|  |  |
| --- | --- |
|  |  |

**RT.DataVision**

**Описание функциональных характеристик**

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |

**Оглавление**

[1 Термины и определения 3](#_Toc165045673)

[2 Функциональные характеристики 3](#_Toc165045674)

[3 Принципы функционирования RT.DataVision 5](#_Toc165045675)

[3.1 Структура RT.DataVision 5](#_Toc165045676)

[3.2 Функционирование RT.DataVision 6](#_Toc165045677)

[3.2.1 Подключение к СУБД 6](#_Toc165045678)

[3.2.2 Создание диаграмм и дашбордов 6](#_Toc165045679)

[3.2.3 Семантический слой 7](#_Toc165045680)

[3.2.4 Безопасность 7](#_Toc165045681)

[3.2.5 Шаблоны SQL 8](#_Toc165045682)

[3.2.6 Кэширование 9](#_Toc165045683)

[3.2.7 Регистрация событий 9](#_Toc165045684)

[3.2.8 Асинхронные запросы через Celery 9](#_Toc165045685)

[3.2.9 Оповещения и отчеты 9](#_Toc165045686)

[3.2.10 Dremio 9](#_Toc165045687)

[4 Бизнес-процессы 10](#_Toc165045688)

# Термины и определения

| *Термин* | *Определение* |
| --- | --- |
| Пользователь | Лицо или группа лиц, извлекающих пользу из RT.DataVision в процессе её применения. |
| Разработчик | Организация, которая выполняет разработку задач (в том числе анализ требований, проектирование, приёмочные испытания) в процессе жизненного цикла. |

# Функциональные характеристики

RT.DataVision состоит из двух компонентов:

1. RT.DataVision – это современное корпоративное веб-приложение для бизнес-аналитики. Это быстрая, легкая, интуитивно понятная платформа, позволяющая пользователям с любым набором навыков легко исследовать и визуализировать свои данные, от простых круговых диаграмм до высокодетализированных геопространственных диаграмм deck.gl.
2. Dremio, который обеспечивает аналитику самообслуживания для групп обработки данных. Аналитики данных могут исследовать и визуализировать данные с временем ответа на запросы менее секунды, а инженеры данных могут принимать и преобразовывать данные непосредственно в озере данных с полной поддержкой операций DML. Кроме того, аналитики могут объединять данные в озере с данными во внешних базах данных, поэтому им не нужно перемещать данные в объектное хранилище, чтобы извлечь пользу из этих данных. Открытая платформа Dremio Lakehouse, основанная на таких общественных стандартах, как Apache Iceberg и Apache Arrow, позволяет организациям использовать лучшие в своем классе механизмы обработки и устраняет привязку к поставщику.

RT.DataVision – позволяет своим пользователям возможность визуализации в виде различного вида диаграмм наборов данных полученных с помощью SQL запроса из СУБД. RT.DataVision может быть полезен для выполнения бизнес анализа и принятия управленческих решений.

Основными достоинствами RT.DataVision являются то, что он:

1. Мощный, но простой в использовании.

Быстро и легко интегрируйте и исследуйте свои данные, используя либо наш простой конструктор визуализаций без кода, либо современную интегрированную среду разработки SQL.

1. Интегрируется с современными базами данных.

Может подключаться к любому источнику данных на основе SQL через SQLAlchemy, включая современные облачные собственные базы данных и механизмы с петабайтным масштабом.

1. Имеет современную архитектуру.

Является легким и хорошо масштабируемым, используя всю мощь вашей существующей инфраструктуры, не требуя еще одного уровня приема данных.

1. Богатые визуализации и информационные панели.

Поставляется с широким набором красивых визуализаций. Наша архитектура подключаемых модулей визуализации упрощает создание пользовательских визуализаций, которые добавляются непосредственно в RT.DataVision.

Компонент, входящий в состав системы, Dremio обеспечивает аналитику самообслуживания для групп обработки данных. Аналитики данных могут исследовать и визуализировать данные с временем ответа на запросы менее секунды, а инженеры данных могут принимать и преобразовывать данные непосредственно в озере данных с полной поддержкой операций DML. Кроме того, аналитики могут объединять данные в озере с данными во внешних базах данных, поэтому им не нужно перемещать данные в объектное хранилище, чтобы извлечь пользу из этих данных. Открытая платформа Dremio Lakehouse, основанная на таких общественных стандартах, как Apache Iceberg и Apache Arrow, позволяет организациям использовать лучшие в своем классе механизмы обработки и устраняет привязку к поставщику.

RT.DataVision обеспечивает:

1. Интуитивно понятный интерфейс для визуализации наборов данных и создания интерактивных информационных панелей.
2. Широкий спектр красивых визуализаций для демонстрации ваших данных.
3. Построитель визуализации без кода для извлечения и представления наборов данных.
4. SQL IDE мирового класса для подготовки данных к визуализации, включая богатый браузер метаданных.
5. Облегченный семантический слой, который позволяет аналитикам данных быстро определять настраиваемые параметры и показатели.
6. Встроенную поддержку большинства баз данных, поддерживающих SQL.
7. Бесшовное асинхронное кэширование и запросы в памяти.
8. Расширяемую модель безопасности, которая позволяет настраивать очень сложные правила, определяющие, кто может получить доступ к тем или иным функциям продукта и наборам данных.
9. Интеграцию с основными механизмами аутентификации (база данных, OpenID, LDAP, OAuth, REMOTE\_USER и т. д.).
10. Возможность добавления пользовательских плагинов визуализации.
11. API для программной настройки.
12. Облачную архитектуру, специально разработанную для масштабирования.

RT.DataVision является облачным и предназначен для обеспечения высокой доступности. Он был разработан для масштабирования до больших распределенных сред и очень хорошо работает внутри контейнеров. Хотя вы можете легко протестировать RT.DataVision на скромной конфигурации или просто на своем ноутбуке, масштабирование платформы практически не ограничено.

RT.DataVision кроме того, что является облачным, очень гибкий, он позволяет вам выбирать:

1. веб-сервер (Gunicorn, Nginx, Apache),
2. механизм базы данных метаданных (MySQL, Postgres, MariaDB и т. д.),
3. очередь сообщений (Redis, RabbitMQ, SQS и т. д.),
4. серверную часть результатов (S3, Redis, Memcached и т. д.),
5. уровень кэширования (Memcached, Redis и т. д.),

RT.DataVision также хорошо работает с такими сервисами, как NewRelic, StatsD и DataDog, и может выполнять аналитические рабочие нагрузки для большинства популярных технологий баз данных.

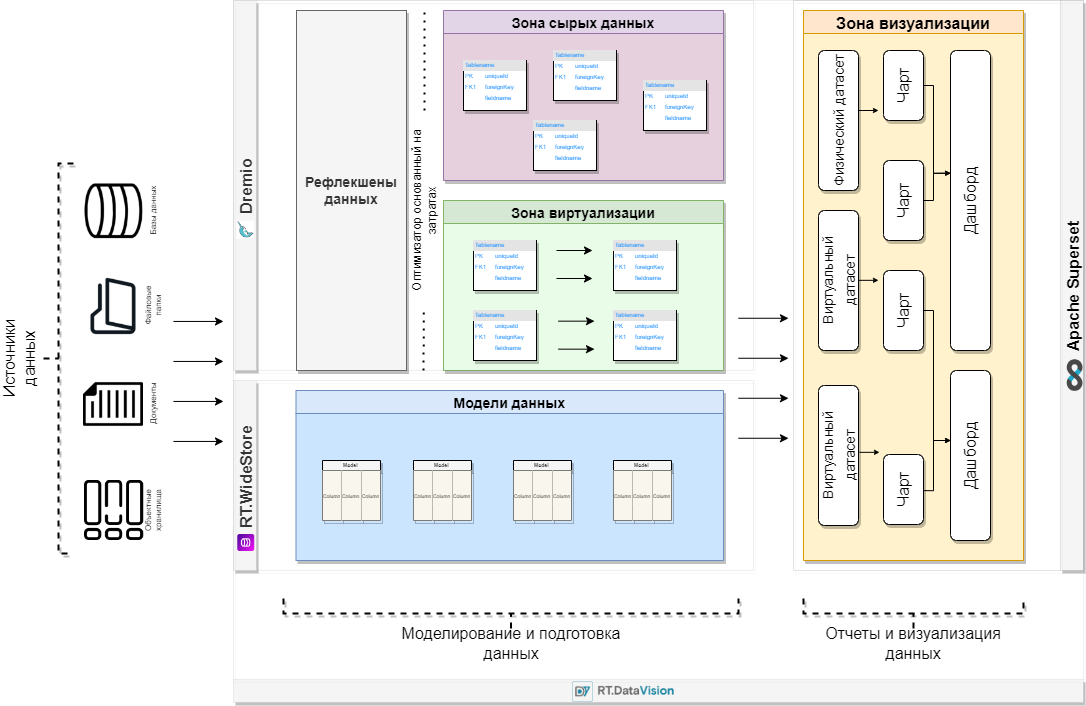
Dremio обеспечивает следующие возможности:

1. По сути является «данными как услуга», т.е. предоставляет возможность быстрого получения и анализа данных.
2. Виртуализация данных: Dremio позволяет создавать виртуальные наборы данных (представления) из подключённых источников данных (озёра данных, базы данных и облачные хранилища), к которому он может в любой момент обратиться и получить доступ с помощью SQL-запроса и других инструментов.
3. Отсутствие необходимости в создании физических слоёв данных в хранилищах и их поддержки ресурсами (дополнительные сервера, штатные единицы), т.е. анализ данных можно производить напрямую в Dremio с использованием представлений.
4. Передача проанализированного и трансформированного набора данных под определённые нужды в BI-системы для построения отчётности на основе уже готового набора.
5. Простой, интуитивно понятный интерфейс, который позволяет снизить порог вхождения аналитика в рабочую деятельность, а для опытных пользователей предоставляет расширенные возможности для быстрого выполнения действий в интерфейсе.
6. Подключение большого количества источников для объединения данных из них: Dremio представляет собой единую платформу для объединения, использования и хранения всех необходимых для деятельности компании данных из источников.
7. Возможности self-service: Dremio предоставляет сервис, интерфейс и права доступа, с помощью которых бизнес сам сможет собрать необходимые ему данные, воспользовавшись только консультацией со стороны ИТ-специалиста.
8. Скорость выполнения заданий: благодаря архитектуре и механизмам Dremio получать и трансформировать данные быстрее, чем с использованием экосистемы ПО, которая трансформирует и хранит множество этапов трансформации данных в физических слоях.

# Принципы функционирования RT.DataVision

## Структура RT.DataVision

Схема взаимодействия RT.DataVision с компонентами представлена на схеме ниже.



## Функционирование RT.DataVision

### Подключение к СУБД

Для RT.DataVision устанавливается драйвер базы данных Python DB-API и диалект SQLAlchemy для каждого хранилища данных, к которому необходимо подключение.

RT.DataVision не поставляется в комплекте с возможностью подключения к базам данных, за исключением SQLite, который является частью стандартной библиотеки Python. Поэтому для использования иных СУБД необходимо установить пакеты для базы данных, которую требуется использовать в качестве базы данных метаданных, а также пакеты, необходимые для подключения к базам данных, к которым требуется получить доступ через RT.DataVision.

RT.DataVision поддерживает многие базы данных, основными критериями которых являются наличие функционального диалекта SQLAlchemy и драйвера Python.

### Создание диаграмм и дашбордов

Для исследования и визуализации данных в RT.DataVision используются диаграммы и дашборды различной сложности.

Для создания диаграммы или дашборда необходимо выполнение следующей последовательности шагов:

* Подключение к новой или существующей СУБД;
* Выбор таблицы (набора данных);
* Настройка свойств полей таблицы;
* Формирование запроса или запросов выборки данных для их визуализации;
* Выбор типа и параметров визуализации данных для создания диаграммы;
* Сохранение диаграммы путём добавления на новый или существующий дашборд;
* Настройка доступа к дашборду других пользователей.

RT.DataVision не имеет уровня хранения для данных, вместо этого RT.DataVision подключается к СУБД, содержащей необходимые данные для визуализации.

RT.DataVision также поддерживает возможность загрузки файла CSV или Excel в базу данных.

Для разграничения доступа к функциям RT.DataVision, созданным наборам данных, дашбордам служит ролевая модель. При регистрации пользователя, ему присваивается определённая роль.

Доступ к дашбордам управляется через владельцев (пользователей, у которых есть права на редактирование дашборда).

Доступом пользователей, не являющихся владельцами, можно управлять двумя способами:

* Разрешения на наборы данных — если добавить к соответствующим ролям разрешения на наборы данных, то будет автоматически предоставлен неявный доступ ко всем дашбордам, использующим эти разрешенные наборы данных.
* Роли дашборда — если включить флаг DASHBOARD\_RBAC, то можно управлять тем, какие роли могут получить доступ к дашборду. Наличие доступа к дашборду неявно предоставляет доступ для чтения к связанным наборам данных.

### Семантический [слой](https://superset.apache.org/docs/Creating%20Charts%20and%20Dashboards/creating-your-first-dashboard#superset-semantic-layer)

RT.DataVision имеет тонкий семантический слой, который добавляет множество улучшений качества для аналитиков.

Семантический слой RT.DataVision может хранить 2 типа вычисляемых данных:

* Виртуальные показатели — возможность написания SQL-запросов, которые объединяют значения из нескольких столбцов (например, SUM(recovered) / SUM(confirmed)) и делают их доступными в виде столбцов для (например, recovery\_rate) визуализации. Агрегатные функции разрешены и рекомендуются для метрик.
* Виртуальные вычисляемые столбцы — возмжность написания SQL-запросов, которые настраивают внешний вид и поведение определенного столбца (например, CAST(recovery\_rate) as float). Агрегатные функции не разрешены в вычисляемых столбцах.

### Безопасность

Безопасность в RT.DataVision обеспечивается Flask AppBuilder (FAB), платформой разработки приложений, созданной поверх Flask. FAB обеспечивает аутентификацию, управление пользователями, разрешения и роли.

RT.DataVision поставляется со следующим набором ролей:

* Admin — администраторы имеют все возможные права, включая предоставление или отзыв прав других пользователей и изменение чужих срезов и дашбордов.
* Alpha — альфа-пользователи имеют доступ ко всем источникам данных, но не могут предоставлять или отзывать доступ другим пользователям. Они также ограничены в изменении объектов, которыми они владеют. Пользователи Alpha могут добавлять и изменять источники данных.
* Gamma — пользователи Gamma имеют ограниченный доступ. Они могут потреблять только данные, поступающие из источников данных, к которым им предоставлен доступ через другую дополнительную роль. У них есть доступ только для просмотра срезов и дашбордов, созданных из источников данных, к которым у них есть доступ. Пользователи Gamma не могут изменять или добавлять источники данных. Предполагается, что они в основном потребители контента, хотя могут создавать срезы и дашборды.

При просмотре дашбордов и срезов пользователи Gamma увидят только те объекты, к которым у них есть доступ.

* [sql\_lab](https://superset.apache.org/docs/security#sql_lab) — предоставляет доступ к Лаборатории SQL. Пользователи с правами Admin по умолчанию имеют доступ ко всем базам данных, пользователям Alpha и Gamma необходимо предоставить доступ для каждой базы данных.
* Public — чтобы предоставить неавторизованным пользователям доступ к некоторым функциям RT.DataVision, можно использовать параметр PUBLIC\_ROLE\_LIKE конфигурации и назначить его другой роли, чьи разрешения необходимо передать этой роли.

Роли состоят из набора разрешений, а RT.DataVision имеет множество категорий разрешений. Ниже представлены различные категории разрешений:

* Модель и действие: модели — это объекты, такие как дашборд, срез или пользователь. Каждая модель имеет фиксированный набор разрешений, например, can\_edit, can\_show, can\_delete, can\_list, can\_add и так далее. Например, можно разрешить пользователю удалять дашборды, добавив can\_delete на сущность дашборда к роли и предоставив данному пользователю эту роль.
* Представления: представления — это отдельные веб-страницы, такие как представление «Исследовать» или представление «Лаборатория SQL». Выдав разрешение к представлению, пользователь увидит его в пунктах меню и сможет загрузить данную страницу.
* Источник данных: для каждого источника данных создается разрешение. Если у пользователя нет разрешения all\_datasource\_access, он сможет только просматривать срезы или исследовать источники данных, которые ему предоставлены.
* База данных: предоставление доступа к базе данных позволяет пользователю получить доступ ко всем источникам данных в этой базе данных и позволяет пользователю запрашивать эту базу данных в Лаборатории SQL при условии, что пользователю предоставлено специальное разрешение Лаборатории SQL.

Также RT.DataVision поддерживает фильтры безопасности на уровне строк, т.е. имеется возможность создавать фильтры, которые назначаются конкретной таблице, а также набору ролей.

### Шаблоны SQL

Лаборатория SQL и Исследовать поддерживает [шаблоны Jinja](https://jinja.palletsprojects.com/en/2.11.x/) в запросах. Когда шаблоны включены, код Python может встраиваться в виртуальные наборы данных и в пользовательский SQL в фильтрах и элементах управления метриками в Исследовать.

### Кэширование

RT.DataVision использует [Flask-Caching](https://flask-caching.readthedocs.io/) для кэширования. Для обеспечения безопасности существуют две отдельные конфигурации кэша для собственных метаданных RT.DataVision (CACHE\_CONFIG) и данных диаграмм, запрашиваемых из подключенных источников данных (DATA\_CACHE\_CONFIG). Однако результаты запросов из Лаборатории SQL хранятся в другом бэкэнде, который называется RESULTS\_BACKEND.

### Регистрация событий

RT.DataVision по умолчанию регистрирует события специальных действий в своей внутренней базе данных. Доступ к этим журналам можно получить в пользовательском интерфейсе. Можно настраивать данные журналы, реализуя собственный класс журнала событий.

### Асинхронные запросы через Celery

В больших аналитических базах данных обычно выполнение запросов осуществляется в течение нескольких минут или часов. Чтобы включить поддержку длительных запросов, которые выполняются за пределами типичного времени ожидания веб-запроса (30-60 секунд), необходимо настроить асинхронный сервер для RT.DataVision, который состоит из:

* Одного или нескольких воркеров RT.DataVision (которые реализованы как воркеры Celery и могут быть запущены с помощью команды celery worker).
* Брокера celery (очередь сообщений), для которого рекомендуется использовать Redis или RabbitMQ.
* Серверной части результатов, которая определяет, где воркер будет сохранять результаты запроса.

### Оповещения и отчеты

Пользователи могут настраивать автоматические оповещения и отчеты для отправки дашбордов или диаграмм получателю через электронную почту или канал Slack.

Существуют следующие варианты:

* Оповещения отправляются при достижении условия SQL.
* Отчеты отправляются по расписанию.

### Dremio

Механизм SQL для открытых платформ, обеспечивающий производительность и возможности на уровне хранилища данных в озере данных, а также возможности самообслуживания, которые делают данные пригодными для использования и совместной работы.

Один механизм для всех рабочих нагрузок SQL. Поддерживайте все рабочие нагрузки SQL, от критически важных BI-дашбордов до специальных/исследовательских рабочих нагрузок, с помощью механизма, который обеспечивает производительность и функциональность хранилища (включая операции DML) непосредственно в Lakehouse.

Обеспечьте бесперебойную работу конечных пользователей с помощью прозрачной оптимизации и ускорения запросов.

Любые данные, где угодно. Обогатите свой анализ данными, которых [еще] нет в озере, с помощью коннекторов к различным внешним базам данных.

Демократизируйте свои данные там, где они находятся, и обеспечьте различные варианты использования с унифицированным представлением ваших данных без копий данных.

Согласованные и консистентные данные. Определите согласованное и безопасное представление данных и бизнес-метрик, которое может быть использовано любым последующим приложением.

Создайте более простую модель данных и устраните необходимость экспорта данных в экстракты Tableau/импорт Power BI, кубы и другие оптимизированные структуры данных.

Курируйте, анализируйте и делитесь наборами данных в режиме самообслуживания. Работайте с наиболее логичным представлением данных, не прибегая к предварительно оптимизированным и агрегированным физическим таблицам.

Лучший в своем классе опыт. Взаимодействуйте со своими данными с помощью интегрированного интерфейса для аналитиков, которые знают и любят SQL.

Запрашивайте данные о озере с помощью SQL Runner от Sonar, лучшей в своем классе интегрированной среды разработки для аналитиков, которая включает в себя автозаполнение, выполнение нескольких операторов, а также возможность сохранять и делиться сценариями SQL.

Понимайте и оптимизируйте производительность запросов с помощью SQL Profiler от Sonar, а также визуализируйте использование и происхождение наборов данных с помощью карты данных Sonar.

Простая интеграция BI и обработки данных. Команды BI могут беспрепятственно подключаться к Dremio Sonar и визуализировать данные с помощью ваших любимых инструментов BI со встроенными коннекторами для Tableau и Power BI, включая поддержку единого входа на основе OAuth 2.0.

Команды по обработке и анализу данных получают выгоду от высокоскоростного подключения к данным. Легко обрабатывайте результаты запросов с помощью блокнотов Python, R и Jupyter благодаря Apache Arrow Flight.

# Бизнес-процессы

RT.DataVision может выполнять автоматизацию следующих бизнес-процессов:

* Специфика RT.DataVision даёт преимущества в таких сферах как аналитика данных, ритейл, банковская сфера и другие, в которых имеется потребность получать быстрые результаты визуализации данных, что позволяет принимать решения на основе данных (data-driven).