

Информационные процессы / Information Processes

Оригинальная статья / Original article

УДК 004.021

<https://doi.org/10.31432/1994-2443.2025.03>**Реляционная модель устройства узла сети связи
для телекоммуникационных систем****А. В. Синицын¹, С. Э. Лукьянов², Р. Ю. Байдушин¹ ✉, П. А. Цветков¹**

¹Федеральное государственное бюджетное образовательное учреждение
высшего образования «МИРЭА — Российский технологический университет»,
проспект Вернадского, 78, г. Москва, 119454, Российская Федерация

²Федеральное государственное автономное научное учреждение «Центр информационных
технологий и систем органов исполнительной власти имени А. В. Старовойтова»
(ФГАНУ ЦИТиС), Пресненский Вал, 19, стр. 1, г. Москва, 123557, Российская Федерация

✉ roman.baydushin@gmail.com

Аннотация. *Актуальность.* Современное развитие телекоммуникационных сетей связи требует технологий построения аналитической системы эксплуатационной поддержки сети с использованием инвентарной, технологической и аналитической информации по узлам связи для проектирования узлов и линий связи, маршрутов и других телекоммуникационных услуг. Планирование и оптимизация структуры сети связи приобретает большое значение для оптимизации нагрузок на сеть при изменении трафика данных, количества пользователей, требований к доступности сервисов и качества предоставляемых услуг связи. *Цель.* Моделирование устройства узла сети связи для прототипа системы поддержки сети, представления его в виде ER-диаграммы. *Материалы и методы.* Использовалось отечественное оборудование DWDM/SDH производства компании «Т8» и ОАО «Супертел» (Россия). *Результаты.* Разработана достаточно универсальная реляционная модель устройства узла связи. *Выводы.* Модель применима к любому используемому оборудованию любой конфигурации для предоставления телекоммуникационных услуг.

Ключевые слова: коммуникационные сети, OSS, SQL, реляционная модель, узел сети связи, телекоммуникационные сети, ER-диаграмма, PostgreSQL, база данных, отказоустойчивость, проектирование системы

Финансирование. Финансирование отсутствовало.

Благодарности. Авторы выражают благодарность за постановку задачи, предоставленные материалы, содержательное обсуждение и консультации Перепеловой Е. В., заместителю начальника НТЦ ФГАНУ ЦИТиС им. А. В. Старовойтова.

Для цитирования: Синицын А.В., Лукьянов С.Э., Байдушин Р.Ю., Цветков П.А. Реляционная модель устройства узла сети связи для телекоммуникационных систем. *Информация и инновации.* 2025;20(1):105-124. <https://doi.org/10.31432/1994-2443.2025.03>

Relational model of a communication network node device for telecommunication systems

Alexander V. Sinitsyn¹, Stanislav E. Lukyanov², Roman Y. Baydushin¹ ✉, Pavel A. Tsvetkov¹

¹Federal State Budgetary Educational Institution of Higher Education "MIREA — Russian Technological University", 78, Vernadsky Avenue, Moscow, 119454, Russian Federation

²Federal State Autonomous Scientific Institution "Center for Information Technologies and Systems of Executive Authorities named after A. V. Starovoytov",

Presnensky Val, 19, building 1, Moscow, 123557, Russian Federation

✉ roman.baydushin@gmail.com

Abstract. *Relevance.* Modern development of telecommunication networks requires technologies for constructing an analytical system of operational support of the network using inventory, technological and analytical information on communication nodes for designing nodes and communication lines, routes and other telecommunication services. Planning and optimization of the communication network structure is of great importance for optimizing network loads when changing data traffic, the number of users, requirements for service availability and the quality of communication services provided. *Purpose.* Modeling of the device of a communication network node for a prototype of a network support system, its representation in the form of an ER diagram. *Materials and methods.* Domestic DWDM/SDH equipment manufactured by T8 and Supertel OJSC (Russia) was used. *Results.* A fairly universal relational model of a communication node device was developed. *Conclusions.* The model is applicable to any equipment of any configuration used to provide telecommunication services.

Keywords: communication networks, OSS, SQL, relational model, communication network node, telecommunication networks, ER diagram, PostgreSQL, data base, fault tolerance, system design

Financing. No funding.

Acknowledgments. The authors would like to thank E. V. Perepelova, Deputy Head of the Scientific Research Center, Federal State Autonomous Research Institution «Starovoytov Center of Information Technologies and Systems for Executive Power Authorities» for the formulation of the problem, materials, informative discussion and consultations.

For citation: Sinitsyn A.V., Lukyanov S.E., Baydushin R.Y., Tsvetkov P.A. Relational model of a communication network node device for telecommunication systems. *Information and Innovations*. 2025;20(1):105-124. (In Russ.). <https://doi.org/10.31432/1994-2443.2025.03>

ВВЕДЕНИЕ

Современное развитие сетей связи [1] характеризуется увеличением трафика данных, количества пользователей и постоянным повышением требований к доступности сервисов и качества предоставляемых услуг связи. Поэтому построение систем эксплуатационной поддержки сети (OSS — Operation Support System) с использованием инвентарной, технологической и аналитической информации по узлам связи для проектирования узлов и линий связи, маршрутов для предоставления телекоммуникационных услуг имеет ключевое значение, а планирование и оптимизация структуры сети связи приобретает большое значение.

Планирование и оптимизация позволяют заранее предотвращать простои в предоставлении услуг связи, а также более эффективно использовать ресурсы в условиях их дефицита, что позволяет сократить затраты на оборудование и последующее обслуживание. Инструменты планирования повышают надежность сети путем выявления ее уязвимостей и точек отказа.

Статья касается части большой задачи по разработке инструментов операционной поддержки сети и решения задач по моделированию сети и правил ее работы.

В статье рассматривается универсальное представление устройства узла сети связи с помощью реляционной модели базы данных, и использование ее на прототипах сети связи. В качестве оборудования для примера построения узла связи использовалось отечественное оборудование DWDM/SDH производства компании «Т8» и ОАО «Супертел» (Россия), также можно рассматривать оборудование компаний Infinera (США), ADVA Optical Networking (Германия), Huawei (Китай) и др.

Прототип рассматриваемого участка транспортной сети с соответствующими связями внутри узла взят как пример для демонстрации принципов работы созданной модели устройства узла связи прототипа сети связи.

Моделирование устройства узла связи необходимо для применения при планировании телекоммуникационных сетей в смысле OSS, что подчеркивается рекомендациями Международного телекоммуникационного союза «Надежность планирования телекоммуникационных сетей». Несмотря на наличие достаточного числа работ по исследованию оценки надежности топологий проводных сетей, задача построения «отказоустойчивых» сетей изучена не полностью. Актуальные и полезные в этом направлении работы [2, 3].

Интересные математические модели, имеющие отношение к теме статьи, были представлена в [4], функциональная модель структуры узла описана в работе [5].

Построение реальной магистральной и региональной сети связи с использованием технологий OTN/DWDM описано в [6].

В статье рассматривается новая удобная отказоустойчивая [7] и универсальная реляционная модель устройства узла связи, позволяющая создавать узлы связи на оборудовании любого вендора и произвольной конфигурации. Модель может применяться для решения таких задач OSS, как:

- поиск возможных маршрутов между заданными узлами для заданной скорости с минимизацией расстояния, стоимости и пр.;
- поиск вариантов равномерной загрузки сети с обнаружением точек отказа;
- выделение кластеров сети для увеличения трафика между ними;

— проверка максимальной пропускной способности для заданного направления.

1. МАТЕРИАЛЫ И МЕТОДЫ

1.1. Основная идея

Сеть связи представляем планарным ориентированным графом с весами, где i -я вершина — это i -й узел связи, а ij -ребро — канал связи между i -м и j -м узлами. Вес ребра — это пропускная способность канала связи между узлами.

Любой узел связи представлен оборудованием, состоящим из корзин, в которые вставляются платы с портами. Корзины могут быть нескольких уровней и разной пропускной способности, где каждый порт верхнего уровня может агрегировать несколько портов нижнего уровня. Соединения между уровнями происходит простым соединением портов.

Основной идеей построения создаваемой модели является принцип “матрешки”. Рассматриваются следующие сущности: допустимые “порты”= $\{port1 \dots portm\}$, допустимые “платы”= $\{board1 \dots boardn\}$ и допустимые “корзины”= $\{basket1 \dots basketk\}$, а также считаем, что “порты” из представленной ниже табл. 1 “вставляются” в “платы”, представленные табл. 2, которые, в свою очередь, “вставляются” в “корзины” из табл. 3. Такое представление соответствует реалиям устройства “плат”, оснащенного модулями и портами SFP, и реалиям устройства “корзин”, оборудованных слотами для установки “плат”.

Каждый допустимый “порт” $port_i$, $i=1, m$ характеризуются своей пропускной способностью $capacity_i$.

Каждая плата $board_i$, $i=1, n$ имеет некоторое количество портов $Nports$, а также список “разрешенных” портов — множество $\{port_i\}$, состоящее из списка допу-

стимых портов, соответствующих каждой плате.

Каждая “корзина” $basket_i$ имеет некоторое количество слотов $Nslots$, в которые могут устанавливаться только совместимые платы — множество $\{boards_i\}$ из списка допустимых “плат”. Каждая “корзина” может быть отнесена к определенному уровню L_i на каждом узле связи. При этом “нижние” корзины относятся только к своему узлу, а “корзины” самого верхнего уровня определяют ребро между некоторыми двумя узлами связи. Если из узла связи выходят несколько ребер, то и корзин верхнего уровня будет несколько для каждого ребра.

Каждый порт может быть “клиентским” или “линейным”. Если клиентский порт принадлежит клиенту, то линейный порт, как правило, находящийся на уровне выше и имеющий более высокую пропускную способность, агрегирует порты нижних уровней.

Таким образом, каждый порт самого верхнего уровня на узле, определяющий ребро (канал связи между узлами связи) может иметь уникальный идентификатор типа SSSS.TTTT.BB.DD.PP. Здесь SSSS — идентификатор исходного узла (Source), например 0001, что соответствует Москве; TTTT — идентификатор узла, до которого есть ребро или канал связи (Target), например 0123, что соответствует соседнему городу Тула; BB — номер корзины на узле SSSS, который может включать и уровень, например 0203 — третья корзина второго уровня; DD — номер слота платы, например 08 — восьмой слот; PP — номер порта в плате, например 04 — четвертый порт.

В рамках объектной модели рассмотрен способ проектирования через вспомогательные поля. Каждая таблица из списка порты, платы, корзины имеют

поля, которые содержат ссылки на связанный объект. Это ограничение задает порядок создания записей в объектной модели. Сначала создаются записи в справочниках, потом, начиная с таблицы node (узел), идет детализация до портов. Просто создать запись в таблице портов без установленной платы нельзя, так как порт обязательно должен быть связан с платой. Такая же логика распространяется на связь плат с корзинами.

1.2. Ограничения предметной области

Перечислим основные ограничения предметной области, которые учитывались при проектировании реляционной модели.

1.2.1. Значения базовой единицы E1

Для рассматриваемых транспортных сетей SDH все поддерживаемые порты

можно представить в виде зависимости от скорости $E_1 = 2.048$ Мбит/с, как представлено в табл. 1. Учитывая наличие заголовков и указателей, обеспечивающих работу транспортных сетей, дополнительно для расчета пришлось использовать математически выведенную универсальную единицу $E_2 = 2.4685$ Мбит/с. Таким образом, при высоких скоростях значение базовой единицы E_1 увеличивается до E_2 для сохранения соответствия пропускной способности (capacity) порта стандартам SDH ITU-T G.707. Количество типов портов GE и число n для портов FE в табл. 1 можно увеличивать в зависимости от возможностей инкапсуляции GE/FE в формат SDH и выбранного производителя оборудования. Этот случай детализировался на примере оборудования SDH ОСМ-КМ и П-317 производства ОАО «Супертел».

Таблица 1. Пример списка доступных портов
Table 1. Example of a list of available ports

id, $i=1, m$	Название port _i	Пропускная способность capacity _i
0	E1	E_1
$n=1-42$	FE _n	$n \cdot E_1$
43	STM1	$63 \cdot E_2$
44	GE1	$63 \cdot E_2$
45	GE2	$126 \cdot E_2$
46	GE3	$189 \cdot E_2$
47	STM4	$252 \cdot E_2$
48	GE4	$252 \cdot E_2$
49	GE5	$315 \cdot E_2$
50	GE6	$378 \cdot E_2$
51	GE7	$441 \cdot E_2$
52	STM16	$1008 \cdot E_2$
53	STM64	$4032 \cdot E_2$
54	Lambda	Согласно соединенному порту

1.2.2. Правила соединений портов с платами и плат с корзинами

На основании описания оборудования SDH OCM-KM и П-317 производства ОАО «Супертел» и DWDM «Волга» производства компании «Т8» в качестве при-

мера взят перечень плат и сформирован список разрешенных для них портов в табл. 2.

В табл. 3 приведены корзины с максимальным количеством слотов в них, а также разрешенные для этих корзин платы.

Таблица 2. Пример списка доступных плат

Table 2. Example of a list of available boards

id, i=1, n	Название board _i	Количество портов N _{ports}	Разрешенные порты {port _i }
0	блок 21E1	21	E1
1	блок Eth10/100	4	FE1-FE42
2	блок CTM-16	1	STM16
3	блок CTM-16M	4	STM16
4	блок Eth1000M	2	GE1-GE7
5	блок CTM-1/4M	8	STM1, STM4
6	блок CTM-64	4	STM64
7	транспондер MD-D3ES	4	STM1, GE1, STM4, STM16
8	транспондер TT-10EP	10	STM64
9	оптоволокно		

Таблица 3. Пример списка доступных корзин

Table 3. Example of a list of available baskets

id, i=1, k	Название basket _i	Количество слотов Slot _{basket_i}	Уровень L _{basket_i}	Разрешенные платы {boards} _{basket_i}
0	SDH П-317	13 (часть зарезервирована)	1	21E1, CTM-16, Eth10/100
1	SDH OCM-KM	10	2	CTM-1/4M, CTM-16M, Eth1000M, CTM-64
2	DWDM	10	3	TT-10EP, MD-D3ES
3	Оптоволокно			

1.2.3. Многоуровневая модель устройства узла связи

В настоящей объектной модели количество уровней равно n, но в качестве примера взят участок сети, состоящий из трех уровней, представленный в таблице 3.

Каждая корзина нижнего уровня соединяется с корзиной верхнего уровня путем соединения клиентского порта в корзине нижнего уровня с линейным портом корзины верхнего уровня. Процесс одинаково повторяется от уровня к уровню.

При этом соединение клиентских портов и линейных на одном уровне происходит внутри корзины по задней панели (backplane).

1.2.4. Понятие “лямбда оптическая”

В нашей модели низкоскоростные порты нижних уровней агрегируются в высокоскоростные порты верхних уровней, а самый верхний уровень представлен “оптической лямбдой”, которая рассматривается как порт самой высокой пропускной способности. Вообще говоря, физически “оптической лямбдой” называют определенную длину волны света, используемую для передачи данных по оптоволоконным линиям связи с применением технологий спектрального уплотнения DWDM/CWDM. В представленной модели каждой «оптической лямбде» соответствует линейный порт, размещенный на транспондерах оборудования DWDM.

В объектной модели предусмотрено ограничение на количество возможных лямбд. В качестве примера количество оптических лямбд принято равным 40, хотя оно может достигать до 160 в зависимости от применяемого оборудования и используемой системы передачи DWDM. Также учтено, что в одном направлении для соединения двух ближайших узлов может быть только 40 лямбд, то есть одна корзина третьего уровня соответствует одному ребру с другой корзиной третьего уровня другого узла связи.

2. РЕЗУЛЬТАТЫ

Приведенная в настоящей статье схема объектной модели (рис. 1) может использоваться как для существующих, так и для проектируемых сетей связи, а также для создания отдельных систем, эмулирующих работу сети связи. В качестве системы управления базами данных для тести-

рования объектной модели был выбран PostgreSQL [8, 9]. Более подробно правила проектирования базы данных описаны в работах [10, 11], полезные материалы также можно найти в работах [12, 13].

Рассмотрим спроектированную схему подробнее. В общей сложности схема состоит из 14 таблиц, связанных между собой. Это именно тот минимальный набор таблиц, из которого должна состоять объектная модель для представления узла сети связи.

Глобально таблицы можно разделить на два типа: (1) объекты, хранящие в себе справочную информацию и (2) объекты, содержащие установленное оборудование на конкретном узле связи.

Таблицы-справочники содержат полную информацию для построения маршрута. Они обеспечивают гибкость схемы, так как любая новая модель может быть добавлена в справочник и потом использована вместо того, чтобы каждый раз вносить одну и ту же информацию вручную. Также хранение справочной информации в отдельной таблице позволяет динамически обновлять все фактические объекты, которые были созданы на основе этой модели. Допустим, у модели платы № 1 изменилась цена и теперь для того, чтобы эта цена применилась на все установленные платы такой модели необходимо вручную с помощью SQL-запросов обновить информацию и пересчитать все маршруты. В нашем случае никаких дополнительных обращений в базу данных не потребуется и достаточно будет пересчитать стоимость для всех существующих маршрутов, а новые будут формироваться исходя из новой цены. На схеме таблицами-справочниками являются: `port_types`, `board_allowed_port_lists`, `board_models`, `basket_allowed_board_lists`, `basket_models`.

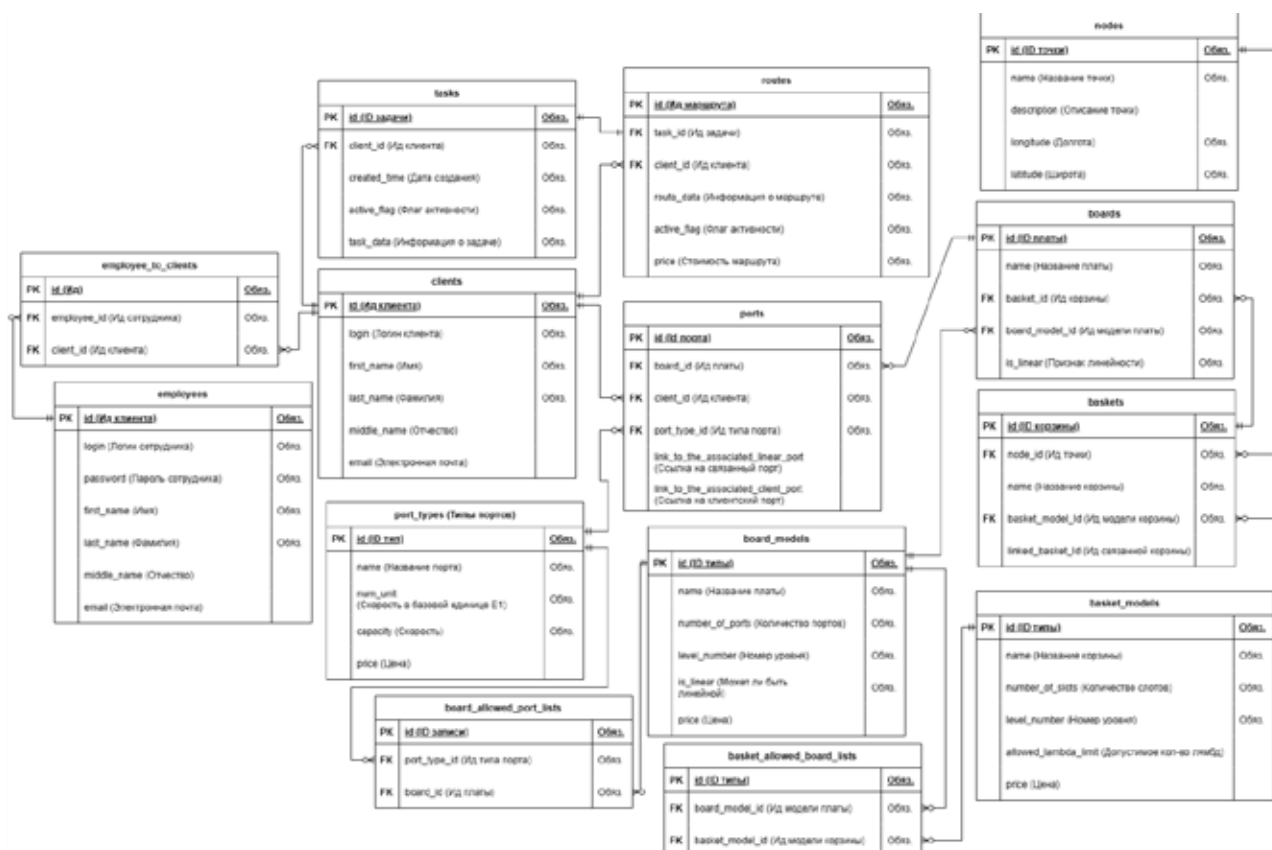


Рис. 1. Представление устройства узла сети связи с помощью объектной модели
Fig. 1. Representation of communication network node using object model

Таблицы, в которых хранится информация о “фактически” существующих объектах, позволяют создавать и записывать данные об оборудовании, которое было установлено/обновлено в результате построения маршрута для того, чтобы при последующем расчете маршрута система могла видеть уже установленные объекты для конкретной точки и, соответственно, учитывать это при принятии решения о необходимости добавления нового объекта. Если для клиента 1 в конкретной точке была установлена плата, которая состоит из двух портов, то при построении еще одного маршрута система будет учитывать, что есть еще один доступный порт, в который можно добавить клиента 2. С помощью этих таблиц реализуется связь между разными объектами, такими

как порты/платы/корзины/точки и клиенты.

2.1. Таблица узлов (nodes)

Таблица узлов (табл. 4) — основополагающая таблица, которая хранит в себе узлы, имеющиеся в системе и использующиеся для построения маршрутов. Именно с создания записей в этой таблице должна начинаться работа приложения, потому что узлы — самый верхнеуровневый объект, без наличия которого не может быть построен ни один маршрут.

2.2. Таблица моделей корзин (basket_models)

Таблица моделей корзин (табл. 5) хранит в себе все доступные модели корзин. На узлы могут быть добавлены корзины,

Таблица 4. Пример заполнения таблицы узлов
Table 4. Example of filled table of nodes

Ключи	Nodes	Обязательность заполнения	Пример
PK	id	Обязательно	5771e82a-e144-490f-acbe-386100d57cfd
	name	Обязательно	Москва
	description		Любое описание для точки
	longitude	Обязательно	37.620405
	latitude	Обязательно	55.7540471

Таблица 5. Пример заполнения таблицы моделей корзин
Table 5. Example of filled table of basket models

Ключи	Nodes	Обязательность заполнения	Пример
PK	id	Обязательно	efda03a4-e43b-4dcf-a628-ef35ebe6735b
	name	Обязательно	sdh_p_317
	number_of_slots	Обязательно	10
	level_number	Обязательно	1
	allowed_lambda_limit		null
	price		null

модель которых представлена в этой таблице. Поле `number_of_slots` позволяет явно указать количество слотов, которые можно подключить к корзине. Благодаря `level_number`, система может явно понимать к какому уровню относится корзина. Атрибут `allowed_lambda_limit` позволяет явно задать количество доступных лямбд на корзине. Это позволит избежать превышения допустимого лимита.

2.3. Таблица корзин (baskets)

Таблица корзин (табл. 6) хранит в себе фактически установленные корзины. Для обеспечения возможности получения справочной информации по моде-

лям корзин в таблице создан внешний ключ `basket_model_id`, который ссылается на `id` модели в справочной таблице `basket_models` (модели корзин). Также имеется внешний ключ `node_id`, который необходим для связи корзины с узлом, на которой корзина установлена. Атрибут `linked_basket_id` позволяет явно указать ид связанной корзины на другом узле. Это сделано для возможности отслеживания пути клиента.

2.4. Таблица моделей плат (board_models)

Таблица моделей плат (табл. 7) хранит в себе все доступные модели плат, которые

Таблица 6. Пример заполнения таблицы корзин
Table 6. Example of filled table of baskets

Ключи	Nodes	Обязательность заполнения	Пример
PK	id	Обязательно	9e24e35b-7ebe-4f1c-9f71-6c77a0d12382
FK	node_id	Обязательно	5771e82a-e144-490f-acbe-386100d57cfd
	name	Обязательно	sdh_p_317
FK	basket_model_id	Обязательно	efda03a4-e43b-4dcf-a628-ef35ebe6735b
	linked_basket_id		null

Таблица 7. Пример заполнения моделей платы
Table 7. Example of filled table of board models

Ключи	Nodes	Обязательность заполнения	Пример
PK	id	Обязательно	847297d0-1bd0-4070-a480-3f12a8ddfb3d
	name	Обязательно	block_21_e1
	number_of_ports	Обязательно	21
	level_number	Обязательно	1
	is_linear	Обязательно	false
	price		null

могут быть установлены в корзины. Поле number_of_ports позволяет явно указать количество портов, которые можно подключить к плате. Благодаря level_number, система может явно понимать к какому уровню относится плата. Атрибут is_linear нужен для указания возможности использования платы в качестве линейной, т. е. для передачи клиентов на уровень выше.

2.5. Таблица плат (boards)

Таблица плат (табл. 8) хранит в себе фактически установленные платы. Для обеспечения возможности получения справочной информации по моделям плат

в таблице создан внешний ключ board_model_id, который ссылается на id модели в справочной таблице board_models (модели плат). Также имеется внешний ключ basket_id, который необходим для связи платы с корзины, на которой эта плата установлена. Атрибут is_linear указывает функциональность установленной платы (используется ли она как клиентская или как линейная).

2.6. Таблица разрешенных плат в корзине (basket_allowed_board_lists)

Таблица разрешенных плат в корзине (табл. 9) необходима для хранения связей

Таблица 8. Пример заполнения таблицы плат
Table 8. Example of filled table of boards

Ключи	Nodes	Обязательность заполнения	Пример
PK	id	Обязательно	e9967e3e-2d07-4b39-abd9-27c218add6eb
	name	Обязательно	block_21_e1
FK	basket_id	Обязательно	9e24e35b-7ebe-4f1c-9f71-6c77a0d12382
FK	board_model_id	Обязательно	847297d0-1bd0-4070-a480-3f12a8ddfb3d
	is_linear	Обязательно	false

Таблица 9. Пример заполнения разрешенных плат в корзине
Table 9. Example of filled table of allowed boards in basket

Ключи	Nodes	Обязательность заполнения	Пример
PK	id	Обязательно	5128d5a7-e39c-40b4-9daa-9a7bc4e35a38
FK	board_model_id	Обязательно	847297d0-1bd0-4070-a480-3f12a8ddfb3d
FK	basket_model_id	Обязательно	efda03a4-e43b-4dcf-a628-ef35ebe6735b

между моделями корзин и допустимых плат, которые можно установить в корзину. Комбинации id платы и корзин в этой таблице должны быть уникальными. Двух полностью одинаковых записей не будет.

2.7. Таблица типов портов (port_types)

Таблица типов портов (табл. 10) необходима для хранения связей между моделями корзин и допустимых плат, которые можно установить в корзину. Поле saracity позволяет указать скорость, которую может обеспечивать определенный тип порта. В дальнейшем это понадобится для определения нужного порта в зависимости от скорости, которая была указана клиентом.

2.8. Таблица портов (ports)

Таблица портов (табл. 11) хранит в себе фактически подключенные порты. Для обеспечения возможности получения справочной информации по типам портов в таблице создан внешний ключ port_type_id, который ссылается на id типа порта в справочной таблице port_types (Типы портов). Также имеется внешний ключ board_id, который необходим для связи порта с платой, на которой этот порт располагается. Для связи порта с клиентом используется внешний ключ client_id, который ссылается на таблицу clients (Клиенты). Для связи клиентского порта с линейным портом используется атрибут link_to_the_associated_linear_port.

Таблица 10. Пример заполнения таблицы типов портов
Table 10. Example of filled table of port types

Ключи	Nodes	Обязательность заполнения	Пример
PK	id	Обязательно	620a7bbe-5ddf-4435-9461-6d64444a621f
	name	Обязательно	e1
	num_unit	Обязательно	1
	capacity		2.048
	price		null

Таблица 11. Пример заполнения таблицы портов
Table 11. Example of filled table of ports

Ключи	Nodes	Обязательность заполнения	Пример
PK	id	Обязательно	c36b58e5-22e2-43f9-88a9-86c7cfa43b42
FK	board_id	Обязательно	e9967e3e-2d07-4b39-abd9-27c218add6eb
FK	client_id	Обязательно	dfb81f7e-2819-404a-8ee8-201ac6472c0d
FK	port_type_id	Обязательно	620a7bbe-5ddf-4435-9461-6d64444a621f
	link_to_the_associated_linear_port		null
	link_to_the_associated_client_port		null

Для связи линейного порта с клиентским портом используется атрибут link_to_the_associated_client_port.

2.9. Таблица разрешенных портов для плат (board_allowed_port_lists)

Таблица разрешенных портов для плат (табл. 12) необходима для хранения связей между типами портов и допустимых плат, в которые можно установить порт.

2.10. Таблица клиентов (clients)

Таблица клиентов (табл. 13) хранит в себе информацию о фактических клиентах.

2.11. Таблица связи между клиентами и сотрудниками (employee_to_clients)

Таблица связи между клиентами и сотрудниками (табл. 14) хранит в себе информацию о том, какой сотрудник относится к каким клиентам. Переходная таблица, которая была создана для того, чтобы обе-

Таблица 12. Пример заполнения таблицы разрешенных портов для плат
Table 12. Example of filled table of allowed ports in board

Ключи	Nodes	Обязательность заполнения	Пример
PK	id	Обязательно	01611c81-e348-4f45-8d92-faa50056928d
FK	port_type_id	Обязательно	620a7bbe-5ddf-4435-9461-6d64444a621f
FK	board_id	Обязательно	847297d0-1bd0-4070-a480-3f12a8ddfb3d

Таблица 13. Пример заполнения таблицы клиентов
Table 13. Example of filled table of clients

Ключи	Nodes	Обязательность заполнения	Пример
PK	id	Обязательно	dfb81f7e-2819-404a-8ee8-201ac6472c0d
	login	Обязательно	test
	first_name	Обязательно	Роман
	last_name	Обязательно	Цветков
	middle_name		
	email		

Таблица 14. Пример заполнения таблицы связей между клиентами и сотрудниками
Table 14. Example of filled connection table between clients and employees

Ключи	Nodes	Обязательность заполнения	Пример
PK	id	Обязательно	dfb81f7e-2819-404a-8ee8-201ac6472c0d
FK	employee_id	Обязательно	test
FK	client_id	Обязательно	Роман

спечить возможность работы с клиентом несколькими сотрудниками или одного сотрудника для нескольких клиентов.

2.12. Таблица сотрудников (employees)

Таблица сотрудников (табл. 15) хранит в себе информацию о сотрудниках системы.

2.13. Таблица задач для построения маршрутов (tasks)

Таблица задач для построения маршрутов (табл. 16) хранит в себе задачи на построение маршрутов. Атрибут active_flag необходим для того, чтобы явно пометить какие задачи активные, а какие нет. Атри-

Таблица 15. Пример заполнения таблицы сотрудников
Table 15. Example of filled table of employees

Ключи	Nodes	Обязательность заполнения	Пример
PK	id	Обязательно	dfb81f7e-2819-404a-8ee8-201ac6472c0d
	login	Обязательно	test
	password	Обязательно	Роман
	first_name		
	last_name		
	middle_name		
	email		

Таблица 16. Пример заполнения таблицы задач для построения маршрутов
Table 16. Example of filled tasks' table for building routes

Ключи	Nodes	Обязательность заполнения	Пример
PK	id	Обязательно	dfb81f7e-2819-404a-8ee8-201ac6472c0d
FK	client_id	Обязательно	test
	created_time	Обязательно	
	active_flag	Обязательно	true
	task_data	Обязательно	Информация, необходимая для построения маршрута: • точка начала маршрута • точка конца маршрута • желаемая скорость

бут task_data состоит из JSON запроса на построение маршрута.

2.14. Таблица задач для хранения построенных маршрутов (routes)

Таблица задач для хранения построенных маршрутов (табл. 17) хранит в себе выбранные маршруты клиента. Для обеспечения связи с задачей, в рамках которой этот маршрут был создан, есть атри-

бут task_id. Атрибут active_flag необходим для того, чтобы явно пометать какие маршруты активные, а какие нет. Атрибут route_data состоит из JSON сформированного маршрута.

2.15. Пример построения маршрута для сети связи

На рис. 2 представлена схема соединения узла А с узлом Б.

Таблица 17. Пример заполнения таблицы задач для хранения построенных маршрутов

Table 17. Example of filled tasks' table for storing built routes

Ключи	Nodes	Обязательность заполнения	Пример
PK	id	Обязательно	dfb81f7e-2819-404a-8ee8-201ac6472c0d
FK	task_id	Обязательно	7de38d32-2350-45d6-8d08-bcc1197ddec1
FK	client_id	Обязательно	6e7db310-e984-4507-8b60-24850d3c00eb
	route_data		Информация построенного маршрута
	active_flag		true
	price		200

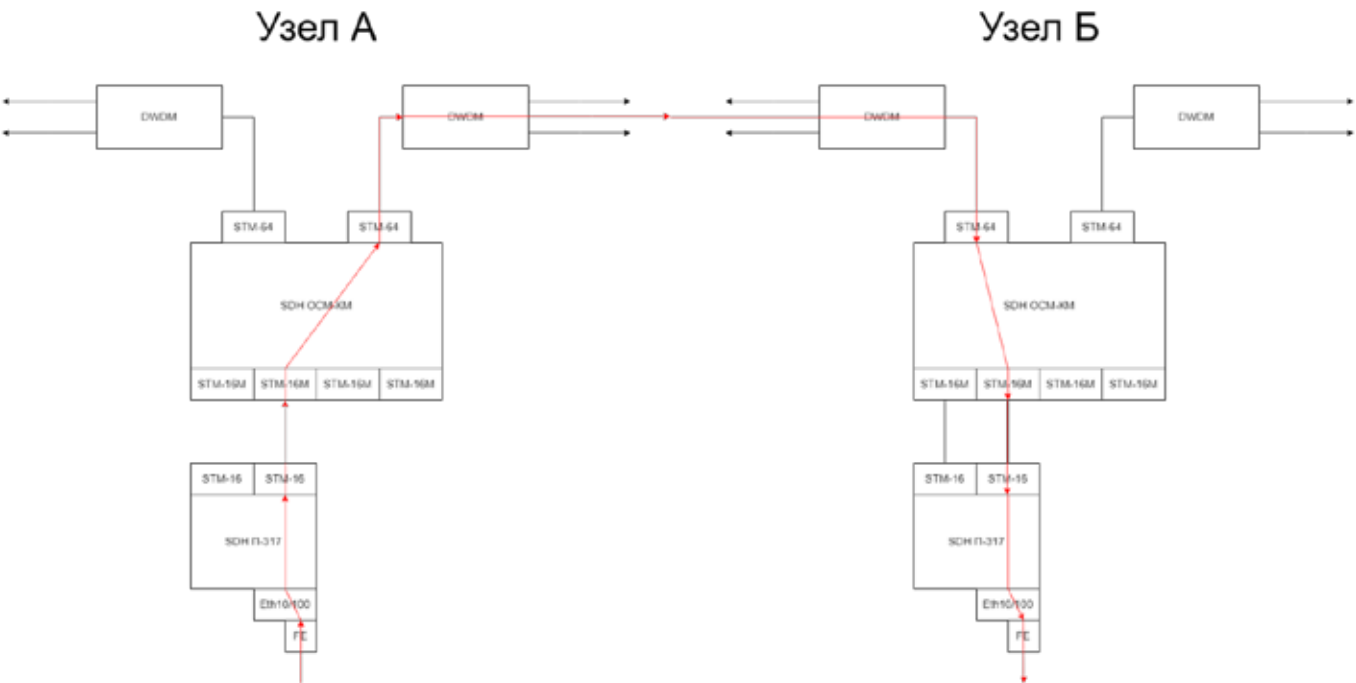


Рис. 2. Пример построения маршрута
Fig. 2. Example of how route is built

Сначала порт FE вставляется в слот платы Eth10/100. Далее этот клиентский порт соединяется с линейным портом платы STM-16. Затем линейный порт соединяется с клиентским портом платы STM-16M,

которая стоит в корзине второго уровня. Затем этот порт также соединяется с линейным портом платы STM-64, которая далее соединяется с клиентским портом транспондера TT-10EP. Этот порт соеди-

няется с линейным портом транспондера и уходит в лямбду (оптоволокно) на транспондер на пункт Б. Далее, аналогично соединениям на пункте А, на пункте Б происходит обратный процесс.

При построении маршрута база данных будет выглядеть следующим образом (рис. 3–9).

	login character varying (50)	first_name character varying (50)	last_name character varying (50)	middle_name character varying (50)	email character varying (50)	id [PK] uuid
1	test	Иван	Иванов	Иванович	test@mail.ru	a4bfe322-7ac0-40b0-aa97-59d65d3aed90

Рис. 3. Записи в таблице клиентов
Fig. 3. Records in the client's table

	id [PK] uuid	name character varying (50)	description character varying (50)	longitude double precision	latitude double precision
1	2ee4d18a-ebcd-4de8-b558-f9f845900...	Узел А	Узел А из примера	39.190289	44.878414
2	8f7f3b73-60f7-43ff-b2e6-24c0aa1f14e6	Узел Б	Узел Б из примера	82.7793606	52.4922513

Рис. 4. Записи в таблице узлов
Fig. 4. Records in the nodes table

	id [PK] uuid	node_id uuid	name character varying (50)	basket_model_id uuid	linked_basket_id uuid
1	bdb82535-9de4-4788-9e21-023e1754b2...	2ee4d18a-ebcd-4de8-b558-f9f845900568	sdh_p_317	efda03a4-e43b-4dcf-a628-ef35ebe6735b	[null]
2	7929fac6-337c-441c-bbe0-b589abc544...	2ee4d18a-ebcd-4de8-b558-f9f845900568	sdh_osm_km	621426e1-ca47-4e97-8483-f1fe3087702b	[null]
3	87fd7b20-1142b7-a02e-92d6dd074957...	8f7f3b73-60f7-43ff-b2e6-24c0aa1f14e6	sdh_p_317	efda03a4-e43b-4dcf-a628-ef35ebe6735b	[null]
4	95912188-c65c-4262-8e6f-a5f680327c18...	8f7f3b73-60f7-43ff-b2e6-24c0aa1f14e6	sdh_osm_km	621426e1-ca47-4e97-8483-f1fe3087702b	[null]
5	39301c88-e756-4fa3-977a-604037edd579...	8f7f3b73-60f7-43ff-b2e6-24c0aa1f14e6	dwdm_voiga	6ac99116-3a1c-4652-a046-0e214b61e4b5	950dc9b4-55a3-4be0-9f36-f4fb5a682853
6	950dc9b4-55a3-4be0-9f36-f4fb5a682853	2ee4d18a-ebcd-4de8-b558-f9f845900568	dwdm_voiga	6ac99116-3a1c-4652-a046-0e214b61e4b5	39301c88-e756-4fa3-977a-604037edd579

Рис. 5. Записи в таблице корзин
Fig. 5. Records in the baskets table

	id [PK] uuid	name character varying (50)	basket_id uuid	board_model_id uuid	is_linear boolean
1	f469eae8-fe9e-4ba2-a9ce-1c15d0810b3f	block_eth10_100	bdb82535-9de4-4788-9e21-023e1754b2...	ff46b2dd-c799-45fb-b771-f580798137...	false
2	c195d761-040b-481a-866a-72d368178d...	block_stm16m	7929fac6-337c-441c-bbe0-b589abc544d3	72fa13f5-0c6f-40bb-8290-3a0e81db9f0b	false
3	9d90e127-cfee-441c-aefe-598c8bf3fea6	block_eth10_100	87fd7b20-1142b7-a02e-92d6dd074957	ff46b2dd-c799-45fb-b771-f580798137...	false
4	02447c79-6145-4af7-8acb-70988b2a58...	block_stm16m	95912188-c65c-4262-8e6f-a5f680327c18	72fa13f5-0c6f-40bb-8290-3a0e81db9f0b	false
5	f0206445-edbe-417b-a6b1-645fc85b995f	block_stm16	bdb82535-9de4-4788-9e21-023e1754b2...	7e28abb7-7f22-44fd-a549-ea324d54ab...	true
6	013e711c-2b70-4d02-a41b-9b9fb131d9d1	block_stm_64	7929fac6-337c-441c-bbe0-b589abc544d3	7cd8d795-1cdc-4f70-b64b-73c1c0d7c...	true
7	e0c5ee28-c57c-42e9-8163-3987bd4382f5	tt_10_ep	950dc9b4-55a3-4be0-9f36-f4fb5a682853	4102cb07-72f8-4ffa-b61b-125756bf28c1	true
8	d987cf33-2314-4a36-abd0-9b599973c8...	block_stm16	87fd7b20-1142b7-a02e-92d6dd074957	7e28abb7-7f22-44fd-a549-ea324d54ab...	true
9	298fd722-f3e3-4d91-b1b2-51e98f108bd7	block_stm_64	95912188-c65c-4262-8e6f-a5f680327c18	7cd8d795-1cdc-4f70-b64b-73c1c0d7c...	true
10	6eb3d993-4ae0-4169-96ed-45fcd953ae46	tt_10_ep	39301c88-e756-4fa3-977a-604037edd579	4102cb07-72f8-4ffa-b61b-125756bf28c1	true

Рис. 6. Записи в таблице плат
Fig. 6. Records in the boards table

id [PK] uuid	downet_id uuid	client_id uuid	port_type_id uuid	link_to_the_server_id link_id uuid	link_to_the_server_id client_id uuid
1	19827115-05a8-4a45-434a-4a4a4a4a4a4a	82204445-4d5a-4770-4d51-4d514d514d51	4a4a4a45-4d5a-4770-4d51-4d514d514d51	3a778a7c-108f-488a-8a72-358b3cc3555a	[null]
2	841a88a2-0179-421a-4849-3a493a493a49	013a7116-2378-40b2-4410-5050137116	4a4a4a45-4d5a-4770-4d51-4d514d514d51	3a778a7c-108f-488a-8a72-358b3cc3555a	[null]
3	14717847-1734-413a-4a73-458a477847	21880221-1a49-4a91-4132-41498118027	4a4a4a45-4d5a-4770-4d51-4d514d514d51	3a778a7c-108f-488a-8a72-358b3cc3555a	[null]
4	871c279a-8a8a-4135-4a14-598a7c108a	091e7c73-233a-4a30-40d3-9059497203b	4a4a4a45-4d5a-4770-4d51-4d514d514d51	3a778a7c-108f-488a-8a72-358b3cc3555a	[null]
5	5c2c5705-4051-4403-4423-473143103b	9a0a8a8b-805a-4042-4404-41300b10b2f	4a4a4a45-4d5a-4770-4d51-4d514d514d51	3a778a7c-108f-488a-8a72-358b3cc3555a	[null]
6	59a55a5c-488a-4a18-4130-054a4a4a7722	01900701-048b-4819-488a-72036b178c	4a4a4a45-4d5a-4770-4d51-4d514d514d51	3a778a7c-108f-488a-8a72-358b3cc3555a	4a1a8a42-5170-422a-4a49-3a493a493a49
7	5a0c180a-103a-4110-8c2f-4a0b10a0b08	024a719a-8143-4a77-8a05-708b8b2a0b	4a4a4a45-4d5a-4770-4d51-4d514d514d51	3a778a7c-108f-488a-8a72-358b3cc3555a	04717847-1734-413a-4a73-458a477847
8	75c12a7c-4a7c-4056-8a96-7a7a555a2	9a0a8a8b-805a-4042-4404-41300b10b2f	4a4a4a45-4d5a-4770-4d51-4d514d514d51	3a778a7c-108f-488a-8a72-358b3cc3555a	9910279a-8a8a-4135-4a14-598a7c108a
9	51a1a2b5-0a57-405a-8043-4a1a5a5a8a	4c5a4a2b-c57c-424a-8163-30a70a6302f	4a4a4a45-4d5a-4770-4d51-4d514d514d51	3a778a7c-108f-488a-8a72-358b3cc3555a	4a1a8a42-5170-422a-4a49-3a493a493a49
10	430c157a-008b-431a-055a-17211a4a7a2c	4a0b5993-4a40-4116-9a4a-455a455a3a4b	4a4a4a45-4d5a-4770-4d51-4d514d514d51	3a778a7c-108f-488a-8a72-358b3cc3555a	51a1a2b5-0a57-405a-8043-4a1a5a5a8a

Рис. 7. Записи в таблице портов

Fig. 7. Records in the ports table

id [PK] uuid	client_id uuid	created_time timestamp without time zone	active_flag boolean	task_data json
1	c331c7d5-0e91-40bb-92f6-17d2b7446e2c	2025-03-30 10:23:54	true	{ "startingPoint": "", "destinationPoint": "", "capacity": "" }

Рис. 8. Записи в таблице задач

Fig. 8. Records in the tasks table

id [PK] uuid	task_id uuid	client_id uuid	route_data json	active_flag boolean	price double precision
1	689b3f66-4767-4a30-add7-d3c06bb0032b	c331c7d5-0e91-40bb-92f6-17d2b7446e2c	4a0b5993-4a40-4116-9a4a-455a455a3a4b	true	200

Рис. 9. Записи в таблице маршрутов

Fig. 9. Records in the routes table

3. ОБСУЖДЕНИЕ

В системе есть параметры, которые являются рассчитываемыми (например, количество свободных портов у платы). При добавлении маршрута для нового клиента количество свободных портов будет уменьшаться. Это логично, так как порты новых клиентов занимают свое место на плате. В статье рассматривается подход, при котором не создаются отдельные поля для хранения информации о количестве доступных портов. Система будет рассчитывать нужные параметры, но их

сохранение в базу данных не предусмотрено.

Второй способ — считать значения, сохранять в базу данных и вместо выполнения логики расчета брать значение из атрибута.

4. ВЫВОДЫ

Построена и реализована программными средствами реляционная модель узла связи, применяемая к любому используемому оборудованию для произвольного уровня агрегации портов.

ВКЛАД АВТОРОВ

А. В. Синицын — концепция, анализ информации.

С. Э. Лукьянов — концепция, редактирование текста.

Р. Ю. Байдушин — сбор данных, анализ информации, подготовка текста.

П. А. Цветков — сбор данных, редактирование текста.

CONTRIBUTION OF THE AUTHORS

Alexander V. Sinitsyn — concept, information analysis.

Stanislav E. Lukyanov — concept, text editing.

Roman Y. Baydushin — data collection, information analysis, text preparation.

Pavel A. Tsvetkov — data collection, text editing.

КОНФЛИКТ ИНТЕРЕСОВ

Авторы заявляют об отсутствии конфликта интересов.

CONFLICT OF INTERESTS

The author declares no relevant conflict of interest.

СПИСОК ИСТОЧНИКОВ/ REFERENCES

1. Кучерявый А.Е., Владыко А.Г., Киричек Р.В., Маколкина М.А., Парамонов А.И., Выборнова А.И., Пирмагомедов Р.Я. Перспективы научных исследований в области сетей связи на 2017–2020 годы. *Информационные технологии и телекоммуникации*. 2016;4(3):1–14. (In Russ.).
Koucheryavy A., Vladyko A., Kirichek R., Makolkina M., Paramonov A., Vybornova A., Pirmagomedov R. The prospects for research in the field of communications networks on the 2017–2020 years. *Telecom IT*. 2016;4(3):1–14 (In Russ.).
2. Cholda P., Mykkeltveit A., Helvik B.E., Wittner O.J., Jajszczyk A. A survey of resilience differentiation frameworks in communication networks. *IEEE Communications Surveys & Tutorials*. 2007;9(4):32–55. <https://doi.org/10.1109/COMST.2007.4444749>
3. Wiatr P., Chen J., Monti P., Wosinska L. Energy efficiency versus reliability performance in optical backbone networks. *Journal of Optical Communications and Networking*. 2025;7(3): A482–A491. <https://doi.org/10.1364/JOCN.7.00A482>
4. Чечин Г.В. Математическая модель для анализа сетей связи с одним или несколькими узлами обмена трафиком. *Электросвязь*. 2019;(7):50–55.
Chechin G.V. Mathematical Model for Analyzing Communication Networks with One or Several Traffic Exchange Nodes. *Electrosvyaz*. 2019;(7):50–55. (In Russ.).
5. Иванов Р.М. Функциональная модель структуры узла связи специального назначения рассредоточенного (модульного) пункта управления. *Известия Тульского государственного университета. Технические науки*. 2021;(9):231–236. <https://doi.org/10.24412/2071-6168-2021-9-231-236>
Ivanov R.M. Functional model of the structure of a special-purpose communication node of a dispersed (modular) control center. *News of Tula State University. Technical sciences*. 2021;(9):231–236. (In Russ.). <https://doi.org/10.24412/2071-6168-2021-9-231-236>
6. Наний О.Е., Стариков П.П., Петров С.В. и др. Инженерно-технические решения по построению магистральной и региональной сети связи с использованием технологии OTN/DWDM. *Информатизация и связь*. 2019. № 1. с. 74–83. <https://doi.org/10.34219/2078-8320-2019-10-1-74-83>. EDN ZCPWYX
Naniy O.E., Starikov P.P., Petrov S.V. and etc. Engineering and technical solutions for building a backbone and regional communication network using OTN/DWDM technology. *Information and communication*. 2019. № 1. p. 74–83. (In Russ.). <https://doi.org/10.34219/2078-8320-2019-10-1-74-83>. EDN ZCPWYX

7. Зайкова Е.А., Моргунов Е.П. Обзор существующих средств резервного копирования базы данных в СУБД PostgreSQL. *Актуальные проблемы авиации и космонавтики*. 2021;2:248–250.
Zaikova E.A., Morgunov E.P. Overview of existing database backup tools in PostgreSQL database management system. *Current problems of aviation and cosmonautics*. 2 (2021): 248–250. (In Russ.).
8. Makris A., Tserpes K., Spiliopoulos G. et al. MongoDB Vs PostgreSQL: A comparative study on performance aspects. *Geoinformatica*. 2021;25(2):243–268. <https://doi.org/10.1007/s10707-020-00407-w>
9. Truskowski W., Klewek R., Skublewska-Paszkowska, M. (2020). Comparison of MySQL, MSSQL, PostgreSQL, Oracle databases performance, including virtualization. *Journal of Computer Sciences Institute*, 16, 279–284. <https://doi.org/10.35784/jcsi.2026>
10. Mohammed M.A., Muhammed D.A., Abdullah J.M. Practical Approaches of Transforming ER Diagram into Tables. *International Journal of Multidisciplinary and Scientific Emerging Research*. 2015;4(22):2349–6037.
11. Старушенкова Е.Е. Основные этапы проектирования баз данных. *E-Scio*. 2021;(11):318–323. EDN OLOOIR
Starushenkova E.E. The main stages of database design. *E-Scio*. 2021;(11):318-323. (In Russ.). EDN OLOOIR
12. Синицын А.В., Синицын А.А., Жаворонкова О.Г. Алгоритм построения оптимальной маршрутизации для отказоустойчивой сети связи. *Информатизация и связь*. 2023;(5):49–56. <https://doi.org/10.34219/2078-8320-2023-14-5-49-56>
Sinitsyn A.V., Sinitsyn A.A., Zhavoronkova O.G. An algorithm for building optimal routing for a fault-tolerant communication network. *Informatization and communication*. 2023;(5):49–56. (In Russ.). DOI: 10.34219/2078-8320-2023-14-5-49-56
13. Иващенко Ю.Н., Бабинцев Д.И., Карнаух С.Э., Лукьянов А.П., Супрун В.Н., Бетин В.А. Прогнозирование трафика телекоммуникационных сетей и их оптимизация. *Информатизация и связь*. 2023;(4):53–58. <https://doi.org/10.34219/2078-8320-2023-14-4-53-58>. EDN CPWWJP
Ivashchenko V.A., Babintsev Y.N., Karnaukh D.I., Lukyanov S.E., Suprun A.P., Betin V.N. Forecasting the traffic of telecommunication networks and their optimization. *Informatization and communications*. 2023;(4):53–58. (In Russ.). <https://doi.org/10.34219/2078-8320-2023-14-4-53-58>. EDN CPWWJP

ИНФОРМАЦИЯ ОБ АВТОРАХ

Александр Владимирович Синицын, кандидат физико-математических наук, РТУ МИРЭА, Институт информационных технологий, г. Москва, Российская Федерация; ORCID: <https://orcid.org/0000-0001-7392-1837>; e-mail: a@sinitsyn.info

Станислав Эмильевич Лукьянов, кандидат технических наук, ведущий научный сотрудник, Федеральное государственное автономное научное учреждение «Центр информационных технологий и систем органов исполнительной власти имени А. В. Старовойтова» (ФГАНУ ЦИТиС), г. Москва, Российская Федерация; ORCID: <https://orcid.org/0009-0007-0596-1942>; e-mail: lukyanov@inevm.ru

Роман Юрьевич Байдушин, РТУ МИРЭА, Институт информационных технологий, г. Москва, Российская Федерация; ORCID: <https://orcid.org/0009-0005-4354-044X>; e-mail: roman.baydushin@gmail.com

Павел Алексеевич Цветков, РТУ МИРЭА, Институт информационных технологий, г. Москва, Российская Федерация; ORCID: <https://orcid.org/0009-0006-1597-6251>; e-mail: pashokts@mail.ru

INFORMATION ABOUT THE AUTORS

Alexander V. Sinitsyn, PhD (Phys.-Math.), RTU MIREA, Institute of Information Technology, Moscow, Russian Federation; ORCID: <https://orcid.org/0000-0001-7392-1837>; e-mail: a@sinitsyn.info

Stanislav E. Lukyanov, Cand. Sci. (Eng.) Federal State Autonomous Scientific Institution "Center for Information Technologies and Systems of Executive Authorities named after A. V. Starovoytov"; ORCID: <https://orcid.org/0009-0007-0596-1942>; e-mail: lukyanov@inevm.ru

Roman Y. Baydushin, RTU MIREA, Institute of Information Technology; ORCID: <https://orcid.org/0009-0005-4354-044X>; e-mail: roman.baydushin@gmail.com

Pavel A. Tsvetkov, RTU MIREA, Institute of Information Technology; ORCID: <https://orcid.org/0009-0006-1597-6251>; e-mail: pashokts@mail.ru

Поступила / Received 01.03.2025

Принята / Accepted 01.04.2025

Подписано в печать.

Печать офсетная

Тираж 500 экз.

Адрес редакции: 125252, Россия, Москва, ул. Куусинена, д. 21-Б

Типография АО «Т8 Издательские Технологии»,

Адрес типографии: 109316, Россия, Москва, Волгоградский пр-т, д. 42, корп. 5.