

УДК 004.62

И.С. Чистякова

ИНТЕГРАЦИЯ АКСИОМАТИКИ ДЕСКРИПТИВНЫХ ЛОГИК С РЕЛЯЦИОННОЙ МОДЕЛЬЮ ДАННЫХ

Работа является логическим продолжением ранее опубликованных исследований, посвященных описанию отображений между дескриптивной логикой и реляционной моделью данных. На основе бинарной реляционной структуры данных, созданной ранее, осуществляется отображение аксиоматики дескриптивной логики ALC в реляционную модель данных (RDM). В работе используются полученные ранее результаты исследований, а именно структура данных RM^2 и отображения базовых концептов логики ALC в RDM.

Ключевые слова: семантическая интеграция данных, семантический веб, отображение данных, реляционная модель данных, онтология, ALC, дескриптивная логика, аксиомы дескриптивных логик.

Введение

Интеграция данных является одним из наиболее важных научных направлений в современном мире. Стремительное развитие информационных технологий привело к накоплению огромных объемов данных в различных источниках, разнородных, распределенных, представляющих информацию различными способами, содержащих взаимосвязанные и взаимно противоречивые данные.

В публикации [1] приведен подробный обзор комплексной проблемы интеграции данных в семантическом Web. В процессе анализа было выделено три её составляющих:

- выработка схем интеграции данных;
- выработка отображений между моделями;
- выработка способов манипулирования.

В данной работе мы продолжаем рассматривать вторую составляющую комплексной проблемы, а именно – отображения между дескриптивной логикой (DL) и реляционной моделью данных (RDM).

Исходя из ряда особенностей, изложенных в обзоре [1], в основу нашего подхода к созданию отображений между DL ALC и RDM легли следующие компоненты:

- центральная схема интеграции данных;
- GAV/LAV представление;

- формально-логический уровень абстракции.

Использование центральной схемы интеграции означает, что в системе присутствует одна глобальная «точка контроля», в основе которой лежит некоторая модель данных. В работе [2] ее называют глобальной схемой, а все остальные – локальными схемами, или схемами источников. Мы придерживаемся той же терминологии. В качестве глобальной модели данных мы рассматриваем дескриптивную логику ALC. Обоснование такого выбора приведено в работе [3]. Критическим моментом централизованной схемы остается разработка отображений между моделями, а именно схемами источников и глобальной схемой.

В данной публикации рассматривается только GAV представление, поскольку созданные нами механизмы отображений [3–5] разрешают взаимодействие от глобальной схемы к источнику. Создание отображений LAV представлений является предметом дальнейших исследований.

Решение проблемы происходит на формально-логическом уровне абстракции. Это означает, что для создания отображений мы не опускаемся на уровень прикладных реализаций (конкретных онтологий и реляционных баз данных), а решаем проблему на уровне моделей, лежащих в основе большинства современных информационных систем. Раздел 1 посвящен описанию аксиоматики дескриптивной логики ALC в контексте применяемого

механизма создания отображений. В разделе 2 рассматривается определение аксиоматики ALC. Раздел 3 описывает представление аксиоматики ALC в модели RM^2 . В разделе 4 можно ознакомиться с основными выводами.

DL ALC и её аксиоматика

Согласно определению [6].

Дескриптивная логика – формальный язык для представления знаний и рассуждений в его терминах.

Рассмотрим более широкое определение.

Дескриптивная логика – это семейство формализмов для представления знаний прикладного домена, с помощью первоначального определения понятий домена (*терминологии*) и последующим использованием этих концептов для специфицирования свойств объектов и индивидов, возникающих в домене.

Мы даем упрощенное и более наглядное определение.

Дескриптивная логика – это семейство языков представления знаний, позволяющих описывать понятия предметной области в формализованном виде, обладающие развитыми выразительными возможностями и являющиеся фрагментом логики предикатов.

Для того, чтобы задать какую-либо DL, необходимо задать её *синтаксис* и *семантику*.

Синтаксис описывает, какие выражения (концепты, роли, аксиомы и т. п.) считаются правильно построенными в данной логике.

Семантика указывает, как интерпретировать эти выражения, т. е. придает им формальный смысл.

Существует множество различных дескриптивных логик, которые обладают своими специфичными синтаксисом и семантикой, применяющиеся в различных знание-ориентированных системах, выполняющих различное целевое назначение. При создании бинарной реляционной модели данных, нами была выбрана DL ALC. В работе [3] приведено обоснование данного выбора, поэтому мы не будем рассматривать это подробно.

Приведем **синтаксис** для логики ALC:

$$\top \mid \perp \mid A \mid \neg C \mid C \sqcap D \mid C \sqcup D \mid \exists R.C \mid \forall R.C,$$

где A – атомарный концепт, C и D – произвольные концепты, R – атомарная роль, \top – концепт истина (Thing), \perp – концепт ложь (Nothing), $\neg C$ – концепт-дополнение концепта C , $C \sqcap D$ и $C \sqcup D$ – концепты пересечение и объединение, $\exists R.C$ и $\forall R.C$ – концепты. Другие выражения не являются концептами.

Семантика логики ALC определяется через понятие интерпретации [7].

Интерпретация есть пара

$$I = (\Delta, \cdot^I),$$

состоящая из непустого множества Δ , называемого *областью* данной интерпретации и интерпретирующей функции \cdot^I , которая сопоставляет:

- каждому атомарному концепту $A \in CN$ – произвольное подмножество $A^I \in \Delta$;
- каждой атомарной роли $R \in RN$ – произвольное подмножество $RI \in \Delta \times \Delta$.

Интерпретирующая функция распространяется на множество всех концептов логики ALC однозначным образом – индукцией по построению концепта:

$$\top^I = \Delta; \perp^I = \emptyset; (\neg C)^I = \Delta \setminus C^I;$$

$$(C \sqcap D)^I = (C \cap D)^I; (C \sqcup D)^I = (C \cup D)^I;$$

$$(\exists R.C)^I = \{e \in \Delta \mid \text{существует } d \in \Delta \text{ такой, что } \langle e, d \rangle \in R^I \text{ и } d \in C^I\};$$

$$(\forall R.C)^I = \{e \in \Delta \mid \text{для всех } d \in \Delta \text{ таких, что } \langle e, d \rangle \in R^I \text{ выполнено } d \in C^I\}.$$

Основу нашего механизма составляет следующий тезис: DL можно рас-

смагивать как модель данных. Обоснование данного утверждения приведено в [3], но необходимо напомнить некоторые существенные аспекты, которые пригодятся нам в текущей публикации.

Следующее определение гласит:

Модель данных – это интегрированный набор, а именно:

- структура данных, представляющая собой набор понятий для описания и обработки данных;
- операция, предоставляющая возможность манипулирования данными;
- ограничения целостности, накладываемых на данные.

Опираясь на данное определение, мы утверждаем, что любую DL можно рассматривать как модель данных, в соответствии со следующими положениями:

1) DL обладают синтаксисом и семантикой. Именно в семантике проявляется её структурная часть (возникают понятия множеств, бинарных отношений и т. д.);

2) DL обладает манипулятивной частью, которую представляют такие её компоненты, как конструкторы концептов и ролей;

3) DL обладает целостной частью, которую представляют такие её компоненты, как *аксиомы*.

Таким образом, в качестве центральной модели в интеграционной схеме была выбрана DL. В качестве локальных источников могут выступать любые другие модели (и онтологические в том числе). Мы остановили свой выбор на реляционной модели данных.

Описывая отображения DL в RDM [3–5], мы останавливались на первых двух пунктах, – это структура и манипулятивная часть. В текущей работе мы будем рассматривать последний компонент – аксиоматику DL.

Концепты DL интересны как инструмент записи знаний о предметной области, которые подразделяются на *интенциональные* и *экстенциональные*.

Интенциональные – общие знания о понятиях и их взаимосвязях.

Экстенциональные – знания об индивидуальных объектах, их свойствах и связях с другими объектами.

В соответствии с этим делением, знания, записываемые с помощью языка DL, подразделяются на:

– набор **терминологических аксиом** или TBox (T);

– набор **утверждений (фактов)** об индивидах или ABox (A);

– набор **аксиом для ролей** RBox (R), который является специальным расширением синтаксиса логики ALC, о котором подробнее будем говорить далее.

Терминологической аксиомой называется выражение вида $C \sqsubseteq D$ или $C \equiv D$, где C и D – произвольные концепты. Терминологией (или TBox) называется произвольный конечный набор аксиом данного вида.

Таким образом, в синтаксис аксиоматики ALC вводятся две аксиомы: \sqsubseteq и \equiv (вложенность и эквивалентность концептов).

Аксиоматика обладает своей *семантикой*. Аксиома $C \sqsubseteq D$ истинна в интерпретации I , если $C^I \subseteq D^I$, при этом I называют моделью данной аксиомы. Аналогично, $C \equiv D$ истинна в интерпретации I , если $C^I = D^I$.

Интерпретацию I называют **моделью** терминологии T , если I является моделью всех аксиом из T . Терминология T называется выполнимой (или совместной), если она имеет модель.

Концепт имеет модель, если существует такая интерпретация, при которой концепт C имеет модель, если существует такая C иное, при которой оно не равно пустому множеству.

Концепт C **выполним** в терминологии T , если существует модель I терми-

нологии T , такая что $C^I \neq \emptyset$. Концепты C и D эквивалентны в T , если в любой модели I терминологии T имеем $C^I = D^I$. Концепт C **вложен в D** в терминологии T , если в любой модели I терминологии T имеем $C^I \subseteq D^I$. Концепты C и D называются **непересекающимися** в T , если в любой модели I терминологии T имеем $C^I \cap D^I = \emptyset$.

На этом основании, мы делаем следующее утверждение: любой концепт в терминологии T может быть **эквивалентным другому концепту, быть в него вложенным или не пересекаться с ним**.

Как было сказано ранее, мы решаем проблему создания отображений на формально-логическом уровне абстракции, то есть – на уровне моделей, которые лежат в основе большинства современных информационных систем. В работах [3–5] мы говорили об отображениях семантики DL ALC, а также ее расширений в РМД, где описывали, что собой представляет модель ALC. Задача настоящей публикации определить, что собою представляют отображения для аксиом эквивалентности, вложенности и непересекаемости концептов.

Несмотря на то, что аксиома непересекаемости концептов выражается через аксиомы эквивалентности и вложенности, мы рассматриваем отображение для аксиомы непересекаемости, чтобы бинарная реляционная модель данных имела большую выразительность.

Системой фактов (или $AVox$) называется конечное множество A утверждений вида $a:C$ и aRb , где $a, b \in IN$ есть индивиды, C – произвольный концепт, R – роль.

Говорим, что факт $a:C$ или aRb верен в интерпретации I , если $a^I \in C^I$ или $\langle a^I, b^I \rangle \in R^I$ соответственно, при этом I – модель этого факта.

Для системы фактов $AVox$ иногда вводится «соглашение об уникальности имен», которое означает, что разным име-

нам индивидов интерпретация должна сопоставлять различные элементы из области интерпретации. Если соглашение не выдвигается, то есть допускается что один объект может иметь множество имен, то тогда вводится **аксиома равенства индивидов**, с помощью которой мы указываем, что некоторая пара имен именуется один индивид. Она обозначается « \equiv » и составляет аксиоматику $AVox$ (отображения самой системы фактов было рассмотрено в работе [3]). Эта аксиома показывает, что у одного индивида может быть множество имен, то есть не выполняется принцип уникальности. Однако по-прежнему одно имя именуется только один индивид. Отображение этой аксиомы составляет одну из задач настоящей публикации.

В работах [4–5] мы рассматривали отображения для так называемых расширенных дескриптивных логик. До этого момента мы рассматривали расширения путем добавления в синтаксис дескриптивных логик новых концептов или ролей. Однако, существует еще один способ расширения DL – это введение в логику аксиом для ролей, составляющих $RVox$.

Аксиомы $RVox$ бывают следующие:

- **иерархия ролей (H)**: допускаются аксиомы вида $R \sqsubseteq S$, где R и S – произвольные роли;
- **транзитивные роли (S)**: допускаются аксиомы вида $Tr(R)$, где R – произвольная роль;
- **эквивалентность роли (\equiv)** можно считать сокращением для двух аксиом $R \sqsubseteq S$ и $S \sqsubseteq R$.

Если I – интерпретация, то по определению полагаем:

- $I \models R \sqsubseteq S \Leftrightarrow R^I \subseteq S^I$;
- $I \models Tr(R) \Leftrightarrow$ отношение R^I является транзитивным;
- $I \models R \equiv S \Leftrightarrow R^I = S^I$.

Когда дескриптивная логика расширяется аксиомами для ролей, то для

ее именованія действуют те же принципы, что и для логик, расширенных путем добавления новых концептов или ролей. А именно, если в логике присутствует аксиома иерархии ролей, то буква H добавляется к имени логики. Но если в логике присутствуют транзитивные роли, то буква S пишется вместо букв ALC. Так, например, логика ALC с обратными ролями (I) и иерархией ролей (H) обозначается как ALCH_I, а если расширить ее транзитивными ролями (S), то логика будет называться SH_I.

В задачу текущей публикации входит нахождение отображений аксиом для ролей.

Отображение аксиоматики ALC в RM²

Для того, чтобы начать описание отображений аксиоматики ALC, следует напомнить, что основополагающей частью данного механизма является центральная схема.

Мы уже приводили ее в работе [3] вместе с описанием основополагающих сущностей, однако, в контексте решаемой задачи, схема претерпела существенные изменения. Она показана на рисунке. Дадим некоторые пояснения.

В структуре присутствуют только такие сущности, которые представляют унарные и бинарные отношения, так как в DL ALC n-арные отношения отсутствуют, а также их экземпляры. Следует отметить, что DL с n-арными отношениями существуют, однако они представляют собой специальные расширения ALC, поэтому на данном этапе мы решили отказаться от них, сосредоточившись только на самой ALC.

В отличие от классической RDM, в нашей структуре представлена одновременно и модельная, и метамодельная часть. К метамодельной части мы относим сущности Concept и Role, которые представляют собой перечни имен концептов и ролей соответственно. Следует отметить, что в DL не может существовать отдельно роль, как самостоятельная единица. В нашей структуре роль может

существовать отдельно, будучи представлена именем самой роли, именем концепта-домена и именем концепта-диапазона.

К модельной части относятся сущности ConceptIndividual и RoleIndividual, которые представляют собой индивиды концептов и индивиды ролей. В явном виде таких элементов, как индивиды ролей нет в DL, однако мы вводим данную сущность для связывания индивидов концептов с именами ролей.

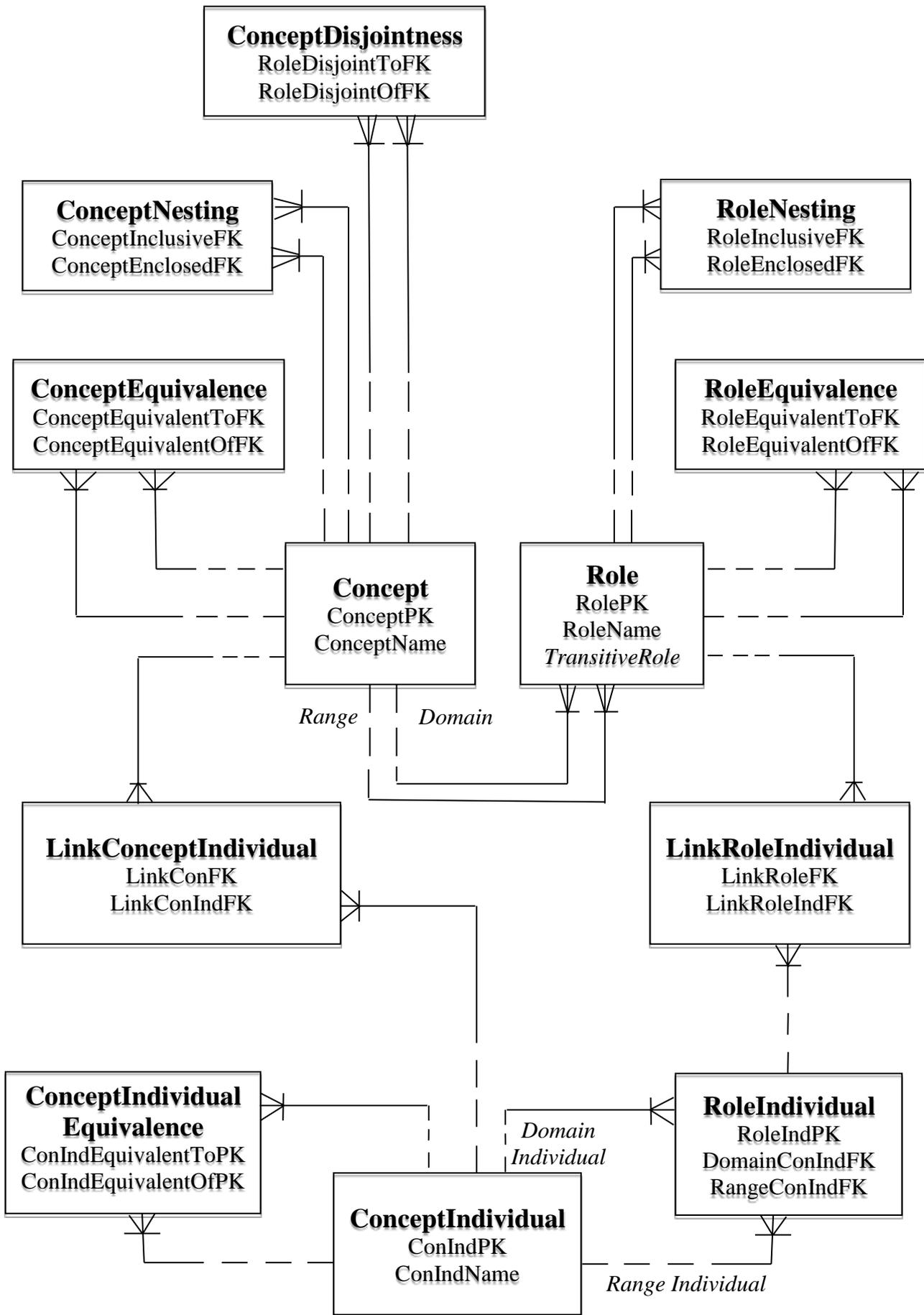
Поскольку один концепт может иметь множество индивидов, а один и тот же индивид может принадлежать нескольким концептам, то для разрешения связи многие-ко-многим мы вводим дополнительную сущность LinkConceptIndividual. Аналогично, сущность LinkRoleIndividual разрешает связь многие-ко-многим для роли и индивидов, которые она связывает. Каждая роль имеет много экземпляров, связывающих два индивида. С другой стороны, каждая конкретная пара индивидов может быть связана несколькими ролями.

В отличие от схемы, которую мы приводили в предыдущих работах, на текущей модели присутствует несколько полноценных сущностей, которые не были представлены в прошлый раз, а именно:

- ConceptEquivalence;
- ConceptNesting;
- ConceptDisjointness;
- ConceptIndividualEquivalence;
- RoleEquivalence;
- RoleNesting.

Эти сущности – есть не что иное, как отображение аксиоматики ALC, о чем мы упоминали в работе [4], в процессе описания схемы. Рассмотрим по отдельности каждую из них.

ConceptEquivalence – сущность, в которой содержатся кортежи, которые связывают два, эквивалентных между собой, концепта. Определяется парой внешних ключей:



Рисунок

1) **ConceptEquivalentToFK** – внешний ключ на имя концепта, которому ставится в эквивалентность другой концепт;

2) **ConceptEquivalentOfFK** – внешний ключ на имя концепта, который ставится в качестве эквивалента первому концепту.

ConceptNesting – сущность, в которой содержатся кортежи, которые связывают два иерархических (\sqsubseteq) концепта. Определяется парой внешних ключей:

1) **ConceptInclusiveFK** – внешний ключ на концепт-родитель;

2) **ConceptEnclosedFK** – внешний ключ на концепт-потомок.

ConceptDisjointness – сущность, в которой содержатся кортежи, которые связывают два, непересекающихся между собой, концепта. Определяется парой внешних ключей:

1) **ConceptDisjointToFK** – внешний ключ на концепт, который не пересекается с другим концептом;

2) **ConceptDisjointOfFK** – внешний ключ на концепт, с которым не пересекается первый концепт.

ConceptIndividualEquivalence – сущность, в которой содержатся кортежи, связывающие, равные между собой, индивиды, подпадающие под соглашение о не уникальности имен. Определяется парой внешних ключей:

1) **ConIndEquivalentToFK** – внешний ключ на индивид, который равен другому индивиду;

2) **ConIndEquivalentOfFK** – внешний ключ на индивид, который ставится в качестве равного первому индивиду.

Следует напомнить, что для индивидов ролей (**RoleIndividual**) мы не вводим такую аксиому, потому что на данный момент этот вопрос недостаточно исследован, и у нас нет никаких оснований утверждать, что одна и та же связь между одними и теми же индивидами может иметь несколько имен. Поэтому, в нашей структуре правильно лишь утверждение о том, что экземпляр роли существует и представляет собой пару индивидов концептов, связанную определенной ролью.

RoleEquivalence – сущность, в которой содержатся кортежи, которые связывают две, эквивалентных между собой, роли. Определяется парой внешних ключей:

1) **RoleEquivalentToFK** – внешний ключ на имя роли, которой ставится в эквивалентность другая роль;

2) **RoleEquivalentOfFK** – внешний ключ на имя роли, которая ставится в качестве эквивалента первой роли.

RoleNesting – сущность, в которой содержатся кортежи, которые связывают две иерархических (\sqsubseteq) роли. Определяется парой внешних ключей:

1) **RoleInclusiveFK** – внешний ключ на роль-родитель;

2) **RoleEnclosedFK** – внешний ключ на роль-потомок.

TransitiveRole – атрибут отношения **Role**, который указывает, является ли данная роль транзитивной или нет. На рисунке выделен курсивом, чтобы наглядно обозначить, что данный атрибут является отображением аксиомы транзитивности, и не присутствовал в модели в предыдущих публикациях.

Выводы

В результате исследований, изложенных в данной работе, получен определенный результат в области интеграции семейств дескриптивных логик, в основе которых лежит DL ALC, в реляционную модель данных. Он охватывает создание бинарной реляционной структуры данных, описание механизмов отображения для базовых операций DL ALC, а также основных её расширений.

Текущая работа была посвящена описанию отображений аксиоматики дескриптивных логик. Показано, что ряд аксиом, вводимых в логику, специальным образом расширяют логику ALC, тем самым образуя новую дескриптивную логику, более выразительную по отношению к базовой. Однако, такие логики не требуют расширения набора операций традиционной реляционной алгебры.

Среди открытых вопросов остаются следующие:

– как отобразить n-арные расширения в бинарную реляционную структуру данных;

– как отобразить операции реляционной алгебры в конструкторы дескриптивной логики.

Интерпретация операций реляционной алгебры в конструкторы дескриптивной логики является объектом дальнейших исследований.

1. Чистякова И.С. Онтолого-ориентированная интеграция данных в семантическом вебе. Проблемы програмування. 2014. – № 2–3. С. 188–196.
2. Lenzerini M. Data Integration: A Theoretical Perspective. Proc. of the 21st ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems (PODS 2002). N. Y.: ACM Press, 2002. P. 233–246.
3. Резниченко В.А., Чистякова И.С. Отображение дескриптивной логики ALC в бинарную реляционную структуру данных. Проблемы програмування. 2015. – № 4. – С. 13–30.
4. Резниченко В.А., Чистякова И.С. Интеграция семейства расширенных дескриптивных логик с реляционной моделью данных. Проблемы програмування. 2016. № 2–3. С. 38–47.
5. Чистякова И.С. Интеграция логик с операциями над ролями с реляционной моделью данных. Проблемы програмування. 2016. № 4. С. 58–65.
6. Baader F., Calvanese D., McGuinness D., Nardi D., and Patel-Schneider P. F., editors. The Description Logic Handbook. Cambridge University Press, 2003.
7. Evgeny Zolin (2009) Description logics (lectures). [Online] Available from: <http://lpcs.math.msu.su/~zolin/dl/> [Accessed: 2009].

References

1. Chystiakova I.S. (2014). Ontology-oriented data integration on the Semantic Web (Онтолого-ориентированная интеграция данных в семантическом вебе). Problems in programming. N 2–3. P. 188–196.

2. Lenzerini M. (2002). Data Integration: A Theoretical Perspective. Proceedings of the 21st ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems. New York, NY, USA. 2002. New York: ACM.
3. Reznichenko V.A. Chystiakova I.S. (2015). Mapping of the Description Logics ALC into the Binary Relational Data Structure (Отображение дескриптивной логики ALC в бинарную реляционную структуру данных). Problems in programming. N 4. P. 13–30.
4. Reznichenko V.A. Chystiakova I.S. (2016). Integration of the family of extended description logics with relational data model (Онтолого-ориентированная интеграция данных в семантическом вебе). Problems in programming. N 2–3. P. 38–47.
5. CHYSTIAKOVA I.S. (2016). Integration of the description logics with extensions into relational data model (Интеграция логик с операциями над ролями с реляционной моделью данных). Problems in programming. N 4. P. 58–65.
6. BAADER F. et al. (2003). The Description Logic Handbook. P. 47–65.
7. EVGENY ZOLIN (2009) Description logics (lections). [Online] Available from: <http://lpcs.math.msu.su/~zolin/dl/> [Accessed: 2009].

Получено 08.11.2016

Об авторе:

Чистякова Инна Сергеевна,
младший научный сотрудник.
Количество научных публикаций в
украинских изданиях – 10.
orcid.org/0000-0001-7946-3611.

Место работы автора:

Институт программных систем
НАН Украины.
03187, г. Киев,
проспект Академика Глушкова, 40.
E-mail: inna_islyamova@ukr.net.
Тел.: +38(066) 847 7784.