ФЕДЕРАЛЬНОЕ АГЕНТСТВО

ПО ТЕХНИЧЕСКОМУ РЕГУЛИРОВАНИЮ И МЕТРОЛОГИИ

НАЦИОНАЛЬНЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

ГОСТ Р ИСО 18629-12 — 2010

Системы промышленной автоматизации и интеграция

ЯЗЫК СПЕЦИФИКАЦИЙ ПРОЦЕССА

Часть 12

Внешнее ядро

ISO 18629-12:2005
Industrial automation systems and integration — Process specification language — Part 12: Outer core
(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 21 декабря 2010 г. № 894-ст

4 Настоящий стандарт идентичен международному стандарту ИСО 18629-12:2005 «Системы промышленной автоматизации и интеграция. Язык спецификаций процесса. Часть 12. Внешнее ядро» (ISO 18629-12:2005 «Industrial automation systems and integration — Process specification language — Part 12: Outer core»).

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 . . . 5
4 Общие положения комплекса стандартов ИСО 18629 . . . 5
5 Общие положения настоящего стандарта . . . 5
5.1 Расширения, входящие во внешнее ядро . . . 5
6 Теория ядра подчиненной деятельности . . . 6
6.1 Примитивные отношения . . . 6
6.2 Установленные отношения . . . 6
6.3 Связь с другими наборами аксиом . . . 6
6.4 Неформальная семантика . . . 6
6.5 Определения . . . 7
6.6 Аксиомы . . . 7
7 Теория ядра дерева случаев . . . 8
7.1 Примитивные отношения . . . 8
7.2 Примитивные функции . . . 8
7.3 Связь с другими наборами аксиом . . . 8
7.4 Неформальная семантика . . . 8
7.5 Аксиомы . . . 9
8 Теория ядра дискретного состояния . . . 11
8.1 Примитивные категории . . . 11
8.2 Примитивные отношения . . . 11
8.3 Связь с другими наборами аксиом . . . 12
8.4 Неформальная семантика . . . 12
8.5 Аксиомы . . . 12
9 Теория ядра атомарной деятельности . . . 13
9.1 Примитивные отношения . . . 13
9.2 Примитивные функции . . . 14
9.3 Связь с другими наборами аксиом . . . 14
9.4 Неформальная семантика . . . 14
9.6 Аксиомы . . . 14
10 Теория ядра комплексной деятельности . . . 15
10.1 Примитивные отношения . . . 15
10.2 Установленные отношения . . . 15
10.3 Связь с другими наборами аксиом . . . 16
10.4 Неформальная семантика . . . 16
10.5 Определения . . . 17
10.6 Аксиомы . . . 18
11 Теория ядра случаев деятельности . . . 19
11.1 Примитивные отношения . . . 20
11.2 Установленные отношения в теории ядра случаев деятельности . . . 20
11.3 Связь с другими наборами аксиом . . . 20
11.4 Неформальная семантика . . . 20
11.5 Определения . . . 21
11.6 Аксиомы . . . 22
Приложение А (обязательное) Идентификатор ASN.1, присвоенный ИСО 18629-12 . . . 25
Приложение В (справочное) Связь с ситуационным исчислением . . . 26
Приложение С (справочное) Пример описания процесса с помощью внешнего ядра PSL . . . 27
Приложение ДА (справочное) Сведения о соответствии ссылочных международных стандартов ссылочным национальным стандартам Российской Федерации . . . 32
Библиография . . . 33

Введение

Комплекс стандартов ИСО 18629 устанавливает требования к интерпретируемому компьютером обмену информацией о производственном процессе. Стандарты комплекса определяют родовой язык, используемый для описания производственного процесса, осуществляемого одной и той же промышленной компанией или несколькими ее промышленными секторами, или разными компаниями, независимо от любой частной модели представления. Характер этого языка подходит для совместного использования информации, относящейся к производству, на всех стадиях производственного процесса.

В настоящем стандарте приведено описание базовых элементов ядра, определенного в комплексе стандартов ИСО 18629.

Стандарты комплекса ИСО 18629 не зависят от какого-либо представления специфического процесса или модели, предложенной в приложении программного обеспечения в домене управления производством. Комплекс стандартов ИСО 18629 определяет структурную основу, обеспечивающую возможность взаимодействия приложений программного обеспечения.

НАЦИОНАЛЬНЫЙ СТАНДАРТ РОССИЙСКОЙ ФЕДЕРАЦИИ

Системы промышленной автоматизации и интеграция

ЯЗЫК СПЕЦИФИКАЦИЙ ПРОЦЕССА

Часть 12

Внешнее ядро

Industrial automation systems and integration. Process specification language. Part 12. Outer core

Дата введения — 2011—09—01

1 Область применения

Настоящий стандарт в форме аксиом и определений устанавливает понятия, относящиеся к внешнему ядру языка, что позволяет аксиоматизировать семантику терминологии, используемой для описания внешнего ядра в соответствии с требованиями комплекса стандартов ИСО 18629.

Настоящий стандарт распространяется на:

- деятельность и ее агрегирование:
- дискретные деятельности и их состояния;
- ограничения деятельности;
- параллельные примитивные деятельности;
- составные деятельности;
- примеры составных деятельностей.

2 Нормативные ссылки

В настоящем стандарте использованы нормативные ссылки на следующие 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) — Specification of basic notation)

ИСО 10303-1 Системы промышленной автоматизации и интеграция. Представление данных о продукции и обмен данными. Часть 1. Обзор и основные принципы (ISO 10303-1, Industrial automation systems and integration — Product data representation and exchange — Part 1: Overview and fundamental principles)

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

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

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

Издание официальное

3 Термины, определения и сокращения

3.1 Термины и определения

В настоящем стандарте применены следующие термины с соответствующими определениями.

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

[ИСО 18629-1:2004, статья 3.1.1]

3.1.2 **консервативное определение** (conservative definition): Определение, устанавливающее необходимые и достаточные условия для полного соответствия термина, а также не позволяющее выводить новые умозаключения из теории.

[ИСО 18629-1:2004, статья 3.1.2]

3.1.3 **теория ядра** (core theory): Набор аксиом для реляционных и функциональных символов, обозначающих примитивные понятия.

[ИСО 18629-1:2004, статья 3.1.3]

3.1.4 **данные** (data): Представление информации в формальном виде, подходящем для ее передачи, интерпретации или обработки людьми или на электронно-вычислительных машинах.

[ИСО 18629-1:2004, статья 3.1.4]

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

Примечание — Установленные лексикой символы подразделяют на постоянные, функциональные и относительные.

[ИСО 18629-1:2004, статья 3.1.5]

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

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

[ИСО 18629-1:2004, статья 3.1.6]

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 дефинициальными расширениями.

[ИСО 18629-1:2004, статья 3.1.9]

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

[ИСО 18629-1:2004, статья 3.1.10]

3.1.9 **идемпотент** (idempotent): Элемент (a) множества, удовлетворяющий условию: a OP a = a, где OP — внутренняя бинарная операция.

Примечание — Функция является идемпотентной, если каждый элемент в области ее определения является идемпотентом по отношению к данной функции.

Пример — Теоретико-множественное пересечение и объединение являются идемпотентами, так как $S \cap S = S$ ***и*** $S \cup S = S$.

3.1.10 **информация** (information): Факты, концепции или инструкции.

[ИСО 18629-1:2004, статья 3.1.11]

3.1.11 **интерпретация** (interpretation): Ракурс рассмотрения всех предложений теории и присвоения этим предложениям значений истинности (TRUE или FALSE).

Примечание — Пример интерпретации приведен в приложении С.

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

[ИСО 18629-1:2004, статья 3.1.12]

3.1.13 **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 18629-1:2004, статья 3.1.13]

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

[ИСО 15531-1:2004, статья 3.6.22]

3.1.15 **производственный процесс** (manufacturing process): Структурированный комплекс видов деятельности или работ, выполняемых с материалом для его перевода 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., основанные на стратегическом использовании и размещении материально-производственных запасов.

[ИСО 15531-1:2004, статья 3.6.25]

3.1.16 **модель** (model): Сочетание набора элементов и истинного назначения, удовлетворяющее всем правильно построенным формулировкам в теории.

Примечание 1 — В настоящем стандарте определение термина «модель» отличается от используемого в научной и другой литературе: если предложение является верным в определенной интерпретации, то можно сказать, что интерпретация — это модель предложения. Виды семантик, представленных в настоящем стандарте, часто называют теоретически смоделированными семантиками.

Примечание 2 — Модель обычно представляют в виде совокупности дополнительных структур (частично упорядоченных, в качестве структурного или векторного пространства). В этом случае модель определяет значения для терминологии и понятия истины для предложений языка в условиях данной модели. Задавая модель, основной набор аксиом математических структур, используемый в наборе аксиом, используют как основу для определения понятий, представленных в терминах языка, и их логических взаимосвязей, в результате чего набор моделей создает формальные семантики онтологии.

[ИСО 18629-1:2004, статья 3.1.16]

3.1.17 **онтология** (ontology): Лексика специализированной терминологии, дополненная необходимой спецификацией значений терминов.

Примечание 1 — Онтология — это структурированный набор относительных терминов, представленный с описанием значений терминов на формальном языке. Описание значения объясняет, как и почему термины соотносятся, и определяет условия сегментирования и структурирования набора терминов.

Примечание 2 — Основополагающим компонентом языка технологических спецификаций ИСО 18629 является онтология. Примитивные концепции в онтологии, соответствующей определению ИСО 18629, достаточны для описания основных производственных и инженерных процессов, а также бизнес-процессов.

Примечание 3 — Основное внимание онтологии направлено не только на термины, но и на их значения. Произвольный набор терминов включён в онтологию, но эти термины могут применяться только в том случае, если их значения согласованы. Такие предполагаемые семантики терминов могут быть утверждены и использованы.

Примечание 4 — Любой термин, используемый без точного определения, может быть причиной неясности и путаницы. Сложность для онтологии заключается в том, что структура нуждается в создании терминов, имеющих точное значение. Для онтологии, соответствующей определению ИСО 18629, необходимо предоставить математически строгую характеристику информационного процесса, а также четкое выражение основных логических свойств этой информации на языке, указанном в ИСО 18629.

[ИСО 18629-1:2004, статья 3.1.17]

3.1.18 **внешнее ядро** (outer core): Набор теорий ядра, которые являются расширениями ядра PSL и настолько обобщены и распространены в своем применении, что каждая теория может быть представлена отдельно.

Примечание — На практике расширения включают в себя аксиомы внешнего ядра.

[ИСО 18629-1:2004, статья 3.1.18]

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

[ИСО 18629-1:2004, статья 3.1.19]

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

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

[ИСО 18629-1:2004, статья 3.1.20]

3.1.21 **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 15531-1:2004, статья 3.6.29]

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

[ИСО 18629-1:2004, статья 3.1.23]

3.1.23 **теория доказательств** (proof theory): Совокупность теорий и лексических элементов, необходимых для интерпретации семантики языка.

Примечание — Теория доказательств состоит из трех компонентов: ядра PSL, внешнего ядра и расширений.

[ИСО 18629-1:2004, статья 3.1.25]

3.1.24 **ядро PSL** (PSL-core): Набор аксиом для понятий деятельности, события деятельности, момента времени и объекта.

Примечание — Мотивацией для ядра PSL является наличие любых двух приложений, имеющих отношение к процессу, которые должны совместно использовать упомянутые аксиомы с целью обмена информацией о процессе. Поэтому ядро PSL является адекватным для описания основных концепций производственных процессов. Следовательно, эта характеристика основных процессов имеет несколько допущений в отношении их характеристик, за исключением тех, которые необходимы для описания процессов. Поэтому ядро PSL ограничено с точки зрения выражения логической возможности. При этом ядро PSL обеспечивает определение многих вспомогательных понятий, которые необходимы для описания всех интуитивных понятий в производственном процессе.

[ИСО 18629-1:2004, статья 3.1.26]

3.1.25 **псевдоструктура** (semilattice): Частичное упорядочение, в котором для каждых двух элементов существует единственная наименьшая верхняя граница.

3.1.26 **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 18629-1:2004, статья 3.1.31]

3.1.27 **область обсуждения** (universe of discourse): Совокупность конкретных или абстрактных вещей, относящихся к области реального мира, выбранных в соответствии с интересом, который они представляют для системы, подлежащей моделированию, и ее окружения.

[ИСО 15531-1:2004, статья 3.6.50]

3.2 Сокращения

FOL — логика первого порядка (First-Order Logic);
BNF — форма Бэкуса-Наура (Backus-Naur form);
KIF — формат обмена знаниями (Knowledge Interchange Format);
PSL — язык спецификаций процесса (Process Specification Language).

4 Общие положения комплекса стандартов ИСО 18629

Комплекс стандартов ИСО 18629 устанавливает требования к лексике, онтологии и грамматике языка, используемого для описания производственных процессов.

Примечание 1 — PSL — это язык, используемый для спецификации производственных процессов, основанный на математически точно определенных словаре и грамматике. Он отличается от языка, используемого для описаний схем и продукта в ИСО 10303, ИСО 13584 и ИСО 15926, а также от языка характеризации процессов, приведенного в ИСО 15531, но тесно с ним связан и дополняет его. При обмене информацией между двумя процессами язык PSL определяет каждый процесс независимо от его поведения (режима работы). Например, объект, рассматриваемый как ресурс для одного процесса, считают таким же объектом, даже если его рассматривают в качестве продукта второго процесса.

Примечание 2 — PSL основан на логике первого порядка и использует другие методы спецификации семантики, используемые в ИСО 10303. Значения понятий, определяемых в рамках PSL, вытекают из набора аксиом и определений, представленных каждым расширением ядра PSL. Примечания и примеры, приведенные в стандартах, должны способствовать пониманию языка.

В частях с 11 по 19 комплекса стандартов ИСО 18629 приведены основные теории, необходимые для представления точных определений и аксиом элементарных концепций. Основные теории обеспечивают точные семантические переводы между разными схемами.

Части с 11 по 19 распространяются на:

- представление основных элементов языка;
- обеспечение стандартизованных наборов аксиом, соответствующих интуитивным элементарным семантическим концепциям описания основных процессов;
- свод правил, необходимых для разработки других основных теорий или расширений в соответствии с ядром языка спецификации процесса.

Данные части комплекса стандартов ИСО 18629 не распространяются на:

- представление информации, относящейся к концепциям, не являющимся частью основных теорий.

5 Общие положения настоящего стандарта

В настоящем разделе приведены фундаментальные теории, составляющие внешнее ядро.

5.1 Расширения, входящие во внешнее ядро

Внешнее ядро включает в себя следующие теории:

- subactivity.th (теория ядра подчиненной деятельности);
- occtree.th (теория ядра дерева случаев);
- disc_state.th (теория ядра дискретных состояний);
- atomic.th (теория ядра атомарной деятельности);
- complex.th (теория ядра составной деятельности);
- act_occ.th (теория ядра случаев деятельности).

Связи между теориями внешнего ядра изображены на рисунке 1. Стрелками обозначены зависимости между теориями. Все теории внешнего ядра являются расширениями ядра PSL. Теория ядра атомарной деятельности является расширением как теории подчиненной деятельности, так и теории дерева случаев, а теория ядра дискретных состояний является расширением только теории ядра дерева случаев. Теория ядра случаев деятельности — это расширение теории ядра составной деятельности, которая, в свою очередь, является расширением теории атомарной деятельности.

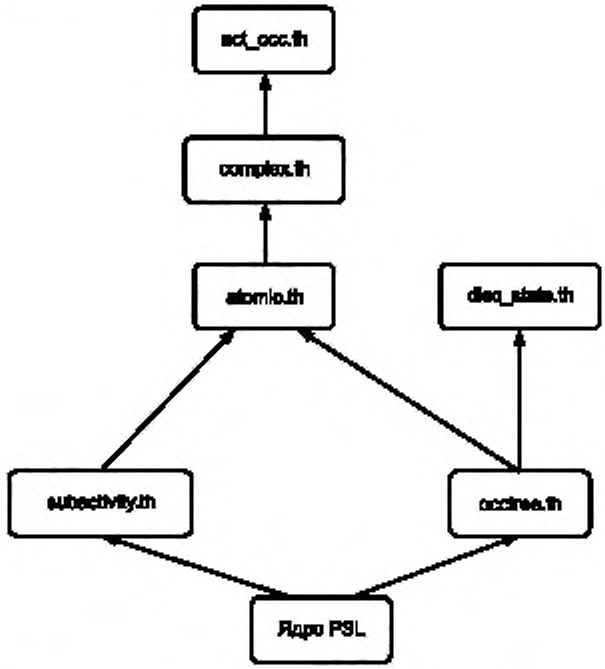

Рисунок 1 — Зависимости между наборами аксиом внешнего ядра PSL

6 Теория ядра подчиненной деятельности

Подчиненная деятельность (subactivity) предоставляет аксиомы для понятий композиции процессов. Единственным ограничением данной теории ядра является то, что отношение подчиненной деятельности изоморфно дискретному частичному упорядочению. В других теориях ядра установлены дополнительные ограничения.

6.1 Примитивные отношения

Нелогическая лексика теории ядра подчиненной деятельности включает в себя следующий символ отношения:

- subactivity (отношение подчиненной деятельности).

6.2 Установленные отношения

Нелогическая лексика теории ядра подчиненной деятельности включает в себя следующий символ установленного отношения:

- primitive (примитивный элемент).

6.3 Связь с другими наборами аксиом

В теории ядра подчиненной деятельности используется теория pslcore.th, установленная в ИСО 18629-11.

Лексика дефиниций для теории ядра подчиненной деятельности не требуется.

6.4 Неформальная семантика

6.4.1 Подчиненная деятельность

Обозначение KIF символа subactivity:

(subactivity ?a1 ?a2)

Неформальная семантика символа subactivity:

(subactivity ?al ?a2) принимает значение TRUE в интерпретации теории ядра подчиненной деятельности только в том случае, если деятельность ?al является подчиненной деятельностью деятельности ?a2.

Отношение подчинения деятельности представляет собой дискретное частичное упорядочение множества деятельностей.

6.4.2 Примитивная деятельность

Обозначение KIF символа primitive:

(primitive ?a)

Неформальная семантика для символа primitive:

(primitive ?a) принимает значение TRUE в интерпретации теории ядра подчиненной деятельности только в том случае, если деятельность ?al не имеет подчиненных деятельностей.

6.5 Определения

Деятельность является примитивной только в том случае, если она не имеет подчиненных деятельностей.

6.5.1 Определение 1

```
(forall (?a) (iff (primitive ?a)
      (forall (?al)
         (implies (subactivity ?al ?a)
               (= ?al ?a))))))
```

6.6 Аксиомы

6.6.1 Аксиома 1

subactivity — это отношение между деятельностями:

```
(forall (?a1 ?a2)
    (implies (subactivity ?a1 ?a2)
             (and (activity ?a1)
                  (activity ?a2)))))
```

6.6.2 Аксиома 2

Отношение subactivity является рефлексивным:

```
(forall (?a)
   (implies (activity ?a)
            (subactivity ?a ?a)))
```

6.6.3 Аксиома 3

Отношение subactivity является антисимметричным:

```
   (forall (?a1 ?a2)
       (implies (and (subactivity ?a1 ?a2)
                     (subactivity ?a2 ?a1))
                (= ?a1 ?a2)))
```

6.6.4 Аксиома 4

Отношение subactivity является транзитивным:

```
(forall (?a1 ?a2 ?a3)
      (implies (and (subactivity ?a1 ?a2)
                    (subactivity ?a2 ?a3))
               (subactivity ?a1 ?a3)))
```

6.6.5 Аксиома 5

Отношение subactivity является дискретным упорядочением, поэтому для каждой деятельности существует подчиненная деятельность, расположенная в структуре этого упорядочения.

```
(forall (?a1 ?a2)
      (implies (subactivity ?a1 ?a2)
               (exists (?a3)
               (and (subactivity ?a1 ?a3)
```

```
(subactivity ?a3 ?a2)
(forall (?a4)
(implies (and (subactivity ?a1 ?a4)
              (subactivity ?a4 ?a3)
         (or (= ?a4 ?a1)
             (= ?a4? a3))))))))
```

6.6.6 Аксиома 6

Отношение subactivity является дискретным упорядочением. Для каждой деятельности существует подчиненная деятельность, расположенная в структуре этого упорядочения.

```
(forall (?a1 ?a2)
    (implies (subactivity ?a1 ?a2)
    (exists (?a3)
    (and    (subactivity ?a1 ?a3)
            (subactivity ?a3 ?a2)
                (forall (?a4)
            (implies (and (subactivity ?a3 ?a4)
                          (subactivity ?a4 ?a2)
                     (or (= ?a4 ?a2)
                         (=?a4?a3))))))))
```

7 Теория ядра дерева случаев

Occurrence tree представляет собой множество всех дискретных последовательностей случаев деятельности, изоморфных подструктурам дерева ситуаций, подлежащих ситуационному исчислению (см. приложение С). Основное отличие дерева ситуаций от последних заключается в том, что для каждого дерева случаев существует единственный начальный случай деятельности, а не единственная начальная ситуация. В ситуационном исчислении используется отношение poss, позволяющее определять ограничения ко всем случаям деятельности в дереве случаев. Так как в дерево случаев входят последовательности, которые разработчиками модели домена будут признаны невозможными, отношение poss «обрезает» ветви дерева случаев, соответствующие невозможным случаям деятельности.

Дерево случаев не является структурой, представляющей подчиненные деятельности какой-либо деятельности. Оно представляет все случаи всех деятельностей в домене, а не конкретный случай деятельности.

7.1 Примитивные отношения

Нелогическая лексика теории ядра дерева случаев включает в себя следующие символы примитивных отношений:

- earlier (ранее);
- initial (начальный);
- legal (допустимый).

7.2 Примитивные функции

Нелогическая лексика теории ядра дерева случаев включает в себя следующий символ примитивной функции:

- successor (следующий элемент).

7.3 Связь с другими наборами аксиом

В теории ядра дерева случаев используется теория pslcore.th, установленная в ИСО 18629-11.
Лексика дефиниций для теории ядра дерева случаев не требуется.

7.4 Неформальная семантика

7.4.1 Отношение earlier

Обозначение KIF символа earlier:

(earlier ?occl ?occ2)

Неформальная семантика для отношения earlier:

(earlier ?occl ?occ2) принимает значение TRUE в интерпретации теории ядра дерева случаев только в том случае, если два случая деятельности ?occ1 и ?occ2 принадлежат одной и той же ветви дерева, а ?occ1 находится ближе к корневому элементу дерева, чем ?occ2.

В интерпретациях деревьев случаев множество всех последовательностей случаев деятельности образует дерево, а отношение earlier задает частичное упорядочение всех случаев деятельности, принадлежащих этому дереву.

7.4.2 Отношение initial

Обозначение KIF символа initial:

(initial ?occ)

Неформальная семантика отношения initial:

(initial ?occ) принимает значение TRUE в интерпретации теории ядра дерева случаев только в том случае, если случай деятельности ?occ является корневым элементом дерева случаев.

7.4.3 Отношение legal

Обозначение KIF отношения legal:

(legal ?occ)

Неформальная семантика отношения legal:

(legal ?occ) принимает значение TRUE в интерпретации теории ядра дерева случаев только в том случае, если случай деятельности ?occ является элементом допустимого дерева случаев.

7.4.4 Отношение poss

Обозначение KIF символа отношения poss:

(poss ?occ1 ?occ2)

Неформальная семантика символа отношения poss:

(poss ?a ?occ) принимает значение TRUE в интерпретации теории ядра дерева случаев только в том случае, если для деятельности ?a существует случай, являющийся следующим элементом случая деятельности ?occ в дереве случаев.

7.4.5 Отношение precedes

Обозначение KIF символа отношения precedes:

(precedes ?occ1 ?occ2)

Неформальная семантика символа отношения precedes:

(precedes ?occ1 ?occ2) принимает значение TRUE в интерпретации теории ядра дерева случаев только в том случае, если случай деятельности ?occ1 расположен раньше случая деятельности ?occ2 в дереве случаев, а все случаи деятельности между этими случаями соответствуют возможным деятельностям. Данное отношение определяет поддерево дерева случаев, в котором каждый случай деятельности является случаем возможной деятельности.

7.4.6 Отношение successor

Обозначение KIF символа отношения successor:

(successor ?a ?occ)

Неформальная семантика символа отношения successor:

(= (successor ?a ?occ) ?occ2) принимает значение TRUE в интерпретации теории ядра дерева случаев только в том случае, если ?occ2 обозначает случай ?a, следующий за случаем деятельности ?occ в дереве случаев.

7.5 Аксиомы

Теория ядра дерева событий включает в себя следующие аксиомы:

7.5.1 Аксиома 1

Отношение earlier применимо только для следующих случаев деятельности:

```
(forall (?occ1 ?occ2)
        (implies (earlier ?occ1 ?occ2)
                 (and (activity_occurrence ?occ1)
                      (activity_occurrence ?occ2))))
```

7.5.2 Аксиома 2

Отношение earlier нерефлексивно для множества случаев:

```
(forall (?occ1 ?occ2)
    (implies (earlier ?occ1 ?occ2)
             (not (earlier ?occ2 ?occ1))))
```

7.5.3 Аксиома 3

Отношение earlier является транзитивным для множества случаев:

```
(forall (?occ1 ?occ2 ?occ3)
    (implies (and (earlier ?occ 1 ?occ2)
                  (earlier ?occ2 ?occ3))
             (earlier ?occ1 ?occ3)))
```

7.5.4 Аксиома 4

Ветвь дерева случаев представляет собой полностью упорядоченное множество случаев деятельности:

```
(forall (?occ1 ?occ2 ?occ3)
    (implies (and (earlier ?occ 1 ?occ2)
                  (earlier ?occ3 ?occ2)
             (or  (earlier ?occ1 ?occ3)
                  (earlier ?occ3 ?occ1)
                  (= ?occ3 ?occ1))))
```

7.5.5 Аксиома 5

Ни один случай не может произойти раньше начального случая:

```
(forall (?occ1 ?occ2)
    (implies (initial ?occ1)
             (not (exists (?occ2)
                  (earlier ?occ2 ?occ1)))))
```

7.5.6 Аксиома 6

Каждая ветвь дерева случаев имеет начальный случай:

```
(forall (?occ1 ?occ2)
    (implies (earlier ?occ1 ?occ2)
             (exists (?occp)
                 (and (initial ?occp)
                 (or (earlier ?occp ?occ1)
                     (=?occp?occ1))))))
```

7.5.7 Аксиома 7

Каждая деятельность имеет начальный случай:

```
(forall (?a)
    (implies (activity ?a)
        (exists (?s)
            (and (occurrence_of ?s ?a)
                 (initial ?s)))))
```

7.5.8 Аксиома 8

Два начальных случая деятельности дерева случаев не являются случаями одной и той же деятельности:

```
(forall (?occ1 ?occ2 ?a)
    (implies (and (initial ?occ1)
                  (initial ?occ2)
                  (occurrence_of ?occ1 ?a)
                  (occurrence_of ?occ2 ?a))
             (= ?occ1 ?occ2)))
```

7.5.9 Аксиома 9

Элемент случая деятельности, представляющий случай деятельности:

```
(forall (?a ?occ)
    (implies (and (activity_occurrence ?occ)
                  (activity ?a))
             (occurrence_of (successor ?a ?occ) ?a))))
```

7.5.10 Аксиома 10

Каждый случай, не являющийся начальным, представляет собой следующий элемент другого случая:

```
(forall (?occ)
    (implies (not (initial ?occ))
        (exists (?a ?occp)
            (= ?occ (successor ?a ?occp)))))
```

7.5.11 Аксиома 11

Случай ?occ1 происходит ранее следующего случая ?occ2 только в том случае, если случай ?occ2 происходит позже случая ?occ1.

```
(forall (?a ?occ1 ?occ2)
    (iff (earlier ?occ1 (successor ?a ?occ2))
        (or (earlier ?occ 1 ?occ2)
            (= ?occ1 ?occ2)))))
```

7.5.12 Аксиома 12

Отношение legal является ограничением к случаям деятельности:

```
(forall (?occ)
    (implies (legal ?occ)
        (activity_occurrence ?occ)))
```

7.5.13 Аксиома 13

Если случай деятельности допускается, то допускаются и все предшествующие ему случаи деятельности:

```
(forall (?occ1 ?occ2)
    (implies (and (legal ?occ1)
                  (earlier ?occ2 ?occ1))
        (legal ?occ2)))
```

7.5.14 Аксиома 14

Момент завершения endof случая деятельности предшествует моменту начала beginof следующего случая деятельности:

```
(forall (?occ1 ?occ2)
    (implies (earlier ?occ1 ?occ2)
        (before (endof ?occ1) (beginof ?occ2))))
```

7.5.15 Определение 1

poss — это отношение между деятельностями и случаями деятельности:

```
(forall (?a ?occ) (iff (poss ?a ?occ)
(legal (successor ?a ?occ))))
```

7.5.16 Определение 2

Случай деятельности ?occ1 предшествует случаю деятельности ?occ2 только в том случае, если ?occ1 происходит ранее ?occ2 в отношении упорядочения earlier, и каждый случай между этими случаями является возможным:

```
(forall (?occ1 ?occ2) (iff (precedes ?occ1 ?occ2)
    (and (earlier ?occ 1 ?occ2)
        (legal ?occ2))))
```

8 Теория ядра дискретного состояния

В Discrete state введено понятие состояний (флюентов). Флюенты изменяются только при наступлении случаев деятельности, но остаются неизменными во время случаев примитивной деятельности. При этом у деятельностей есть предпосылки (флюенты, существующие до наступления случая) и последствия (флюенты, возникающие после наступления случая).

8.1 Примитивные категории

Нелогическая лексика теории ядра дискретного состояния включает в себя следующую примитивную категорию:

- state.

8.2 Примитивные отношения

Нелогическая лексика теории ядра дискретного состояния включает в себя следующие символы примитивных отношений:

- holds;
- prior.

8.3 Связь с другими наборами аксиом

В теории ядра дискретного состояния используются следующие теории:
- pslcore.th, установленная в ИСО 18629-11;
- occtree.th (теория ядра дерева случаев).

Лексика дефиниций для теории ядра дискретного состояния не требуется.

8.4 Неформальная семантика

8.4.1 Отношение state

Обозначение KIF символа state:

(state ?f)

Неформальная семантика символа state.

(state ?f) принимает значение TRUE в интерпретации теории ядра дискретного состояния только в том случае, если ?f представляет собой множество состояний ракурса рассмотрения интерпретации. Состояния образуют подкатегорию объекта.

Примечание — Объект определен в ИСО 18629-11.

Состояния представляют те свойства и отношения в домене, которые могут изменяться в результате произошедших случаев деятельности.

8.4.2 Отношение holds

Обозначение KIF символа отношения holds:

(holds ?f ?occ)

Неформальная семантика символа отношения holds:

(holds ?f ?occ) принимает значение TRUE в интерпретации теории ядра дискретного состояния только в том случае, если состояние ?f является истинным после случая деятельности ?occ.

8.4.3 Отношение prior

Обозначение KIF символа отношения:

(prior ?f ?occ)

Неформальная семантика для отношения prior.

(prior ?f ?occ) принимает значение TRUE в интерпретации теории ядра дискретного состояния только в том случае, если состояние ?f является истинным до случая деятельности ?occ.

8.5 Аксиомы

Теория ядра дискретного состояния включает в себя следующие аксиомы.

8.5.1 Аксиома 1

Состояния являются объектами:

```
(forall (?f)
    (implies (state ?f)
             (object ?f)))
```

8.5.2 Аксиома 2

Отношение holds действует только между состояниями и случаями деятельности. Это означает, что состояние является истинным после случая деятельности ?o:

```
(forall (?f ?occ)
    (implies (holds ?f ?occ)
             (and (state ?f)
             (activity_occurrence ?occ))))
```

8.5.3 Аксиома 3

Отношение prior действует только между состояниями и случаями деятельности. Это означает, что состояние является истинным после случая деятельности ?o:

```
(forall (?f?occ)
    (implies (prior ?f ?occ)
        (and (state ?f)
            (activity_occurrence ?occ))))
```

8.5.4 Аксиома 4

Состояния, существовавшие до начальных случаев, согласованы друг с другом:

```
(forall (?occ1 ?occ2 ?f)
        (implies (and (initial ?occ1)
                      (initial ?occ2))
                 (iff  (prior ?f ?occ1)
                       (prior ?f ?occ2))))
```

8.5.5 Аксиома 5

Состояние может возникнуть после случая только в том случае, если оно существует до следующего случая:

```
(forall (?a ?occ)
    (iff (holds ?f?occ)
       (prior ?f (successor ?a ?occ))))
```

8.5.6 Аксиома 6

Если состояние возникает после случая какой-либо деятельности, существует самый ранний случай деятельности на ветви дерева, где находится это состояние:

```
(forall (?f ?occ1)
    (implies (holds ?f ?occ1)
             (exists (?occ2)
                  (and (precedes ?occ2 ?occ1)
             (holds ?f ?occ2)
             (or     (initial ?occ2)
                     (not (prior ?f ?occ2)))
             (forall (?occ3)
                  (implies (and (precedes ?occ2 ?occ3)
                                (precedes ?occ3 ?occ1))
                           (holds ?f?occ3)))))))
```

8.5.7 Аксиома 7

Если состояние не возникает после случая какой-либо деятельности, существует самый ранний случай деятельности на ветви, где этого состояния нет:

```
(forall (?f?occ1)
    (implies (not (holds ?f?occ1))
             (exists (?occ2)
                  (and (precedes ?occ2 ?occ1)
                       (not (holds ?f?occ2))
                  (or    (initial ?occ2)
                         (prior ?f ?occ2))
                  (not (exists (?occ3)
                       (and (precedes ?occ2 ?occ3)
                            (precedes ?occ3 ?occ1))
                            (holds ?f?occ3)))))))))
```

9 Теория ядра атомарной деятельности

Atomic activity предоставляет аксиомы, относящиеся к интуитивным представлениям об одновременном параллельном агрегировании примитивных деятельностей, которое, как правило, представляют в терминах осуществления/происхождения одновременных деятельностей, а не в терминах одновременного осуществления/происхождения деятельностей.

9.1 Примитивные отношения

Нелогическая лексика теории ядра атомарной деятельности включает в себя следующий символ примитивного отношения:

- atomic.

9.2 Примитивные функции

Нелогическая лексика теории ядра атомарной деятельности включает в себя следующий символ примитивной функции:

- conc.

9.3 Связь с другими наборами аксиом

В теории ядра атомарной деятельности используются следующие теории:

- pslcore.th, установленная в ИСО 18629-11 (ядро PSL);
- occtree.th (теория ядра дерева случаев);
- subactivity.th (теория ядра подчиненной деятельности).

Лексика дефиниций для теории ядра атомарной деятельности не требуется.

9.4 Неформальная семантика

9.4.1 Отношение atomic

Обозначение KIF символа отношения atomic:

(atomic ?a)

Неформальная семантика символа отношения atomic:

(atomic ?a) принимает значение TRUE в интерпретации теории ядра атомарной деятельности только в том случае, если ?a является либо примитивной деятельностью, либо суперпозицией множества параллельных примитивных деятельностей.

9.4.2 Отношение conc

Обозначение KIF символа отношения conc:

(conc ?a1 ?a2)

Неформальная семантика символа отношения conc:

(= ?a3 (conc ?a1 ?a2)) принимает значение TRUE в интерпретации теории ядра атомарной деятельности только в том случае, если ?a3 является атомарной деятельностью, являющейся суперпозицией двух параллельных атомарных деятельностей ?a1 и ?a2.

9.5 Определения

Данная теория ядра не включает в себя определений.

9.6 Аксиомы

Теория ядра атомарной деятельности включает в себя следующие аксиомы.

9.6.1 Аксиома 1

Любая примитивная деятельность является атомарной:

```
(forall (?a)
        (implies (primitive ?a)
                 (atomic ?a)))
```

9.6.2 Аксиома 2

Функция conc является идемпотентной:

```
(forall (?a)
    (= ?a (conc ?a ?a)))
```

9.6.3 Аксиома 3

Функция conc является коммутативной:

```
(forall (?a1 ?a2)
    (= (conc ?a1 ?a2) (conc ?a2 ?a1)))
```

9.6.4 Аксиома 4

Функция conc является ассоциативной:

```
(forall (?a1 ?a2 ?a3)
    (= (conc ?a1 (conc ?a2 ?a3)) (conc (conc ?a1 ?a2) ?a3)))
```

9.6.5 Аксиома 5

Параллельная агрегация атомарной деятельности является атомарной деятельностью:

```
(forall (?a1 ?a2)
    (iff (atomic (conc ?a1 ?a2))
        (and (atomic ?a1)
             (atomic ?a2))))
```

9.6.6 Аксиома 6

Атомарная деятельность ?a1 является подчиненной деятельностью атомарной деятельности ?a2 только в случае, если ?a2 является идемпотентной для ?a1:

```
(forall (?a1 ?a2)
    (implies (and (atomic ?a1)
                  (atomic ?a2))
             (iff (subactivity ?a1 ?a2)
                  (= ?a2 (conc ?a1 ?a2)))))
```

9.6.7 Аксиома 7

Атомарная деятельность включает в себя подчиненную деятельность только в том случае, если существует другая атомарная деятельность, допускающая параллельное агрегирование с этой деятельностью:

```
(forall (?a1 ?a2)
    (implies (atomic ?a2)
             (iff (subactivity ?a1 ?a2)
                  (exists (?a3)
                          (= ?a2 (conc ?a1 ?a3))))))
```

9.6.8 Аксиома 8

Псевдоструктура атомарных деятельностей является дистрибутивной:

```
(forall (?a ?b0 ?b1)
    (implies (and  (subactivity ?a (conc ?b0 ?b1))
                   (not (primitive ?a)))
             (exists (?a0 ?a1)
                  (and (subactivity ?a0 ?a)
                       (subactivity ?a1 ?a)
                       (= ?a (conc ?a0 ?a1)))))))
```

9.6.9 Аксиома 9

Только случаи атомарной деятельности могут быть элементами допустимого дерева случаев:

```
(forall (?a ?occ)
    (implies (and (occurrence_of ?occ ?a)
                  (legal ?occ))
             (atomic ?a)))
```

10 Теория ядра комплексной деятельности

Complex activity является основой для представлений и логических выводов в отношении составной деятельности, а также отношений между случаями деятельности и случаями ее подчиненных деятельностей. В моделях теории ядра составной деятельности случаи составной деятельности соответствуют поддеревьям дерева случаев. Деятельность может включать в себя подчиненные деятельности, которые не происходят. Единственным ограничением является соответствие любого случая подчиненной деятельности поддереву дерева случаев, описывающего случаи данной деятельности. Не каждый случай деятельности является случаем подчиненной деятельности. Во время случая деятельности могут происходить другие внешние деятельности. Различные подчиненные деятельности могут находиться на разных ветках дерева деятельности, так что для различных случаев деятельности могут происходить различные случаи подчиненных деятельностей.

10.1 Примитивные отношения

Нелогическая лексика теории ядра составной деятельности включает в себя следующие примитивные отношения:

- min_precedes;
- root.

10.2 Установленные отношения

Нелогическая лексика теории ядра составной деятельности включает в себя следующие установленные отношения:

- subtree;
- do;
- leaf;
- next_subocc;
- sibling.

10.3 Связь с другими наборами аксиом

В теории составной деятельности используются следующие теории ядра:
- pslcore.th, установленная в ИСО 18629-11;
- occtree.th;
- subactivity.th;
- atomic.th.

Для теории ядра составной деятельности не требуются лексиконы дефиниций.

10.4 Неформальная семантика

10.4.1 Отношение min_precedes

Обозначение KIF символа отношения min_precedes:
(min_precedes ?occ1 ?occ2 ?a)
Неформальная семантика символа отношения min_precedes:
(min_precedes ?occ1 ?occ2 ?a) принимает значение TRUE в интерпретации теории ядра составной деятельности только в том случае, если ?occ1 и ?occ2 являются случаями подчиненной деятельности в дереве деятельности ?a, а ?occ1 предшествует ?occ2 в поддереве. Любой случай деятельности ?a соответствует дереву деятельности, являющемуся поддеревом дерева случаев. Случаи деятельности в этом поддереве являются случаями подчиненной деятельности, соответствующими случаю деятельности ?a.

10.4.2 Отношение root

Обозначение KIF символа отношения root:
(root ?occ ?a)
Неформальная семантика символа отношения root:
(root ?occ ?a) принимает значение TRUE в интерпретации теории ядра составной деятельности только в том случае, если случай деятельности ?occ является корневым элементом дерева деятельности ?a.

10.4.3 Отношение subtree

Обозначение KIF для отношения subtree:
(subtree ?occ ?a1 ?a2)
Неформальная семантика символа отношения subtree:
(subtree ?occ ?a1 ?a2) принимает значение TRUE в интерпретации теории ядра составной деятельности только в случае, если каждый случай атомарной подчиненной деятельности в дереве деятельности ?a1 с корневым элементом ?occ является элементом дерева деятельности ?a2.

10.4.4 Отношение leaf

Обозначение KIF символа отношения leaf:
(leaf ?occ ?a)
Неформальная семантика символа отношения leaf:
(leaf ?occ ?a) принимает значение TRUE в интерпретации теории ядра составной деятельности только в том случае, если случай деятельности ?occ является элементом дерева деятельности ?a.

10.4.5 Отношение do

Обозначение KIF символа отношения do:
(do ?a ?occ1 ?occ2)
Неформальная семантика символа отношения do:
(do ?a ?occ1 ?occ2) принимает значение TRUE в интерпретации теории ядра составной деятельности только в том случае, если ?occ1 является корневым элементом дерева деятельности, а ?occ2 — элементом того же дерева, так что оба случая деятельности являются элементами одной и той же ветви дерева деятельности.

10.4.6 Отношение next_subocc

Обозначение KIF символа отношения next_subocc:
(next_subocc ?occ1 ?occ2 ?a)
Неформальная семантика символа отношения next_subocc:

(next_subocc ?occ1 ?occ2 ?a) принимает значение TRUE в интерпретации теории ядра составной деятельности только в том случае, если ?occ1 предшествует ?occ2 на дереве деятельности, на котором не существует случая подчиненной деятельности между двумя данными случаями.

10.5 Определения

Теория ядра составной деятельности содержит следующие определения.

10.5.1 Определение 1

Дерево деятельности ?a1 с корневым элементом, представленным случаем ?occ1, является поддеревом дерева деятельности ?a2 в том случае, если случай каждой подчиненной атомарной деятельности дерева деятельности ?a1 является элементом дерева деятельности ?a2:

```
(forall (?occ1 ?a1 ?a) (iff (subtree ?occ1 ?a1 ?a2)
(and (root ?occ1 ?a1)
     (exists (?occ2)
          (and (root ?occ2 ?a2)
               (min_precedes ?occ1 ?occ2 ?a1)))
     (exists (?occ3)
          (and (min_precedes ?occ1 ?occ3 ?a1)
               (not (min_precedes ?occ2 ?occ3 ?a2)))))))
```

10.5.2 Определение 2

Случай является элементом дерева деятельности только в том случае, если существует более ранний случай подчиненной атомарной деятельности, но не существует более позднего случая подчиненной атомарной деятельности:

```
(forall (?occ ?a) (iff (leaf ?occ ?a)
(exists (?occ1)
     (and (min_precedes ?occ1 ?occ ?a)
          (not (exists (?occ2)
               (min_precedes ?occ ?occ2 ?a)))))))
```

10.5.3 Определение 3

Отношение do устанавливает начальный и конечный случаи атомарной деятельности, подчиненной этой деятельности:

```
(forall (?a ?occ1 ?occ2) (iff (do ?a ?occ1 ?occ2)
(and (min_precedes ?occ1 ?occ2 ?a)
     (not (exists (?occ3)
          (min_precedes ?occ3 ?occ1 ?a)))
     (not (exists (?occ4)
          (min_precedes ?occ2 ?occ4 ?a)))))))
```

10.5.4 Определение 4

Случай деятельности ?occ2 является следующим случаем подчиненной деятельности после случая ?occ1 в дереве деятельности ?a только в том случае, если ?occ1 предшествует ?occ2 на дереве деятельности и не существует случая подчиненной деятельности между этими случаями:

```
(forall (?occ1 ?occ2 ?a) (iff (next_subocc ?occ1 ?occ2 ?a)
(and (min_precedes ?occ1 ?occ2 ?a)
     (not (exists (?occ3)
          (and (min_precedes ?occ1 ?occ3 ?a)
               (min_precedes ?occ3 ?occ2 ?a)))))))
```

10.5.5 Определение 5

Случаи подчиненной атомарной деятельности ?occ1 и ?occ2 являются родственными по отношению к деятельности ?a либо когда у них есть общий предшествующий элемент в дереве деятельности ?a, либо когда они оба являются корневыми элементами дерева деятельности ?a, имеющими общий предшествующий элемент в дереве событий:

```
(forall (?occ1 ?occ2 ?a) (iff (sibling ?occ1 ?occ2 ?a)
(or  (exists (?occ3)
     (and (next_subocc ?occ3 ?occ1 ?a)
          (next_subocc ?occ3 ?occ2 ?a)))
     (and (root ?occ1 ?a)
```

```
(root ?occ2 ?a)
(or (and (initial ?occ1)
         (initial ?occ2))
    (exists (?occ4 ?a1 ?a2)
        (and (= ?occ1 (successor ?a1 ?occ4))
             (= ?occ2 (successor ?a2 ?occ4))))))))
```

10.6 Аксиомы

Теория ядра составной деятельности включает в себя следующие аксиомы.

10.6.1 Аксиома 1

Любые случаи деятельности на дереве деятельности соответствуют случаям атомарной деятельности, подчиненной этой деятельности:

```
(forall (?a ?occ1 ?occ2)
    (implies (min_precedes ?occ1 ?occ2 ?a)
        (exists (?a1 ?ap)
            (and (subactivity ?a1 ?a)
                 (atomic ?ap)
                 (subactivity ?a1 ?ap)
                 (occurrence_of ?occ2 ?ap)))))
```

10.6.2 Аксиома 2

Любые случаи деятельности на дереве деятельности соответствуют случаям атомарной деятельности, подчиненной этой деятельности:

```
(forall (?a ?occ1 ?occ2)
    (implies (min_precedes ?occ1 ?occ2 ?a)
        (exists (?a2 ?ap)
            (and (subactivity ?a2 ?a)
                 (atomic ?ap)
                 (subactivity ?a2 ?ap)
                 (occurrence_of ?occ1 ?ap)))))
```

10.6.3 Аксиома 3

Случаи корневого элемента на дереве деятельности соответствуют случаям атомарной деятельности, подчиненной этой деятельности:

```
(forall (?a ?occ1)
    (implies (root ?occ1 ?a)
        (exists (?a2 ?ap)
            (and (subactivity ?a2 ?a)
                 (atomic ?ap)
                 (subactivity ?a2 ?ap)
                 (occurrence_of ?occ1 ?ap)))))
```

10.6.4 Аксиома 4

На всех деревьях деятельности существует случай корневой подчиненной деятельности:

```
(forall (?occ1 ?occ2 ?a)
    (implies (min_precedes ?occ1 ?occ2 ?a)
        (exists (?occ3)
            (and (root ?occ3 ?a)
                 (or (min_precedes ?occ3 ?occ1 ?a)
                     (= ?occ3 ?occ1)))))))
```

10.6.5 Аксиома 5

Ни один случай подчиненной деятельности на дереве деятельности не может произойти раньше случая корневой подчиненной деятельности:

```
(forall (?occ ?a)
    (implies (root ?occ ?a)
        (not (exists (?occ2)
            (min_precedes ?occ2 ?occ ?a)))))
```

10.6.6 Аксиома 6

Дерево деятельности представляет собой поддерево дерева случаев:

```
(forall (?occ1 ?occ2 ?a)
    (implies (min_precedes ?occ1 ?occ2 ?a)
        (exists (?occ0)
            (and (initial ?occ0)
                (or (precedes ?occ0 ?occ1)
                    (= ?occ0 ?occ1))
                (precedes ?occ1 ?occ2)))))
```

10.6.7 Аксиома 7

Случаи корневых элементов являются элементами дерева случаев:

```
(forall (?occ ?a)
    (implies (root ?occ ?a)
        (exists (?occ0)
            (and (initial ?occ0)
                (or (precedes ?occ0 ?occ)
                    (= ?occ0 ?occ))))))
```

10.6.8 Аксиома 8

Каждый случай атомарной деятельности является деревом деятельности, содержащим только один случай:

```
(forall (?a1 ?a2 ?occ)
    (implies (and (atomic ?a1)
                (occurrence_of ?occ ?a1)
                (subactivity ?a2 ?a1))
        (root ?occ ?a2)))
```

10.6.9 Аксиома 9

Деревья деятельности являются дискретными:

```
(forall (?occ1 ?occ2)
    (implies (min_precedes ?occ1 ?occ2 ?a)
        (exists (?occ3)
            (and (next_subocc ?occ1 ?occ3 ?a)
                (or (min_precedes ?occ3 ?occ2 ?a)
                    (= ?occ3 ?occ2))))))
```

10.6.10 Аксиома 10

Случаи подчиненной деятельности, находящиеся на одной ветви дерева случаев, принадлежат одной ветви дерева деятельности:

```
(forall (?a ?occ1 ?occ2 ?occ3)
    (implies (and (min_precedes ?occ1 ?occ2 ?a)
                (min_precedes ?occ1 ?occ3 ?a)
                (precedes ?occ2 ?occ3))
        (min_precedes ?occ2 ?occ3 ?a)))
```

10.6.11 Аксиома 11

Дерево деятельности случая составной подчиненной деятельности является поддеревом дерева деятельности случая.

```
(forall (?a1 ?a2)
    (implies (subactivity ?a1 ?a2)
        (not (exists (?occ)
            (subtree ?occ ?a2 ?a1)))))
```

11 Теория ядра случаев деятельности

Activity occurrence включает в себя аксиомы, ограничивающие свойства случаев атомарной подчиненной деятельности. Теория ядра случаев деятельности сводит интуитивные представления к произвольным составным элементам подчиненной деятельности.

11.1 Примитивные отношения

Нелогическая лексика теории ядра случаев деятельности включает в себя символы следующих примитивных отношений:

- subactivity_occurrence;
- mono.

11.2 Установленные отношения в теории ядра случаев деятельности

Нелогическая лексика теории ядра случаев деятельности включает в себя символы следующих установленных отношений:

- root_occ;
- leaf_occ;
- same_grove;
- iso_occ;
- equiv_occ;
- hom.

11.3 Связь с другими наборами аксиом

Применяют следующие теории ядра:

- psl_core.th, установленную в ИСО 18629-11;
- occtree.th;
- subactivity.th;
- atomic.th;
- complex.th.

Лексика дефиниций для теории ядра случаев деятельности не требуется.

11.4 Неформальная семантика

11.4.1 Отношение subactivity_occurrence

Обозначение KIF символа отношения subactivity_occurrence:

(subactivity_occurrence ?occ1 ?occ2)

Неформальная семантика сотношения subactivity_occurrence:

Существует однозначное соответствие случаев деятельности и ветвей деревьев данной деятельности.

Неформальное описание отношения subactivity_occurrence:

(subactivity_occurrence ?occ1 ?occ2) принимает значение TRUE в интерпретации теории ядра случаев деятельности только в том случае, если ветвь, соответствующая случаю деятельности ?occ1, является подмножеством ветви, соответствующей случаю деятельности ?occ2.

11.4.2 Отношение mono

Обозначение KIF символа отношения mono:

(mono ?occ1 ?occ2 ?a)

Неформальная семантика символа отношения mono:

(mono ?occ1 ?occ2 ?a) принимает значение TRUE в интерпретации теории ядра случаев деятельности только в том случае, если существует однозначное соответствие между ветвями дерева деятельности ?a, при котором случай атомарной подчиненной деятельности ?occ1 отображается в случай атомарной деятельности ?occ2.

11.4.3 Отношение root_occ

Обозначение KIFсимвола отношения root_occ:

(root_occ ?occ1 ?occ2 ?a)

Неформальная семантика символа отношения root_occ:

(root_occ ?occ 1 ?occ2 ?a) принимает значение TRUE в интерпретации теории ядра случаев деятельности только в том случае, если случай деятельности ?occ1 является случаем корневого элемента ветви дерева деятельности ?a, соответствующего случаю деятельности ?occ2.

11.4.4 Отношение leaf_occ

Обозначение KIFсимвола отношения leaf_occ:

(leaf_occ ?occ1 ?occ2 ?a)

Неформальная семантика символа отношения leaf_occ:

(leaf_occ ?occ1 ?occ2 ?a) принимает значение TRUE в интерпретации теории ядра случаев деятельности только в том случае, если случай деятельности ?occ1 является элементом ветви дерева деятельности ?a, соответствующего случаю деятельности ?occ2.

11.4.5 Отношение iso_occ

Обозначение KIF символа отношения iso_occ:

(iso_occ ?occ1 ?occ2)

Неформальная семантика символа отношения iso_occ:

(iso_occ ?occ1 ?occ2) принимает значение TRUE в интерпретации теории ядра случаев деятельности только в том случае, если ?occ1 и ?occ2 являются случаями атомарной деятельности, включающей в себя общую подчиненную деятельность.

11.4.6 Отношение hom

Обозначение KIFсимвола отношения hom:

(hom ?occ 1 ?occ2 ?a)

Неформальная семантика символа отношения hom:

(hom ?occ1 ?occ2 ?a) принимает значение TRUE в интерпретации теории ядра случаев деятельности только в случае, если существует отображение между ветвями дерева деятельности ?a, при котором случай атомарной подчиненной деятельности ?occ1 отображается в случай атомарной подчиненной деятельности ?occ2.

11.4.7 Отношение same_grove

Обозначение KIF символа отношения same_grove:

(same_grove ?occ1 ?occ2 ?a)

Неформальная семантика символа отношения same_tree:

(same_grove ?occ1 ?occ2 ?a) принимает значение TRUE в интерпретации теории ядра случаев деятельности только в том случае, если случаи ?occ1 и ?occ2 деятельности ?a соответствуют ветвям одного и того же дерева деятельности ?a.

11.5 Определения

Теория ядра случаев деятельности содержит следующие определения.

11.5.1 Определение 1

Случай ?occ1 является корневым случаем случая деятельности ?a только в том случае, если он является случаем подчиненной деятельности и корневым элементом дерева деятельности ?a:

```
(forall (?occ1 ?occ2) (iff (= ?occ2 (root_occ ?occ1)
(exists (?a)
    (and (occurrence ?occ1 ?a)
        (subactivity_occurrence ?occ2 ?occ1)
        (root ?occ2 ?a)))))
```

11.5.2 Определение 2

Случай ?occ 1 является элементом случая деятельности ?a только в том случае, если он является случаем подчиненной деятельности и элементом дерева деятельности ?a.

```
(forall (?s ?occ) (iff (leaf_occ ?s ?occ)
(exists (?a)
    (and (occurrence_of ?occ ?a)
        (subactivity_occurrence ?s ?occ)
        (leaf ?s ?a)))))
```

11.5.3 Определение 3

Случай деятельности ?occ1 является изоморфенным случаю деятельности ?occ2, если оба случая являются случаями атомарной деятельности, содержащей общую подчиненную деятельность:

```
(forall (?s1 ?s2) (iff ( ИСО_occ ?s1 ?s2)
(exists (?a1 ?a2)
    (and (occurrence_of ?s1 ?a1)
        (occurrence_of ?s2 ?a2)
        (subactivity ?a1 ?a2)))))
```

11.5.4 Определение 3

Два случая деятельности эквивалентны только в том случае, если они являются случаями атомарных деятельностей, имеющих общую подчиненную деятельность:

```
(forall (?occ1 ?occ2) (iff (equiv_occ ?occ1 ?occ2)
(exists (?a1 ?a2 ?a3)
      (and (occurrence_of ?occ 1 (conc ?a1 ?a2))
      (occurrence_of ?occ2 (conc ?a1 ?a3)))))))
```

11.5.5 Определение 4

```
(forall (?s1 ?s2 ?a) (iff (hom ?s1 ?s2 ?a)
(exists (?occ1 ?occ2)
      (and (iso_occ ?s1 ?s2)
            (subactivity_occurrence ?s1 ?occ1)
            (subactivity_occurrence ?s2 ?occ2)
            (occurrence_of ?occ1 ?a)
            (occurrence_of ?occ2 ?a)
            (not(= ?occ1 ?occ2))))))
```

11.5.6 Определение 5

Два случая деятельности принадлежат к одному множеству деревьев деятельности только в том случае, если они имеют общий корневой случай:

```
(forall (?occ1 ?occ2) (iff (same_grove ?occ1 ?occ2)
(exists (?a)
      (and (occurrence_of ?occ1 ?a)
            (occurrence_of ?occ2 ?a)
            (or (and (initial (root_occ ?occ1))
                  (initial (root_occ ?occ2)))
               (exists (?s4 ?a1 ?a2)
                  (and (= (root_occ ?occ1) (successor ?a1 ?s4))
                        (= (root_occ ?occ2) (successor ?a2 ?s4)))))))))
```

11.6 Аксиомы

Теория ядра случаев деятельности включает в себя следующие аксиомы.

11.6.1 Аксиома 1

Для каждой ветви дерева деятельности ?a существует случай деятельности ?a. Все случаи атомарной подчиненной деятельности на ветви являются случаями подчиненной деятельности, соответствующими этому случаю ?a.

```
(forall (?a ?occ1 ?occ2)
            (implies (min_precedes ?occ1 ?occ2 ?a)
                  (exists (?occ)
                        (and (occurrence ?occ ?a)
                              (subactivity_occurrence ?occ1 ?occ)
                              (subactivity_occurrence ?occ2 ?occ))))))
```

11.6.2 Аксиома 2

Для каждой ветви дерева деятельности ?a существует случай деятельности ?a. Корневой случай подчиненной деятельности на ветви является случаем подчиненной деятельности, соответствующим случаю деятельности ?a:

```
(forall (?a ?s)
      (implies (root ?s ?a)
      (exists (?occ)
                  (occurrence_of ?occ ?a)
                  (subactivity_occurrence ?s ?occ))))
```

11.6.3 Аксиома 3

Каждый случай деятельности ?a включает в себя случаи атомарной подчиненной деятельности, которые являются элементами дерева деятельности ?a:

```
(forall (?occ ?a)
      (implies (exists ?s1 ?s2)
            (and (subactivity_occurrence ?s1 ?occ)
                  (subactivity_occurrence ?s2 ?occ)
                  (root ?s2 ?a)
                  (or (min_precedes ?s2 ?s1 ?a)
                      (= ?s1 ?s2))))))
```

11.6.4 Аксиома 4

Разные случаи деятельности соответствуют разным ветвям дерева деятельности:

```
(forall (?a ?s1 ?occ1 ?occ2)
    (implies (and (occurrence_of ?occ1 ?a)
                  (occurrence_of ?occ2 ?a)
                  (not (= ?occ 1 ?occ2))
                  (subactivity_occurrence ?s1 ?occ1)
                  (subactivity_occurrence ?s1 ?occ2))
             (exists (?s2)
                (and (min_precedes ?s1 ?s2 ?a)
                     (subactivity_occurrence ?s1 ?occ1)
                     (not (subactivity_occurrence ?s 1 ?occ2))))))
```

11.6.5 Аксиома 5

Все случаи атомарной подчиненной деятельности случая составной деятельности являются элементами одной и той же ветви дерева деятельности:

```
(forall (?a ?a1 ?a2 ?occ ?s1 ?s2)
    (implies (and (occurrence_of ?occ ?a)
                  (occurrence_of ?s1 ?a1)
                  (atomic ?a1)
                  (occurrence_of ?s2 ?a2)
                  (atomic ?a2)
                  (subactivity_occurrence ?s1 ?occ)
                  (subactivity_occurrence ?s2 ?occ))
             (or (min_precedes ?s1 ?s2 ?a)
                 (min_precedes ?s2 ?s1 ?a)
                 (= ?s1 ?s2))))
```

11.6.6 Аксиома 6

Все элементы одной ветви дерева деятельности являются случаями атомарной подчиненной деятельности одних и тех же случаев деятельности:

```
(forall (?a ?s1 ?s2 ?occ)
    (implies (and (min_precedes ?s1 ?s2 ?a)
                  (subactivity_ooccurrence ?s2 ?occ))
             (subactivity_occurrence ?s1 ?occ))
```

11.6.7 Аксиома 7

Отношение subactivity_occurrence не противоречит отношению subactivity:

```
(forall (?a1 ?a2 ?occ1 ?occ2)
    (implies (and (occurrence_of ?occ1 ?a1)
                  (occurrence_of ?occ2 ?a2)
                  (subactivity_occurrence ?occ1 ?occ2))
             (subactivity ?a1 ?a2)))
```

11.6.8 Аксиома 8

Отношение subactivity_occurrence является транзитивным:

```
(forall (?occ1 ?occ2 ?occ3)
    (implies (and (subactivity_occurrence ?occ1 ?occ2)
                  (subactivity_occurrence ?occ2 ?occ3))
             (subactivity_occurrence ?occ1 ?occ3)))
```

11.6.9 Аксиома 9

Случаи подчиненных деятельностей являются случаями одной подчиненной деятельности, если они удовлетворяют условию включения ветви:

```
(forall (?a1 ?a2 ?occ1 ?occ2)
    (implies (and (occurrence_of ?occ1 ?a1)
                  (occurrence_of ?occ2 ?a2)
                  (subactivity ?a1 ?a2)
                  (not (subactivity_occurrence ?occ 1 ?occ2)))
             (exists (?s)
                (and (subactivity_occurrence ?s ?occ2)
                     (not (subactivity_occurrence ?s ?occ1)))))
```

11.6.10 Аксиома 10

Момент времени beginof составной деятельности совпадает с моментом времени beginof отношения root:

```
(forall (?occ)
    (implies (activity_occurrence ?occ)
        (= (beginof ?occ) (beginof (root_occ ?occ)))))
```

11.6.11 Аксиома 11

Момент времени endof составной деятельности совпадает с моментом времени endof отношения leaf:

```
(forall (?s ?occ)
    (implies (leaf_occ ?s ?occ)
        (= (endof ?occ) (endof ?s))))
```

11.6.12 Аксиома 12

Отношение mono является гомоморфизмом ветвей:

```
(forall (?s1 ?s2 ?a)
    (implies (mono ?s1 ?s2 ?a)
        (hom ?s1 ?s2 ?a)))
```

11.6.13 Аксиома 13

Если случай атомарной подчиненной деятельности имеет образ при гомоморфном отображении ветвей, то существует другой случай атомарной подчиненной деятельности, который находится в отношении mono с этим случаем:

```
(forall (?s1 ?s2 ?a)
    (implies (hom ?s1 ?s2 ?a)
        (exists (?s3)
            (and (mono ?s3 ?s2 ?a)
                (or (min_precedes ?s1 ?s3 ?a)
                    (min_precedes ?s3 ?s1 ?a)
                    (= ?s1 ?s3))))))
```

11.6.14 Аксиома 14

Отношение mono исчерпывается однозначными гомоморфизмами между различными ветвями дерева деятельности:

```
(forall (?s1 ?s2 ?s3 ?a)
    (implies (and (mono ?s1 ?s2 ?a)
                  (mono ?s3 ?s2 ?a))
        (not (or (min_precedes ?s1 ?s3 ?a)
                 (min_precedes ?s3 ?s1 ?a)))))
```

11.6.15 Аксиома 15

```
(forall (?s1 ?s2 ?s3 ?s4 ?occ ?a)
    (implies (and (min_precedes ?s1 ?s2 ?a)
                  (mono ?s1 ?s3 ?a)
                  (mono ?s2 ?s4 ?a)
                  (subactivity_occurrence ?s3 ?occ)
                  (subactivity_occurrence ?s4 ?occ)
                  (iso_occ ?s1 ?s2))
        (min_precedes ?s3 ?s4 ?a)))
```

Приложение А
(обязательное)

Идентификатор ASN.1, присвоенный ИСО 18629-12

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

ISO standard 18629 part 12 version 1

Смысл этого значения определен в ИСО/МЭК 8824-1 и описан в ИСО 18629-1.

Приложение В
(справочное)

Связь с ситуационным исчислением

Теории дерева случаев, дискретных состояний и атомарной деятельности являются дополнениями к более ранним работам по исследованию представления знаний, проводимому членами сообщества исследователей в области искусственного интеллекта. В частности, понятие дерева случаев представляет собой обобщение ситуационного исчисления. Ситуационное исчисление — это формализм логических выводов о действиях, приведенных в [2]. Исходная онтология, приведенная в [2], включает в себя ситуации, флюенты и действия. Несмотря на то что в более ранних исследованиях отсутствовала аксиоматика ситуационного исчисления, в них содержались неформальные интуитивно понятные представления, пригодные для интерпретации данных основных категорий. Ситуации рассматривались как «моментальные снимки мира», флюенты представляли свойства, которые могли быть разными для разных ситуаций, а действия вызывали изменения значений флюентов.

Язык ситуационного исчисления по [2] содержит символ отношения holds(f,s) (для указания того, что значение флюента f является истинным в ситуации s) и символ функции do(a,s) (для обозначения ситуации, которая является результатом выполнения действия a в ситуации s). Предварительные условия и результаты действия представлены аксиомами, определяющими содержание флюентов при выполнении действий.

В [4] приведен набор аксиом ситуационного исчисления. В частности, Рейтер (Reiter) предложил использовать для ситуационного исчисления аксиоматику второго порядка, в которую входит новый символ отношения poss(a,s) для обозначения возможности действия a в ситуации s, а также новый символ константы S0 для обозначения начальной ситуации. Моделями данной теории ядра являются деревья с корневым элементом S0, ветви которых представляют возможные ситуации в будущем с описанием всех возможных вариантов эволюции событий в мире. При этом произвольная последовательность действий определяет ветвь дерева ситуаций, так что две разные последовательности действий приводят к возникновению разных ситуаций. При данном подходе ситуации на интуитивном уровне представляются в виде последовательности случаев действий. Следует отметить, что деревья ситуаций не описывают переходы между состояниями, а представляют возможные в будущем гипотетические «состояния мира» в качестве результатов конкретных последовательностей действий.

Аксиоматика ситуационного исчисления, приведенная в [4], не включает в себя теорию ядра для времени и не предусматривает средств представления параллелизма. В [3] приведена расширенная аксиоматика путем включения этих концепций.

В настоящем стандарте аксиомы, приведенные в [4] и [3], модифицированы. Вместо деревьев, в которых ситуации представлены в виде последовательностей действий, рассмотрены деревья, в которых ситуации в общих чертах аналогичны случаям деятельности. Так как начальная ситуация S0 не соответствует случаю какой-либо деятельности, потребовалось введение случаев начальной деятельности. Кроме того, аксиоматика, приведенная в настоящем стандарте, является аксиоматикой первого порядка.

Приложение С
(справочное)

Пример описания процесса с помощью внешнего ядра PSL

В настоящем приложении приведено подробное описание сценария, в котором формализм PSL в соответствии с комплексом стандартов ИСО 18629 применяется для задач совместного использования информации при выполнении нескольких производственных функций.

Данный сценарий представляет собой сценарий интероперабельности производственной деятельности. Это означает, что он предназначен для демонстрации того, как можно использовать язык PSL для упрощения механизма обмена знаниями о технологическом процессе в производственной среде. Точнее говоря, этот сценарий ориентирован на передачу знаний от планировщика технического процесса диспетчеру предприятия мелкосерийного производства.

В настоящем приложении приведено дополнение к примеру, приведенному в приложении С ИСО 18629-11, иллюстрирующее применение концепций внешнего ядра при спецификации процесса изготовления изделия GT-350.

С.1 Процесс изготовления GT-350

В этом разделе процессы на различных производственных участках объединены на самом высоком уровне в набор видов деятельности, необходимых для изготовления изделия GT-350. В соответствии с описанием конструкции изделия GT-350 (см. ИСО 18629-11, приложение С, таблица С.1) компоненты изделия либо закупают, либо передают для изготовления изделия по субподряду, либо изготавливают изделие на самом предприятии. Описание процесса относится к деятельности, осуществляемой для изготовления внутренних компонентов. Обзор производственного процесса сверху вниз позволяет получить общую картину абстрактной деятельности по изготовлению изделия GT-350, которая расширяется вниз на детальные уровни цехов.

В соответствии с рисунком С.1 процесс изготовления изделия GT-350 распределен между шестью главными стадиями. Работы на первых пяти стадиях (изготовление внутренних элементов, привода, отделочных элементов, двигателя и шасси) не упорядочены по отношению друг к другу, но должны быть завершены до начала окончательной сборки изделия.

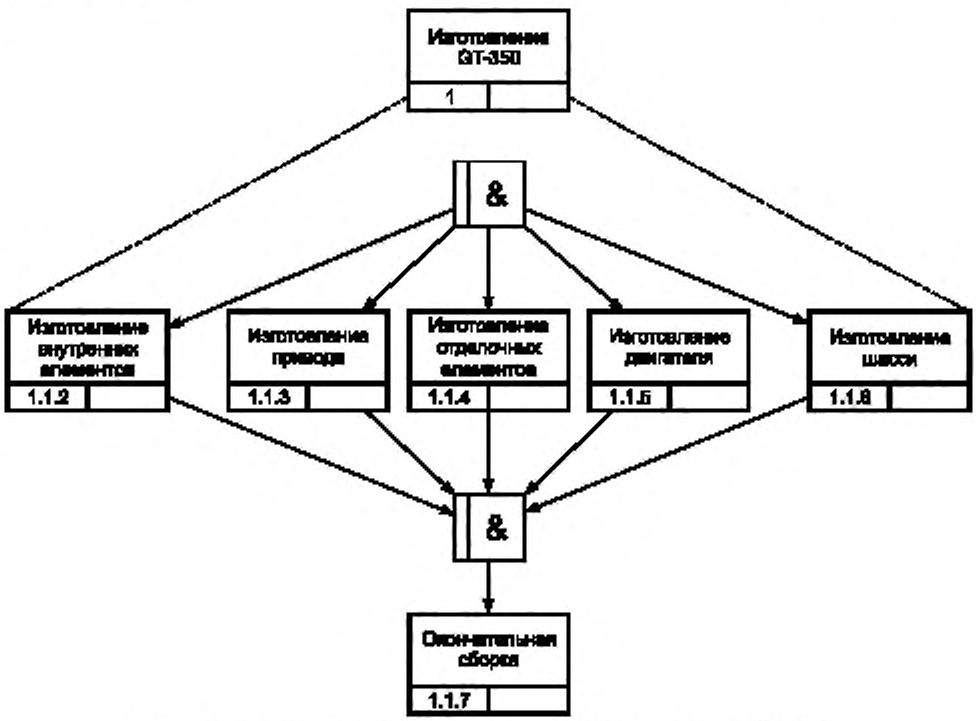

Рисунок С.1 — Верхний уровень процесса изготовления изделия GT-350 [1]

В терминах внешнего ядра PSL представление верхнего уровня производственного процесса выглядит следующим образом:

```
(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)
(forall (?s1 ?s2 ?s3 ?s4 ?s5 ?s6)
(implies (and (leaf_occ ?s1 ?occ1)
(leaf_occ ?s2 ?occ2)
(leaf_occ ?s3 ?occ3)
(leaf_occ ?s4 ?occ4)
(leaf_occ ?s5 ?occ5)
(root_occ ?s6 ?occ6))
(min_precedes ?s1 ?s6 make_gt350)
(min_precedes ?s2 ?s6 make_gt350)
(min_precedes ?s3 ?s6 make_gt350)
(min_precedes ?s4 ?s6 make_gt350)
(min_precedes ?s5 ?s6 make_gt350)))))))
```

Каждая абстрактная деятельность, указанная выше, может быть детализирована, однако в настоящем примере данная детализация не рассматривается.

На основе представления в терминах процесса IDEF3 [1] для абстрактных деятельностей на разных этапах производственного процесса приведен ряд примеров описания процесса с использованием внешнего ядра PSL, представленного в настоящем стандарте.

С.2 Абстрактная деятельность make-engine

Двигатель 350-Engine собирают из компонентов, изготовленных разными отделениями компании CMW. Схема процесса производства изображена на рисунке С.2. Деталь состоит из блока двигателя, жгута и проводки. Сборку двигателя 350-Engine выполняют на сборочной площадке A004. На сборку одного изделия требуется 5 мин.

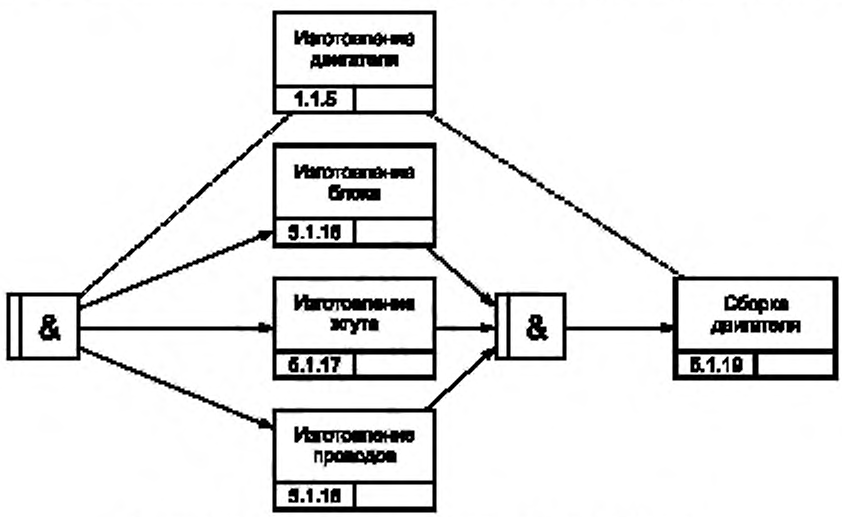

Рисунок С.2 — Процесс изготовления двигателя 350-Engine

В терминах внешнего ядра PSL представление отдельных видов деятельности и соответствующей технологической информации на стадии изготовления двигателя будет следующим:

```
(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 ?occ 1 make_block)
(occurrence_of ?occ2 make_harness)
(occurrence_of ?occ3 make_wires)
(occurrence_of ?occ4 assemble_engine)
(subactivity_occurrence ?occ1 ?occ)
(subactivity_occurrence ?occ2 ?occ)
(subactivity_occurrence ?occ3 ?occ)
(subactivity_occurrence ?occ4 ?occ)
(forall(?s1 ?s2 ?s3 ?s4)
(implies (and (leaf_occ ?s1 ?occ1)
(leaf_occ ?s2 ?occ2)
(leaf_occ ?s3 ?occ3)
(root_occ ?s4 ?occ4))
(min_precedes ?s1 ?s4 make_engine)
(min_precedes ?s2 ?s4 make_engine)
(min_precedes ?s3 ?s4 make_engine)))))
```

С.2.1 Изготовление блока

Блок 350-Block изготавливают на стадии сборки узла двигателя 350-Engine. При этом объединяют результаты работ литейного и механического цехов согласно рисунку С.3.



Рисунок С.3 — Процесс изготовления блока 350-Block [1]

В терминах внешнего ядра PSL представление отдельных видов деятельности и соответствующей технологической информации будет следующим:

```
(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 ?occ 1 produce_molded_metal)
(occurrence_of ?occ2 machine_block)
(min_precedes ?occ1 ?occ2 make_block)))))
```

С.2.2 Изготовление жгута

Жгут 350-Harness (рисунок С.4) изготавливают на стадии сборки узла двигателя 350-Engine. При этом используют результаты работ цеха по производству проводов и кабелей. Сборку жгута 350-Harness выполняют на станке для изготовления проводов и кабелей. На сборку одного комплекта требуется 10 мин.



Рисунок С.4 — Процесс изготовления жгута 350-Harness

В терминах внешнего ядра PSL представление отдельных видов деятельности и связанной технологической информации будет следующим:

```
(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)
(min_precedes ?occ3 ?occ2 make_harness)))))
```

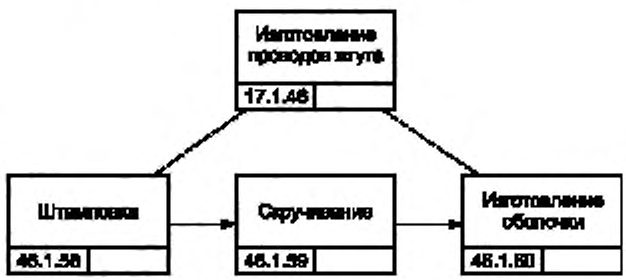

Рисунок С.5 — Процесс изготовления проводов жгута

С.2.3 Изготовление проводов жгута

Комплект проводов 350-Wire-Set изготавливают на стадии сборки узла двигателя 350-Engine. При этом используют результаты работ цеха по производству проводов и кабелей.



Рисунок С.6 — Процесс изготовления проводов 350-Wire

В терминах внешнего ядра PSL представление отдельных видов деятельности и соответствующей технологической информации будет следующим:

```
(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)
(and    (occurrence_of ?occ 1 extrude)
(occurrence_of ?occ2 twist)
(occurrence_of ?occ3 jacket)
(min_precedes ?occ1 ?occ2 make_harness_wire)
(min_precedes ?occ2 ?occ3 make_harness_wire))))
```

Приложение ДА
(справочное)

Сведения о соответствии ссылочных международных стандартов ссылочным 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»
ИСО 10303-1	IDT	ГОСТ Р ИСО 10303-1 — 99 «Системы автоматизации производства и их интеграция. Представление данных об изделии и обмен этими данными. Часть 1. Общие представления и основополагающие принципы»
ИСО15531-1	IDT	ГОСТ Р ИСО 15531-1 — 2008 «Промышленные автоматизированные системы и интеграция. Данные по управлению промышленным производством. Часть 1. Общий обзор»
ИСО 18629-1	—	*
ИСО 18629-11	—	*

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

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

- IDT — идентичные стандарты.

Библиография

[1] Федеральные стандарты по обработке информации. Публикация 184. Определение интеграции для информационного моделирования (IDEF3), FIPS PUB 184, Национальный институт стандартов и технологий, декабрь 1993 г., IDEF3[1)]
(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

[2] McCarthy, J. и Hayes, P. Некоторые философские проблемы с точки зрения искусственного интеллекта. Журнал «Машинный интеллект», 4, под ред. B. Meltzer и D. Michie, Edinburgh University Press, Эдинбург, 1969 г., стр. 463 — 502
(McCarthy, J., and Hayes, P. Some philosophical problems from the standpoint of artificial intelligence. In Machine Intelligence 4, B. Meltzer and D. Michie, eds. Edinburgh University Press, Edinburgh, 1969, pp. 463 — 502)

[3] Pinto, J. Временное рассмотрение в ситуационном исчислении, технический отчет KRR-TR-94-1, Отделение компьютерных наук, Университет Торонто, 1994 г.
(Pinto, J. Temporal Reasoning in the Situation Calculus, Technical Report
KRR-TR-94-1, Department of Computer Science, University of Toronto, 1994)

[4] Reiter, R. Проблема систем отсчета в ситуационном исчислении: простое (частное) решение и полнота результата для целевой регрессии. В сб. под ред. Vladimir Lifschitz. Искусственный интеллект и математическая теория вычислений: сборник статей в честь John McCarthy, 1991 г., стр. 418 — 440, Academic Press, Сан-Диего.
(Reiter, R. The frame problem in the situation calculus: a simple solution (sometimes) and a completeness result for goal regression. In Vladimir Lifschitz, editor, Artificial Intelligence and Mathematical Theory of Computation: Papers in Honor of John McCarthy, 1991, pp. 418 — 440. Academic Press, San Diego)

[1)] Документ доступен в сети Интернет по адресу <http://www.idef.com >.

УДК 65.011:56.681.3:006.354 ОКС 25.040.40 Т 58

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

Редактор *Т. А. Леонова*
Технический редактор *В. Н. Прусакова*
Корректор *С. В. Смирнова*
Компьютерная верстка *В. Н. Романовой*

Сдано в набор 27.02.2014. Подписано в печать 03.06.2014. Формат 60×84 1/8. Бумага офсетная. Гарнитура Ариал. Печать офсетная. Усл. печ. л. 4,65. Уч.-изд. л. 3,40. Тираж 64 экз. Зак. 401.

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