синхронизация представляет собой передачу материала, то используется стрелка для указания планируемого направления движения материала. Если линий нет, то используется уникальный идентификатор, указывающий на конкретное взаимодействие с каждым рецептурным элементом, задействованным в синхронизации.



Add A — добавить компонент а; heat — нагрев; transfer to reactor — передача материала в реактор; transfer from pre-mix — передача после предварительного смешивания

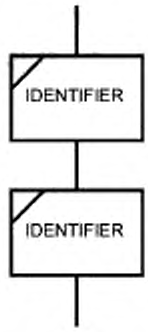
Рисунок 26 — Пример синхронизации элементов

6.1.1.5 Переход рецептуры

Процедурные функциональные диаграммы описывают два типа перехода: неявный и явный переходы.

6.1.1.5.1 Неявный переход

Направленная связь, состоящая из отдельной линии, идущей между рецептурными сущностями (см. рисунок 27), указывает на переход, единственным условием которого является то, что сущности до перехода должны закончить свою работу. Для данного типа перехода дополнительных логических условий не требуется.

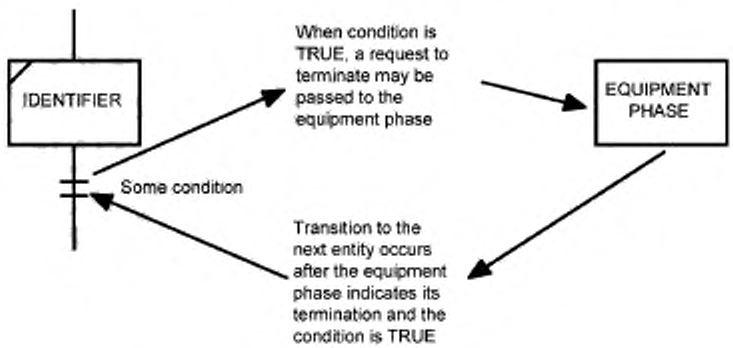


Identifier — идентификатор

Рисунок 27 — Неявный переход

6.1.1.5.2 Явный переход

Направленная связь обозначается отдельной линией, идущей между рецептурными сущностями. Для указания явного перехода рецептуры используются две короткие близко расположенные черты, идущие перпендикулярно линии связи (см. рисунок 28).



Identifier — идентификатор; When condition is TRUE, a request to terminate may be passed to the equipment phase — если условие равно TRUE, то на фазу оборудования может быть передан запрос на прекращение работы; equipment phase — фаза оборудования; Some condition — некоторое условие, transition to the next entity occurs after the equipment phase indicates its termination and the condition is TRUE — переход к следующей сущности имеет место, если произошло прекращение работы фазы оборудования и значение условия равно TRUE

Рисунок 28 — Явный переход

Данный переход определен выражением, равным либо TRUE, либо FALSE. Условие перехода начинает непрерывно оцениваться, как только непосредственно предшествующая сущность активируется.

Переход необходим для реализации двух функций:

a) прерывание выполнения ветви процедурной логики рецептуры;

b) запрос на прекращение работы всех непосредственно предшествующих переходу процедурных элементов (например, процедур технологической установки, операций, фаз).

Прекращение работы непосредственно предшествующих процедурных элементов может быть формулируемым условием. Вместе с тем, условный язык формулировки в настоящем стандарте не рассматривается.

Если условие перехода равно TRUE, то активные процедурные элементы, непосредственно предшествующие переходу, должны прекратить работу. Если непосредственно предшествующие процедурные элементы прекращают работу до того как условие перехода равно TRUE, то функция перехода продолжает оценивать значение логической функции до тех пор, пока оно равно TRUE.

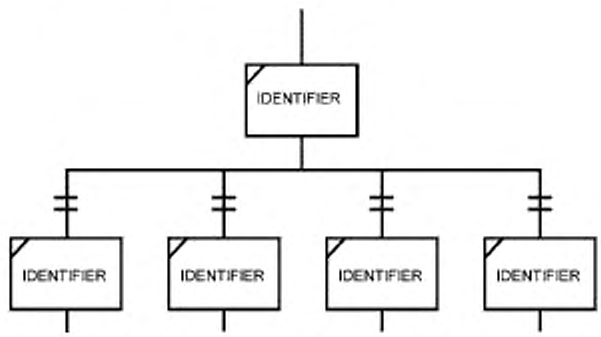
Сущность, непосредственно следующая за переходом, активируется после реализации условия перехода, равного TRUE, после прекращения работы процедурного элемента, предшествующего переходу.

6.1.1.6 Базовые структуры

Данные структуры определяют планируемую последовательность выполнения рецептурных элементов. Простейший случай — это набор процедурных элементов рецептуры, активируемых один за другим. Более сложные структуры включают выбор указанной последовательности выполнения работ и рассмотрение параллельных последовательностей.

6.1.1.6.1 Начало выбора последовательности выполнения работ

Начало выбора последовательности иллюстрируется на рисунке 29. Каждая ветвь выбора последовательности начинается с перехода. Выбор одной из нескольких возможностей представлен совокупностью переходов (например, *под* горизонтальной линией) по числу возможностей. Ниже указанной линии выбирается только одна последовательность из нескольких возможных. Переходы оцениваются приоритетно слева направо. Последовательность, расположенная на блок-схеме ниже перехода, для которой условие перехода становится равным TRUE первым (при оценке с указанным приоритетом), становится искомой «выбранной» последовательностью.

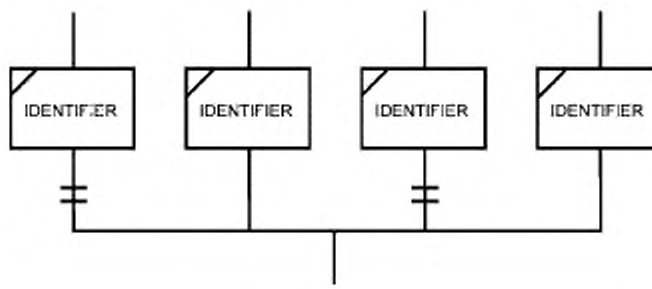


Identifier — идентификатор

Рисунок 29 — Начало выбора последовательности

6.1.1.6.2 Окончание выбора последовательности

Окончание выбора последовательности указывает на возможность присоединения других последовательностей выполнения работ путем их выбора (см. рисунок 30).

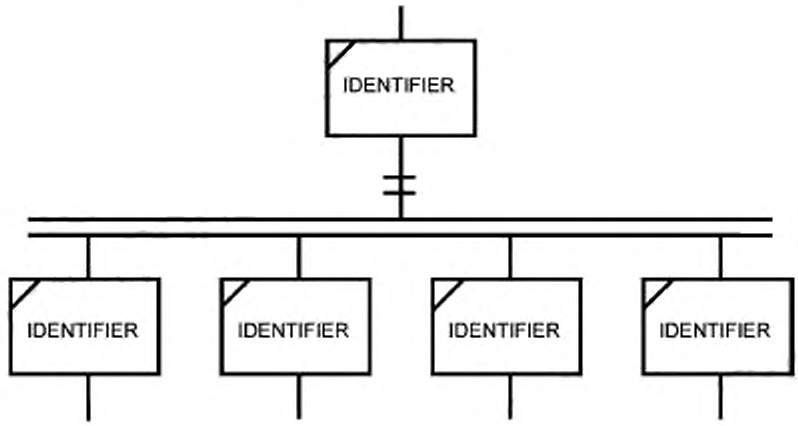


Identifier — идентификатор

Рисунок 30 — Окончание выбора последовательности

6.1.1.6.3 Начало параллельных последовательностей

Начало параллельных последовательностей (см. рисунок 31) указывает начало независимых последовательностей выполнения рецептурных элементов. В начале процедуры выбора имеется одна последовательность выполнения работ для каждого пути. Все возможные последовательности выполнения начинаются от одной последовательности выполнения, определенной в рецептурной сущности. Начало и окончание последовательностей выполнения могут не сочетаться. Если нужен явный переход, то он выполняется над параллельными линиями.

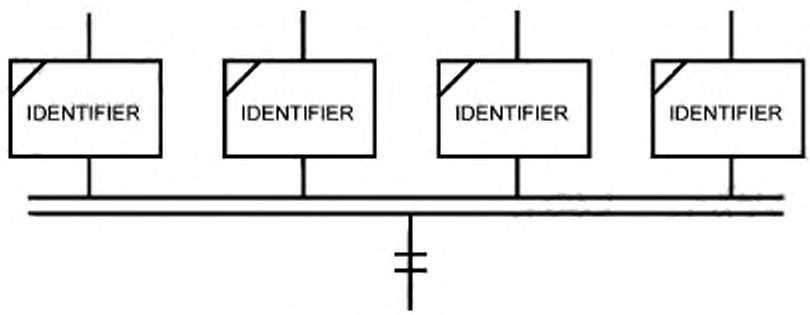


Identifier — идентификатор

Рисунок 31 — Начало параллельной последовательности

6.1.1.6.4 Окончание параллельной последовательности

Окончание параллельных последовательностей указывает порядок сопряжения независимых последовательностей выполнения рецептурного элемента (см. рисунок 32). Переход, следующий непосредственно за параллельными линиями, оценивается только при условии, что все сущности, непосредственно предшествующие рассматриваемым параллельным линиям, либо активны, либо уже закончили работу.

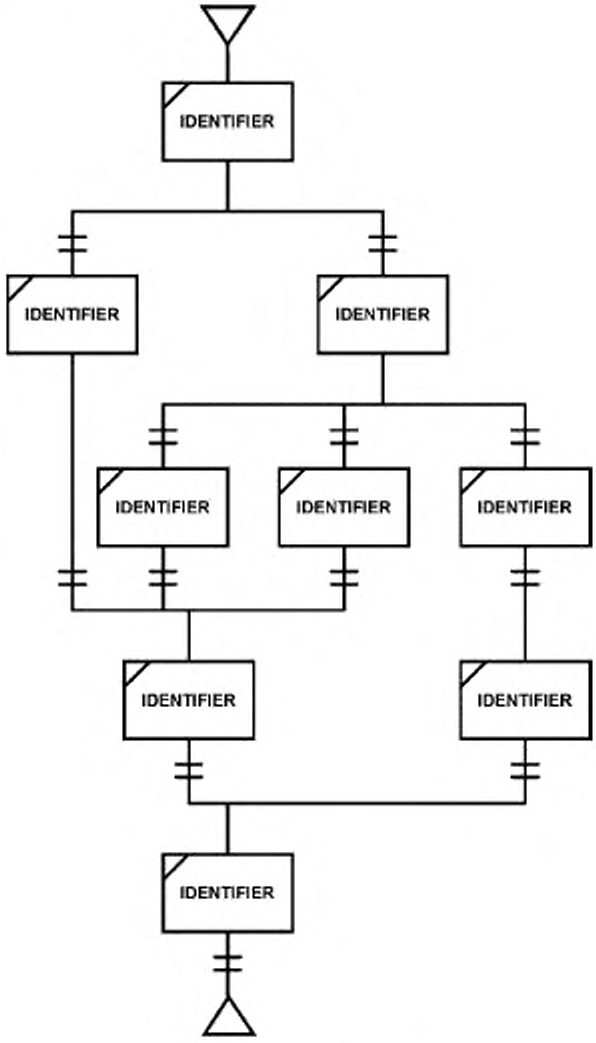


Identifier — идентификатор

Рисунок 32 — Окончание параллельных последовательностей

6.1.1.6.5 Правила для корректных диаграмм

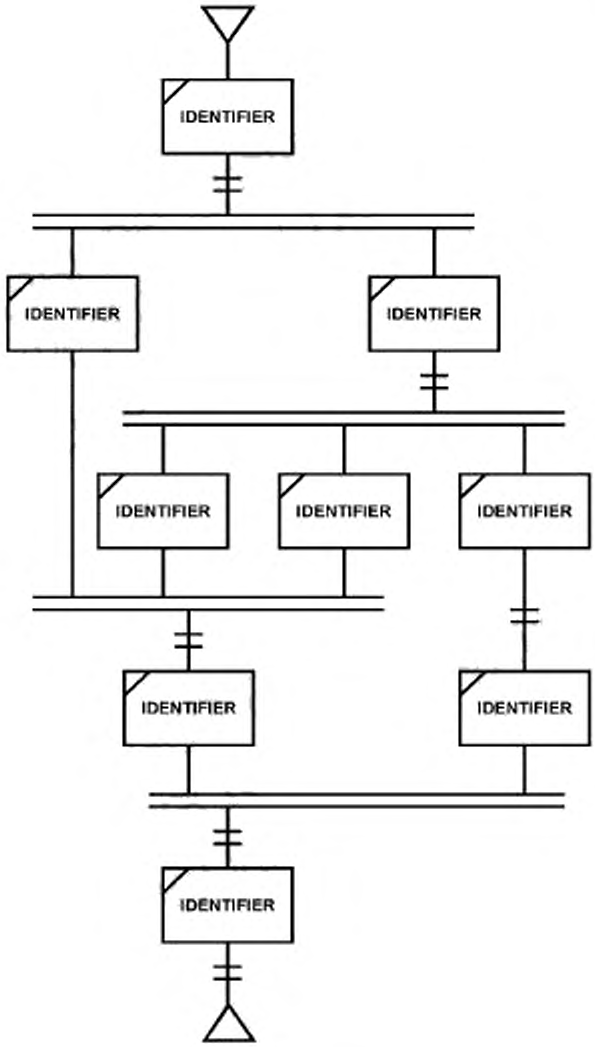
Корректные диаграммы должны отвечать согласованным правилам, установленным для последовательностей выполнения работ. Независимые параллельные последовательности выполнения работ объединяются. Окончание выбора последовательностей не может использоваться для объединения параллельных последовательностей выполнения работ. Рисунок 33 содержит пример корректного сегмента диаграммы с выбором последовательности и окончанием последовательности работ.



Identifier — идентификатор

Рисунок 33 — Диаграмма корректного выбора последовательности

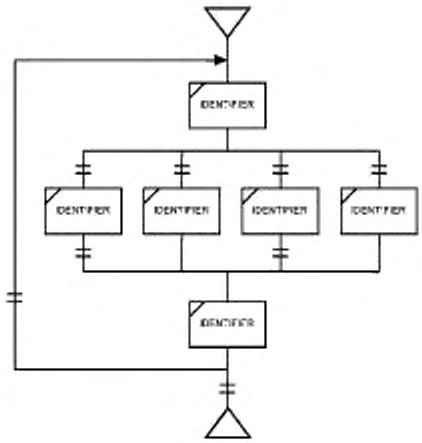
Рисунок 34 содержит пример корректного сегмента диаграммы, указывающего начало работ и окончание параллельной последовательности.



Identifier — идентификатор

Рисунок 34 — Корректная диаграмма параллельной последовательности

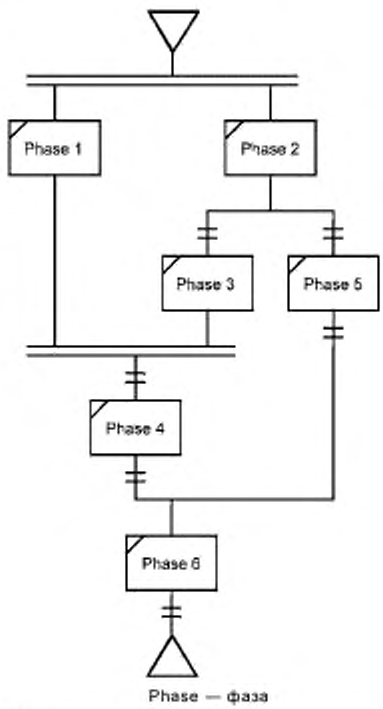
Замыкание блок-схемы обеспечивает повторное выполнение сущностей, основанное на условиях перехода (см. рисунок 35). Замыкание допускает динамическое выполнение сущностей с учетом различных условий.



Identifier — идентификатор

Рисунок 35 — Замыкание с явными процедурными элементами рецептуры

Настоящий стандарт не может определить все корректные и некорректные процедурные функциональные диаграммы. Рассматриваемые PFC-диаграммы могут иметь недостижимые процедурные сущности или сущности с некорректными последовательностями выполнения работ (например, (см. рисунок 36) последовательность с «фазой 1» может никогда не закончиться, если выполняется последовательность с «фазой 5»).

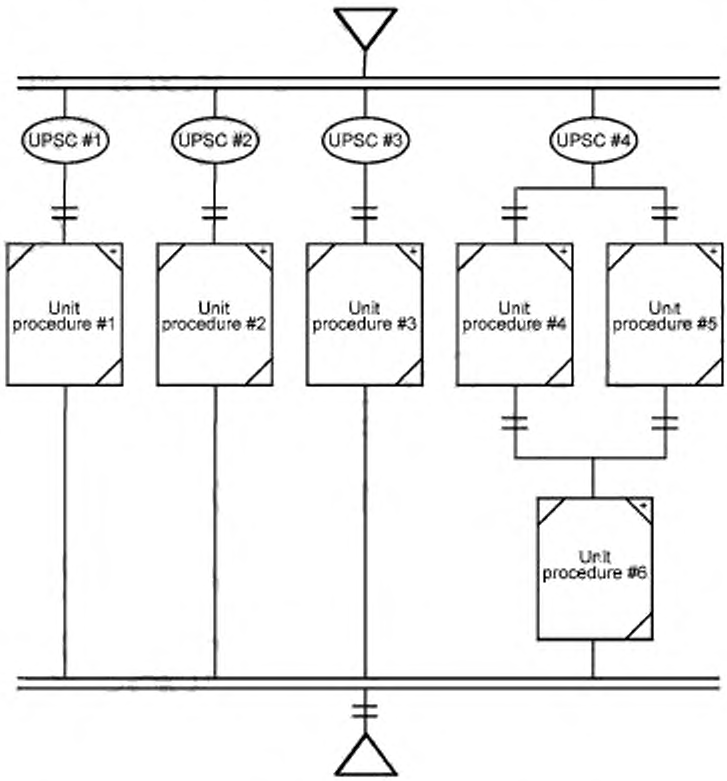


Phase — фаза

Рисунок 36 — Некорректная процедурная функциональная диаграмма

6.1.2 Процедура технологической установки и ее инициализация

Отображение начала рецептурной процедуры приведено на рисунке 37. На каждом уровне, расположенном ниже уровня процедуры технологической установки, направленные соединительные звенья ясно указывают порядок активизации (инициализации) процедурных элементов рецептуры. Инициализация рецептурной процедуры ставится в соответствие требованиям разработки календарного плана. Следовательно, необходимо принимать во внимание правила начала работ, некоторые из которых основываются на календарном плане выполнения работ. Процедура технологической установки активируется после того, как переход (идущий после символа выделения ресурса) принимает значение TRUE.



Unit procedure — процедура технологической установки

Рисунок 37 — Отображение процедуры технологической установки и ее инициализации

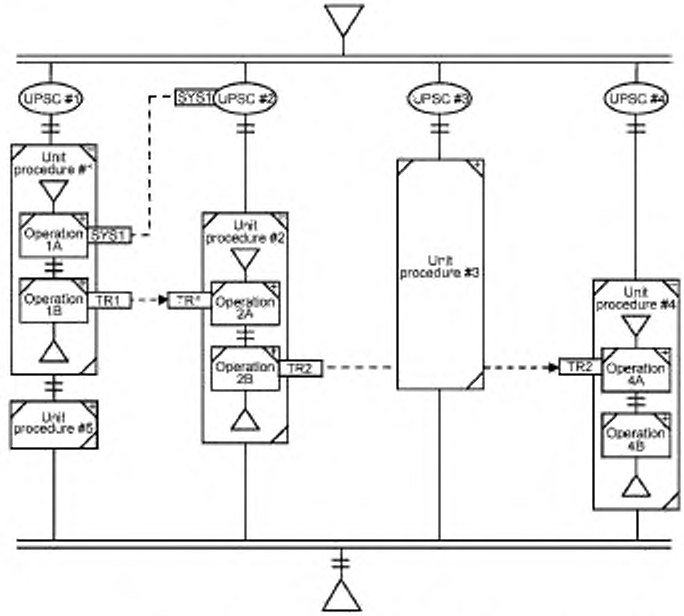
6.1.2.1 Процедура, процедура технологической установки, завершение операции

Если в процедурной функциональной диаграмме достигнут символ окончания работ, то работа инкапсулирующего процедурного элемента завершается.

6.1.2.2 Отношения между процедурными сущностями

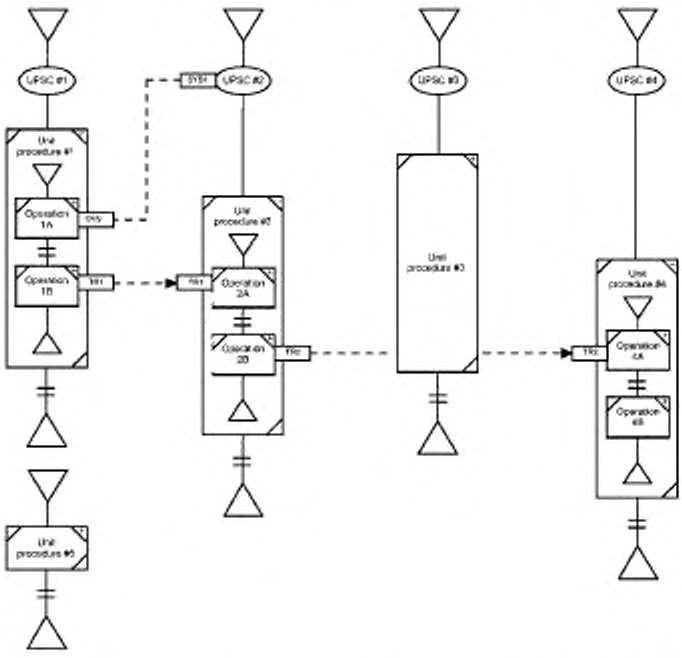
Рисунки 38 и 39 иллюстрируют два метода отображения относительного отношения (relative relationship) между процедурными элементами в процедурных функциональных диаграммах. Данное отношение может быть выполнено путем расположения процедурных элементов вертикально по отношению друг к другу. Вертикальный размер процедурного элемента также может изменяться для корректного отображения относительных отношений между указанными элементами. Горизонтальные штриховые линии обозначают синхронизацию, имеющую место между операциями внутри каждой процедуры технологической установки.

Соответствующая система должна обеспечивать практическую реализацию, по крайней мере, одной из методик, описанных на рисунках 38 и 39.



Unit procedure — процедура технологической установки; operation — операция

Рисунок 38 — Отношения между процедурными сущностями



Unit procedure — процедура технологической установки; operation — операция

Рисунок 39 — Отношения между процедурными сущностями — Альтернатива 1

Если процедурные элементы на двух различных уровнях (например, процедуры технологической установки и операции) представляются на рисунке данного типа, то прямоугольник блок-схемы, указывающий группировку процедурного элемента более высокого уровня (например, процедуры технологической установки), включает и процедурные элементы нижнего уровня (например, операции). Процедура технологической установки #1 (см. рисунок 38) иллюстрирует процедурный элемент высокого уровня, включающий PFC-диаграмму нижнего уровня.

6.1.3 Непроцедурная информация о технологической рецептуре

Прочая информация, являющаяся частью технологической рецептуры, поставлена в соответствие конкретному элементу (обозначению) процедуры технологической рецептуры. Настоящий стандарт не описывает порядок практической реализации соответствующего отношения (связи). В «бумажном» варианте, например, ссылка может быть выполнена аналогично сноске в конец страницы. Информация может расшифровываться параллельно с рассмотрением указанного процедурного элемента. В электронном варианте для реализации процедуры выбора можно использовать «всплывающие окна» и некоторые прочие (пока отсутствующие) механизмы. Вместе с тем, рассматриваемое отношение должно быть четко указано. Оно ставится в соответствие каждому рассматриваемому приложению.

6.1.3.1 Формула технологической рецептуры

Информация о формуле включает входные сигналы, параметры и выходные сигналы технологического процесса.

Информация о формуле должна быть представимой во всей своей целостности (например, должна ассоциироваться с рецептурной процедурой) как для отдельных частей (например, только входного сигнала технологического процесса, только для конкретной процедуры технологической установки), так и для сводного анализа формул нижнего уровня в соответствии с имеющимся контекстом и требованиями приложения. Формула должна ассоциироваться с процедурным элементом рецептуры в соответствии с указанием.

Например, количество продукта, изготавливаемого по рецептуре, может ассоциироваться с конкретной рецептурной процедурой, если количество материала, добавляемого в реактор, может ассоциироваться с конкретной технологической фазой. Использование формул позволяет просуммировать все параметры процедурных элементов рецептуры, идентифицируемые как входные сигналы технологического процесса. Таким образом, может быть сформирован список всех входных сигналов технологического процесса для всех рецептур, процедур технологической установки и операций.

6.1.3.2 Требования технологической рецептуры к оборудованию

Требования к оборудованию в части выполнения процедурных элементов рецептуры могут иметь свои особенности. Для их представления используется метод, позволяющий выделить требования к оборудованию, как ассоциированные индивидуально с каждым процедурным элементом, так и ассоциированные со всеми указанными элементами в совокупности.

6.1.3.3 Заголовок и прочая информация

Категория «информация заголовка и прочая информация» — это категория рецептурной информации, которая может быть поставлена в соответствие как рецептуре в целом (например, идентификатор рецептуры, ее организационно-правовой статус), так и конкретной рецептурной процедурной сущности (например, требования безопасности оборудования, информация о химических угрозах). Вся информация заголовка и прочая информация должна быть представима как в целом, так путем ассоциирования с процедурными сущностями, которым данная информация поставлена в соответствие.

6.2 Отображение рецептуры управления

Если для отображения рецептуры управления используются процедурные функциональные диаграммы, то следует использовать известные принципы, определенные для PFC-диаграмм, отображающих технологические рецептуры. В данном случае, отображение рецептуры управления может превратиться в активное отображение текущей информации в автоматизированных системах. Так как отношение между процедурными элементами рецептуры и процедурными элементами оборудования известно (в ходе выполнения рецептуры управления), то для указания статуса процедурного элемента могут быть использованы цвета и/или прочие обозначения.

6.3 Работа в исключительных ситуациях

Режим работы в исключительных ситуациях может быть включен в рецептуру (для обеспечения технологической безопасности) в дополнение к любой другой логике работы оборудования, используемой в реальных обстоятельствах. Указанные исключительные ситуации, связанные с производством осо-

бых продуктов, обычно ставятся в соответствие имеющемуся методу изготовления продукта и требуемому качеству продукта. Указанные ситуации не ставятся в соответствие конкретному оборудованию. Указанный режим работы «вставляется» в нормальную технологическую процедуру. Его трудно отличить от штатного режима работы. Особые режимы работы активируются только в исключительных ситуациях.

Область применения режима работы в исключительной ситуации (в рецептурной процедуре) обычно ограничивается процедурами технологической установки, так как технологические установки работают независимо. В некоторых случаях, особые команды, состояния и/или режимы распространяются на другие задействованные технологические установки (например, передача материала, его параллельная обработка с общими временными ограничениями). В некоторых обстоятельствах вся рецептура полностью может быть переведена в особое состояние и/или в особый режим работы.

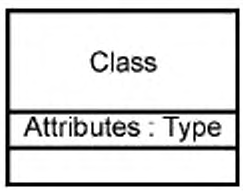
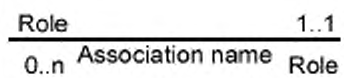
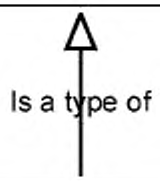
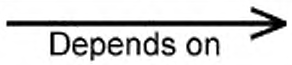
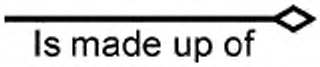
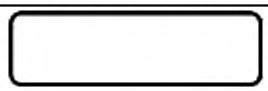
Приложение А
(обязательное)

Методика моделирования данных

А.1 Нотация языка UML

В таблице А.1 приведена нотация языка UML, используемая в настоящем стандарте.

Таблица А.1 — Нотация языка UML

Обозначение	Определение
	Определяет класс объекта. Каждый объект имеет свой тип и свои атрибуты. Каждый объект уникально идентифицируется и нумеруется. Для указанных классов операции или методы здесь не описаны. Если перед атрибутом стоит символ '-', то данный атрибут не является обязательным при любом использовании данного класса
	Это ассоциация между элементами данного класса и элементами другого или этого же класса. Каждая ассоциация идентифицируется. Она может иметь некоторое ожидаемое количество (диапазон) членов подкласса. Число 'n' указывает, что значение не определено (например, 0,n означает, что могут существовать нуль и более членов рассматриваемого подкласса)
	Данное обозначение (стрелка указывает на суперкласс) указывает, что элемент рассматриваемого класса имеет особый тип суперкласса
	Обозначение зависимости (например, интенсивная взаимосвязь между пунктами) указывает, что элемент данного класса зависит от элементов другого класса
	Обозначение агрегации (например, указание на то, что данный элемент состоит из нескольких других элементов) указывает, что элемент данного класса составлен из элементов других классов
	Класс объектов, являющихся реализациями объектов другого класса

А.2 Определения

А.2.1 **класс** (class): Описание множества объектов, имеющих одинаковые атрибуты, поведение, отношения и семантику.

А.2.2 **инкапсуляция** (encapsulation): Методика, позволяющая отделить внешние аспекты объекта от его внутренних аспектов. Описывает подробности практической реализации объекта (другое название – «сокрытие информации»).

А.2.3 **реализация; экземпляр** (instance): Термин, используемый для ссылки на объект, принадлежащий некоторому классу. Сам термин не является классом или подклассом. Например, термин «реактор401» является реализацией класса «реактор».

А.2.4 **модель** (model): Формальное абстрактное представление системы. Модель обычно представляется как совокупность диаграмм и словаря данных.

А.2.5 **объект** (object): Сущность, состоящая из состояния и поведения. Состояние – это значения всех атрибутов в заданный момент времени. Атрибут – это единица информации, определяющая объект. Поведение объекта — это функциональность, содержащаяся в объекте. Она необходима для выполнения манипуляций с атрибутами.

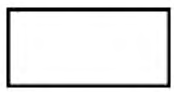
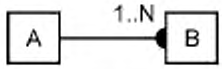
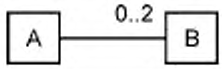
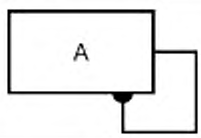
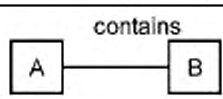
А.2.6 **подкласс** (subclass): Это класс, представляющий собой частный случай более общего класса (например, «реактор со стеклянной облицовкой» - это подкласс класса «реакторов»).

А.2.7 **универсальный язык моделирования; UML** (unified modeling language (UML)): Язык, используемый для описания, визуализации, конструирования и документирования элементов программного обеспечения систем. Используется как для коммерческого моделирования, так и для систем без программного обеспечения.

А.3 Нотация диаграммы отношений сущностей, ERD

Таблица А.2 содержит нотацию ERD, используемую в настоящем стандарте.

Таблица А.2 — Нотация ERD

Обозначение	Определение
	Определяет сущность
	Для каждого события А имеет место одно и только одно событие В. Ассоциация с В может быть обозначена № 1.
	Численно определенная ассоциация. В данном примере, для каждого события А может иметь место одно или несколько событий В. Другим примером является обозначение 0..N. Если числовая ассоциация отсутствует, то имеет место вариант 0..N
	Численно определенная ассоциация: от 0 до некоторого положительного значения. В данном примере, для каждого события А имеет место от 0 до 2 событий В
	Зацикленная ассоциация. Для каждого события А может иметь место нуль и более событий для сущностей того же типа. Ассоциация является необязательной, если событие для сущности А включает нуль и более событий для сущностей того же типа.
	Ассоциация между сущностями помечена для указания природы имеющегося отношения. Метка относится к ближайшей сущности. Ассоциация считывается следующим образом: каждое событие А содержит одно или несколько событий В

Приложение В
(обязательное)

Листинг определений Языка структурированных запросов SQL

Данное приложение содержит примеры составления текстов программ для формирования таблиц (в соответствии с ИСО/МЭК 9075:1992), приведенных в разделе 5.

```
CREATE TABLE BXT_Exchange (
        ExchangeID                      CHAR (32)       NOT NULL,
        ExchangeValue                   CHAR (128)      NOT NULL,
        PRIMARY KEY (ExchangeID))

INSERT INTO BXT_Exchange (ExchangeID, ExchangeValue)
        VALUES ('Schema', 'IEC 61512-2 2001')
        VALUES ('Delimiter', '/')
        VALUES ('ToolID', 'ToolName')
        VALUES ('ToolVersion', '4.0')
        VALUES ('ToolSchema', '1.2')

CREATE TABLE BXT_EnumerationSet (
        EnumSet                         CHAR (32)       NOT NULL,
        Description                     CHAR (255),
        PRIMARY KEY (EnumSet))

CREATE TABLE BXT_Enumeration (
        EnumSet                         CHAR (32)       NOT NULL,
        EnumValue                       INTEGER         NOT NULL,
        EnumString                      CHAR (32),
        Description                     CHAR (255),
        PRIMARY KEY (EnumSet, EnumValue))

INSERT INTO BXT_EnumerationSet (EnumSet, Description)
        VALUES ('Boolean', 'Defines a set of Boolean values')
        VALUES ('DirectionType',
                'Defines how a parameter is intended to be handled')
        VALUES ('EquipmentLevel',
                'Defines the equipment hierarchical level for equipment elements')
        VALUES ('EquipmentType',
                'Defines the type of equipment record for equipment elements')
        VALUES ('EvaluationRule',
                'Defines the evaluation rules for equipment properties')
        VALUES ('FormulaSubType', 'Defines the recipe formula types')
        VALUES ('FormulaType', 'User supplied formula sub type definitions')
        VALUES ('LinkDepiction',
                'Defines how links between recipe elements are to be depicted')
        VALUES ('LinkToType', 'Defines if a link references a step or a transition')
        VALUES ('LinkType', 'Defines the type of link')
        VALUES ('RE_Type',
                'Defines the recipe element, either recipe procedure level or allocation entity')
        VALUES ('RE_Use', 'Defines how a recipe element is used in a recipe')
        VALUES ('RecipeStatus', 'Defines the possible status of a recipe'
        VALUES ('RecordSet',
                'Defines the enumeration set used to classify a record into a category of batch history information.')
        VALUES ('RecordSetControlRecipe',
                'Provides further history record classification under the category of ControlRecipe. ')
      VALUES ('RecordSetMasterRecipe',
                'Provides further history record classification under the category of MasterRecipe.')
      VALUES ('RecordSetExecutionInfo',
```

```
                'Provides further history record classification under the category of ExecutionInfo. ')
        VALUES ('RecordSetMaterialInfo',
                'Provides further history record classification under the category of MaterialInfo. ')
        VALUES ('RecordSetContinuousData',
                'Provides further history record classification under the category of ContinuousData. ')
        VALUES ('RecordSetEvents',
                'Provides further history record classification under the category of Events. ')
        VALUES ('RecordSetOperatorChange',
                'Provides further history record classification under the category of OperatorChange. ')
        VALUES ('RecordSetOperatorComment',
                'Provides further history record classification under the category of OperatorComment.')
        VALUES ('RecordSetAnalysisData',
                'Provides further history record classification under the category of AnalysisData.')
        VALUES ('RecordSetLateRecord',
                'Provides further history record classification under the category of LateRecord.')
        VALUES ('RecordSetRecipeData',
                'Provides further history record classification under the category of RecipeData. ')
        VALUES ('RecordSetRecipeSpecified',
                'Provides further history record classification under the category of RecipeSpecified.')
        VALUES (.RecordSetSummaryData.,
                'Provides further history record classification under the category of SummaryData. ')
        VALUES ('ScheduleAction', 'Defines the intended action of the schedule entry ')
        VALUES ('ScheduleMode',
                'Defines the mode which the schedule entry begins execution in ')
        VALUES ('ScheduleStatus', 'Defines the possible status of a schedule')
        VALUES ('SE_Type',
                'Defines the type of entity in a schedule record')
        VALUES ('ValueDataType',
                'Defines how a value is represented (for example Boolean, float, etc.) ')
        VALUES ('ValueType', 'Defines how a value string is interpreted')
INSERT INTO BXT_Enumeration (EnumSet, EnumValue, EnumString, Description)
        VALUES ('Boolean', 0, 'FALSE', 'Defines a Boolean value')
        VALUES ('Boolean', 1, 'TRUE., '')
        VALUES ('DirectionType', 0, 'Invalid', 'Entry not valid')
        VALUES ('DirectionType', 1, 'Internal',
                'Identifies how a parameter is handled.')
        VALUES ('DirectionType', 2, 'Input',
                'The Recipe Element receives the value from an external source. ')
        VALUES ('DirectionType', 3, 'Output',
                'The Recipe Element creates the value and makes it available for external use. ')
        VALUES ('DirectionType', 4, 'Input/Output',
                'The Recipe Element and external element exchange the value, and may change its value.')
        VALUES ('EquipmentLevel', 0, 'Invalid', 'Entry not valid')
        VALUES ('EquipmentLevel', 1, 'Enterprise',
                'Identifies the equipment hierarchical level for BXT_EquipElement')
        VALUES ('EquipmentLevel', 2, 'Site', '')
        VALUES ('EquipmentLevel', 3, 'Area', '')
        VALUES ('EquipmentLevel', 4, 'Process Cell', '')
        VALUES ('EquipmentLevel', 5, 'Unit', '')
        VALUES ('EquipmentLevel', 6, 'Equipment Module', '')
        VALUES ('EquipmentLevel', 7, 'Control Module', '')
        VALUES ('EquipmentType', 0, 'Invalid', 'Entry not valid')
        VALUES ('EquipmentType', 1, 'Class',
                'Identifies the record type for BXT_EquipElement')
        VALUES ('EquipmentType', 2, 'Element', '')
        VALUES ('EvaluationRule', 0, 'Invalid', 'Entry not valid')
        VALUES ('EvaluationRule', 1, '=',
                'Equals comparison operator for equipment properties')
        VALUES ('EvaluationRule', 2, '<>',
                'Not equals comparison operator for equipment properties')
        VALUES ('EvaluationRule', 3, '<',
```

```
        'Less than comparison operator for equipment properties')
VALUES ('EvaluationRule', 4, '>',
        'Greater than comparison operator for equipment properties')
VALUES ('EvaluationRule', 5, '<=',
        'Less than or equals comparison operator for equipment properties')
VALUES ('EvaluationRule', 6, '>=',
        'Greater than or equals comparison operator for equipment properties')
VALUES ('EvaluationRule', 7, 'Member',
        'Is a member of comparison operator for equipment properties')
VALUES ('EvaluationRule', 8, 'Not member',
        .Is not a member of comparison operator for equipment properties.)
VALUES ('EvaluationRule', 9, 'Not',
        'Not comparison operator for equipment properties')
VALUES ('FormulaType', 0, 'Invalid', 'Entry not valid')
VALUES ('FormulaType', 1, 'Process Input', 'Recipe Formula type')
VALUES ('FormulaType', 2, 'Process Output', '')
VALUES ('FormulaType', 3, 'Process Parameter', '')
VALUES ('FormulaSubType', 0, 'Invalid', 'Entry not valid')
VALUES ('LinkDepiction', 0, 'Invalid', 'Entry not valid')
VALUES ('LinkDepiction', 1, 'None', 'No link depiction')
VALUES ('LinkDepiction', 2, 'Line', 'Link shown with line only')
VALUES ('LinkDepiction', 3, 'ID', 'Link shown with identifier only')
VALUES ('LinkDepiction', 4, 'Line & ID',
        'Link shown with line and identification')
VALUES ('LinkDepiction', 5, 'Line & Arrow',
        'Link shown with line and material flow arrow')
VALUES ('LinkDepiction', 6, 'Line, Arrow, & ID',
        'Link shown with line, material flow arrow and identification')
VALUES ('LinkToType', 0, 'Invalid', 'Entry not valid')
VALUES ('LinkToType', 1, 'Recipe Element',
        'Link is referencing an entry in the BXT_MRecipeElement table')
VALUES ('LinkToType', 2, 'Transition',
        'Link is referencing an entry in the BXT_MRecipeTransition table')
VALUES ('LinkType', 0, 'Invalid', 'Entry not valid')
VALUES ('LinkType', 1, 'ControlLink',
        'Defines a link between recipe elements that indicates a flow of procedural control'')
VALUES ('LinkType', 2, 'TransferLink',
        'Defines a link between recipe elements that indicates a material transfer'')
VALUES ('LinkType', 3, 'SynchronizationLink',
        'Defines a link between recipe elements where there is some form of synchronization'')
VALUES ('RE_Type', 0, 'Invalid', 'Entry not valid')
VALUES ('RE_Type', 1, 'Master Recipe',
        'Specifies the type of recipe element'')
VALUES ('RE_Type', 2, 'Procedure', '')
VALUES ('RE_Type', 3, 'Unit Procedure', '')
VALUES ('RE_Type', 4, 'Operation', '')
VALUES ('RE_Type', 5, 'Phase', '')
VALUES ('RE_Type', 6, 'Allocation', '')
VALUES ('RE_Type', 7, 'Begin', '')
VALUES ('RE_Type', 8, 'End', '')
VALUES ('RE_Type', 9, 'Start Parallel', '')
VALUES ('RE_Type', 10, 'End Parallel', '')
VALUES ('RE_Type', 11, 'Start Branch', '')
VALUES ('RE_Type', 12, 'End Branch', '')
VALUES ('RE_Use', 0, 'Invalid', 'Entry not valid')
VALUES ('RE_Use', 1, 'Linked',
        'A recipe element (RE) may have several referencing RE Steps'')
VALUES ('RE_Use', 2, 'Embedded',
        'An RE has only one referencing RE, one RE is defined for each use of the RE'')
VALUES ('RE_Use', 3, 'Copied',
        'The same as Embedded, but the specific RE was modified from its original definition'')
```

```
VALUES ('RecipeStatus', 0, 'Invalid', 'Entry not valid')
VALUES ('RecipeStatus', 1, 'Approved for Production',
        'Recipe was approved for production'')
VALUES ('RecipeStatus', 2, 'Approved for Test',
        'Recipe was only approved for test'')
VALUES ('RecipeStatus', 3, 'Not Approved',
        'Recipe was not approved for production or test'')
VALUES ('RecipeStatus', 4, 'Inactive', 'Recipe was not active'')
VALUES ('RecipeStatus', 5, 'Obsolete', 'Recipe was obsolete')
VALUES ('RecordSet', 0, 'Invalid', 'Entry not valid')
VALUES ('RecordSet', 1, 'RecordSetControlRecipe',
        'Defines that a batch history information record is part of the ControlRecipe category'')
VALUES ('RecordSet', 2, 'RecordSetMasterRecipe', '')
VALUES ('RecordSet', 3, 'RecordSetExecutionInfo', '')
VALUES ('RecordSet', 4, 'RecordSetMaterialInfo', '')
VALUES ('RecordSet', 5, 'RecordSetContinuousData', '')
VALUES ('RecordSet', 6, 'RecordSetEvents', '')
VALUES ('RecordSet', 7, 'RecordSetOperatorChange', '')
VALUES ('RecordSet', 8, 'RecordSetOperatorComment', '')
VALUES ('RecordSet', 9, 'RecordSetAnalysisData', '')
VALUES ('RecordSet', 10, 'RecordSetLateRecord', '')
VALUES ('RecordSet', 11, 'RecordSetRecipeData', '')
VALUES ('RecordSet', 12, 'RecordSetRecipeSpecified', '')
VALUES ('RecordSet', 13, 'RecordSetSummaryData', '')
VALUES ('RecordSetControlRecipe', 0, 'Invalid', 'Entry not valid')
VALUES ('RecordSetControlRecipe', 1, 'Entire Control Recipe',
        'History record is related to the entire control recipe'')
VALUES ('RecordSetMasterRecipe', 0, 'Invalid', 'Entry not valid')
VALUES ('RecordSetMasterRecipe', 1, 'Entire Master Recipe',
        'History record is related to the entire master recipe'')
VALUES ('RecordSetExecutionInfo', 0, 'Invalid', 'Entry not valid')
VALUES ('RecordSetExecutionInfo', 1, 'Allocation', '')
VALUES ('RecordSetExecutionInfo', 2, 'De-allocation', '')
VALUES ('RecordSetExecutionInfo', 3, 'State Change', '')
VALUES ('RecordSetExecutionInfo', 4, 'State Command', '')
VALUES ('RecordSetExecutionInfo', 5, 'Mode Change', '')
VALUES ('RecordSetExecutionInfo', 6, 'Mode Command', '')
VALUES ('RecordSetExecutionInfo', 7, 'Procedural Entity Message', '')
VALUES ('RecordSetExecutionInfo', 8, 'Procedural Entity Alarm', '')
VALUES ('RecordSetExecutionInfo', 9, 'Procedural Entity Version', '')
VALUES ('RecordSetExecutionInfo', 10, 'Procedural Entity Prompt', '')
VALUES ('RecordSetExecutionInfo', 11, 'Procedural Entity Prompt Response', '')
VALUES ('RecordSetMaterialInfo', 0, 'Invalid', 'Entry not valid')
VALUES ('RecordSetMaterialInfo', 1, 'Material Consumption', '')
VALUES ('RecordSetMaterialInfo', 2, 'Material Production', '')
VALUES ('RecordSetMaterialInfo', 3, 'Material Allocation', '')
VALUES ('RecordSetMaterialInfo', 4, 'Material De-allocation', '')
VALUES ('RecordSetContinuousData', 0, 'Invalid', 'Entry not valid')
VALUES ('RecordSetContinuousData', 1, 'Continuous Data Value', '')
VALUES ('RecordSetContinuousData', 2, 'Trend Association', '')
VALUES ('RecordSetContinuousData', 3, 'Trend Disassociation', '')
VALUES ('RecordSetEvents', 0, 'Invalid', 'Entry not valid')
VALUES ('RecordSetEvents', 1, 'General Event', '')
VALUES ('RecordSetOperatorChange', 0, 'Invalid', 'Entry not valid')
VALUES ('RecordSetOperatorChange', 1, 'General Operator Intervention', '')
VALUES ('RecordSetOperatorComment', 0, 'Invalid', 'Entry not valid')
VALUES ('RecordSetOperatorComment', 1, 'General Operator Comment', '')
VALUES ('RecordSetAnalysisData', 0, 'Invalid', 'Entry not valid')
VALUES ('RecordSetAnalysisData', 1, 'General Analysis Message', '')
VALUES ('RecordSetLateRecord', 0, 'Invalid', 'Entry not valid')
VALUES ('RecordSetLateRecord', 1, 'General Late Record', '')
```

```
VALUES ('RecordSetRecipeData', 0, 'Invalid', 'Entry not valid')
VALUES ('RecordSetRecipeData', 1, 'Generic Recipe Data', '')
VALUES ('RecordSetRecipeData', 2, 'Recipe Parameter Value Change', '')
VALUES ('RecordSetRecipeData', 3, 'Recipe Result Data', '')
VALUES ('RecordSetRecipeSpecified', 0, 'Invalid', 'Entry not valid')
VALUES ('RecordSetRecipeSpecified', 1, 'Generic Recipe Specified Data', '')
VALUES ('RecordSetSummaryData', 0, 'Invalid', 'Entry not valid')
VALUES ('RecordSetSummaryData', 1, 'Generic Summary Data', '')
VALUES ('RecordSetSummaryData', 2, 'Utilities Consumption', '')
VALUES ('RecordSetSummaryData', 3, 'Equipment Run Time', '')
VALUES ('ScheduleChange', 0, 'Invalid', 'Entry not valid')
VALUES ('ScheduleChange', 1, 'New', 'Schedule record change action', '')
VALUES ('ScheduleChange', 2, 'Update', '')
VALUES ('ScheduleChange', 3, 'Delete', '')
VALUES ('ScheduleMode', 0, 'Invalid', 'Entry not valid')
VALUES ('ScheduleMode', 1, 'Automatic', 'Schedule record mode')
VALUES ('ScheduleMode', 2, 'Semi-Automatic', '')
VALUES ('ScheduleMode', 3, 'Manual', '')
VALUES ('ScheduleMode', 4, 'Not Specified', '')
VALUES ('ScheduleStatus', 0, 'Invalid', 'Entry not valid')
VALUES ('ScheduleStatus', 1, 'Complete', 'Batch schedule record status')
VALUES ('ScheduleStatus', 2, 'In-progress', '')
VALUES ('ScheduleStatus', 3, 'Scheduled', '')
VALUES ('ScheduleStatus', 4, 'Schedule Hold', '')
VALUES ('ScheduleStatus', 5, 'Not Specified', '')
VALUES ('SE_Type', 0, 'Invalid', 'Entry not valid')
VALUES ('SE_Type ', 1, 'Campaign',
        ' Defines the type of Scheduled Entry')
VALUES ('SE_Type ', 2, 'Batch', '')
VALUES ('SE_Type ', 3, 'Unit Procedure', '')
VALUES ('SE_Type ', 4, 'Operation', '')
VALUES ('SE_Type ', 5, 'Phase', '')
VALUES ('ValueDataType', 0, 'Invalid', 'Entry not valid')
VALUES ('ValueDataType', 1, 'Boolean',
        'Defines the data type that is expected for an associated value'')
VALUES ('ValueDataType', 2, '8-Bit string', '')
VALUES ('ValueDataType', 3, '16-Bit string', '')
VALUES ('ValueDataType', 4, '32-Bit string', '')
VALUES ('ValueDataType', 5, '8-Bit unsigned integer', '')
VALUES ('ValueDataType', 6, '16-Bit unsigned integer', '')
VALUES ('ValueDataType', 7, '32-Bit unsigned integer', '')
VALUES ('ValueDataType', 8, '8-Bit signed integer', '')
VALUES ('ValueDataType', 9, '16-Bit signed integer', '')
VALUES ('ValueDataType', 10, '32-Bit signed integer', '')
VALUES ('ValueDataType', 11, '32-Bit float', '')
VALUES ('ValueDataType', 12, 'Double float', '')
VALUES ('ValueDataType', 13, 'Octet string', '')
VALUES ('ValueDataType', 14, 'DateTime', '')
VALUES ('ValueType', 0, 'Invalid', 'Entry not valid')
VALUES ('ValueType', 1, 'Constant',
        'Defines how a value string is interpreted' It contains a fixed value as a string'')
VALUES ('ValueType', 2, 'Reference',
        'Defines how a value string is interpreted' It points to the source of the value'')
VALUES ('ValueType', 3, 'Equation',
        'Defines that a value string is interpreted as an expression to be evaluated in order to
        determine the value'')
VALUES ('ValueType', 4, 'External',
        'Value is supplied by some external means, and it is not contained in the recipe (i'e', value
        may be supplied by an operator entry or by a scheduling system)'')
```

```
CREATE TABLE BXT_MRecipeElement (

        RE_ID                   CHAR (128)          NOT NULL,
        REVersion                       CHAR (16)   NOT NULL,
        VersionDate                     DATETIME,
        ApprovalDate            DATETIME,
        EffectiveDate           DATETIME,
        ExpirationDate          DATETIME,
        Author                  CHAR (32),
        ApprovedBy                      CHAR (32),
        ProcessCellID           CHAR (32),
        ProductID                       CHAR (32),
        UsageConstraint         CHAR (255),
        Description                     CHAR (255),
        Status                          INTEGER,
        RE_Type                         INTEGER,
        RE_Function                     CHAR (255),
        RE_Use                  INTEGER,
        DerivedRE                       CHAR (128),
        DerivedVersion          CHAR (16),
        PRIMARY KEY (RE_ID, REVersion))

CREATE TABLE BXT_MRecipeStep (
        ParentRE                        CHAR (128)  NOT NULL,
        ParentVersion           CHAR (16)           NOT NULL,
        StepID                  CHAR (128)          NOT NULL,
        RE_ID                   CHAR (128)          NOT NULL,
        REVersion                       CHAR (16)   NOT NULL,
        VerticalStart                   FLOAT,
        VerticalStop                    FLOAT,
        HorizontalStart         FLOAT,
        HorizontalStop          FLOAT,
        ScaleReference          FLOAT,
        ScaleEngrUnits          CHAR (32),
        MaximumScale            FLOAT,
        MinimumScale            FLOAT,
        PRIMARY KEY (ParentRE, ParentVersion, StepID),
        FOREIGN KEY (RE_ID, REVersion)
            REFERENCES BXT_MRecipeElement (RE_ID, REVersion))

CREATE TABLE BXT_MRecipeTransition (
        RE_ID                   CHAR (128)          NOT NULL,
        REVersion                       CHAR (16)   NOT NULL,
        TransitionID                    CHAR (128)  NOT NULL,
        Condition                       CHAR (255),
        VerticalStart                   FLOAT,
        VerticalStop                    FLOAT,
        HorizontalStart         FLOAT,
        HorizontalStop          FLOAT,
        PRIMARY KEY (RE_ID, REVersion, TransitionID),
        FOREIGN KEY (RE_ID, REVersion)
            REFERENCES BXT_MRecipeElement (RE_ID, REVersion))

CREATE TABLE BXT_MRecipeLink (
        RE_ID                   CHAR (128)          NOT NULL,
        REVersion                       CHAR (16)   NOT NULL,
        LinkID                          CHAR (32)   NOT NULL,
        FromType                        INTEGER,
        FromElement             CHAR (128),
```

```
        ToType                  INTEGER,
        ToElement                       CHAR (128),
        LinkType                INTEGER,
        VerticalStart                   FLOAT,
        VerticalStop                    FLOAT,
        HorizontalStart         FLOAT,
        HorizontalStop          FLOAT,
        Depiction                       INTEGER,
        EvaluationOrder         INTEGER,
        PRIMARY KEY (RE_ID, REVersion, LinkID),
        FOREIGN KEY (RE_ID, REVersion)
            REFERENCES BXT_MRecipeElement (RE_ID, REVersion))

CREATE TABLE BXT_MRecipeElementParameter (
        RE_ID                   CHAR (128)              NOT NULL,
        REVersion                       CHAR (16)       NOT NULL,
        ParameterID                     CHAR(32)        NOT NULL,
        ParentParamID           CHAR(32),
        DataInterpretation              INTEGER,
        DataDirection           INTEGER,
        DefaultValue                    CHAR(128),
        Description                     CHAR(255),
        EngrUnits                       CHAR(32),
        EnumSet                         CHAR (32),
        DefaultScaling          INTEGER,
        ParamType                       INTEGER,
        ParamSubType            INTEGER,
        PRIMARY KEY (RE_ID, REVersion, ParameterID),
        FOREIGN KEY (RE_ID, REVersion)
            REFERENCES BXT_MRecipeElement (RE_ID, REVersion))

CREATE TABLE BXT_MRecipeStepParameter (
        ParentRE                        CHAR (128)      NOT NULL,
        ParentVersion           CHAR (16)               NOT NULL,
        StepID                  CHAR (128)              NOT NULL,
        ParameterID                     CHAR (32)       NOT NULL,
        ParentParamID           CHAR (32),
        ParameterValue          CHAR (128),
        DataInterpretation              INTEGER,
        Scaled                  INTEGER,
        PRIMARY KEY (ParentRE, ParentVersion, StepID, ParameterID),
        FOREIGN KEY (ParentRE, ParentVersion, StepID)
            REFERENCES BXT_MRecipeStep (ParentRE, ParentVersion, StepID))

CREATE TABLE BXT_MRecipeOtherInformation (
        RE_ID                   CHAR (128)              NOT NULL,
        REVersion                       CHAR (16)       NOT NULL,
        StepID                  CHAR (128)              NOT NULL,
        DataID                  CHAR (32)               NOT NULL,
        DataType                        CHAR (32),
        DataValue                       CHAR (255),
        Description                     CHAR (255),
        PRIMARY KEY (RE_ID, REVersion, DataID),
        FOREIGN KEY (RE_ID, REVersion)
            REFERENCES BXT_MRecipeElement (RE_ID, REVersion))

CREATE TABLE BXT_MRecipeElementEquip (
        RE_ID                   CHAR (128)              NOT NULL,
        REVersion                       CHAR (16)       NOT NULL,
```

```
        PropertyID                  CHAR (32)       NOT NULL,
        DefaultValue                CHAR (128),
        DataInterpretation          INTEGER,
        EvaluationRule      INTEGER,
        EngrUnits                   CHAR (32),
        Description                 CHAR (255),
        PRIMARY KEY (RE_ID, REVersion, PropertyID),
        FOREIGN KEY (RE_ID, REVersion)
            REFERENCES BXT_MRecipeElement)

CREATE TABLE BXT_MRecipeStepEquip (
        ParentRE                    CHAR (128)      NOT NULL,
        ParentVersion       CHAR (16)               NOT NULL,
        StepID              CHAR (128)              NOT NULL,
        PropertyID                  CHAR (32)       NOT NULL,
        PropertyValue       CHAR (128),
        PRIMARY KEY (ParentRE, ParentVersion, StepID, PropertyID),
        FOREIGN KEY (ParentRE, ParentVersion, StepID)
            REFERENCES BXT_MRecipeStep (ParentRE, ParentVersion, StepID))

CREATE TABLE BXT_EquipElement (
        EquipmentID                 CHAR (32)       NOT NULL,
        EE_Type                     INTEGER,
        EE_Level                    INTEGER,
        ContainedIn                 CHAR (32),
        Description                 CHAR (255),
        PRIMARY KEY (EquipmentID))

CREATE TABLE BXT_EquipLink (
        EquipmentID                 CHAR (32)       NOT NULL,
        ToEquipmentID       CHAR (32)               NOT NULL,
        Description                 CHAR (255),
        PRIMARY KEY (EquipmentID , ToEquipmentID),
        FOREIGN KEY (EquipmentID)
            REFERENCES BXT_EquipElement,
        FOREIGN KEY (ToEquipmentID)
            REFERENCES BXT_EquipElement)

CREATE TABLE BXT_EquipInclude (
        EquipmentID                 CHAR (32)       NOT NULL,
        ClassEquipmentID            CHAR (32)       NOT NULL,
        Description                 CHAR (255),
        PRIMARY KEY (EquipmentID, ClassEquipmentID),
        FOREIGN KEY (EquipmentID)
            REFERENCES BXT_EquipElement,
        FOREIGN KEY (ClassEquipmentID)
            REFERENCES BXT_EquipElement)

CREATE TABLE BXT_EquipProperty (
        EquipmentID                 CHAR (32)       NOT NULL,
        PropertyID                  CHAR (32)       NOT NULL,
        PropertyValue       CHAR (255),
        EngrUnits                   CHAR (32),
        Description                 CHAR (255),
        PRIMARY KEY (EquipmentID, PropertyID),
        FOREIGN KEY (EquipmentID)
            REFERENCES BXT_EquipElement)
CREATE TABLE BXT_EquipInterface (
        EquipmentID                 CHAR (32)       NOT NULL,
```

```
    EPI_ID                  CHAR (32)           NOT NULL,
    EPI_Definition          CHAR (32)           NOT NULL,
    Description                 CHAR (255),
    PRIMARY KEY (EPI_ID, EquipmentID),
    FOREIGN KEY (EquipmentID)
        REFERENCES BXT_EquipElement)

CREATE TABLE BXT_EquipInterfaceDefinition (
    EPI_Definition          CHAR (32)           NOT NULL,
    Description                 CHAR (255),
    PRIMARY KEY (EPI_Definition))

CREATE TABLE BXT_EquipInterfaceParameter (
    EPI_Definition          CHAR (32)           NOT NULL,
    ParameterID                 CHAR (32)       NOT NULL,
    ParentParamID           CHAR (32),
    Type                        INTEGER         NOT NULL,
    EngrUnits                   CHAR (32),
    EnumSet                     CHAR (32),
    Scaled                  INTEGER,
    DefaultValue                CHAR (128),
    Description                 CHAR (255),
    PRIMARY KEY (EPI_Definition, ParameterID),
    FOREIGN KEY (EPI_Definition)
        REFERENCES BXT_EquipInterfaceDefinition)

CREATE TABLE BXT_ScheduleEntry (
    ScheduleEntryID         CHAR (64)           NOT NULL,
    ParentSchedID           CHAR (64),
    ExternalID                  CHAR (64),
    RE_ID                   CHAR (128),
    REVersion                   CHAR (16),
    SE_Type                 INTEGER,
    BatchID                 CHAR (128),
    LotID                       CHAR (128),
    CampaignID                  CHAR (128),
    ProductID                   CHAR (32),
    OrderID                 CHAR (128),
    SE_Action                   INTEGER,
    SchedStatus                 INTEGER,
    StartCondition          CHAR (255),
    InitialMode                 INTEGER,
    SchedStartTime          DATETIME,
    SchedEndTime            DATETIME,
    BatchPriority               INTEGER,
    BatchSize                   FLOAT,
    EngrUnits                   CHAR (32),
    SENote                  CHAR (255),
    Description                 CHAR (255),
    PRIMARY KEY (ScheduleEntryID))

CREATE TABLE BXT_ScheduleEquip (
    ScheduleEntryID         CHAR (64)           NOT NULL,
    RequirementID           CHAR (32)           NOT NULL,
    Description                 CHAR (255),
    PRIMARY KEY (ScheduleEntryID, RequirementID))

CREATE TABLE BXT_ScheduleProperty (
    ScheduleEntryID         CHAR (64)           NOT NULL,
```

```
        RequirementID           CHAR (32)           NOT NULL,
        PropertyName            CHAR (32)           NOT NULL,
        PropertyValue           CHAR (255),
        EngrUnits                   CHAR (32),
        Description             CHAR (255),
        PRIMARY KEY (ScheduleEntryID, RequirementID, PropertyName))

    CREATE TABLE BXT_ScheduleParameter (
        ScheduleEntryID         CHAR (64)           NOT NULL,
        ParameterID                 CHAR (32)       NOT NULL,
        ParentParameterID           CHAR (32),
        ParameterValue          CHAR (255),
        EngrUnits                   CHAR (32),
        ItemLocation                CHAR (128),
        EnumSet                     CHAR (32),
        Description                 CHAR (255),
        PRIMARY KEY (ScheduleEntryID, ParameterID))

    CREATE TABLE BXT_HistoryElement (
        HistoryElementID        INTEGER             NOT NULL,
        BatchID                 CHAR (128),
        MasterRecipeID          CHAR (128),
        MasterRecipeVersion     CHAR (16),
        ControlRecipeID         CHAR (28),
        ReferenceEquipProcedure INTEGER,
        RecipeProcedure         CHAR (128),
        UnitProcedure           CHAR (128),
        UnitProcedureCounter    INTEGER,
        Operation                   CHAR (128),
        OperationCounter            INTEGER,
        Phase                       CHAR (128),
        PhaseCounter            INTEGER,
        EquipmentID                 CHAR (32),
        EPI_ID                  CHAR (32),
        PRIMARY KEY (HistoryElementID))

    CREATE TABLE BXT_HistoryLog (
        RecordID                    INTEGER         NOT NULL,
        UTC                         DATETIME,
        LocalTime                   DATETIME        NOT NULL,
        BatchID                 CHAR (128),
        HistoryElementID        INTEGER,
        EquipmentID                 CHAR (32),
        EPI_ID                  CHAR (32),
        UserID                  CHAR (64),
        RecordSet                   INTEGER         NOT NULL,
        RecordSubSet            INTEGER,
        RecordAlias                 CHAR (32),
        NewValue                    CHAR (128),
        OldValue                    CHAR (128),
        EngrUnits                   CHAR (32),
        PRIMARY KEY (RecordID))
```

Приложение С
(справочное)

Сокращения

В настоящем стандарте используются следующие сокращения:

BXT — Таблица обмена информацией о партии изделий (Batch exchange table);

EPE — Процедурный элемент оборудования (Equipment procedural element);

ID — Идентификатор (Identification);

ISA — Американское Общество Инженеров-Приборостроителей (Instrument Society of America);

MR — Технологическая рецептура (Master recipe);

PFC — Процедурная функциональная диаграмма (Procedure function chart);

RE — Рецептурный элемент (Recipe element);

SFC — Последовательная функциональная диаграмма (Sequential function chart);

SOP — Стандартная рабочая процедура (Standard operating procedure);

SQL — Язык структурированных запросов (Structured query language);

UML — Универсальный язык моделирования (Unified modeling language);

UTC — Универсальное координированное время (Universal coordinated time).

Приложение D
(справочное)

Руководство по языку

D.1 Общие положения

Язык – это множество символов и правил их использования в процессе общения. Раздел 6 содержит руководство по созданию указанных обозначений и правил, необходимых для обеспечения связи элементов производства партии изделий.

Процесс общения, задействующий машины и рабочих, имеет место для всех шести управляющих действий, описанных в МЭК 61512-1. Представители обеих сторон процесса общения должны понимать используемые обозначения, должны использовать одни и те же установленные правила.

Процесс общения между рабочими и электронными системами производства партии изделий обычно организуется с помощью видеоустройств, графических устройств, различных указателей, устройств ввода информации. Для интенсивного общения часто используют текстовые устройства. Визуальные символы могут оказаться более эффективными для передачи сложной информации.

Главной отличительной особенностью управления серийным производством является наличие рецептуры. Графические обозначения, поставленные в соответствие рассматриваемой рецептуре, и правила их использования определены в разделе 6. Прочие компоненты рецептуры (например, формулы) содержат подробности, которые лучше представить в визуальной форме, а не в текстовой.

D.2 Построение PFC-диаграмм

В ISTR88.00.03:1996 [2] рассмотрены три способа отображения информации: табличный формат, нотация (система обозначений) диаграмм Ганта, последовательные функциональные диаграммы (SFC-диаграммы).

Табличный формат привлекает своей простотой, очевидностью интерпретации и гибкостью (например, поддержка дополнительных атрибутов производится в понятной табличной форме, имеется поддержка вставок). Вместе с тем, область применения табличного формата ограничена линейными процедурами из-за трудностей адресации выбора и параллельных шагов.

Диаграмма Ганта обычно используется для описания действий во времени. Она может быть расширена для описания рецептурных процедур в более сложных случаях, чем табличный формат. Диаграмма Ганта не сводится только к отображению условных решений.

SFC-диаграммы могут использоваться для отображения условных решений процедуры. Вместе с тем, имеется ряд конкретных аспектов процедур, не представляемых корректно с помощью SFC-диаграмм. Некоторые представления с помощью SFC-диаграмм выглядят излишне усложненными.

Элементы этих трех указанных методов отображения информации объединены в одно целое для создания новой системы обозначений, называемой PFC-диаграммой. Данный вид диаграмм устанавливает графическое описание процедурной части рассматриваемой рецептуры и является производным от нотации функциональных диаграмм, определенных в МЭК 60848. PFC-диаграммы модифицированы для обеспечения корректного отображения рецептуры с использованием преимуществ диаграмм Ганта и табличного формата.

D.3 Рецептурная процедура

Отделение рецептурной процедуры (определяющей требуемую функциональность процесса) от процедуры оборудования (определяющей эффективность управления) обеспечивает достижение одной из целей, поставленной в МЭК 61512-1: рецептура должна создаваться без вовлечения рутинного аппарата управления производством. Поставленная цель оправдывает разделение усилий между функцией инженерного управления и функцией рецептурной авторизации. Поставленная цель требует, чтобы функция инженерного управления определяла и обеспечивала практическую реализацию процедурных элементов оборудования (например, фазы оборудования), обеспечивала представление авторского вклада разработчика технологической рецептуры с соответствующими ограничениями.

МЭК 61512-1 определяет четыре уровня для процедурных элементов:

- собственно процедура;
- процедура технологической установки;
- операция;
- фаза.

Собственно процедура включает несколько процедур технологической установки. Процедура технологической установки включает упорядоченное множество операций, отображающих смежные производственные последовательности, имеющие место внутри технологической установки. Операция представляет собой упорядоченное множество фаз. Количество фаз (одновременно активируемых в технологической установке) не ограничено. Но только одна операция может активироваться в технологической установке в любой момент времени. Процедуры

технологической установки в значительной степени независимы. При этом они включают (ссылаются на) операции и фазы нижнего уровня, которые могут взаимодействовать с операциями и фазами других процедур рассматриваемой технологической установки.

D.4 Требования к отображению элементов процедурного управления

В настоящем стандарте определен метод описания элементов процедурного управления в рецептурных процедурах. Для корректного отображения необходимо удовлетворять следующим требованиям:

- простота восприятия информации оператором;
- простота построения: количество синтаксических требований и условных обозначений должно быть минимальным;
- четкое определение границ: использование стандартных графических обозначений начала инструкции и окончания инструкции;
- однозначное отображение порядка выполнения действия: последовательность, параллелизм, выбор (дифференциация), сходимость;
- выражение координатных отношений: порядок передачи материала, время ожидания, требования синхронизации;
- иерархический уровень: стандартные обозначения собственно процедур, процедур технологической установки, операций, фаз;
- существование различных уровней: стандартные графические обозначения возможного разложения иерархических элементов на составные части;
- применимость к технологическим рецептурам и рецептурам управления;
- применимость ко всем уровням: аналогичные множества обозначений и правил используются на всех уровнях рецептуры;
- независимость от способа представления: одинаковая степень используемости и возможности понимания как для «бумажного» представления, так и для электронного представления с полным набором цветов и возможностей компьютерной графики.

Приложение Е
(справочное)

Пример обработки процедурных функциональных диаграмм

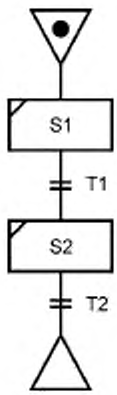
Многие правила, касающиеся обработки PFC-диаграмм, зависят от используемой производственной системы. Следующие примеры иллюстрируют возможные варианты обработки рассматриваемой диаграммы.

В указанных примерах точка используется для идентификации активных символов. Точка не является частью обозначения. Она только поясняет ситуацию. Символ является «активным», если он (в рассматриваемый момент времени) подвергается оценке или выполнению в процессе обработки PFC-диаграммы.

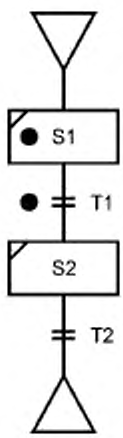
Если процедурный элемент является активным, то PFC-диаграмма (представляемый процедурный элемент оборудования) нижнего уровня в настоящий момент времени подвергается обработке в соответствии с используемой моделью состояния. В рассматриваемом примере используется пример модели состояния из МЭК 61512-1. Элемент является неактивным, если его обозначения не оцениваются (не задействованы) системой обработки PFC-диаграммы. Если процедурный элемент находится в резерве, то PFC-диаграмма (рассматриваемый процедурный элемент оборудования) нижнего уровня соответствует используемой модели состояния. В этом случае элемент не обрабатывается, и своего состояния изменить не может.

Если активируется явный переход, то рассматриваемое выражение оценивается системой обработки PFC-диаграммы. Если значение выражения равно TRUE, то соответствующее уведомление отсылается на PFC-диаграмму (рассматриваемую процедурную сущность оборудования) нижнего уровня, представляемую предшествующим рецептурным элементом.

Пример 1 — Фаза заканчивается, если значение рассматриваемого перехода равно TRUE.

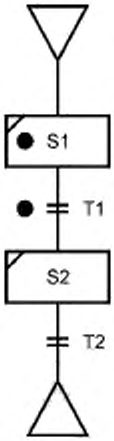


PFC-диаграмма запущена. Символ начала работ активирован.



Рецептура фазы S1 активирована. Соответствующая фаза оборудования (не показана) также активируется.

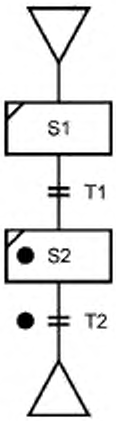
Явный переход T1 становится активным. Он непрерывно оценивается, так как рецептура фазы S1 активирована.



Значение T1 равно TRUE.

Фаза оборудования S1 получает входной сигнал: значение T1 равно TRUE.

Фаза оборудования S1 остается активной до тех пор, пока не активирована система внутренней логической оценки ее состояния.



Фаза оборудования S1 переходит в состояние завершения работ и деактивируется. Завершение работ может также являться результатом подачи внешнего сигнала на вход T1 и успешного завершения управляющих действий.

Рецептурная фаза S1 неактивна, так как неактивной является фаза оборудования S1.

Оценка состояния T1 больше не производится, элемент становится неактивным.

Рецептурная фаза S2 и соответствующая фаза оборудования S2 активируются.

Явный переход T2 становится активным, включается режим непрерывной оценки состояния данного элемента.

Приложение ДА
(справочное)

Сведения о соответствии ссылочных международных стандартов национальным стандартам

Таблица ДА.1

Обозначение ссылочного международного стандарта	Степень соответствия	Обозначение и наименование соответствующего национального стандарта
IEC 60848	—	*
IEC 60050-351	—	*
IEC 61131-3	—	*
IEC 61512-1	—	*
ISO/IEC 9075 (все части)	—	*
* Соответствующий национальный стандарт отсутствует.		

Библиография

[1] Rumbaugh, J. Jacobsen, I and J Booch, The unified modelling Language Manual, 1999, Addison-Wesley

[2] ISA TR88.0.03:1996, Possible recipe procedure presentation formats

УДК 658.52.011.56:006.354 ОКС 35.240.50 IDT

Ключевые слова: серийное производство, управление серийным производством, технологические процессы серийного производства, модели и терминология серийного производства, обмен информацией серийного производства

Редактор *Е. В. Дрюк*
Технический редактор *В. Н. Прусакова*
Корректор *И. А. Королева*
Компьютерная верстка *А. С. Тыртышного*

Сдано в набор 17.10.2016. Подписано в печать 16.11.2016. Формат 60 × 84 1/8. Гарнитура Ариал.
Усл. печ. л. 10,70. Уч.-изд. л. 9,68. Тираж 26 экз. Зак. 2808.
Подготовлено на основе электронной версии, предоставленной разработчиком стандарта

Издано и отпечатано во ФГУП «СТАНДАРТИНФОРМ», 123995 Москва, Гранатный пер., 4.
www.gostinfo.ru gost.e-ecolog.ru

gost.e-ecolog.ru