ФЕДЕРАЛЬНОЕ АГЕНТСТВО

ПО ТЕХНИЧЕСКОМУ РЕГУЛИРОВАНИЮ И МЕТРОЛОГИИ

НАЦИОНАЛЬНЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

ГОСТ Р ИСО 18629-43 — 2011

Системы промышленной автоматизации и интеграция

ЯЗЫК СПЕЦИФИКАЦИЙ ПРОЦЕССА

Часть 43

Дефинициональные расширения: упорядочивание действий и расширение продолжительности

ISO 18629-43:2006
Industrial automation systems and integration — Process specification language — Part 43:
Definitional extension: Activity ordering and duration extensions
(IDT)

Издание официальное

Москва
Стандартинформ
2014

gost.e-ecolog.ru

Предисловие

1 ПОДГОТОВЛЕН Научно-техническим центром «ИНТЕК» на основе собственного аутентичного перевода на русский язык международного 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 22 декабря 2011 г. № 1612-ст

4 Настоящий стандарт идентичен международному стандарту ИСО 18629-43:2006 «Системы промышленной автоматизации и интеграция. Язык спецификаций процесса. Часть 43. Дефиниционные расширения: упорядочивание действий и расширение продолжительности» (ISO 18629-43:2006 «Industrial automation systems and integration — Process specification language — Part 43: Definitional extension: Activity ordering and duration extensions»).

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 ГОСТ Р 1.0 — 2012 (раздел 8). Информация об изменениях к настоящему стандарту публикуется в ежегодном (по состоянию на 1 января текущего года) информационном указателе «Национальные стандарты», а официальный текст изменений и поправок — в ежемесячном информационном указателе «Национальные стандарты». В случае пересмотра (замены) или отмены настоящего стандарта соответствующее уведомление будет опубликовано в ближайшем выпуске ежемесячного информационного указателя «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.ru)

© Стандартинформ, 2014

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, определения и сокращения . . . 2
3.1 Термины и определения . . . 2
3.2 Сокращения . . . 4
4 Общая информация об ИСО 18629 . . . 4
5 Структура настоящего стандарта . . . 4
6 Сильные частично упорядоченные действия . . . 5
6.1 Примитивная лексика сильных частично упорядоченных действий . . . 5
6.2 Описываемая лексика понятий сильных частично упорядоченных действий . . . 5
6.3 Теории ядра, обусловленные сильными частично упорядоченными действиями . . . 5
6.4 Дефинициональные расширения, обусловленные сильными частично упорядоченными действиями . . . 5
6.5 Определение понятия для сильных частично упорядоченных действий . . . 5
6.6 Грамматика описаний процесса для сильных частично упорядоченных действий . . . 7
7 Ограничения продолжительности событий . . . 8
7.1 Примитивная лексика ограничений продолжительности событий . . . 8
7.2 Лексика ограничений продолжительности событий . . . 8
7.3 Теории ядра, обусловленные ограничениями продолжительности событий . . . 8
7.4 Дефинициональные расширения, обусловленные ограничениями продолжительности событий 8
7.5 Определение ограничений продолжительности событий . . . 8
7.6 Грамматика ограничений продолжительности событий . . . 9
8 Продолжительность, основанная на состоянии . . . 10
8.1 Примитивная лексика продолжительности, основанной на состоянии . . . 10
8.2 Определяемые соотношения для продолжительности, основанной на состоянии . . . 10
8.3 Теории ядра, обусловленные продолжительностью, основанной на состоянии . . . 10
8.4 Дефинициональные расширения, обусловленные продолжительностью, основанной на состоянии . . . 10
8.5 Определения продолжительности, основанной на состоянии . . . 10
8.6 Грамматика продолжительности, основанной на состоянии . . . 11
9 Временная продолжительность . . . 11
9.1 Примитивная лексика временной продолжительности . . . 11
9.2 Определяемые соотношения временной продолжительности . . . 11
9.3 Теории ядра, обусловленные временной продолжительностью . . . 11
9.4 Дефинициональные расширения, обусловленные временной продолжительностью . . . 11
9.5 Определения временной продолжительности . . . 11
9.6 Грамматика описаний процесса для временной продолжительности . . . 12
10 Временная продолжительность и продолжительность, основанная на состоянии . . . 12
10.1 Примитивная лексика временной продолжительности и продолжительности, основанной на состоянии . . . 13
10.2 Описываемая лексика временной продолжительности и продолжительности, основанной на состоянии . . . 13
10.3 Теории ядра, обусловленные временной продолжительностью и продолжительностью, основанной на состоянии . . . 13
10.4 Дефинициональные расширения, обусловленные временной продолжительностью и продолжительностью, основанной на состоянии . . . 13
10.5 Определения временной продолжительности и продолжительности, основанной на состоянии 13
10.6 Грамматика временной продолжительности и продолжительности, основанной на состоянии . 14
11 Упорядочивание и ограничения продолжительности событий . . . 14
11.1 Примитивная лексика упорядочивания и ограничений продолжительности событий . . . 14
11.2 Описываемая лексика упорядочивания и ограничений продолжительности событий . . . 14
11.3 Теории ядра, обусловленные упорядочиванием и ограничениями продолжительности событий 14

11.4 Дефиниціональные расширения, обусловленные упорядочиванием и ограничениями продолжительности событий . . . 15
11.5 Определения упорядочивания и ограничений продолжительности событий . . . 15
11.6 Грамматика описаний процесса для упорядочивания и ограничений продолжительности событий . . . 15
12 Упорядочивание и ограничения продолжительности встроенных событий . . . 16
12.1 Примитивная лексика упорядочивания и ограничений продолжительности встроенных событий . . . 16
12.2 Описываемая лексика упорядочивания и ограничений продолжительности встроенных событий . . . 16
12.3 Теории ядра, обусловленные упорядочиванием и ограничениями продолжительности встроенных событий . . . 16
12.4 Дефиниціональные расширения, обусловленные упорядочиванием и ограничениями продолжительности встроенных событий . . . 16
12.5 Определения упорядочивания и ограничений продолжительности встроенных событий . . . 16
12.6 Грамматика упорядочивания и ограничений продолжительности встроенных событий . . . 17
13 Входные условия нарушения выполнения действий . . . 17
13.1 Примитивная лексика входных условий нарушения выполнения действий . . . 17
13.2 Описываемая лексика входных условий нарушения выполнения действий . . . 18
13.3 Теории, обусловленные входными условиями нарушения выполнения действий . . . 18
13.4 Дефиниціональные расширения, обусловленные входными условиями нарушения выполнения действий . . . 18
13.5 Определения входных условий нарушения выполнения действий . . . 18
13.6 Грамматика описаний процесса для входных условий нарушения выполнения действий . . . 19
14 Запланированные встраивающие ограничения . . . 19
14.1 Примитивная лексика запланированных встраивающих ограничений . . . 19
14.2 Описываемая лексика запланированных встраивающих ограничений . . . 19
14.3 Теории ядра, обусловленные запланированными встраивающими ограничениями . . . 19
14.4 Дефиниціональные расширения, обусловленные запланированными встраивающими ограничениями . . . 20
14.5 Определения запланированных встраивающих ограничений . . . 20
14.6 Грамматика запланированных встраивающих ограничений . . . 20
15 Эффекты, основанные на продолжительности . . . 21
15.1 Примитивная лексика эффектов, основанных на продолжительности . . . 21
15.2 Описываемая лексика эффектов, основанных на продолжительности . . . 21
15.3 Теории ядра, обусловленные эффектами, основанными на продолжительности . . . 21
15.4 Дефиниціональные расширения, обусловленные эффектами, основанными на продолжительности . . . 21
15.5 Определения эффектов, основанных на продолжительности . . . 21
15.6 Грамматика эффектов, основанных на продолжительности . . . 22
16 Эффекты действий, основанные на продолжительности и времени . . . 22
16.1 Примитивная лексика эффектов действий, основанных на продолжительности и времени . . . 22
16.2 Описываемая лексика эффектов действий, основанных на продолжительности и времени . . . 22
16.3 Теории ядра, обусловленные эффектами действий, основанными на продолжительности и времени . . . 23
16.4 Дефиниціональные расширения, обусловленные эффектами действий, основанными на продолжительности и времени . . . 23
16.5 Определения эффектов действий, основанных на продолжительности и времени . . . 23
16.6 Грамматика эффектов действий, основанных на продолжительности и времени . . . 24
17 Упорядочивание соотношений в комплексной последовательности . . . 24
17.1 Примитивная лексика упорядочивания соотношений в комплексной последовательности . . . 24
17.2 Описываемая лексика упорядочивания соотношений в комплексной последовательности . . . 24
17.3 Теории, обусловленные упорядочиванием соотношений в комплексной последовательности 24

17.4 Дефинициональные расширения, обусловленные упорядочиванием соотношений в комплексной последовательности 24
17.5 Определения упорядочивания соотношений в комплексной последовательности 24
Приложение А (справочное) ASN.1 Идентификатор настоящего стандарта 26
Приложение В (справочное) Пример описания технологического процесса в соответствии с настоящим стандартом 27
Приложение ДА (справочное) Сведения о соответствии ссылочных международных стандартов национальным стандартам Российской Федерации 33
Библиография 34

Введение

ИСО 18629 — это комплекс стандартов на компьютерно-интерпретируемый обмен данными обеспечения технологического процесса. Все части комплекса стандартов ИСО 18629 устанавливают групповой язык программирования для описания конкретного технологического процесса, рассматриваемого как часть всего процесса изготовления изделия либо внутри одной промышленной компании, либо сразу в нескольких промышленных секторах (компаниях) вне его связи с какой-либо моделью компьютерного представления. Природа данного языка программирования такова, что он обеспечивает доступ к спецификациям технологического процесса и технологическим данным изделия на всех стадиях процесса его изготовления.

В настоящем стандарте установлены описания дефинициональных расширений языка программирования, относящихся к расширениям действий в соответствии с комплексом стандартов ИСО 18629.

Все части комплекса ИСО 18629 не связаны с какой-либо конкретной моделью компьютерного представления технологического процесса в рассматриваемом техническом приложении. Все вместе указанные части ИСО 18629 обеспечивают структурную технологическую взаимосвязь процессов производства для улучшения оперативной совместимости рассматриваемых технических приложений.

НАЦИОНАЛЬНЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

Системы промышленной автоматизации и интеграция

ЯЗЫК СПЕЦИФИКАЦИЙ ПРОЦЕССА

Часть 43

Дефинициональные расширения: упорядочивание действий и расширение продолжительности

Industrial automation systems and integration. Process specification language.
Part 43. Definitional extension. Activity ordering and duration extensions

Дата введения — 2012—09—01

1 Область применения

Настоящий стандарт устанавливает спецификацию непримитивных понятий языка программирования. При этом используется набор определений, написанных на языке, установленном в ИСО 18629. Данные определения устанавливают аксиомы для терминологии в соответствии с ИСО 18629.

Область применения настоящего стандарта включает определения понятий с использованием терминологии в соответствии с ИСО 18629-13.

Область применения настоящего стандарта не включает определения новых временных и основанных на состоянии понятий в соответствии с ИСО 18629-11 и ИСО 18629-12.

2 Нормативные ссылки

В настоящем стандарте использованы нормативные ссылки на следующие международные стандарты, которые необходимо учитывать при использовании настоящего стандарта. В случае ссылок на документы, у которых указана дата утверждения, необходимо пользоваться только указанной редакцией. В случае, когда дата утверждения не приведена, следует пользоваться последней редакцией ссылочных документов, включая любые поправки и изменения к ним:

ИСО/МЭК 8824-1 Информационные технологии. Нотация абстрактного синтаксиса версии 1 (ASN.1). Часть 1. Спецификация базовой нотации (ISO/IEC 8824-1, Information technology — Abstract Syntax Notation One (ASN.1) — Part 1:Specification of basic notation)

ИСО 15531-1 Системы промышленной автоматизации и интеграция. Управляющая информация промышленным производством. Часть 1. Общий обзор (ISO 15531-1, Industrial automation systems and integration — Industrial manufacturing management data — Part 1: General overview)

ИСО 15531-42 Системы промышленной автоматизации и интеграция. Управляющая информация промышленным производством. Часть 42. Модель времени (ISO 15531-42, Industrial automation systems and integration — Industrial manufacturing management data — Part 42: Time Model)

ИСО 18629-1:2004 Системы промышленной автоматизации и интеграция. Язык спецификаций процесса. Часть 1. Обзор и основные принципы (ISO 18629-1:2004, Industrial automation systems and integration — Process specification language — Part 1: Overview and basic principles)

ИСО 18629-11:2005 Системы промышленной автоматизации и интеграция. Язык спецификаций процесса. Часть 11. Ядро PSL (ISO 18629-11:2005, Industrial automation systems and integration — Process specification language — Part 11: PSL core)

ИСО 18629-12 Системы промышленной автоматизации и интеграция. Язык спецификаций процесса. Часть 12. Внешнее ядро (ISO 18629-12, Industrial automation systems and integration — Process specification language — Part 12: Outer core)

Издание официальное

ИСО 18629-13 Системы промышленной автоматизации и интеграция. Язык спецификаций процесса. Часть 13. Теории длительности и упорядочения (ISO 18629-13, Industrial automation systems and integration — Process specification language — Part 13: Duration and ordering theories)

3 Термины, определения и сокращения

3.1 Термины и определения

В настоящем стандарте применены следующие термины с соответствующими определениями:

3.1.1 **автоморфизм** (automorphism): Отображение «один к одному» элементов на множество, сохраняющее соотношения и функции в некоторой модели.

[ИСО 18629-13]

3.1.2 **аксиома** (axiom): Точно сформулированное аналитическое выражение на формальном языке, устанавливающее ограничения к интерпретации символов в словаре языка.

[ИСО 18629-1]

3.1.3 **установленная лексика** (defined lexicon): Набор символов в нелогической лексике, обозначающих установленные понятия.

Примечание — Установленная лексика включает константы, функции и символы соотношений.

Пример — Термины с консервативными определениями [ИСО 18629-1].

3.1.4 **дефинициональное расширение** (definitional extension): Расширение ядра PSL, представляющее новые лингвистические понятия, которые могут быть определены с помощью терминов ядра PSL.

Примечание — Дефинициональные расширения не добавляют выразительную силу ядру PSL и используются для подробного описания семантики и терминологии в области применения.

[ИСО 18629-1]

3.1.5 **продолжительность интервала времени** (duration interval of time): Длина интервала времени, представленная в заданных единицах измерения.

[ИСО 15531-42]

3.1.6 **эндоморфизм** (endomorphism): Отображение множества на некоторое подмножество, сохраняющее соотношения и функции в некоторой модели.

[ИСО 18629-13]

3.1.7 **расширение** (extension): Расширение ядра PSL, содержащее дополнительные аксиомы.

Примечание 1 — Ядро PSL представляет собой относительно простой набор аксиом, достаточный для представления широкого круга основных процессов. Однако для представления более сложных процессов требуются дополнительные ресурсы, отсутствующие в ядре PSL. Ядро PSL с каждым понятием следует использовать для описаний того или иного процесса, а для описания разнообразных модульных расширений следует использовать расширение и дополнения ядра PSL. В этом случае пользователь может использовать такой язык, который соответствует требованиям к выразительности.

Примечание 2 — Все расширения являются теориями ядра или дефинициональными расширениями.

[ИСО 18629-1]

3.1.8 **грамматика** (grammar): Правила совместного использования логических символов и словарных терминов для составления точно сформулированных аналитических выражений.

[ИСО 18629-1]

3.1.9 **гомоморфизм** (homomorphism): Отображение между множествами, сохраняющее некоторые соотношения на элементах множества.

[ИСО 18629-13]

3.1.10 **язык** (language): Сочетание лексики и грамматики.

[ИСО 18629-1]

3.1.11 **лексика** (lexicon): Набор символов и терминов.

Примечание — Лексика состоит из логических (например, булевы выражения и квантификаторы) и нелогических символов. В комплексе стандартов ИСО 18629 нелогическая часть лексики состоит из выражений (констант, функциональных символов и реляционных символов), необходимых для представления основных понятий онтологии.

[ИСО 18629-1]

3.1.12 **производство** (manufacturing): Функция или действие, предусматривающие перевод или превращение материала из сырья или заготовки в завершенное состояние.

[ИСО 15531-1]

3.1.13 **производственный процесс** (manufacturing process): Структурированный комплекс видов деятельности или работ, выполняемых с материалом для перевода его из сырья или заготовки в завершенное состояние.

Примечание — Производственные процессы могут быть представлены в виде технологической схемы процесса, схемы движения продукта, в виде табличной схемы или схемы фиксированного расположения. К планируемым производственным процессам могут относиться изготовление продукта для складирования, на заказ и для сборки на заказ и т. д., основанным на стратегическом использовании и размещении материально-производственных запасов.

[ИСО 15531-1]

3.1.14 **мономорфизм** (monomorphism): Отображение «один к одному» между множествами, сохраняющее некоторые соотношения на элементах множества.

[ИСО 18629-13]

3.1.15 **примитивная концепция** (primitive concept): Лексический термин, не имеющий консервативного определения.

[ИСО 18629-1]

3.1.16 **примитивная лексика** (primitive lexicon): Набор символов в нелогическом словаре, обозначающих элементарные понятия.

Примечание — Примитивная лексика включает в себя постоянные, функциональные и реляционные символы.

[ИСО 18629-1]

3.1.17 **процесс** (process): Структурированный ряд видов деятельности, включающий в себя различные сущности предприятия, предназначенный и организованный для достижения конкретной цели.

Примечание — Данное определение аналогично определению, приведенному в ИСО 10303-49. Тем не менее ИСО 15531 нуждается в понятии структурированного набора деятельностей без какого-либо предопределенного отношения ко времени или этапам. С точки зрения управления потоком некоторые свободные процессы могут требовать синхронизации в отношении цели, хотя в действительности они ничего не выполняют (задачи-призраки).

[ИСО 15531-1]

3.1.18 **продукт** (product): Изделие, материал или вещество, изготовленное в процессе производства.

[ИСО 10303-1]

3.1.19 **ресурс** (resource): Любые устройство, инструмент и средства, за исключением сырья и компонентов конечной продукции, имеющиеся в расположении предприятия для производства товаров и услуг.

Примечание 1 — Рассматриваемое понятие ресурса адаптировано по отношению к ИСО 15531-1. Понятие ресурса, введенное в ИСО 15531-1, не включает сырьевые материалы, продукты и компоненты, являющиеся (с точки зрения системной теории) элементами окружающей среды и, таким образом, не являющиеся частью системы. В настоящем стандарте данное допущение снято. Более того, определение, принятое в ИСО 15531-1, во многом использует определение, принятое в ИСО 10303-49, при этом оно включается в определение, принятое в настоящей части ИСО 18629. В дополнение к понятию ресурса, принятому в ИСО 15531, понятие ресурса, принятое в настоящем стандарте, включает сырьевые и расходуемые материалы в соответствии с ИСО 18629-14.

Примечание 2 — Ресурсы в соответствии с приведенным выше определением включают также рабочую силу, рассматриваемую как особое средство с заданными возможностями и заданной производительности. Указанные средства рассматриваются как целесообразные для использования в процессе производства на основании технического задания. Данное определение не включает какого-либо моделирования индивидуального или группового поведения человеческого ресурса, за исключением его способности выполнять заданную работу в процессе производства (например, преобразование сырого материала или полуфабриката, обеспечение логистических услуг и т. п.). Это означает, что человеческие ресурсы, как и другие, рассматриваются с точки зрения их функций, их возможностей и их состояния (например, занят, свободен). При этом исключается какое-либо моделирование или представление какого-либо аспекта индивидуального или группового социального поведения.

[ИСО 15531-1]

3.1.20 **теория** (theory): Набор аксиом и определений, относящийся к данному понятию или набору понятий.

Примечание — Данное определение отражает подход искусственного интеллекта, где теория — это набор предположений, на которых основано значение соответствующего понятия.

[ИСО 18629-1]

3.2 Сокращения

- KIF — формат взаимного обмена знаниями (Knowledge Interchange Format).

4 Общая информация об ИСО 18629

Части с 41 по 49 комплекса международных стандартов ИСО 18629 определяют дефинициональные расширения, необходимые для формулировки точных определений и родственных аксиом непримитивных понятий ИСО 18629. Дефинициональные расширения определены ИСО 18629-11 и ИСО 18629-12, где введены новые элементы лексики. Данные элементы дефинициональных расширений могут быть полностью определены в соответствии с ИСО 18629-11 и ИСО 18629-12. Дефинициональные расширения дают точные семантические определения элементов, используемых в спецификациях индивидуальных технических приложений или типов технических приложений, при обеспечении совместных работ. Дефинициональные расширения существуют в следующих категориях:

- расширения действий;
- временные расширения и расширения, основанные на состоянии;
- упорядочивание действий и расширение продолжительности;
- назначения ресурса,
- наборы ресурсов;
- расширения действий процессора.

Индивидуальным (групповым) пользователям ИСО 18629 может потребоваться расширение ИСО 18629 для спецификации понятий, отсутствующих в настоящее время в частях с 41 по 49 комплекса международных стандартов ИСО 18629. Для этих целей они должны использовать элементы, определенные в ИСО 18629. Пользовательские расширения и их определения устанавливают дефинициональные расширения, которые не должны быть включены в части с 41 по 49 ИСО 18629.

Примечание — Пользовательские расширения должны удовлетворять требованиям ИСО 18629 и соответствовать ИСО 18629-1:2004 (подразделы 5.1 и 5.2).

Предметом рассмотрения частей с 41 по 49 комплекса стандартов ИСО 18629 являются:

- семантические определения (на основе понятий, установленных в ИСО 18629-11 и ИСО 18629-12), элементы которых являются характерными для шести понятий, определенных выше;
- набор аксиом, ограничивающих использование элементов в дефинициональных расширениях.

Предметом рассмотрения частей с 41 по 49 комплекса стандартов ИСО 18629 не являются:

- определения и аксиомы для понятий, определенных ИСО 18629-11 и ИСО 18629-12;
- элементы, не определенные в соответствии с ИСО 18629-11 и ИСО 18629-12;
- пользовательские расширения.

5 Структура настоящего стандарта

Фундаментальные теории, описанные в настоящей части ИСО 18629:

- сильные частично упорядоченные действия;
- ограничения продолжительности событий;
- продолжительность, основанная на состоянии;
- временная продолжительность;
- временная продолжительность и продолжительность, основанная на состоянии;
- упорядочивание и ограничения продолжительности событий;
- упорядочивание и ограничения продолжительности встроенных событий;
- входные условия нарушения выполнения действий;
- запланированные встраивающие ограничения;

- эффекты, основанные на продолжительности;
- эффекты действий, основанные на продолжительности и состоянии;
- упорядочивание соотношений в комплексной последовательности.

Все дефиниционные расширения в настоящем стандарте являются расширениями ИСО 18629-13, которые, в свою очередь, являются расширениями ИСО 18629-12 и ИСО 18629-11.

6 Сильные частично упорядоченные действия

Данный раздел характеризует все определения, обусловленные сильными частично упорядоченными действиями.

6.1 Примитивная лексика сильных частично упорядоченных действий

Лексика сильных частично упорядоченных действий не требует никаких примитивных соотношений.

6.2 Описываемая лексика понятий сильных частично упорядоченных действий

В данном подразделе определены следующие соотношения:
- (same_bag ?s1 ?s2 ?a);
- (snapshot ?s1 ?s2 ?a);
- (rotate ?s ?a);
- (reflect ?s ?a);
- (flip?s?a);
- (turn?s?a);
- (bag ?occ);
- (strong_poset ?occ);
- (choice_poset ?occ);
- (complex_poset ?occ).

Каждое понятие определяется неформальной семантикой и аксиомами KIF.

6.3 Теории ядра, обусловленные сильными частично упорядоченными действиями

Для данного расширения необходимо иметь:
- soo.th;
- act_occ.th;
- complex.th;
- atomic.th;
- subactivity.th;
- occtree.th;
- psl_core.th.

6.4 Дефинициональные расширения, обусловленные сильными частично упорядоченными действиями

Для сильных частично упорядоченных действий никакие дефинициональные расширения не требуются.

6.5 Определение понятия для сильных частично упорядоченных действий

Для сильных частично упорядоченных действий определены нижеследующие понятия.

6.5.1 same_bag

Данное соотношение используется для спецификации событий, соединенных оператором типа «и».

```
(forall (?s1 ?s2 ?a) (iff (same_bag ?s1 ?s2 ?a)
(exists (?s3 ?s4)
    (and (next_subocc ?s1 ?s3 ?a)
        (iso_occ ?s3 ?s2)
        (next_subocc ?s2 ?s4 ?a)
        (iso_occ ?s4 ?s1)))))
```

6.5.2 snapshot

Событие ?s1 на дереве действий для ?a является упорядоченно-гомоморфическим для элемента ?s2 упорядочивания события.

```
(forall (?s1 ?s2 ?a) (iff (snapshot ?s1 ?s2 ?a)
(and (iso_occ ?s1 ?s2)
     (soo ?s2 ?a)
     (implies (root_soo ?s2 ?a)
          (root ?s1 ?a))
     (forall (?s3)
          (implies (soo_precedes ?s3 ?s2 ?a)
               (exists (?s4)
                    (and (min_precedes ?s4 ?s1 ?a)
                         (or (mono ?s3 ?s4 ?a)
                              (= ?s3 ?s4)))))))))
```

6.5.3 rotate

На дереве действий для ?a, содержащем ?s, множество последующих событий является копией множества вершин дерева для ?s, имеющих общего родителя.

```
(forall (?s ?a) (iff (rotate ?s ?a)
(and (forall (?s1)
     (implies (next_subocc ?s ?s1 ?a)
          (exists (?s2)
               (and (sibling ?s ?s2 ?a)
                    (iso_occ ?s1 ?s2)))))
(forall (?s3)
     (implies (sibling ?s ?s3 ?a)
          (same_bag ?s ?s3 ?a))))))
```

6.5.4 reflect

На дереве действий для ?a, содержащем ?s, множество последующих событий является копией множества последующих элементов ?s упорядочивания событий.

```
(forall (?s ?a) (iff (reflect ?s ?a)
(forall (?s1)
     (iff (next_subocc ?s ?s1 ?a)
          (exists (?s2 ?s3)
               (and (next_soo ?s2 ?s3 ?a)
                    (snapshot ?s ?s2 ?a)
                    (iso_occ ?s1 ?s3)))))))
```

6.5.5 flip

На дереве действий для ?a, содержащем ?s, множество последующих событий является копией множества последующих элементов ?s упорядочивания событий вместе с копией вершин дерева для ?s (имеющих общего родителя).

```
(forall (?s ?a) (iff (flip ?s ?a)
(forall (?s1)
     (iff (next_subocc ?s ?s1 ?a)
          (or (exists (?s2 ?s3)
               (and (snapshot ?s ?s2 ?a)
                    (next_soo ?s2 ?s3 ?a)
                    (iso_occ ?s1 ?s3)))
          (exists (?s2)
               (and (sibling ?s ?s2 ?a)
          (iso_occ ?s1 ?s2))))))))
```

6.5.6 turn

На дереве действий для ?a, содержащем ?s, множество последующих событий является копией множества последующих элементов ?s упорядочивания событий вместе с копией вершин дерева для ?s (имеющих общего родителя), удовлетворяющих условию «same_bag».

```
(forall (?s ?a) (iff (turn ?s ?a)
(and (exists (?s5)
     (and (sibling ?s5 ?a)
          (same_bag ?s ?s5 ?a)))
```

```
(forall (?s1)
  (implies (next_subocc ?s ?s1 ?a)
        (or (exists (?s2 ?s3)
              (and (snapshot ?s ?s2 ?a)
                   (next_soo ?s2 ?s3 ?a)
                   (iso_occ ?s1 ?s3)))
            (exists (?s2)
              (and (sibling ?s ?s2 ?a)
                (iso_occ ?s1 ?s2)))))))
```

6.5.7 bag

«bag» — это дерево действий, все элементы которого обладают свойством «rotated» (см. 6.5.3). Интуитивно ясно, что данное понятие соответствует карте технологического процесса, содержащей только соотношения типа «и» без линейного упорядочивания.

```
(forall (?occ) (iff (bag ?occ)
(forall (?a ?s ?occ1)
    (implies (and (same_grove ?occ ?occ1)
                  (occurrence_of ?occ ?a)
                  (subactivity_occurrence ?s ?occ1))
             (rotate ?s ?a)))))
```

6.5.8 choice_poset

«choice_poset» — это дерево действий, все элементы которого обладают свойством «reflected» (см.6.5.4). Интуитивно ясно, что данное понятие соответствует карте технологического процесса, содержащей только соотношения «или».

```
(forall (?occ) (iff (choice_poset ?occ)
(forall (?a ?s ?occ1)
    (implies (and (same_grove ?occ ?occ1)
          (occurrence_of ?occ ?a)
          (subactivity_occurrence ?s ?occ1))
       (reflect ?s ?a)))))
```

6.5.9 strong_poset

«strong_poset» — это дерево действий, все элементы которого обладают свойством «flip» (см. 6.5.5). Интуитивно ясно, что данное понятие соответствует карте технологического процесса, содержащей линейное упорядочивание внутри соотношений типа «и».

```
(forall (?occ) (iff (strong_poset ?occ)
(forall (?a ?s ?occ1)
    (implies (and (same_grove ?occ ?occ1)
          (occurrence_of ?occ ?a)
          (subactivity_occurrence ?s ?occ1))
      (flip ?s ?a)))))
```

6.5.10 complex_poset

«complex_poset» — это дерево действий, все элементы которого обладают свойством «turn» (см. 6.5.6). Интуитивно ясно, что данное понятие соответствует карте технологического процесса, содержащей как соотношения «и», так и соотношения «или».

```
(forall (?occ) (iff (complex_poset ?occ)
(forall (?a ?s ?occ1)
    (implies (and (same_grove ?occ ?occ1)
           (occurrence_of ?occ ?a)
           (subactivity_occurrence ?s ?occ1))
         (turn ?s ?a)))))
```

6.6 Грамматика описаний процесса для сильных частично упорядоченных действий

Нижеследующие грамматические утверждения дают описания технологического процесса и вспомогательные правила, определенные в KIF для сильных частично упорядоченных действий.

Примечание — Функция и важность грамматических утверждений поясняется в ИСО 18629-1:2004 (пункты 3.3.8, 4.2.4 и 5.1).

```
                    < strongposet_spec > ::= (and (strong_poset ?occ)
                                < soo_axiom >)
                < tree_formula > ::= (exists (< variable >+)
                    (and (same_tree < variable > ?occ)
                        (subactivity_occurrence < variable > < variable >)))
< precedes_formula > ::= (soo_precedes < variable > < variable > < term >) |
                                (and < precedes_formula >+)
                    < parallel_formula > ::= (parallel < variable > < variable > <term >) |
                                (and < precedes_formula >+)
            < soo_axiom > ::=
            (forall (?occ < variable >*)
                (implies (exists (< variable >+)
                    (and < precedes_formula >*
                            < parallel_formula >*
                            < tree_formula >))))
```

7 Ограничения продолжительности событий

Данный раздел характеризует все определения, обусловленные ограничениями продолжительности событий.

7.1 Примитивная лексика ограничений продолжительности событий

Лексика ограничений продолжительности событий не требует никаких примитивных соотношений.

7.2 Лексика ограничений продолжительности событий

В данном подразделе определены нижеследующие соотношения:
- (dur ?occ);
- (delay ?occ1 ?occ2);
- (dur_equiv ?occ1 ?occ2);
- (delay_equiv?occ1 ?occ2);
- (constant ?a);
- (interval_duration ?a);
- (variable ?a).

Каждое понятие определяется неформальной семантикой и аксиомами KIF.

7.3 Теории ядра, обусловленные ограничениями продолжительности событий

Для данных дефинициональных расширений необходимы:
- duration.th;
- psl_core.th.

7.4 Дефинициональные расширения, обусловленные ограничениями продолжительности событий

Для ограничений продолжительности событий никакие дефинициональные расширения не требуются.

7.5 Определение ограничений продолжительности событий

Для ограничений продолжительности событий определены нижеследующие понятия.

7.5.1 dur

Функция «dur» задает продолжительность между моментом начала некоторого события и моментом окончания некоторого события.

```
(forall (?occ)
    (= (dur ?occ) (duration (beginof ?occ) (endof ?occ))))
```

7.5.2 delay

Функция «delay» задает продолжительность между моментом начала одного события и моментом начала другого события.

```
(forall (?occ1 ?occ2)
    (= (delay ?occ1 ?occ2) (duration (beginof ?occ1) (beginof ?occ2))))
```

7.5.3 dur_equiv

Два события являются эквивалентными по продолжительности тогда и только тогда, когда их продолжительность одинакова.

```
(forall (?occ1 ?occ2) (iff (dur_equiv ?occ1 ?occ2)
(= (dur ?occ1) (dur ?occ2)))))
```

7.5.4 delay_equiv

Два события эквивалентны по задержке тогда и только тогда, когда существует третье событие с такой же задержкой.

```
(forall (?occ1 ?occ2) (iff (delay_equiv ?occ1 ?occ2)
(exists (?occ)
    (= (delay ?occ ?occ1) (delay ?occ ?occ2)))))
```

7.5.5 constant

Некоторое действие является константой тогда и только тогда, когда все события имеют одинаковую продолжительность.

```
(forall (?a) (iff (constant ?a)
(forall (?occ1 ?occ2)
    (implies (and (occurrence ?occ1 ?a)
            (occurrence ?occ2 ?a))
        (dur_equiv ?occ1 ?occ2)))))
```

7.5.6 interval_duration

Некоторое действие имеет продолжительность интервала тогда и только тогда, когда существуют события, имеющие одинаковую продолжительность.

```
(forall (?a) (iff (interval_duration ?a)
(forall (?occ1)
    (implies (occurrence ?occ1 ?a)
        (exists (?occ2)
            (and (occurrence ?occ2 ?a)
                (dur_equiv ?occ1 ?occ2)))))))
```

7.5.7 variable

Некоторое действие является переменным тогда и только тогда, когда все события имеют разную продолжительность.

```
(forall (?a) (iff (variable ?a)
(not (exists (?occ1 ?occ2)
    (and (occurrence ?occ1 ?a)
        (occurrence ?occ2 ?a)
        (dur_equiv ?occ1 ?occ2))))))
```

7.6 Грамматика ограничений продолжительности событий

Нижеследующие грамматические утверждения дают описания технологического процесса и вспомогательных правил, определенных в KIF для ограничений продолжительности событий.

```
    < constant_spec > ::= (forall (?occ)
            (implies (occurrence ?occ < term >)
                < duration_literal >))
    < interval_spec > ::= (forall (?occ)
            (implies (occurrence ?occ < term >)
                < interval_axiom >))
< duration_literal > ::= (= (dur ?occ) < term >)
< interval_literal > ::= (lesser (dur ?occ) < term >) |
                (lesser < term > (dur ?occ))
< interval_axiom > ::= < interval_literal > |
                (not < interval_axiom >) |
                ({and | or} < interval_axiom >*) |
                ({implies | iff} < interval_axiom >)
```

8 Продолжительность, основанная на состоянии

Данный раздел характеризует все определения, обусловленные продолжительностью, основанной на состоянии.

8.1 Примитивная лексика продолжительности, основанной на состоянии

Лексика продолжительности, основанной на состоянии, не требует никаких примитивных соотношений.

8.2 Определяемые соотношения для продолжительности, основанной на состоянии

В данном подразделе определены следующие соотношения:
- (conditional_duration ?a);
- (context_duration ?a);
- (unconditional_duration ?a).

Каждое понятие определяется неформальной семантикой и аксиомами KIF.

8.3 Теории ядра, обусловленные продолжительностью, основанной на состоянии

Для данных дефинициональных расширений необходимы:
- duration.th;
- disc_state.th;
- occtree.th;
- psl_core.th.

8.4 Дефинициональные расширения, обусловленные продолжительностью, основанной на состоянии

Нижеследующие дефинициональные расширения обусловлены продолжительностью, основанной на состоянии:
- actdur.def;
- state_precond.def.

8.5 Определения продолжительности, основанной на состоянии

Для расширения основанной на состоянии продолжительности определены нижеследующие понятия.

8.5.1 conditional_duration

Некоторое действие имеет условную продолжительность при ограничении: если когда-либо два события согласованы по состоянию, то они согласованы и по продолжительности.

```
(forall (?a) (iff (conditional_duration ?a)
(forall (?occ1 ?occ2)
    (implies (and (occurrence ?occ1 ?a)
            (occurrence ?occ2 ?a)
            (state_equiv ?occ1 ?occ2))
        (dur_equiv ?occ1 ?occ2)))))
```

8.5.2 context_duration

Некоторое действие имеет контекстную продолжительность тогда и только тогда, когда существуют сохраняющие продолжительность переменные автоморфизмы.

```
(forall (?a) (iff (context_duration ?a)
(and (exists (?occ1)
    (forall (?occ2)
        (implies (and (occurrence ?occ1 ?a)
                (occurrence ?occ2 ?a)
                (state_equiv ?occ1 ?occ2))
            (dur_equiv ?occ1 ?occ2))))
    (exists (?occ3 ?occ4)
        (and (occurrence ?occ3 ?a)
```

```
        (occurrence ?occ3 ?a)
        (state_equiv ?occ3 ?occ4)
        (not (dur_equiv ?occ3 ?occ4)))))))
```

8.5.3 unconditional_duration

Некоторое действие имеет безусловную продолжительность тогда и только тогда, когда сохраняющие продолжительность переменные автоморфизмы тривиальны.

```
(forall (?a) (iff (unconditional_duration ?a)
(forall (?occ1)
    (implies (occurrence ?occ1 ?a)
        (exists (?occ2)
            (and (occurrence ?occ2 ?a)
                (state_equiv ?occ1 ?occ2)
                (not (dur_equiv ?occ1 ?occ2))))))))
```

8.6 Грамматика продолжительности, основанной на состоянии

Нижеследующие грамматические утверждения дают описания технологического процесса и вспомогательных правил, определенных в KIF для продолжительности, основанной на состоянии.

```
< cond_dur_spec > ::= (forall (?occ)
        (implies (and (occurrence ?occ < term >)
                < simple_state_axiom >)
            < duration_literal >))
< context_dur_spec > ::= (forall (?occ)
        (implies (and (occurrence ?occ < term >)
                < state_axiom >)
            < duration_literal >)) |
        (forall (?occ)
            (implies (and (occurrence ?occ < term >)
                    < simple_state_axiom >)
                < interval_literal >))
```

9 Временная продолжительность

Данный раздел характеризует все определения, обусловленные временной продолжительностью.

9.1 Примитивная лексика временной продолжительности

Лексика временной продолжительности не требует никаких примитивных соотношений.

9.2 Определяемые соотношения временной продолжительности

В данном подразделе определены следующие соотношения:
- (rushhour ?a);
- (weekend ?a);
- (gridlock ?a).

Каждое понятие определяется неформальной семантикой и аксиомами KIF.

9.3 Теории ядра, обусловленные временной продолжительностью

Для данных дефинициональных расширений необходимы:
- duration.th;
- occtree.th;
- psl_core.th.

9.4 Дефинициональные расширения, обусловленные временной продолжительностью

Для данного расширения необходимо иметь actdur.def и time_precond.def.

9.5 Определения временной продолжительности

Для временной продолжительности определены нижеследующие понятия.

9.5.1 rushhour

Некоторое действие является пиковым при условии: если когда-либо два события согласованы по моментам времени их начала, то они согласованы и по продолжительности.

```
(forall (?a) (iff (rushhour ?a)
(forall (?occ1 ?occ2)
      (implies (and (occurrence ?occ1 ?a)
                    (occurrence ?occ2 ?a)
                    (begin_equiv ?occ1 ?occ2))
               (dur_equiv ?occ1 ?occ2))))
```

9.5.2 weekend

Некоторое действие является сверхурочным тогда и только тогда, когда существуют сохраняющие продолжительность автоморфизмы начала.

```
(forall (?a) (iff (weekend ?a)
(and (exists (?occ1)
       (forall (?occ2)
              (implies (and (occurrence ?occ1 ?a)
                            (occurrence ?occ2 ?a)
                            (begin_equiv ?occ1 ?occ2))
                       (dur_equiv ?occ1 ?occ2))))
  (exists (?occ3 ?occ4)
       (and (occurrence ?occ3 ?a)
            (occurrence ?occ3 ?a)
            (begin_equiv ?occ3 ?occ4)
            (not (dur_equiv ?occ3 ?occ4)))))))
```

9.5.3 gridlock

Некоторое действие является тупиковым тогда и только тогда, когда сохраняющий продолжительность автоморфизм начала является тривиальным.

```
(forall (?a) (iff (gridlock ?a)
(forall (?occ1)
      (implies (occurrence ?occ1 ?a)
               (exists (?occ2)
                       (and (occurrence ?occ2 ?a)
                            (begin_equiv ?occ1 ?occ2)
                            (not (dur_equiv ?occ1 ?occ2))))))))
```

9.6 Грамматика описаний процесса для временной продолжительности

Нижеследующие грамматические утверждения дают описания технологического процесса, определенные в KIF для временной продолжительности.

```
< rushhour_spec > ::= (forall (?occ)
      (implies (and (occurrence ?occ < term >)
                    < simple_time_axiom >)
               < duration_literal >))
< weekend_spec > ::= (forall (?occ)
      (implies (and (occurrence ?occ < term >)
                    < time_axiom >)
               < duration_literal >)) |
    (forall (?occ)
      (implies (and (occurrence ?occ < term >)
                    < simple_time_axiom >)
               < interval_literal >))
```

10 Временная продолжительность и продолжительность, основанная на состоянии

Данный раздел характеризует все определения, обусловленные временной продолжительностью и продолжительностью, основанной на состоянии.

10.1 Примитивная лексика временной продолжительности и продолжительности, основанной на состоянии

Лексика временной продолжительности и продолжительности, основанной на состоянии, не требует никаких примитивных соотношений.

10.2 Описываемая лексика временной продолжительности и продолжительности, основанной на состоянии

В данном подразделе определены нижеследующие соотношения:
- (mixed_duration ?a);
- (nondet_mixed_duration ?a);
- (rigid_mixed_duration ?a).

Каждое понятие определяется неформальной семантикой и аксиомами KIF.

10.3 Теории ядра, обусловленные временной продолжительностью и продолжительностью, основанной на состоянии

Для данных дефинициональных расширений необходимы:
- duration.th;
- disc_state;
- occtree.th;
- psl_core.th.

10.4 Дефинициональные расширения, обусловленные временной продолжительностью и продолжительностью, основанной на состоянии

Для данного расширения необходимо иметь нижеследующие дефинициональные расширения:
- actdur.def;
- time_precond.def;
- state_precond.def.

10.5 Определения временной продолжительности и продолжительности, основанной на состоянии

Определены нижеследующие понятия для временной продолжительности и продолжительности, основанной на состоянии.

10.5.1 mixed_duration

Некоторое действие является действием со смешанной продолжительностью при условии: если когда-либо два события согласованы по величинам и моментам времени их начала, то они согласованы и по продолжительности.

```
(forall (?a) (iff (mixed_duration ?a)
(forall (?occ1 ?occ2)
    (implies (and (occurrence ?occ1 ?a)
            (occurrence ?occ2 ?a)
            (state_equiv ?occ1 ?occ2)
            (begin_equiv ?occ1 ?occ2))
        (dur_equiv ?occ1 ?occ2))))
```

10.5.2 nondet_mixed_duration

Некоторое действие является недетерминированным действием со смешанной продолжительностью тогда и только тогда, когда существуют сохраняющие продолжительность переменный автоморфизм и автоморфизм начала.

```
(forall (?a) (iff (nondet_mixed_duration ?a)
(and (exists (?occ1)
    (forall (?occ2)
        (implies (and (occurrence ?occ1 ?a)
                (occurrence ?occ2 ?a)
                (state_equiv ?occ1 ?occ2)
                (begin_equiv ?occ1 ?occ2))
            (dur_equiv ?occ1 ?occ2))))
```

```
(exists (?occ3 ?occ4)
    (and (occurrence ?occ3 ?a)
        (occurrence ?occ3 ?a)
        (begin_equiv ?occ3 ?occ4)
        (state_equiv ?occ1 ?occ2)
        (not (dur_equiv ?occ3 ?occ4)))))
```

10.5.3 rigid_mixed_duration

Некоторое действие является жестким действием со смешанной продолжительностью тогда и только тогда, когда сохраняющие продолжительность переменный автоморфизм и автоморфизм начала являются тривиальными.

```
(forall (?a) (iff (rigid_mixed_duration ?a)
(forall (?occ1)
    (implies (occurrence ?occ1 ?a)
        (exists (?occ2)
            (and (occurrence ?occ2 ?a)
                (begin_equiv ?occ1 ?occ2)
                (state_equiv ?occ1 ?occ2)
                (not (dur_equiv ?occ1 ?occ2))))))))
```

10.6 Грамматика временной продолжительности и продолжительности, основанной на состоянии

Нижеследующие грамматические утверждения дают описания технологического процесса, определенные в KIF для временной продолжительности и продолжительности, основанной на состоянии.

```
< mix_dur_spec > ::= (forall (?occ)
    (implies (and (occurrence ?occ < term >)
        < simple_mix_axiom >)
            < duration_literal >))
< nondet_mix_dur_spec > ::= (forall (?occ)
        (implies (and (occurrence ?occ < term >)
            < mix_formula >)
            < duration_literal >)) |
        (forall (?occ)
            (implies (and (occurrence ?occ < term >)
                < simple_mix_axiom >)
                < interval_literal >))
```

11 Упорядочивание и ограничения продолжительности событий

Данный раздел характеризует все определения, обусловленные упорядочиванием и ограничениями продолжительности событий.

11.1 Примитивная лексика упорядочивания и ограничений продолжительности событий

Лексика упорядочивания и ограничений продолжительности событий не требует примитивных соотношений.

11.2 Описываемая лексика упорядочивания и ограничений продолжительности событий

В данном подразделе определены нижеследующие соотношения:

- (ordered_duration ?a);
- (partial_ordered_duration ?a);
- (unordered_duration ?a).

Каждое понятие определяется неформальной семантикой и аксиомами KIF.

11.3 Теории ядра, обусловленные упорядочиванием и ограничениями продолжительности событий

Для данных дефинициональных расширений необходимы:

- duration.th;

- act_occ.th;
- complex.th;
- atomic.th;
- subactivity.th;
- occtree.th;
- psl_core.th.

11.4 Дефинициональные расширения, обусловленные упорядочиванием и ограничениями продолжительности событий

Для данного расширения необходимо иметь:
- permute.def.

11.5 Определения упорядочивания и ограничений продолжительности событий

Для упорядочивания и ограничений продолжительности событий определены нижеследующие понятия.

11.5.1 ordered_duration

Некоторое действие имеет упорядоченную продолжительность тогда и только тогда, когда все ветви автоморфных событий имеют одинаковую продолжительность.

```
(forall (?a) (iff (ordered_duration ?a)
(forall (?occ1 ?occ2)
    (implies (and (occurrence ?occ1 ?a)
                  (occurrence ?occ2 ?a)
                  (branch_automorphic ?a ?occ1 ?occ2))
             (duration_equiv ?occ1 ?occ2)))))
```

11.5.2 partial_ordered_duration

Некоторое действие имеет контекстную продолжительность тогда и только тогда, когда существуют события, для которых все ветви автоморфных событий имеют одинаковую продолжительность.

```
(forall (?a) (iff (partial_ordered_duration ?a)
(and (exists (?occ1)
    (forall (?occ2)
        (implies (and (occurrence ?occ1 ?a)
                      (occurrence ?occ2 ?a)
                      (branch_automorphic ?a ?occ1 ?occ2))
                 (dur_equiv ?occ1 ?occ2))))
    (exists (?occ3 ?occ4)
    (and (occurrence ?occ3 ?a)
         (occurrence ?occ3 ?a)
         (branch_automorphic ?a ?occ1 ?occ2)
         (not (dur_equiv ?occ3 ?occ4)))))))
```

11.5.3 unordered_duration

Некоторое действие имеет неупорядоченную продолжительность тогда и только тогда, когда для каких-либо событий существуют ветви автоморфных событий, имеющие различную продолжительность.

```
(forall (?a) (iff (unordered_duration ?a)
(exists (?occ1 ?occ2)
    (and (occurrence ?occ1 ?a)
         (occurrence ?occ2 ?a)
         (branch_automorphic ?a ?occ1 ?occ2)
         (not (duration_equiv ?occ1 ?occ2))))))
```

11.6 Грамматика описаний процесса для упорядочивания и ограничений продолжительности событий

Нижеследующие грамматические утверждения дают описания технологического процесса, определенные в KIF для упорядочивания и ограничений продолжительности событий.

```
< order_dur_spec > ::= (forall (?occ)
                        (implies (and (occurrence ?occ < term >)
```

< ordered_sentence >)
< duration_literal >))
< partial_order_dur_spec > ::=
(forall (?occ)
(implies (and (occurrence ?occ < term >)
< ordered_formula >)
< duration_literal >)) |
(forall (?occ)
(implies (and (occurrence ?occ < term >)
< ordered_sentence >)
< interval_literal >))

12 Упорядочивание и ограничения продолжительности встроенных событий

Данный раздел характеризует все определения, обусловленные упорядочиванием и ограничениями продолжительности встроенных событий.

12.1 Примитивная лексика упорядочивания и ограничений продолжительности встроенных событий

Лексика упорядочивания и ограничений продолжительности встроенных событий не требует никаких примитивных соотношений.

12.2 Описываемая лексика упорядочивания и ограничений продолжительности встроенных событий

В данном подразделе определены нижеследующие соотношения:
- (embed_duration ?a);
- (partial_embed_duration ?a);
- (nonembed_duration ?a).

Каждое понятие определяется неформальной семантикой и аксиомами KIF.

12.3 Теории ядра, обусловленные упорядочиванием и ограничениями продолжительности встроенных событий

Для данных теорий необходимы:
- duration.th;
- act_occ.th;
- complex.th;
- atomic.th;
- subactivity.th;
- occtree.th;
- psl_core.th.

12.4 Дефинициональные расширения, обусловленные упорядочиванием и ограничениями продолжительности встроенных событий

Для упорядочивания и ограничений продолжительности встроенных событий необходимо иметь:
- permute.def.

12.5 Определения упорядочивания и ограничений продолжительности встроенных событий

Для упорядочивания и ограничений продолжительности встроенных событий определены нижеследующие понятия.

12.5.1 embed_duration

Некоторое действие имеет встроенную продолжительность тогда и только тогда, когда все ветви автоморфных событий этого же дерева имеют одинаковую продолжительность.

(forall (?a) (iff (embed_duration ?a)
(forall (?occ1 ?occ2)

```
(implies (and (occurrence ?occ1 ?a)
              (occurrence ?occ2 ?a)
              (same_grove ?occ1 ?occ2)
              (branch_automorphic ?occ1 ?occ2))
         (duration_equiv ?occ1 ?occ2)))))
```

12.5.2 partial_embed_duration

Некоторое действие имеет частично встроенную продолжительность тогда и только тогда, когда существуют события, для которых все ветви автоморфных событий этого же дерева имеют одинаковую продолжительность.

```
(forall (?a) (iff (partial_embed_duration ?a)
(and (exists (?occ1)
       (forall (?occ2)
              (implies (and (occurrence ?occ1 ?a)
                            (occurrence ?occ2 ?a)
                       (same_grove ?occ1 ?occ2)
                       (branch_automorphic ?occ1 ?occ2))
                      (dur_equiv ?occ1 ?occ2))))
(exists (?occ3 ?occ4)
      (and (occurrence ?occ3 ?a)
           (occurrence ?occ3 ?a)
         (same_grove ?occ1 ?occ2)
         (branch_automorphic ?occ1 ?occ2)
         (not (dur_equiv ?occ3 ?occ4)))))))
```

12.5.3 nonembed_duration

Некоторое действие имеет невстроенную продолжительность тогда и только тогда, когда для какого-либо события существуют ветви автоморфных событий этого же дерева, имеющие разную продолжительность.

```
(forall (?a) (iff (nonembed_duration ?a)
(exists (?occ1 ?occ2)
        (and (occurrence ?occ1 ?a)
             (occurrence ?occ2 ?a)
             (same_grove ?occ1 ?occ2)
             (branch_automorphic ?occ1 ?occ2)
             (not (duration_equiv ?occ1 ?occ2))))))
```

12.6 Грамматика упорядочивания и ограничений продолжительности встроенных событий

Грамматика описаний процесса для упорядочивания и ограничений продолжительности встроенных событий.

```
< embed_dur__spec > ::= (forall (?occ)
                        (implies (and (occurrence ?occ < term >)
                                      (same_tree < variable > ?occ))
                                 < duration_literal >))
< embed_interval_spec > ::= (forall (?occ)
                        (implies (and (occurrence ?occ < term >)
                                      (same_tree < variable > ?occ))
                                 < interval_axiom >))
```

13 Входные условия нарушения выполнения действий

Данный раздел характеризует все определения, обусловленные входными условиями нарушения выполнения действий.

13.1 Примитивная лексика входных условий нарушения выполнения действий

Лексика входных условий нарушения выполнения действий не требует никаких примитивных соотношений.

13.2 Описываемая лексика входных условий нарушения выполнения действий

В данном подразделе определены нижеследующие соотношения:
- (spoilage ?a);
- (possible_spoilage ?a);
- (nonspoilage ?a).

Каждое понятие определяется неформальной семантикой и аксиомами KIF.

13.3 Теории, обусловленные входными условиями нарушения выполнения действий

Для данной теории необходимы:
- duration.th;
- occtree.th;
- psl_core.th.

13.4 Дефинициональные расширения, обусловленные входными условиями нарушения выполнения действий

Для входных условий нарушения выполнения действий необходимы нижеследующие дефинициональные расширения:
- occ_precond.def;
- time_precond.def;
- precond.def.

13.5 Определения входных условий нарушения выполнения действий

Для входных условий нарушения выполнения действий определены нижеследующие понятия.

13.5.1 spoilage

Выполнение некоторого действия нарушено тогда и только тогда, когда допустимые события зависят от задержки между другими событиями. В особенности это события, находящиеся как на одной орбите эндоморфизмов дерева событий, так и на одной орбите автоморфизмов задержки.

```
(forall (?a) (iff (spoilage ?a)
(forall (?s1 ?s2)
    (implies (and (occurrence_of ?s1 ?a)
            (occurrence_of ?s2 ?a)
            (delay_equiv ?s1 ?s2)
            (tree_equiv ?s1 ?s2))
        (legal_equiv ?s1 ?s2)))))
```

13.5.2 possible_spoilage

Выполнение некоторого действия возможно нарушено тогда и только тогда, когда существуют события, находящиеся как на одной орбите эндоморфизмов дерева событий, так и на одной орбите автоморфизмов задержки.

```
(forall (?a) (iff (possible_spoilage ?a)
(exists (?s1)
    (and (occurrence_of ?s1 ?a)
        (forall (?s2)
            (implies (and (occurrence_of ?s2 ?a)
                    (delay_equiv ?s1 ?s2)
                    (tree_equiv ?s1 ?s2))
                (legal_equiv ?s1 ?s2)))))))
```

13.5.3 nonspoilage

Выполнение некоторого действия не нарушено тогда и только тогда, когда отсутствует связь между допустимыми событиями и задержкой между другими событиями. При этом единственный эндоморфизм дерева событий, сохраняющий задержку между другими событиями, является тривиальным.

```
(forall (?a) (iff (nonspoilage ?a)
(forall (?s1)
    (implies (occurrence_of ?s1 ?a)
        (exists (?s2)
```

```
                (and (occurrence_of ?s2 ?a)
                     (delay_equiv ?s1 ?s2)
                     (tree_equiv ?s1 ?s2)
                     (not (legal_equiv ?s1 ?s2))))))))
```

13.6 Грамматика описаний процесса для входных условий нарушения выполнения действий

Грамматика описаний процесса для входных условий нарушения выполнения действий.

```
< occ_constrained_precond > ::=
(forall (?s)
    ({implies | iff} (and (occurrence ?s < term >)
                          (legal ?s)
                          (ubiquitous < term > < term >))
                     {< leaf_delay_axiom > | < inner_delay_axiom}))
          < leaf_delay_formula > ::= (exists (?occ ?s1)
                (and (occurrence ?occ < term >)
                     (leaf_occ ?s1 ?occ)
                     < delay_literal >
                     (= ?s (successor < term > ?s1))))
< leaf_delay_axiom > ::= < leaf_delay_formula > |
                (not < leaf_delay_formula >)
< inner_delay_formula > ::= (exists (?occ ?s1 ?s2)
                (and (occurrence ?occ < term >)
                     < delay_literal >
                     (subactivity_occurrence ?s1 ?occ)
                     (subactivity_occurrence ?s2 ?occ)
                     (precedes ?s1 ?s)
                     (precedes ?s ?s2)))
< inner_delay_axiom > ::= < inner_delay_formula > |
                (not < inner_delay_formula >)
```

14 Запланированные встраивающие ограничения

Данный раздел характеризует все определения, обусловленные запланированными встраивающими ограничениями.

14.1 Примитивная лексика запланированных встраивающих ограничений

Лексика запланированных встраивающих ограничений не требует никаких примитивных соотношений.

14.2 Описываемая лексика запланированных встраивающих ограничений

В данном подразделе определены нижеследующие соотношения:
- (scheduled ?occ);
- (partial_scheduled ?occ);
- (unscheduled ?occ).

Каждое понятие определяется неформальной семантикой и аксиомами KIF.

14.3 Теории ядра, обусловленные запланированными встраивающими ограничениями

Для данной теории необходимы:
- duration.th;
- act_occ.th;
- complex.th;
- atomic .th;
- subactivity.th;
- occtree.th;
- psl_core.th.

14.4 Дефиниціональные расширения, обусловленные запланированными встраивающими ограничениями

Для данного расширения необходимы нижеследующие дефиниціональные расширения:
- embedding.def;
- time_precond.def.

14.5 Определения запланированных встраивающих ограничений

Для запланированных встраивающих ограничений определены нижеследующие понятия.

14.5.1 scheduled

Некоторое событие ?occ запланировано тогда и только тогда, когда каждое событие из ?occ ограничено задержкой.

```
(forall (?occ) (iff (scheduled ?occ)
(forall (?a ?s1 ?s2 ?s3)
    (implies (and (occurrence_of ?occ ?a)
            (root_occ ?s3 ?occ)
            (subactivity_occurrence ?s1 ?occ)
            (iso_occ ?s1 ?s2)
            (embed_tree ?s1 ?s2 ?s3 ?a)
            (delay_equiv ?s1 ?s2))
        (subocc_equiv ?s1 ?s2 ?s3 ?a)))))
```

14.5.2 partial_scheduled

Некоторое событие ?occ запланировано частично тогда и только тогда, когда некоторые подсобытия из ?occ ограничены задержкой.

```
(forall (?occ) (iff (partial_scheduled ?occ)
(exists (?a ?s ?s1 ?s3 ?s4)
    (and (occurrence_of ?occ ?a)
            (root_occ ?s ?occ)
            (subactivity_occurrence ?s1 ?occ)
            (forall (?s2)
                (implies (and (iso_occ ?s1 ?s2)
                        (embed_tree ?s1 ?s2 ?s ?a)
                        (delay_equiv ?s1 ?s2))
                    (subocc_equiv ?s1 ?s2 ?s ?a))))
            (subactivity_occurrence ?s3 ?occ)
            (iso_occ ?s3 ?s4)
            (delay_equiv ?s3 ?s4)
            (embed_tree ?s3 ?s4 ?s ?a)
            (not (subocc_equiv ?s3 ?s4 ?s ?a))))))
```

14.5.3 unscheduled

Некоторое событие ?occ не запланировано тогда и только тогда, когда ни одно из подсобытий ?occ не ограничено задержкой.

```
(forall (?occ) (iff (unscheduled ?occ)
(forall (?s ?s1 ?a)
    (implies (and (occurrence_of ?occ ?a)
            (root_occ ?s ?occ)
            (subactivity_occurrence ?s1 ?occ))
        (exists (?s2)
            (and (iso_occ ?s1 ?s2)
                (embed_tree ?s1 ?s2 ?s ?a)
                (delay_equiv ?s1 ?s2)
                (not (subocc_equiv ?s1 ?s2 ?s ?a)))))))
```

14.6 Грамматика запланированных встраивающих ограничений

Грамматика описаний процесса для запланированных встраивающих ограничений:

```
< schedule_axiom > ::= (forall (?s ?occ < variable >*)
```

```
                    < schedule_formula >)
< partial_schedule_axiom > ::= (forall (?s ?occ < variable >+)
                    < partial_schedule_sentence >))
< schedule_formula > ::= ({implies | and} < subocc_formula >
                    < delay_literal >)
< schedule_axiom > ::= < schedule_formula> |
            (and < schedule_axiom > < schedule_axiom >+)
< partial_schedule_formula > ::= ({implies | and} < subocc_formula >
                        < delay_interval_axiom >)
< partial_schedule_sentence > ::=
    < partial_schedule_formula> |
    (and < partial_schedule_sentence > < partial_schedule_sentence >+)
```

15 Эффекты, основанные на продолжительности

Данный раздел характеризует все определения, обусловленные эффектами, основанными на продолжительности.

15.1 Примитивная лексика эффектов, основанных на продолжительности

Лексика эффектов, основанных на продолжительности, не требует никаких примитивных соотношений.

15.2 Описываемая лексика эффектов, основанных на продолжительности

В данном подразделе определены следующие соотношения:
- (duration_effects ?a);
- (partial_duration_effects ?a);
- (nonduration_effects ?a).

Каждое понятие определяется неформальной семантикой и аксиомами KIF.

15.3 Теории ядра, обусловленные эффектами, основанными на продолжительности

Для данной теории необходимы:
- duration.th;
- disc_state.th;
- occtree.th;
- psl_core.th.

15.4 Дефинициональные расширения, обусловленные эффектами, основанными на продолжительности

Для эффектов, основанных на продолжительности, необходимы нижеследующие дефинициональные расширения:
- actdur.def;
- effects.def.

15.5 Определения эффектов, основанных на продолжительности

Для эффектов, основанных на продолжительности, определены нижеследующие понятия.

15.5.1 duration_effects

Некоторое действие имеет эффекты, основанные на продолжительности, при условии: если когда-либо какие-либо два события согласованы по продолжительности, то они согласованы и по эффектам (то есть по состоянию, сохраняющемуся после события).

```
(forall (?a) (iff (duration_effects ?a)
(forall (?s1 ?s2)
    (implies (and (occurrence ?s1 ?a)
            (occurrence ?s2 ?a)
            (dur_equiv ?s1 ?s2))
        (effects_equiv ?a ?s1 ?s2)))))
```

15.5.2 partial_duration_effects

Некоторое действие частично ограничено по продолжительности тогда и только тогда, когда существуют сохраняющие эффект автоморфизмы продолжительности.

```
(forall (?a) (iff (partial_duration_effects ?a)
(and (exists (?s1)
       (forall (?s2)
             (implies (and (occurrence ?s1 ?a)
                           (occurrence ?s2 ?a)
                           (dur_equiv ?s1 ?s2))
                      (effects_equiv ?a ?s1 ?s2))))
     (exists (?s3 ?s4)
        (and (occurrence ?s3 ?a)
             (occurrence ?s4 ?a)
             (dur_equiv ?s3 ?s4)
               (not (effects_equiv ?a ?s3 ?s4)))))))
```

15.5.3 nonduration_constraints

Некоторое действие имеет эффекты непродолжительности тогда и только тогда, когда единственный сохраняющий эффект автоморфизм продолжительности является тривиальным.

```
(forall (?a) (iff (nonduration_effects ?a)
(forall (?s1)
     (implies (occurrence ?s1 ?a)
           (exists (?s2)
                 (and (occurrence ?s2 ?a)
                      (dur_equiv ?s1 ?s2)
                      (not (effects_equiv ?a ?s1 ?s2)))))))))
```

15.6 Грамматика эффектов, основанных на продолжительности

Грамматика описаний процесса для эффектов, основанных на продолжительности.

```
< duration_effect_axiom > ::= (forall (?s)
                      (implies (and (occurrence ?s <term >)
                                    < duration_literal >)
                               < simple_holds_axiom >)))
< partial_dur_effect > ::= (forall (< variable >*)
                      (implies (and (occurrence < variable > < term >)
                                    < interval_axiom >)
                               < simple_holds_axiom >)))
```

16 Эффекты действий, основанные на продолжительности и времени

Данный раздел характеризует все определения, обусловленные эффектами действий, основанными на продолжительности и времени.

16.1 Примитивная лексика эффектов действий, основанных на продолжительности и времени

Лексика эффектов действий, основанных на продолжительности и времени, не требует никаких примитивных соотношений.

16.2 Описываемая лексика эффектов действий, основанных на продолжительности и времени

В данном подразделе определены следующие соотношения:

- (maintain_effects ?a);
- (partial_maintain ?a);
- (nonmaintain ?a).

Каждое понятие определяется неформальной семантикой и аксиомами KIF.

16.3 Теории ядра, обусловленные эффектами действий, основанными на продолжительности и времени

Для данной теории необходимы:
- duration.th;
- disc_state.th;
- occtree.th;
- psl_core.th.

16.4 Дефинициональные расширения, обусловленные эффектами действий, основанными на продолжительности и времени

Для данного расширения необходимы нижеследующие дефинициональные расширения:
- actdur.def;
- effects.def;
- state_precond.def.

16.5 Определения эффектов действий, основанных на продолжительности и времени

Для эффектов действий, основанных на продолжительности и времени, определены нижеследующие понятия.

16.5.1 maintain_effects

Некоторое действие поддерживает эффект при условии: если когда-либо какие-либо два события согласованы по продолжительности и состоянию, то они согласованы и по эффектам (то есть по состоянию, сохраняющемуся после события).

```
(forall (?a) (iff (maintain_effects ?a)
(forall (?s1 ?s2)
    (implies (and (occurrence ?s1 ?a)
                  (occurrence ?s2 ?a)
                  (state_equiv ?s1 ?s2)
                  (dur_equiv ?s1 ?s2))
             (effects_equiv ?a ?s1 ?s2)))))
```

16.5.2 partial_maintain

Некоторое действие поддерживает эффект частично тогда и только тогда, когда существуют сохраняющие эффект автоморфизмы, сохраняющие как продолжительность, так и состояние.

```
(forall (?a) (iff (partial_maintain ?a)
(and (exists (?s1)
       (forall (?s2)
           (implies (and (occurrence ?s1 ?a)
                         (occurrence ?s2 ?a)
                         (state_equiv ?s1 ?s2)
                         (dur_equiv ?s1 ?s2))
                    (effects_equiv ?a ?s1 ?s2))))
     (exists (?s3 ?s4)
         (and (occurrence ?s3 ?a)
              (occurrence ?s4 ?a)
              (dur_equiv ?s3 ?s4)
                 (not (effects_equiv ?a ?s3 ?s4)))))))
```

16.5.3 nonmaintain

Некоторое действие не поддерживает эффект тогда и только тогда, когда сохраняющие эффект автоморфизм продолжительности и автоморфизм переменных величин являются тривиальными.

```
(forall (?a) (iff (nonmaintain ?a)
(forall (?s1)
    (implies (occurrence ?s1 ?a)
             (exists (?s2)
                 (and (occurrence ?s2 ?a)
                      (dur_equiv ?s1 ?s2)
                      (not (effects_equiv ?a ?s1 ?s2)))))))))
```

16.6 Грамматика эффектов действий, основанных на продолжительности и времени

Грамматика описаний процесса для эффектов действий, основанных на продолжительности и времени.

```
< maintain_effect_axiom > ::= (forall (?s)
                    (implies (and (occurrence ?s <term >)
                                  < duration_literal >
                                  < simple_state_axiom >)
                                < simple_holds_axiom >)))
< partial_maintain_axiom > ::= (forall (< variable >)
                    (implies (and (occurrence < variable > < term >)
                                  < duration_literal >
                                  < state_axiom >)
                                < simple_holds_axiom >))) |
                  (forall (< variable >*)
                    (implies (and (occurrence < variable > < term >)
                                  < interval_literal >
                                  < simple_state_axiom >)
                                < simple_holds_axiom >))) |
```

17 Упорядочивание соотношений в комплексной последовательности

Данный раздел характеризует все определения, обусловленные упорядочиванием соотношений в комплексной последовательности.

17.1 Примитивная лексика упорядочивания соотношений в комплексной последовательности

Лексика упорядочивания соотношений в комплексной последовательности не требует никаких примитивных соотношений.

17.2 Описываемая лексика упорядочивания соотношений в комплексной последовательности

В данном подразделе определены нижеследующие соотношения:
- (coo_precedes ?occ1 ?occ2 ?a);
- (strong_parallel ?occ1 ?occ2 ?a);
- (atomocc ?occ).

Каждое понятие определяется неформальной семантикой и аксиомами KIF.

17.3 Теории, обусловленные упорядочиванием соотношений в комплексной последовательности

Для данной теории необходимы:
- soo.th;
- actocc.th;
- complex.th;
- atomic .th;
- subactivity.th;
- occtree.th;
- psl_core.th.

17.4 Дефинициональные расширения, обусловленные упорядочиванием соотношений в комплексной последовательности

Для данного расширения необходимо нижеследующее дефинициональное расширение:
- strongposets.def.

17.5 Определения упорядочивания соотношений в комплексной последовательности

Для упорядочивания соотношений в комплексной последовательности определены нижеследующие понятия.

17.5.1 coo_precedes

Некоторое комплексное событие ?occ1 имеет свойство «coo_precedes» по отношению к комплексному событию ?occ2, если каждое неделимое подсобытие ?occ1 также имеет свойство «coo_precedes» по отношению к каждому неделимому подсобытию ?occ2.

```
(forall (?occ1 ?occ2 ?a) (iff (coo_precedes ?occ1 ?occ2 ?a)
(forall (?s1 ?s2)
    (implies (and       (subactivity_occurrence ?s1 ?occ1)
                        (subactivity_occurrence ?s2 ?occ2)
                        (atomocc ?s1)
                        (atomocc ?s2))
        (soo_precedes ?s1 ?s2 ?a)))))
```

17.5.2 strong_parallel

Некоторое комплексное событие ?occ1 параллельно комплексному событию ?occ2, если какое-либо неделимое подсобытие ?occ1 обладает свойством «same_bag» в отношении неделимого подсобытия ?occ2.

```
(forall (?occ1 ?occ2 ?a) (iff (strong_parallel ?occ1 ?occ2 ?a)
(forall (?s1 ?s2)
    (implies (and       (subactivity_occurrence ?s1 ?occ1)
                        (subactivity_occurrence ?s2 ?occ2)
                        (atomocc ?s1)
                        (atomocc ?s2))
        (same_bag ?s1 ?s2 ?a)))))
```

17.5.3 atomocc

Некоторое событие обладает свойством «atomocc», если оно заключается в выполнении неделимого действия.

```
(forall (?s) (iff (atomocc ?s)
(exists (?a)
    (and (atomic ?a)
        (occurrence_of ?s ?a)))))
```

Приложение А
(справочное)

ASN.1 Идентификатор настоящего стандарта

Для однозначной идентификации информационного объекта в открытой системе настоящему стандарту присвоен следующий идентификатор:

iso standard 18629 part 43 version 1

Значение данного идентификатора определено в ИСО/МЭК 8824-1 и детально описано в ИСО 18629-1.

Приложение В
(справочное)

Пример описания технологического процесса в соответствии с настоящим стандартом

В данном приложении рассмотрен подробный сценарий использования языка спецификаций процесса PSL (Process Specification Language) в соответствии с ИСО 18629. Далее рассмотрен частный случай использования программного описания технологического процесса.

Данный сценарий включает совместное выполнение нескольких операций. Целью является повышение эффективности использования данных о технологическом процессе при изготовлении изделия. Подчеркнем, что данный язык программирования прежде всего позволяет повысить эффективность компьютерного обмена данными между сотрудниками планового отдела и руководством производственного подразделения.

В данном приложении рассмотрено расширение примера, использованного в ИСО 18629-11:2005 (приложение С). Пример иллюстрирует технические приложения внешних понятий для спецификации процесса изготовления изделия GT-350.

В.1 Процесс изготовления изделия GT-350

В данном разделе различные производственные процессы объединены в набор действий высокого уровня, необходимых для создания изделия GT-350. В соответствии с технологической картой изделия GT-350 (см. ИСО 18629-11:2005, таблица С.1) компоненты данного изделия либо покупают по контракту, либо изготовляют внутри самого предприятия. Рассматриваемые описания технологических процессов связаны с конкретными действиями, выполняемыми внутри предприятия для изготовления компонентов изделия. Данное рассмотрение технологического процесса в направлении «сверху – вниз» дает общую картину происходящего как комплексного действия по производству изделия GT-350 вплоть до составляющих его действий, выполняемых на уровне более мелких подразделений предприятия.

В соответствии с рисунком В.1 весь процесс изготовления изделия GT-350 организован в шести основных секторах. В первых пяти из них (изготовление интерьера, изготовление привода, изготовление кузова, изготовление двигателя и изготовление шасси) работы могут быть выполнены независимо друг от друга. Одно условие: они должны быть закончены к моменту начала общей сборки изделия.

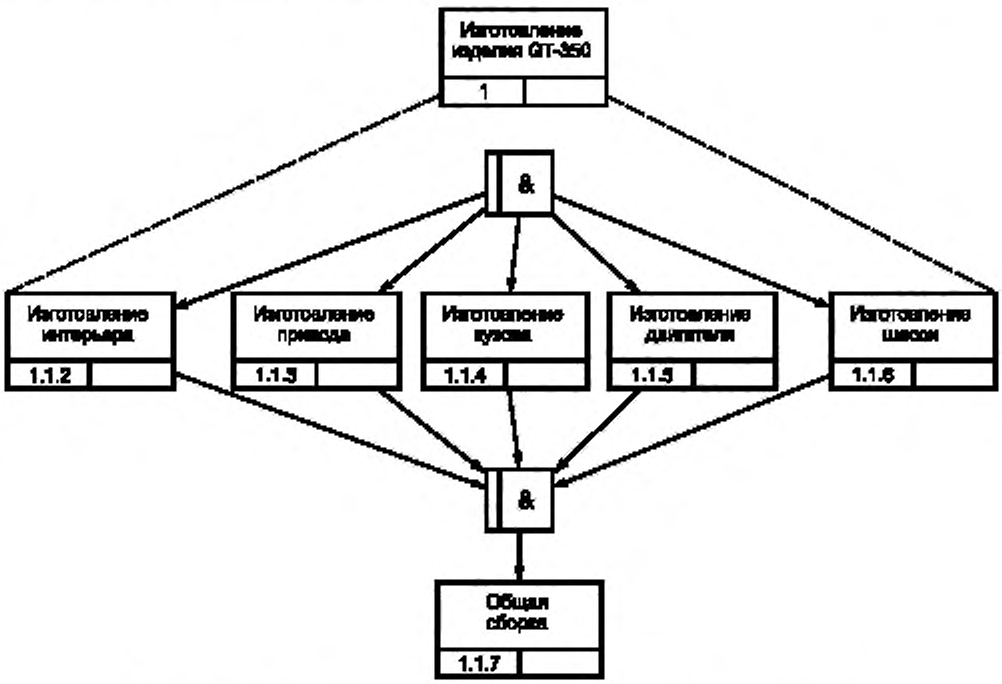

Рисунок В.1 — Верхний уровень процесса изготовления изделия GT-350 [5]

Представление верхнего уровня технологического процесса на языке PSL имеет вид:

```
(subactivity make-chassis make_gt350)
(subactivity make-interior make_gt350)
(subactivity make-drive make_gt350)
(subactivity make-trim make_gt350)
(subactivity make-engine make_gt350)
(subactivity final-assembly make_gt350)
(forall (?occ)
    (<=> (occurrence_of ?occ make_gt350)
        (exists (?occ1 ?occ2 ?occ3 ?occ4 ?occ5 ?occ6)
            (and (occurrence_of ?occ1 make_chassis)
                (occurrence_of ?occ2 make_interior)
                (occurrence_of ?occ3 make_drive)
                (occurrence_of ?occ4 make_trim)
                (occurrence_of ?occ5 make_engine)
                (occurrence_of ?occ6 final_assembly)
                (subactivity_occurrence ?occ1 ?occ)
                (subactivity_occurrence ?occ2 ?occ)
                (subactivity_occurrence ?occ3 ?occ)
                (subactivity_occurrence ?occ4 ?occ)
                (subactivity_occurrence ?occ5 ?occ)
                (subactivity_occurrence ?occ6 ?occ)
                (soo_precedes (soomap ?occ1) (soomap ?occ6) make_gt350)
                (soo_precedes (soomap ?occ2) (soomap ?occ6) make_gt350)
                (soo_precedes (soomap ?occ3) (soomap ?occ6) make_gt350)
                (soo_precedes (soomap ?occ4) (soomap ?occ6) make_gt350)
                (soo_precedes (soomap ?occ5) (soomap ?occ6) make_gt350)
                (strong_parallel ?occ1 ?occ2 make_gt350)
                (strong_parallel ?occ2 ?occ3 make_gt350)
                (strong_parallel ?occ3 ?occ4 make_gt350)
                (strong_parallel ?occ4 ?occ5 make_gt350)))))
(forall (?occ)
    (implies (occurrence_of make_gt350)
        (strong_poset ?occ)))
```

В данном представлении соотношение «soo_precedes» используется для спецификации упорядочивающих ограничений для событий «make_chassis» (изготовление шасси), «make_interior» (изготовление интерьера), «make_drive» (изготовление привода), «make_engine» (изготовление двигателя) и «final_assembly» (окончательная сборка). Каждая стрелка на рисунке В.1 соответствует формуле «soo_precedes». Функция «soomap» используется для выделения возможных многозначных событий.

Все поддействия «make_chassis», «make_interior», «make_drive» и «make_engine» выполняются параллельно. Поэтому их упорядочивающие ограничения описываются соотношением «strong_parallel». Событие «make_gt350» (изготовление изделия GT-350) является сильным «poset» (частично упорядоченным множеством), так как все параллельные поддействия должны закончиться до начала «final_assembly» (окончательной сборки).

Каждое указанное абстрактное действие можно детализировать далее. Однако в рамках примера в данном приложении указанная детализация не рассматривается.

На базе представления IDEF3 (в терминах представления технологического процесса) для краткого описания действий, встречающихся на различных стадиях процесса изготовления изделия, в настоящем стандарте приведены некоторые примеры использования языка программирования PSL-Outercore в соответствии с ИСО 18629-12.

В.2 Абстрактное действие «make_engine» (изготовление двигателя)

Двигатель изделия GT-350 собирается из агрегатов, изготовленных в нескольких подразделениях предприятия. Схема процесса изготовления дана на рисунке В.2. Агрегат состоит из двигательного блока, жгутов и кабелей. Составляющие процессы детально рассмотрены в подразделах ниже. Двигатель изделия GT-350 собирается на сборочном стенде А004. Сборка одного двигателя требует 5 мин.

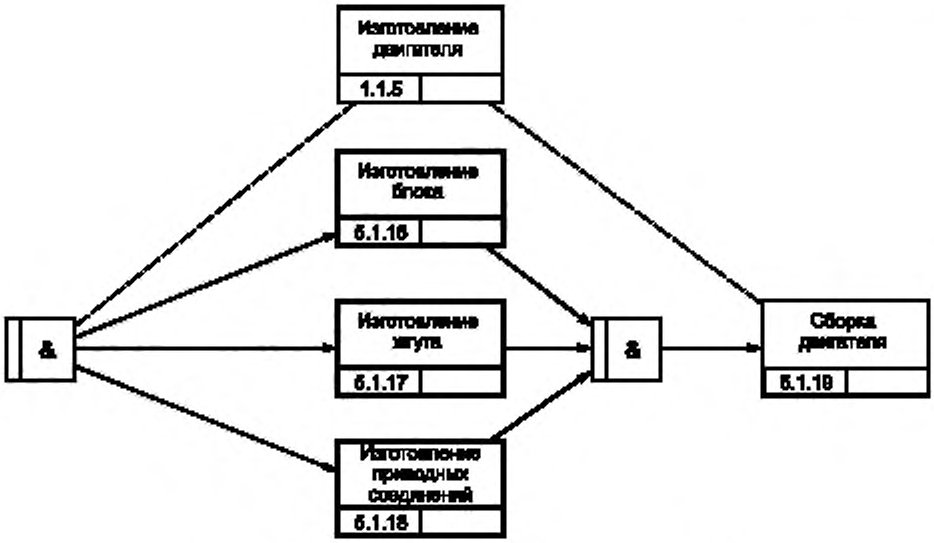

Рисунок В.2 — Процесс изготовления двигателя GT-350 [5]

Ниже некоторые действия и данные технологического процесса изготовления двигателя представлены на языке программирования PSL-Outercore:

```
(subactivity make_block make_engine)
(subactivity make-harness make_engine)
(subactivity make-wires make_engine)
(subactivity assemble_engine make_engine)
(forall (?occ)
(<=>      (occurrence_of ?occ make_engine)
            (exists (?occ1 ?occ2 ?occ3 ?occ4)
              (and (occurrence_of ?occ1 make_block)
                        (occurrence_of ?occ2 make_harness)
                        (occurrence_of ?occ3 make_wires)
                        (occurrence_of ?occ4 assemble_engine)
                        (= (duration (beginof ?occ1) (endof ?occ1)) 10)
                        (= (duration (beginof ?occ2) (endof ?occ2)) 5)
                        (= (duration (beginof ?occ3) (endof ?occ3)) 12)
                        (= (beginof ?occ4) (time_add (endof ?occ3) 10))
                        (subactivity_occurrence ?occ1 ?occ)
                        (subactivity_occurrence ?occ2 ?occ)
                        (subactivity_occurrence ?occ3 ?occ)
                        (subactivity_occurrence ?occ4 ?occ)
                        (soo_precedes (soomap ?occ1) (soomap ?occ4) make_gt350)
                        (soo_precedes (soomap ?occ2) (soomap ?occ4) make_gt350)
                        (soo_precedes (soomap ?occ3) (soomap ?occ4) make_gt350)))))
                        (strong_parallel ?occ1 ?occ2 make_gt350)
                        (strong_parallel ?occ2 ?occ3 make_gt350)))))))
(forall (?occ)
      (implies (occurrence_of make_engine)
            (strong_poset ?occ)))
```

В данном представлении соотношение «soo_precedes» используется для спецификации упорядочивающих ограничений для событий «make_block» (изготовление блока), «make_harness» (изготовление жгута), «make_wires» (изготовление проводов) и «make_engine» (изготовление двигателя). Каждая стрелка на рисунке В.2 соответствует формуле «soo_precedes». Функция «soomap» используется для выделения возможных многозначных событий.

Поддействия «make_block», «make_harness» и «make_wire» выполняются параллельно. Поэтому их упорядочивающие ограничения задаются соотношениями «strong_parallel». Событие «make_engine» является сильным «posets» (частично упорядоченным множеством), так как все параллельные поддействия должны завершиться перед действием «assemble_engine» (сборка двигателя).

Кроме того, данное представление устанавливает, что продолжительность события «make_block» равна 10, продолжительность события «make_harness» равна 5 и продолжительность события «make_wires» равна 12.

Окончательно данное представление использует функцию «time_add», которая устанавливает, что событие «assemble_engine» начинается через 10 временных единиц после окончания события «make_wires».

B.3 Событие «make_block»

Блок изделия GT-350 выполняется как агрегат для сборки двигателя изделия GT-350. Изготовление блока требует выполнения всех технологических операций, начиная от литья заготовки и ее механической обработки (см. рисунок B.3).

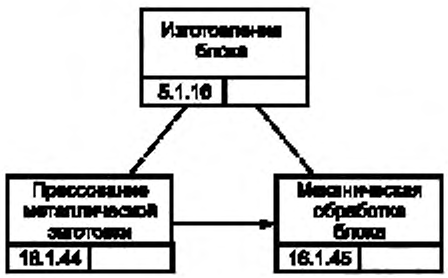

Рисунок B.3 — Процесс изготовления блока изделия GT-350 [5]

Представление некоторых действий и технологических данных на языке PSL-Outercore:

```
(subactivity produce_molded_metal make_block)
(subactivity machine_block make_block)
(primitive machine_block)
(primitive produce_molded_metal)
(forall (?occ)
        (<=> (occurrence_of ?occ make_block)
                (exists (?occ1 ?occ2)
                        (and (occurrence_of ?occ1 produce_molded_metal)
                                (occurrence_of ?occ2 machine_block)
                                (= (beginof ?occ2) (time_add (endof ?occ1) 12))
                                (soo_precedes (soomap ?occ1) (soomap ?occ2) make_block)))))
```

Данное представление использует функцию «time_add» для указания, что событие «machine_block» (механическая обработка блока) начинается через 12 временных единиц после окончания события «produce_ molded_ metal» (прессование металлической заготовки).

B.4 Событие «make_harness»

Жгут изделия GT-350 (см. рисунок B.4) изготовляют как сборочный агрегат двигателя изделия GT-350. Данный технологический процесс организован в цехе кабелей и проводов. Рисунок B.5 представляет процесс изготовления провода жгута. Жгут изделия GT-350 собирается на особом стенде из проводов и кабелей. Сборка одного жгута требует 10 мин.

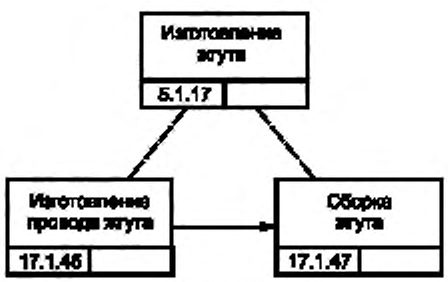

Рисунок B.4 — Процесс изготовления жгута изделия GT-350 [5]

Ниже дано представление некоторых действий и соответствующих технологических данных на языке программирования PSL-Outercore:

```
(subactivity make_harness_wire make_harness)
(subactivity assemble_harness make_harness)
(primitive assemble_harness)
(forall (?occ)
        (<=> (occurrence_of ?occ make_harness)
                (exists (?occ1 ?occ2 ?occ3)
                        (and (occurrence_of ?occ1 make_harness_wire)
                                (occurrence_of ?occ2 assemble_harness)
                                (leaf_occ ?occ3 ?occ1)
                                (soo_precedes (soomap ?occ3) (soomap ?occ2) make_harness)))))
```

Данное представление формализует спецификацию технологического процесса на рисунке В.5. В данном представлении соотношение «soo_precedes» используется для спецификации упорядочивающих ограничений для событий «make_harness_wires» (изготовление проводов жгута) и «assemble_wires» (сборка проводов). Каждая стрелка на рисунке В.5 соответствует формуле «soo_precedes». Функция «soomap» используется для выделения возможных многозначных событий.



Рисунок В.5 — Процесс изготовления провода жгута [5]

В.5 Изготовление проводов жгута

Набор проводов изделия GT-350 изготовляют как сборочный агрегат изделия GT-350. Технологический процесс организуют в цехе проводов и кабелей.



Рисунок В.6 — Процесс изготовления проводов изделия GT-350 [5]

Ниже дано представление некоторых действий и соответствующих технологических данных на языке программирования PSL-Outercore:

```
(subactivity extrude make_harness_wire)
(subactivity twist make_harness_wire)
(subactivity jacket make_harness_wire)
(primitive extrude)
(primitive twist)
(primitive jacket)
```

```
(forall (?occ)
        (<=> (occurrence_of ?occ make_harness_wire)
             (exists (?occ1 ?occ2 ?occ3 ?occ4)
                     (and (occurrence_of ?occ1 extrude)
                          (occurrence_of ?occ2 twist)
                          (occurrence_of ?occ3 jacket)
                        (occurrence_of ?occ4 assemble)
                          (soo_precedes (soomap ?occ1) (soomap ?occ2) make_harness_wire)
                          (soo_precedes (soomap ?occ3) (soomap ?occ4) make_harness_wire)
                          (soo_precedes (soomap ?occ2) (soomap ?occ3) make_harness_wire))))
```

Данное представление формализует процесс спецификации на рисунке В.6. В данном представлении соотношение «soo_precedes» используется для спецификации упорядочивающих ограничений для событий «extrude», «twist», «jacket» и «assemble». Каждая стрелка на рисунке В.6 соответствует формуле «soo_precedes». Функция «soomap» используется для выделения возможных многозначных событий.

Приложение ДА
(справочное)

Сведения о соответствии ссылочных международных стандартов национальным стандартам Российской Федерации

Таблица ДА.1

Обозначение ссылочного международного стандарта	Степень соответствия	Обозначение и наименование соответствующего национального стандарта
ИСО/МЭК 8824-1	IDT	ГОСТ Р ИСО/МЭК 8824-1—2001 «Информационная технология. Абстрактная синтаксическая нотация версии один (АСН.1). Часть 1. Спецификация основной нотации»
ИСО15531-1	IDT	ГОСТ Р ИСО 15531-1:2008 «Промышленные автоматизированные системы и интеграция. Данные по управлению промышленным производством. Часть 1. Общий обзор»
ИСО15531-42:2005	IDT	ГОСТ Р ИСО 15531-42—2010 «Системы промышленной автоматизации и интеграция. Данные по управлению промышленным производством. Часть 42. Модель времени»
ИСО 18629-1:2004	—	*
ИСО 18629-11:2005	—	*
ИСО 18629-12:2005	—	*
ИСО 18629-13:2006	—	*

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

Примечание — В настоящей таблице использовано следующее условное обозначение степени соответствия стандартов:

- IDT — идентичные стандарты.

Библиография

[1] ИСО 10303-1 Системы промышленной автоматизации и интеграция. Представление данных о продукции и обмен данными. Часть 1. Обзор и основные принципы

[2] ИСО 10303-49 Системы промышленной автоматизации и интеграция. Представление данных о продукции и обмен данными. Часть 49. Интегрированные родовые ресурсы: структура и свойства процесса

[3] ИСО 18629-14 Системы промышленной автоматизации и интеграция. Язык спецификаций процесса. Часть 14. Теории ресурсов

[4] ИСО 18629-44 Системы промышленной автоматизации и интеграция. Язык спецификаций процесса. Часть 44. Дефинициональное расширение: расширение ресурсов

[5] Federal Information Processing Standards Publication 184, Integration Definition for Information Modeling (IDEF3), FIPS PUB 184, National Institute of Standards and Technology, December 1993. IDEF3. Available from the Internet: <gost.e-ecolog.ru

[6] Публикация федерального стандарта переработки информации 184, Определение интеграции для информационного моделирования IDEF3), FIPS PUB 184, Национальный институт стандартов и технологии, декабрь 1993. IDEF3. Интернет: (<gost.e-ecolog.ru

УДК 65.011:56.681.3 ОКС 25.040.40 Т58

Ключевые слова: автоматизированные промышленные системы, интеграция, жизненный цикл систем, управление производством

Редактор *А. Д. Чайка*
Технический редактор *В. Н. Прусакова*
Корректор *Л. Я. Митрофанова*
Компьютерная верстка *В. Н. Романовой*

Сдано в набор 22.11.2013. Подписано в печать 03.04.2014. Формат 60×84 1/8. Бумага офсетная. Гарнитура Ариал. Печать офсетная. Усл. печ. л. 4,65. Уч.-изд. л. 4,05. Тираж 74 экз. Зак. 156

ФГУП «СТАНДАРТИНФОРМ». 123995 Москва, Гранатный пер., 4.
gost.e-ecolog.ru gost.e-ecolog.ru
Набрано и отпечатано в Калужской типографии стандартов, 248021 Калуга, ул. Московская, 256.

gost.e-ecolog.ru