



Московский государственный университет имени М.В. Ломоносова

Факультет вычислительной математики и кибернетики

Кафедра автоматизации систем вычислительных комплексов

РЕФЕРАТ

Информационно-ориентированные сети: система именования текстовых объектов

Исполнитель:

Колосов Алексей Михайлович

Москва, 2019

Оглавление

Введение.....	4
Цель работы.....	5
1. Построение информационно-ориентированных сетей	6
1.1 Функционирование ИОС.....	7
1.2 Применение ИОС.....	7
2. Система именования ИОС	8
2.1 Понятие информационного объекта.....	8
2.2 Понятие имени	8
2.3 Операция системы именования	8
2.4 Операции системы разыменования	8
3. Обзор свойств системы именования	9
3.1 Требования системы именования	9
3.2 Тип поиска	10
3.2.1 Поиск по четко сформулированным запросам.....	10
3.2.2 Разведочный поиск.....	11
3.3 Тип имени	12
3.3.1 Машинное имя.....	12
3.3.2 Пользовательское имя.....	12
3.4 Доступная для построения имени информация.....	13
3.4.1 Локальная информация.....	14
3.4.2 Глобальная информация.....	14
3.5 Переименование при коллизии.....	15
3.5.1 Нарастивание имени.....	16

3.5.2 Построение имени заново	16
3.6 Сложность поиска	17
3.7 Выводы.....	17
4. Предлагаемая система именования	18
4.1 Модель информационного объекта	19
4.2 Имя информационного объекта.....	19
4.3 Соответствие требованиям.....	20
4.4 Свойства.....	21
4.5 Реализация	22
5. О влиянии системы именования на построение ИОС.....	23
6. О единой системе именования и едином имени	25
7. Концепция семантической маршрутизации	26
Заключение	29
Список литературы	30

Основные понятия

ИО — информационный объект

Тип ИО — один из типов графический, аудио, видео и текстовый ИО

ИОС — информационно-ориентированная сеть

Задача ИОС — одна из задач Именованное, Поиск, Передача, Хранение

Служба ИОС — одна из служб Публикации, Подписки, Анонсирования

Уровень функционирования ИОС — один из уровней Представления данных и Доступа к данным

Введение

В настоящее время объём данных, передаваемых по сети Интернет, постоянно увеличивается. В то же время пропускная способность каналов связи также растёт. Однако становится больше и конечных устройств. В результате всё более актуальны вопросы, что предпочтительнее: переносить вычисления к данным или перемещать данные к вычислениям; обрабатывать данные на стороне клиента или удалённо; передавать уже подготовленные к использованию данные или данные, которые ещё нужно преобразовывать.

Ответить на вопрос про перенос вычислений и перемещение данных позволяет кэширование, благодаря которому данные остаются там, где над ними проводятся вычисления; бессерверные вычисления и удалённый вызов процедур позволяют ответить на вопрос про сторону обработки данных, когда вызов функции производится на стороне клиента, но вычисления происходят удалённо; наконец, graphics streaming позволяет не передавать по сети видеопоток или проводить рендеринг полностью на клиенте: вычислительная часть также происходит удалённо, но часть, связанная с отображением графики, выполняется на стороне клиента, таким образом по сети передаются только инструкции, состоящие из функций графического API, и данные, содержащие предвычисленные аргументы.

Graphics streaming является примером использования кэширования, бессерверных вычислений и подготовленных к использованию данных. Отдельно стоит отметить, что использование в graphics streaming кэширования позволяет снизить объём передаваемых по сети данных приблизительно на 80%. [1]

Таким образом, перемещение данных к вычислениям является приоритетной задачей, тем более, что в некоторых случаях возможно предсказать распределение востребованности данных в сети и переместить данные к вычислениям до начала вычислений.

В случае перемещения данных к вычислениям данные больше не зависят от устройств, которые когда-либо их содержали, поскольку любые данные могут быть востребованы в любой точке сети и в любой момент времени могут содержаться на любом устройстве. Это означает, что в сети больше нет понятия адреса устройства, за которым данные закреплены на постоянной основе.

Подход, когда обращение к данным производится по их имени, не используя адрес устройства, их содержащего, называется информационно-ориентированным, а сети, построенные с его помощью — информационно-ориентированными сетями(ИОС). В работе предложено построение ИОС, состоящее из системы именования информационных объектов(ИО), системы поиска ИО по имени, системы передачи и хранения ИО.

Работа посвящена разработке системы именования текстовых объектов, поскольку система именования существенно влияет на построение ИОС, так как способ построения имени определяет функционирование ИОС — подробнее см. раздел 6. Важно отметить, что система именования различается у разных типов ИО — подробнее см. раздел 7. Наконец, система именования может быть применена для организации семантической маршрутизации — подробнее см. раздел 8. Таким образом, разработка системы именования текстовых объектов является актуальной.

Цель работы

Целью работы является разработка системы именования текстовых объектов, позволяющей по текстовому объекту определить его имя.

Система именования должна удовлетворять требованиям уникальности имени; возможности восстановления имени, известного частично; положительной корреляции разностей между именами и ИО. Выбор данных требований должен быть обоснован.

1 Построение информационно-ориентированных сетей

Построение ИОС проводится по модели «Поставщик-Потребитель» («Publishing-Subscribing»)[2]. Можно предложить построение ИОС, состоящее из решения задач Именованное, Поиск, Передача, Хранение:

1. В задаче Именованное требуется сопоставить каждому ИО уникальное имя. Для этого необходимо определить систему именования — задать класс ИО, способ его отображения на множество имен, а также определить понятие имени ИО, по которому ИО идентифицируются в ИОС.
2. В задаче Поиск требуется представить механизм определения имени ИО по заданному имени, при этом имя может быть известно не полностью. Для этого необходимо определить систему разыменования.
3. В задаче Передача определяется способ доступа к запрашиваемому ИО. Для этого необходимо определить жизненный цикл запроса ИО, состоящий из формирования запроса, отправление запроса, получение запроса, реализация запроса.
4. В задаче Хранение определяется место ИО. Для этого необходимо определить жизненный цикл ИО, состоящий из первичного размещения ИО, перемещения ИО, обнаружения ИО, выдачи ИО по запросу.



Рисунок 1: Соответствие задач и уровней функционирования ИОС.

Данные задачи соответствуют двум логическим уровням функционирования ИОС (см. Рис. 1): уровню представления данных и уровню доступа к данным. Так, задачи Именованное и Поиск относятся к уровню представления данных, а задачи Передача и Хранение — к уровню доступа к данным.

На уровне доступа к данным решаются задачи, связанные с передачей и хранением данных. В частности, на данном уровне решаются вопросы безопасности, множественной доставки ИО, а также перемещения ИО в места, где они наиболее востребованы, и кэширования ИО.

Информационные объекты «существуют» на уровне представления данных. На уровне доступа к данным они «погружаются» в контейнеры, носители данных, которые обладают своим набором атрибутов — атрибутами носителя, например, сведениями о дате создания; пользователе, добавившем объект в систему; правах доступа к объекту; размере объекта. Отдельно стоит отметить, что идентификатор Поставщика это один из атрибутов носителя, также как и идентификатор Потребителя, который используется для Подписки на ИО.

1.1 Функционирование ИОС

В общем случае ИОС функционирует следующим образом. В модели «Поставщик-Потребитель» пользователь ИОС может совершать два действия — Подписываться на ИО и Публиковать ИО. Тот, кто подписывается на ИО, называется Потребителем, а тот, кто публикует ИО — Поставщиком. Потребитель не может изменять содержимое ИО без разрешения Поставщика. Публикация контента производится через службу Публикации, подписка — через службу Подписки. Подписка бывает одноразовая и постоянная. В первом случае Потребитель получает ИО один раз, во втором — каждый раз, когда ИО обновляется Поставщиком. Подписка на ИО происходит по имени ИО, поэтому если содержимое ИО меняется таким образом, что это приводит к изменению имени ИО, все Потребители, у которых есть постоянная подписка на данный ИО, получают обновление имени ИО.

Для того, чтобы опубликовать ИО, Поставщик выбирает и передает службе Публикации файл-источник. Данная служба формирует имя ИО и загружает его в ИОС. После этого Поставщик получает имя ИО и затем может анонсировать полученное имя Потребителям через службу Анонсирования.

Для того, чтобы подписаться на ИО, необходимо в службе Подписки указать имя ИО. Одному имени соответствует один ИО. В ответ на запрос, служба Подписки возвращает ИО, имя которого соответствует заданному имени. С каждым ИО связан список допустимых действий, которые Потребитель может совершать с данным ИО: одноразовое получение ИО, подписка на обновления ИО, одноразовое получение ИО и подписка на обновления ИО.

1.2 Применение ИОС

Основная область применения ИОС — построение сетевых поисковых систем, в которых поиск осуществляется самой сетью, то есть для обнаружения данных не требуется использование дополнительных систем. В общем случае поиск может одновременно проводиться по ИО всех типов, чего не могут традиционные поисковые системы, в которых поисковая выдача индивидуальна для каждого типа ИО. Стоит отметить, что традиционная сеть это тоже своего рода поисковая система, которая по адресу находит устройство, а если его нет, то сообщает об этом.

Также ИОС могут быть использованы для организации рекомендательных систем; систем доставки контента(Content Delivery System, CDN); систем без обратной связи, например, стриминговых систем; систем с обратной связью, например, систем облачного гейминга.

2 Система именования ИОС

Система именования задаёт отображение множества информационных объектов на множество имен и операцию, реализующую это отображение.

2.1 Понятие информационного объекта

Каждый Информационный объект(ИО) представлен в виде данных, описывающих некоторый факт или явление. Данными могут быть текст, аудио, видео или графический объект. В общем случае любой ИО может быть представлен как бинарный объект. Каждый ИО содержит информацию — интерпретацию данных.

2.2 Понятие имени

Имя позволяет идентифицировать ИО. Чаще всего имя определяется содержанием ИО, однако может и не зависеть от него, например, являясь порядковым номером ИО в системе. В случае когда имя определяется содержанием ИО, оно является признаковым описанием ИО. Задача имени — отразить содержание ИО.

2.3 Операция системы именования

Система именования представлена следующей операцией:

1. `def put(ИО) return name` — по ИО определить его имя

Подробнее данная операция рассмотрена в разделе Предлагаемая система именования.

2.4 Операции системы разыменования

С системой именования связана система разыменования, которая необходима для решения задачи Поиск. Система разыменования задаёт отображение множества имён на множество информационных объектов и операции, реализующие это отображение.

Система разыменования представлена следующими операциями:

1. `def get(name) return ИО` — по имени определить ИО
2. `def find(partly_name) return {name}` — по частично известному имени определить множество соответствующих имён

3. Обзор свойств системы именования

Обзор организован следующим образом. Определены и обоснованы требования к системе именования. Рассмотрены свойства, определяющие систему именования. Свойства сгруппированы в пары. В результате обзора определены те свойства, которые в большей степени соответствуют требованиям системы именования. Данными свойствами должна обладать предлагаемая система именования.

Рассмотрены следующие группы свойств системы именования:

- 1) Тип поиска: поиск по четко сформулированным запросам, разведочный поиск.
- 2) Тип имени: машинное, пользовательское.
- 3) Доступная для построения имени информация: локальная, глобальная.
- 4) Переименование при коллизии: наращивание имени, построение имени заново.

Также рассмотрено дополнительное свойство: сложность поиска. Однако поскольку данное свойство является количественным, а не качественным, для него не проводится обзор соответствия требованиям.

3.1 Требования системы именования

К системе именования предъявляются следующие требования:

- 1) Уникальность имени(далее Требование уникальности)

Имя каждого ИО должно быть уникально в том смысле, что каждому имени соответствует ровно один ИО. При этом одному ИО может соответствовать несколько имен. Данное требование необходимо поскольку если нескольким ИО соответствуют одинаковые имена, эти объекты должны совпадать, в противном случае не будет возможности их различить без получения полностью.

- 2) Возможность восстановления имени, известного частично(далее Требование восстановления по части)

В случае когда имя известно не полностью должна быть возможность определить множество имён ИО, соответствующих имени, известному частично. Данное требование необходимо поскольку чаще всего информационная потребность пользователя не является исчерпывающей. В этом заключается существенное отличие использования системы именования в решении задач Поиск и Хранение — в задаче Хранение доступ осуществляется только по полностью известным именам.

- 3) Положительная корреляция разностей между именами и ИО(далее Требование положительной корреляции)

Данное требование означает, что из разности между именами должна следовать разность между ИО. Обратное утверждение следует из данного, поскольку если ИО отличаются незначительно, имена не могут отличаться значительно. Разность определяется по-разному для машинного и пользовательского имён. Кроме того разность вычисляется не между именами и ИО, а между их моделями. Имя является моделью по построению.

Стоит отметить, что данные требования могут быть применены к системе именования ИО любого типа, не только текстовых объектов. Различаются только модели ИО.

Далее рассмотрены группы свойств системы именования и для входящих в них свойств определена степень соответствия требованиям. Для каждого свойства рассматриваются все требования. Для свойства Доступная для построения имени информация рассматривается дополнительное требование: независимость сложности построения имени от количества ИО. Данное требование является дополнительным, поскольку необходимость его выполнения существенно сужает пространство возможных систем именования. Кроме того, данное требование является качественным, в отличие от остальных требований, которые являются количественными и могут быть использованы как мера качества предлагаемого решения. Выполняемость данного требования также рассматривается в главе Предлагаемое решение.

3.2 Тип поиска

Система именования может быть использована для поиска по четко сформулированному запросу(known-item search), когда запросом является имя или его часть и разведочному поиску(exploratory search), когда запросом является ИО или его часть. [3]

3.2.1 Поиск по четко сформулированным запросам

Примерами поиска по четко сформулированному запросу является поиск по ключевым словам и полнотекстовый поиск. Результатом поиска является точное соответствие.

Соответствие требованиям:

1) Требование уникальности — выполнимо

Для данного свойства возможно выполнить требованием уникальности, поскольку для точного соответствия может быть организовано однозначное отображения множества имён на множество ИО.

2) Требование восстановления по части — выполнимо

Для данного свойства возможно выполнить требование восстановления по части, поскольку могут быть использованы шаблоны, в которых Пользователь указывает известные ему атрибуты, а оставшиеся поля оставляет пустыми.

3) Требование положительной корреляции — выполнимо частично

Для данного свойства возможно лишь частично выполнить требование положительной корреляции, поскольку понятие разности может быть не определено для имен, например, в случае использования категориальных атрибутов.

3.2.2 Разведочный поиск

Пользователь может обладать информационной потребностью в систематизации некоторой области знаний. В этом случае используется разведочный поиск, позволяющий производить изучение новых предметных областей. [4]

В данном типе поиска запросом может быть текст произвольной длины или даже подборка текстов. Результатом поиска может являться «карта» предметной области, позволяющая эффективно организовать обучение.

Соответствие требованиям:

1) Требование уникальности — не выполнимо

Для данного свойства нет возможности выполнить требованием уникальности, поскольку в данном типе поиска нет понятия имени в явном виде.

2) Требование восстановления по части — выполнимо частично

Для данного свойства возможно лишь частично выполнить требование восстановления по части, поскольку из-за отсутствия в данном типе поиска понятия имени в явном виде возможен только вариант восстановления ИО по его части.

3) Требование положительной корреляции — выполнимо

Для данного свойства возможно выполнить требование положительной корреляции, поскольку неявные имена могут удовлетворять данному требованию, так как для них может быть задана необходимая метрика, определяющая понятие разности.

Вывод

В Таблице 1 приведены результаты сравнительного анализа соответствия требованиям свойств из группы Тип поиска.

Из Таблицы 1 делается вывод о том, что Поиск по четко сформулированным запросам в большей степени удовлетворяет требованиям системы именования.

	Поиск по четко сформулированным запросам	Разведочный поиск
Уникальность имени	+	-
Восстановление по части	+	++
Положительная корреляция	++	+

Таблица 1. Сравнительный анализа соответствия требованиям свойств группы Тип поиска.

3.3 Тип имени

Имя может быть машинным в том смысле, что для человека оно неинтерпретируемо, но позволяет вычислительно эффективно проводить операцию именованя и поиска по имени, возможно известном частично. Также имя может быть пользовательским в том смысле, что пользователь по имени может предположить содержание ИО, то есть имя является интерпретируемым для него.

Имя в явном виде из предыдущего раздела это пользовательское имя. Неявное имя — машинное.

Изначально вопросом именованя является вопрос о том, может ли существовать «единое» имя, являющееся одновременно и машинным, и пользовательским. Из истории развития вычислительной техники таких примеров неизвестно. Поэтому скорее всего такого имени существовать не может. Таким образом, предполагается необходимость построения двух имен.

3.3.1 Машинное имя

Примером машинного имени может быть вектор чисел. Машинное имя в некотором смысле интерпретируемо для систем, когда может рассматриваться как координаты.

Соответствие требованиям:

1) Требование уникальности — выполнимо

Для данного свойства возможно выполнить требованием уникальности, поскольку, вектора действительных чисел могут быть попарно различны.

2) Требование восстановления по части — выполнимо частично

Для данного свойства возможно лишь частично выполнить требование восстановления по части, поскольку понятие части может быть не определено, например, в случае когда имя это одно число.

3) Требование положительной корреляции — выполнимо

Для данного свойства возможно выполнить требование положительной корреляции, поскольку для машинных имён могут быть заданы метрики, удовлетворяющие данному требованию, например, для векторов это может быть косинусная метрика.

3.3.2 Пользовательское имя

Примером пользовательского имени может быть аннотация ИО ключевыми словами. По данному имени пользователь имеет возможность предположить содержание ИО.

Соответствие требованиям:

1) Требование уникальности — не выполнимо

Для данного свойства нет возможности выполнить требованием уникальности, поскольку у близких по смыслу объектов могут полностью совпадать части, по которым происходит построение имени.

2) Требование восстановления по части — выполнимо

Для данного свойства возможно выполнить требование восстановления по части, поскольку Пользователь может не указать ряд значений, однако указанные значения образуют множество, для которого возможно определить надмножества, в которое указанное множество входит целиком.

3) Требование положительной корреляции — выполнимо

Для данного свойства возможно выполнить требование положительной корреляции, поскольку в случае, когда имя сохраняет информацию об исходном объекте, похожим с точки зрения пользователя именам соответствуют похожие с точки зрения пользователя объекты.

Вывод

В Таблице 2 приведены результаты сравнительного анализа соответствия требованиям свойств группы Тип имени.

Из Таблицы 2 делается вывод о том, что Машинное имя в большей степени удовлетворяет требованиям системы именования.

	Машинное имя	Пользовательское имя
Уникальность имени	+	-
Восстановление по части	--+	+
Положительная корреляция	+	+

Таблица 2. Сравнительный анализа соответствия требованиям свойств группы Тип имени.

3.4 Доступная для построения имени информация

Для построения имени может быть доступна информация только о том ИО, для которого производится построение имени, локальная информация. Также может быть доступна информация о других, в общем случае всех ИО, включая их имена — глобальная информация.

Кроме того для данного свойства рассматривается дополнительное требование: независимость сложности построения имени от количества ИО.

3.4.1 Локальная информация

Примером локальной информации может быть множество слов, встречающихся в тексте, который именуется, и их частотности.

Соответствие требованиям:

1) Требование уникальности — не выполнимо

Для данного свойства нет возможности выполнить требование уникальности, поскольку в случае использования только локальной информации невозможно гарантировать отсутствие коллизий, поскольку количество ИО не ограничено.

2) Требование восстановления по части — выполнимо

Для данного свойства возможно выполнить требование восстановления по части, поскольку локальная информация индивидуальна и может быть использована для идентификации объекта в случае, если доступна только часть сведений об объекте.

3) Требование положительной корреляции — не выполнимо

Для данного свойства не возможности выполнить требование положительной корреляции, поскольку наличие только функции расстояния не позволит вычислительно эффективно находить ближайший к заданному объект — для этого потребуется полный перебор. Для эффективного решения этой задачи объекты должны обладать координатами, для определения которых необходима глобальная информация.

4) Требование независимости от количества ИО — выполнимо

Для данного свойства возможно выполнить требование независимости от количества ИО, поскольку локальная информация не зависит от количества ИО.

3.4.2 Глобальная информация

Примером глобальной информации может быть множество слов, встречающихся в коллекции текстов и их частотности.

Соответствие требованиям:

1) Требование уникальности — выполнимо

Для данного свойства возможно выполнить требованием уникальности, поскольку коллизии могут разрешаться на этапе построения. При это возникает вопрос, о том какой из объектов переименовывать при коллизии. Данный вопрос рассматривается в следующей секции.

2) Требование восстановления по части — выполнимо частично

Для данного свойства возможно лишь частично выполнить требование восстановления по части, поскольку для частично известной информации может существовать множество вариантов восстановления, поэтому определение всех таких вариантов является вычислительно трудоёмкой задачей.

3) Требование положительной корреляции — выполнимо

Для данного свойства возможно выполнить требование положительной корреляции, поскольку для глобальной информации возможно задание координат объектов, которое позволяет выполнить данное требование.

4) Требование независимости от количества ИО — выполнимо частично

Для данного свойства возможно лишь частично выполнить требование независимости от количества ИО, поскольку для выполнения данного требования необходимо ограничить количество информации, которое используется для построения имен новых объектов и не зависит от их количества.

Вывод

В Таблице 3 приведены результаты сравнительного анализа соответствия требованиям свойств группы Доступная для построения имени информация.

Из Таблицы 3 делается вывод о том, что Глобальная информация в большей степени удовлетворяет требованиям системы именования.

	Локальная информация	Глобальная информация
Уникальность имени	-	+
Восстановление по части	+	+-
Положительная корреляция	-	+

Таблица 3. Сравнительный анализа соответствия требованиям свойств группы Доступная для построения имени информация.

3.5 Переименование при коллизии

В случае изменения набора ИО, имя нового ИО может совпадать с именем ИО, уже содержащегося в наборе, в этом случае происходит коллизия и необходимо определить, какой ИО переименовать для того, что выполнять требование уникальности. Первым вариантом является расширение имени выбранного ИО, его наращивание, с тем, чтобы имя вновь стало уникальным. Другим вариантом является построение имени выбранного ИО заново.

Стоит отметить, что если при расширении множества ИО, коллизий не возникает, количество объектов может быть произвольным. Поэтому важен именно вопрос о переименовании объектов в случае коллизии. Также важен вопрос о том, какой из объектов переименовывать.

3.5.1 Нарращивание имени

Нарращивание имени может быть произведено разными способами, как за счёт удлинения, так и за счёт дополнительных итераций в процессе построения имени.

Соответствие требованиям:

1) Требование уникальности — выполнимо

Для данного свойства возможно выполнить требованием уникальности, поскольку в случае отсутствия ограничений на наращивание, имя может наращиваться до тех пор, пока не станет уникальным.

2) Требование восстановления по части — выполнимо

Для данного свойства возможно выполнить требование восстановления по части, поскольку имея инкрементальную структуру имя может быть представлено в виде уровней, то есть не быть монолитным. В этом случае для него определимо понятие части.

3) Требование положительной корреляции — выполнимо частично

Для данного свойства возможно лишь частично выполнить требование положительной корреляции, поскольку при наращивании имен у объектов могут начать различаться размерности, что будет значить их несравнимость в смысле разности.

3.5.2 Построение имени заново

Соответствие требованиям:

1) Требование уникальности — выполнимо

Для данного свойства возможно выполнить требованием уникальности, поскольку, поскольку новое имя может отличаться от всех остальных имён.

2) Требование восстановления по части — выполнимо частично

Для данного свойства возможно лишь частично выполнить требование восстановления по части, поскольку имя может быть монолитным, например, в случае, когда учитывает связи внутри объекта, поэтому для такого имени понятие части не определено.

3) Требование положительной корреляции — выполнимо

Для данного свойства возможно выполнить требование положительной корреляции, поскольку имена могут строиться таким образом, чтобы обладать одинаковой размерностью и являться координатами.

Вывод

В Таблице 4 приведены результаты сравнительного анализа соответствия требованиям свойств группы Переименование при коллизии.

Из Таблицы 4 делается вывод о том, что Построение имени заново в большей степени удовлетворяет требованиям системы именования.

	Дополнение имени	Построение имени заново
Уникальность имени	+	+
Восстановление по части	+	+-
Положительная корреляция	++	+

Таблица 4. Сравнительный анализа соответствия требованиям свойства группы Переименование при коллизии.

3.6 Сложность поиска

Данное свойство является дополнительным. Сложность поиска является количественным свойством и зависит от того, на каком пространстве реализуется поисковый запрос. Поисковый запрос может реализовываться как на пространстве имен, так и на пространстве ИО. Сложность поиска определяется тем как устроена связь между пространством имен и пространством ИО.

Значение данного свойства рассматривается в главе Предлагаемая система именования.

3.7 Выводы

В результате обзора делается вывод о том, что следующие свойства систем именования в большей степени удовлетворяют требования системы именования:

- 1) Тип поиска: поиск по четко сформулированным запросам.
- 2) Тип имени: машинное.
- 3) Доступная для построения имени информация: глобальная.
- 4) Переименование при коллизии: построение имени заново.

Предлагаемая система именования должна обладать данными свойствами.

4. Предлагаемая система именования

В работе решается задача о существовании системы именования, обладающей свойствами, определёнными в обзоре. Поскольку такая система именования существует, проведение обзора систем именования не требуется. В качестве меры качества предлагаемой системы именования рассматривается степень соответствия требованиям, определяемая с помощью экспериментов.

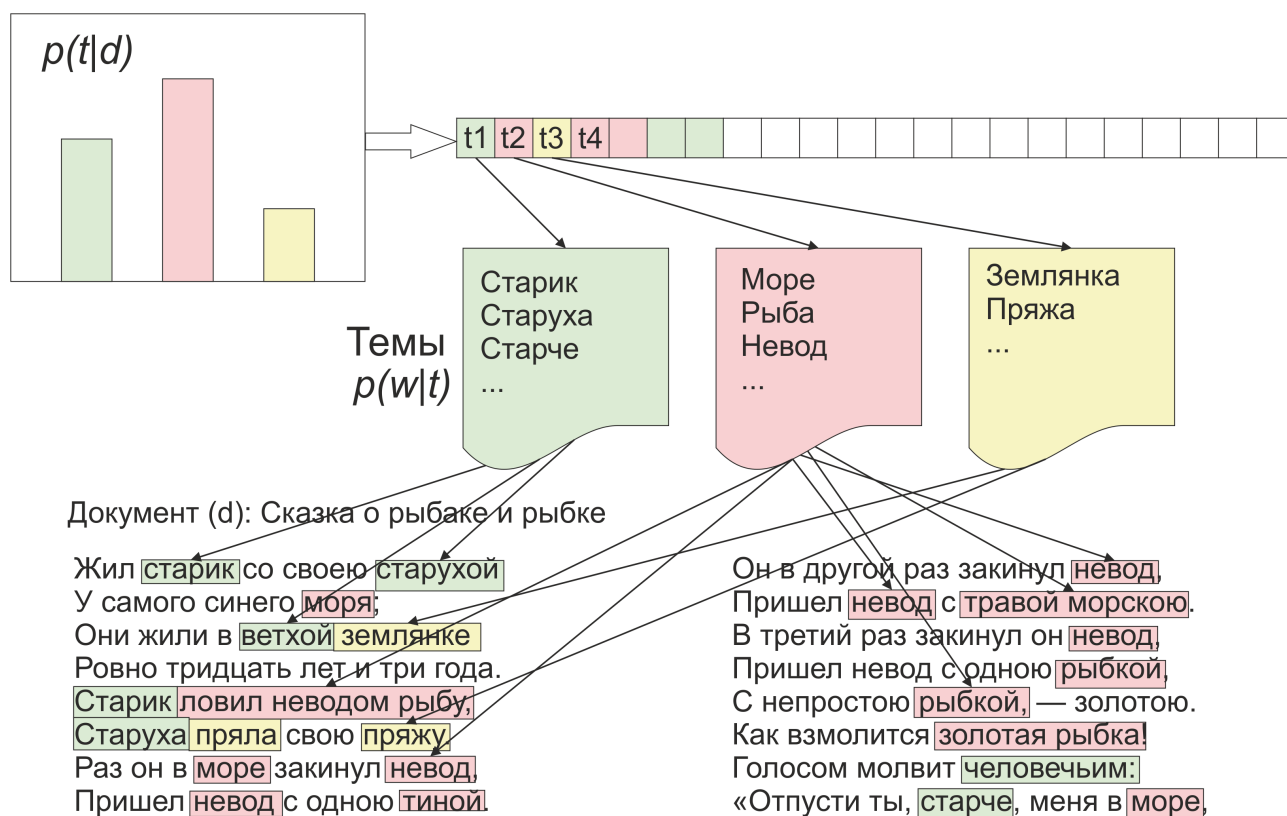


Рисунок 2. Тематическая модель[5].

Для организации системы именования предлагается использовать вероятностное тематическое моделирование[6]. Этот подход позволяет сопоставить документу вектор вероятностей соответствия документа множеству тем(см. Рис. 2). Темы представляют собой распределения характеризующих их слов. Формально решается задача матричного разложения. В исходной матрице в строках находятся слова из словаря коллекции, в столбцах номера документов. Данная матрица раскладывается на две: Фи и Тетта. В матрице Фи в строках находятся слова из словаря коллекции, в столбцах темы, в ячейке вероятность принадлежности слова к соответствующей теме(см. Таблица 5). В матрице Тетта в строках находятся темы, в столбцах номера документов, в ячейке вероятность принадлежности темы к соответствующему документу(см. Таблица 6).

Стоит отметить, что темы являются скрытыми переменными, то есть не могут быть предъявлены в явном виде. Однако их можно достаточно точно определять по семантическим ядрам, т.е. множествам слов, имеющих высокую вероятность принадлежности к данной теме и низкую ко всем остальным.

Также стоит отметить, что тематическая модель осуществляет так называемую «мягкую» кластеризацию, когда один документ может относиться сразу к нескольким темам с соответствующими вероятностями[6]. Но также возможен и случай «жесткой» кластеризации, когда документ относится с вероятностью единица к одной теме и ноль ко всем остальным.

	topic _0	topic _1	topic _2	topic _3	topic _4	topic _5	topic _6	topic _7	topic _8
(@default_class, тестирование_эффект)	0.000 023	0.000 000	0.000 000	0.000 000	0.000 000	0.000 000	0.000 000	0.000 000	0.000 000
(@default_class, ошибка_пользователь)	0.000 000	0.000 000	0.000 000	0.000 024	0.000 000	0.000 019	0.000 000	0.000 000	0.000 000
(@default_class, цвет_нахождение)	0.000 000	0.000 000	0.000 000	0.000 00	0.000 000	0.000 000	0.000 000	2.447 888e -05	0.000 000

Таблица 5. Пример матрицы Фи.

	200	201	202	203	204
topic_0	0.000000	0.0	0.0	0.0	0.0
topic_1	0.000000	0.0	0.0	0.0	0.0
topic_2	0.000000	1.0	0.0	1.0	0.0
topic_3	0.000000	0.0	0.0	0.0	0.0
topic_4	0.000000	0.0	0.0	0.0	1.0
topic_5	0.000000	0.0	0.0	0.0	0.0
topic_6	0.042908	0.0	0.0	0.0	0.0
topic_7	0.000000	0.0	0.0	0.0	0.0
topic_8	0.957092	0.0	1.0	0.0	0.0

Таблица 6. Пример матрицы Тетта.

4.1 Модель информационного объекта

Для представления текста используется модель мешка слов. Данная модель не учитывает порядок слов в тексте, но учитывает количество вхождений слова в текст. [6]

4.2 Имя информационного объекта

Именем документа является вектор вероятностей принадлежности документа к темам. Число тем определяется заранее. Длина вектора равна количеству тем. Каждая компонента вектора

содержит вероятность принадлежности документа к соответствующей данной компоненте теме.

4.3 Соответствие требованиям

В данном разделе описывается возможность соответствия предлагаемой системы именования требованиям системы именования. Оценка степени соответствия данным требованиям проводится в разделе Экспериментальное исследование.

Требование уникальности

Имя является вектором действительных чисел. Поэтому при проведении мягкой кластеризации и достаточно высокой размерности вектора, то есть значительном количестве тем, коллизий быть не должно. Однако в случае жесткой кластеризации и не значительном количестве тем коллизии возможны. Влияние жесткой кластеризации на количество коллизий в зависимости от количества тем рассматривается в экспериментальном исследовании. В качестве модельного примера можно привести случай, когда все документы подвергнуты жесткой кластеризации, документов N , размерность пространства $< \log_2(N)$. В этом случае коллизии неизбежны.

Таким образом, степень соответствия предлагаемой системы именования данному требованию зависит от параметров тематической модели.

Требование восстановления по частично известному имени

В случае тематического моделирования Пользователю может быть известна часть тем имени объекта, либо известна часть объекта, если темы неизвестны.

В первом случае Пользователь ставит единицу в позициях имени, соответствующих известным ему темам. Соответствие позиций с единицами и тем определяется вручную, по значимым словам тем. После этого в матрице Тетта по полученному шаблону производится поиск всех соответствующих данному шаблону имён.

Во втором случае для документа строится имя и затем по функции расстояния определяется ближайшее имя в матрице Тетта. В качестве функции расстояния в тематическом моделировании принято использовать косинусную метрику[4]. Формально это величина схожести двух объектов, однако по формуле $1 - \cos(d1, d2)$ может быть получено расстояние между объектами.

Важным является тот факт, что выполняется дополнительное требование о том, что сложность построения имени для новых объектов не зависит от количества объектов. Сложность построения нового имени определяется размерами матриц Фи и Тетта, которые фиксированы и не зависят от количества документов после обучения модели. Выполнение дополнительного свойства важно для случая, когда известна часть документа, поскольку и когда документ, часть которого известна, присутствует в коллекции и когда отсутствует, необходимо производить построение имени, так как частично известный объект является новым.

Стоит отметить, что для случая когда известна часть документа, обнаружение ближайшего по имени документа приводит к обнаружению близкого по смыслу документа только когда выполнено требование положительной корреляции разностей. Для случая, когда задаётся шаблон имени, данное требования также важно, поскольку если оно выполнено, то документам будут присвоены соответствующие их содержанию темы.

Требование положительной корреляции разностей

В случае если выделены семантические ядра тем, данное требование выполнено по построению.

Таким образом, степень соответствия предлагаемой системы именования данному требованию зависит от качества тематической модели.

4.4 Свойства

В данном разделе определяются свойства предложенной системы именования и их соответствие свойствам, определённым в обзоре.

Тип поиска

С помощью предложенной системы именования возможно осуществление поиска по четко сформулированным запросам. В этом случае Пользователю известна часть тем имени объекта. Все объекты, имена которых соответствуют полученному шаблону, являются результатами поискового запроса. Результаты поиска точно соответствуют поисковому запросу.

Также предложенная система именования может быть использована и для разведочного поиска. В этом случае Пользователю известна часть объекта. Все, объекты, имена которых похожи в заданном смысле на имя частично известного объекта, являются результатами поискового запроса. Поисковая выдача является рекомендациями.

Таким образом предложенная система именования может использоваться как для поиска по чётко сформулированным запросам, так и для разведочного поиска.

Доступная для построения имени информация

Для построения имени используется глобальная, но ограниченная информация — используется информация о всей обучающей выборки. Для новых объектов информация о других новых объектах не учитывается.

С другой стороны имя строится на основе локальной информации об объекте.

Таким образом, предложенная система именования использует для построения имени как глобальную, так и локальную информацию.

Тип имени

Предложенная система именования производит автоматическое построение машинного имени — вектора вероятностей принадлежности документа к темам.

Пользовательское имя может быть получено путем ручного аннотирования тем. В случае появления решения, автоматизирующего решение данной задачи, будет производиться единовременное построение обоих имен.

Таким образом, предложенная система именования позволяет производить последовательное построение машинного и пользовательского имени. При этом машинное имя строится автоматически, а пользовательское вручную.

Переименование при коллизии

В случае коллизии на обучающей выборке возможно как проведение обучения модели заново, так и проведение дообучения. Однако после обучения модели, для новых объектов, в случае появления коллизии, возможен только вариант обучения модели заново. Строго говоря, дообучить модель возможно, но в этом случае потребуется перестраивать имена всех ранее обработанных новых объектов.

При коллизии производится переименование не одного объекта, а всех. Поэтому вопроса о том, какой из объектов, имена которых совпадают, переименовывать, не возникает.

Таким образом, предложенная система именования позволяет производить заново построение имен объектов. Также возможно и проведение наращивания имени, однако не за счёт удлинения, а за счёт дополнительных итераций в обучении модели.

Дополнительное свойство: сложность поиска

Отдельно стоит отметить, что в предложенной системе именования длина имени определяется количеством тем, которое заметно меньше чем количество ключевых слов в документе и гораздо меньше чем размер словаря документа.

Важно отметить, что сложность поиска определяется не количеством позиций в имени, а размером словаря. Поэтому проведение поиска в пространстве тем требуется существенно меньше операций по сравнению с поиском в пространстве имён.

Таким образом, предложенная система именования позволяет эффективно организовать поисковые операции.

4.5 Реализация

Для реализации предложенной системы именования используется библиотека тематического моделирования `bigartm`[7]. Программная реализация, включающая программы для подготовки данных, обучения модели и проведения экспериментов, находится в открытом доступе[8].

5. О влиянии системы именования на построение ИОС

В данном разделе рассматривается влияние предложенной системы именования на решение задач построения ИОС.

Предложенная система именования соответствует дополнительному требованию о том, что сложность построения имени для новых объектов не зависит от числа объектов. Сложность построения имени определяется количеством операций над матрицами Фи и Тетта, которые имеют фиксированный размер. Таким образом, предложенная система именования влияет на задачу Именованье, позволяя минимизировать число операций построения имени для новых объектов.

Предложенная система именования позволяет существенно уменьшить количество поисковых операций, по сравнению с полнотекстовым поиском и поиском по аннотациям ключевыми словами. Для поиска по имени важно не число позиций в имени, а размер словаря, значения которого могут присваиваться данным позициям. Так, словарь тематик имеет существенно ограниченный размер, по сравнению со словарём документа и словарём ключевых слов документа. Поэтому поиск по нему осуществляется заметно быстрее. Особенно это должно быть заметно на больших коллекциях. Таким образом, предложенная система именования влияет на задачу Поиск, позволяя сократить число поисковых операций.

Наличие мягкой кластеризации в предложенной системе именования позволяет определить как сгруппировать объекты для хранения. Новый объект относится к той группе, тематика которой имеет наибольший вес в его имени. Хранение может быть организовано таким образом, что группы объектов свободно перемещаются по сети, в зависимости от вектора запросов Пользователей. Точнее перемещаются центры групп, это не документы, это центроиды кластеров. Таким образом, предложенная система именования влияет на задачу Хранение, позволяя определить каким образом группировать документы для хранения.

Группы объектов могут свободно перемещаться по сети, поскольку в ИОС отсутствует понятие адреса. Однако для организации подобного перемещения необходимо, что количество правил маршрутизации не зависело от числа ИО в сети, поскольку иначе таблицы правил маршрутизации со временем и увеличением количества ИО будут переполнены. Организовать данную независимость можно за счёт маршрутизации по меткам, когда каждой метке соответствует узел сети и таким образом маршрут является последовательностью меток. В этом случае сложность маршрутизации не зависит от количества имен. Сложность маршрутизации определяется количеством правил в таблице маршрутизации.

Предложенная система именования соответствует требованию положительной корреляции разностей между именами из разностей между объектами. Благодаря этому становится возможной "маршрутизация по смыслу", когда запрос поступает из любой точки сети и идёт по центроидам групп в сторону, где находится имя, наиболее близкое к запросу. Возможна аналогия с маршрутизацией по расстоянию Хэмминга. Для организации такой маршрутизации необходимо, чтобы похожие центроиды находились рядом в пространстве имен. Это условие выполнено поскольку предложенная система именования соответствует требованию положительной корреляции, а имена являются координатами.

Так возникает задача отображения пространства координат центроидов групп на физическую топологию. Центроиды находятся рядом в пространстве имен по построению, в случае если тематическое моделирование произведено успешно. Однако два соседних центроида могут быть отображены на не соседние узлы, например, если соответствующие данным узлам Пользователи запросили соответствующий данным центроидам контент. В этом случае задержка до запрошенных объектов у Пользователей уменьшится, однако количество правил маршрутизации для поддержания целостности отображения пространства центроидов на пространства узлов увеличится. Таким образом возникает задача оптимизации, в которой нужно определить баланс между задержкой до Пользователей и количеством правил маршрутизации на узлах. Важно понимать, что не всегда можно «перестроить» существующее отображение координат центроидов на пространство координат узлов, иногда нужно построить его «с нуля». Также стоит отметить, что количество правил маршрутизации минимально, когда из разницы между именами следует разница между узлами, на которые отображаются данные имена; иными словами, близкие имена отображаются на близкие узлы. Таким образом, система именования влияет на задачу Передача, поскольку "маршрутизация по смыслу" возможна только когда соблюдается требование положительной корреляции, соблюдение которого зависит от успешности выделения тематик.

Таким образом, задачи, посвящённые разработке архитектуры и базовой функциональности ИОС, являются развитием данной работы.

6. О единой системе именования и едином имени

Предложенная система именования используется для именования текстовых объектов. В то же время требования, которым она соответствует, также относятся и к системам именования других типов объектов, а именно аудио, видео и изображений.

Возникает вопрос о существовании единой для всех типов ИО системе именования, которая с одной стороны соответствует требованиям системы именования, а с другой позволяет получать унифицированные имена для всех типов ИО.

Возможен следующий подход к определению такой системы именования. Данный подход связан с приведением ИО всех типов к канонической форме, по которой затем строится имя. Данная форма называется синтетическим объектом, это пятый тип ИО. Содержательно это трёхмерная реконструкция сцены, описанной в ИО. Такая сцена может состоять из нескольких синтетических объектов, если исходный объект был составным. Однако более интересен случай, когда каждым из элементов данной сцены является не непосредственно синтетический объект, а его собирательный образ, архиобъект. Может быть проведена аналогия с центроидом в кластере. Исходные ИО могут быть подвергнуты различным преобразованиям, а именно сдвиг, поворот, отражение и масштабирование. Поэтому каноническая форма имени, архиобъект, возможно составной, должна быть инвариантна данным преобразованиям. Такой формы может не существовать. В этом случае может быть предложена функция, которая по двум архиобъектам определяет, является ли первый из них результатом преобразований второго.

Также важен вопрос о едином имени. В предлагаемой системе именования происходит последовательное построение машинного и пользовательского имён. Их одновременное построение возможно в случае автоматического именования тем. Однако единое имя, одновременно являющееся и машинным, и пользовательским, с помощью предлагаемой системы именования получено быть не может, поскольку пара <вероятность принадлежности к теме, название темы> не агрегируется. В то же время для изображений единое имя построено быть может. Примером такого имени является phash[11]. Содержательно это с одной стороны сжатое изображение, а с другой последовательность битов, хэш. Для phash выполняется требование уникальности и требование положительной корреляции, однако для выполнения требования восстановления по части необходима модификация структуры данных, например, использование мешка пикселей.

Таким образом, вопросы о существовании единой системы именования и единого имени для каждого типа ИО, включая синтетические объекты, представляют интерес для дальнейшего исследования.

7. Концепция семантической маршрутизации

Информационно-ориентированные сети(ИОС) — класс компьютерных сетей, в котором нет понятия адреса устройства, содержащего данные; только понятие имени информационного объекта(ИО). Имя можно трактовать как координаты в некотором едином векторном пространстве, таком что похожие с точки зрения смысла ИО имеют близкие в смысле заданной функции расстояния координаты. Возникает задача погружения ИО в данное пространство — данную задачу решает система именования(СИ).

В работе рассмотрен подход к представлению текстов в виде векторов вероятностей принадлежности к тематикам, называемый тематическим моделированием, которое основано на разложении матрицы вхождения слов в документы на произведение матрицы вероятностей принадлежности слов теме и матрицы вероятностей соответствия темы документу.

Существует целое семейство моделей, основанных на подходе с матричным разложением[12]. Данные подходы позволяют строить векторные представления объектов. Объекты могут быть не только текстовыми, но также графическими, аудио и видео. Таким образом, в зависимости от типа объекта, от того «на каком языке он записан», способ построения векторного представления, имени, может быть различным и в общем случае объекты различных типов образуют различные векторные пространства.

Для построения единого векторного пространства текстовых объектов необходимо решить следующие задачи:

- 1) Подготовить данные и провести их предобработку
- 2) В зависимости от выбранного текстового объекта, текста или слова, построить матрицу вхождения слов в документы или матрицу совстречаемости слов
- 3) Обучить тематическую модель и оценить её качество для решения задачи близости(word-similarity) и задачи аналогий

Важно отметить, что любую систему именования можно также воспринимать как язык, а над её словами, именами, возможно производить такие операции как сложение и вычитание, позволяющие получать другие, возможно новые, слова. Кроме того важно, что язык системы именования не зависит от языка описания объекта. Язык описания объекта зависит от типа объекта: текст, изображение, аудио, видео. Имя не зависит от языка описания объекта, поскольку имя кодирует смысл объекта, ИО, а не сам объект. Так, ИО не зависит от языка, описывающего объект, поскольку это интерпретация. ИО может быть представлен в виде синтетического объекта. Также возможно говорить о процедуре, обратной погружению — извлечению, т.е. способу по имени построить соответствующий ему синтетический объект.

Для доступа к данным предлагается использовать подход, называемый семантической маршрутизацией, организация которой состоит из двух этапов: на первом этапе векторное пространство разбивается на области, которые затем ставятся в соответствие физическим узлам, на которых, с помощью правил маршрутизации, организована связность между соседними областями. На данном этапе объекты хранятся на узлах, отвечающих за области, к которым относятся имена этих объектов. Строго говоря, когда речь идёт об имени, имеется ввиду имя информационного объекта, содержащегося в некотором объекте, а не имя самого объекта. На этом этапе имя является адресом и по нему доступен объект, при этом для

потребителя не важно, на каком именно устройстве расположены данные, однако это важно для системы. Доступ к данным на первом этапе называется статической семантической маршрутизацией.

Для организации статической семантической маршрутизации необходимо решить следующие задачи:

- 1) Получить оценку размерности единого векторного пространства
- 2) Определить функцию расстояния в пространстве
- 3) Определить способ разбиения пространства на области
- 4) Определить способ построения отображения областей на физическую топологию с сохранением связности соседних областей
- 5) Определить способ направленного перемещения между областями из заданной области в заданную область
- 6) Определить сложность семантической маршрутизации с точки зрения количества правил маршрутизации
- 7) Определить сложность семантической маршрутизации с точки зрения числа узлов, участвующих в реализации запроса

На втором этапе данные становятся независимы от устройств, их содержащих, и получают возможность перемещаться по сети, например, в места, где данные наиболее востребованы. Именно на данном этапе система становится ИОС в полном смысле этой концепции, т.е. и с точки зрения потребителя, и с точки зрения системы. При этом независимость от устройств имеется ввиду относительно системы, для потребителя доступ к данным по-прежнему прозрачен, однако время доступа к данным может сократиться, поскольку в случае перемещения данных в места, где они наиболее востребованы, в том числе проактивно, система по сути работает как кэш, тем самым уменьшая время доступа к данным. Стоит отметить, что кэшем можно называть как отдельный узел системы, где одни данные выясняют другие и поэтому время жизни данных на узле ограничено, так и всю систему в целом, так как данные при вытеснении не удаляются, а перемещаются на другой узел, то есть время жизни данных в системе не ограничено. Кроме того данные могут дублироваться, например, если они востребованы в разных концах сети. В общем случае требуется отслеживать перемещение данных таким образом, что когда узел, отвечающий за область, содержащую имя запрошенных данных, получает запрос к ним, он может перенаправить данный запрос к узлу, на котором хранятся данные. В качестве запроса используется имя ИО, возможно, известное не полностью, при этом при перенаправлении запрос может модифицироваться, например, к нему может добавляться указатель на область или узел, куда переместились данные.

Кроме того важным является вопрос об устойчивости отображения областей на узлы, производимого на первом этапе, поскольку перемещение данных между узлами создаёт издержки с точки зрения количества правил маршрутизации, необходимых для обнаружения покинувших «свою» область данных. В некоторый момент данные издержки могут стать критическими и привести к перегрузке системы, например, в случае переполнения таблиц маршрутизации — в этом случае часть запросов может быть не реализована. Для того, чтобы избежать данной ситуации, вводится процедура стабилизации, при которой производится повторное отображение областей на узлы, однако с учетом хранящихся на узлах данных. Доступ к данным на втором этапе называется динамической семантической маршрутизацией.

Для организации динамической семантической маршрутизации необходимо решить следующие задачи:

- 1) Определить процедуру проверки устойчивости и процедуру стабилизации отображений областей на узлы
- 2) Определить способ построения указателя на данные, которые «покинули свою область»
- 3) Определить способ организации запроса к данным
- 4) Определить условие «перебалансировки областей» — то есть повторного построения отображения областей на узлы
- 5) Определить способ построения отображения областей на физическую топологию с сохранением связности соседних областей, с учетом того, что некоторые области необходимо отобразить на заданные узлы

Схема организации доступа к данным с помощью семантической маршрутизации представлена на Рисунке 3.

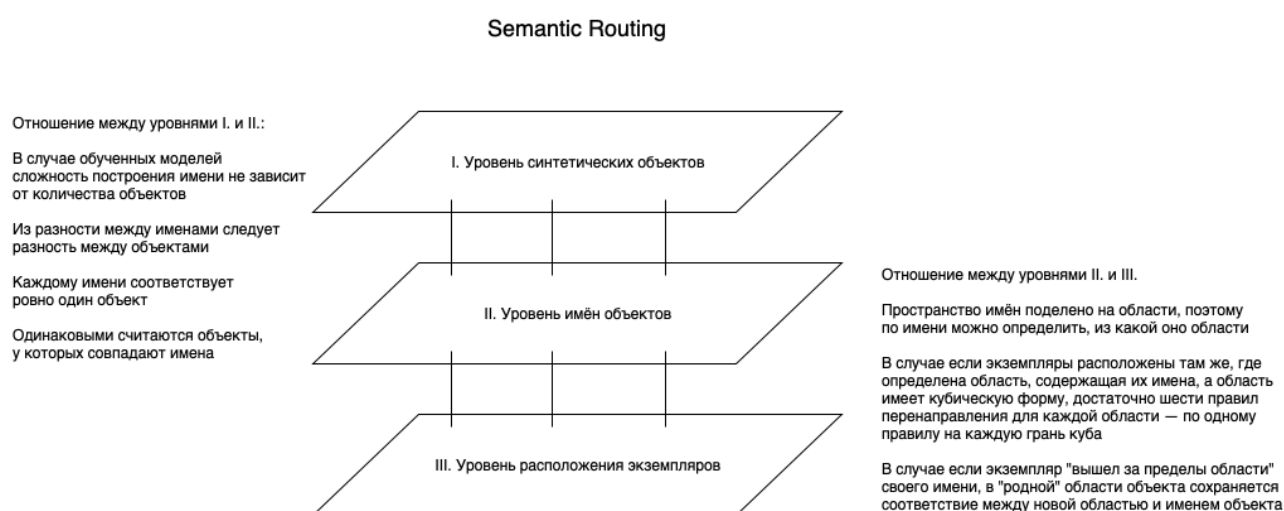


Рисунок 3. Семантическая маршрутизация.

Заключение

В работе получены следующие результаты:

- 1) Сформулированы требования к системе именования
- 2) Предложен подход к формированию имени на основе семантического анализа содержимого информационного объекта
- 3) Рассмотрен подход к организации доступа к данным с помощью семантической маршрутизации и определены задачи, которые необходимо решить для реализации данного подхода

Список литературы

- [1] Xiaofei Liao; Li Lin; Guang Tan; Hai Jin; Xiaobin Yang; Wei Zhang. Bo LiLiveRender: A Cloud Gaming System Based on Compressed Graphics Streaming. 2016.
<https://ieeexplore.ieee.org/document/7166339>
- [2] Anders Eriksson; Borje Ohlman; Karl-Ake Persson. What are the Services of an Information-centric Network, and Who Provides Them? 2012.
<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.473.5419&rep=rep1&type=pdf>
- [3] Еремеев М.А.; Янина А.О. Разведочный поиск на основе тематического моделирования. 2019.
https://lomonosov-msu.ru/archive/Lomonosov_2019/data/16103/uid341423_report.pdf
- [4] Воронцов К. В. BigARTM: от лего-конструктора тематических моделей к сервисам разведочного поиска. 2019.
<http://www.machinelearning.ru/wiki/images/0/0d/Voron-2019-05-11-bigartm.pdf>
- [5] Serg Karpovich. Тематическая модель. 2017.
https://commons.wikimedia.org/wiki/File:Тематическая_модель.png
- [6] Воронцов К. В. Обзор вероятностных тематических моделей. 2019.
<http://www.machinelearning.ru/wiki/images/d/d5/Voron17survey-artm.pdf>
- [7] Проект BigARTM. Библиотека тематического моделирования. 2019.
<http://bigartm.org>
- [8] Колосов Алексей. Система именования текстовых объектов. 2019.
<http://github.com/cheptil/nicn>
- [9] John Langford. Vowpal Wabbit. 2017.
http://hunch.net/~nyoml/vowpal_wabbit.pdf
- [10] Джоэл Грас. Data Science. Наука о данных с нуля. 2017.
Санкт-Петербург. «БХВ-Петербург». Стр. 186-190.
- [11] Zauner, Christoph: Implementation and Benchmarking of Perceptual Image Hash Functions. 2010.
https://www.phash.org/docs/pubs/thesis_zauber.pdf
- [12] Alexander Fonarev. Matrix factorization methods for training embeddings in selected machine learning problems. 2018.
<https://www.skoltech.ru/app/data/uploads/2018/09/Thesis-Fonarev-Final.pdf>