**СОДЕРЖАНИЕ**

[ТЕРМИНЫ И ОПРЕДЕЛЕНИЯ 5](#_Toc193673807)

[**ВВЕДЕНИЕ** 6](#_Toc193673808)

[**1 Исследование предметной области** 7](#_Toc193673809)

[**1.1 Исследование особенностей систем агрегирования данных** 7](#_Toc193673810)

[**1.1.1 Определение и назначение систем агрегирования данных** 7](#_Toc193673811)

[**1.1.2 Классификация систем агрегирования данных** 7](#_Toc193673812)

[**1.1.3 Архитектурные принципы систем агрегирования данных** 7](#_Toc193673813)

[**1.2 Исследование методов выполнения аналитических запросов** 8](#_Toc193673814)

[**1.2.1 Введение в аналитические запросы** 8](#_Toc193673815)

[**1.2.2 Основные методы выполнения аналитических запросов** 8](#_Toc193673816)

[**1.2.2.1 OLAP (Online Analytical Processing)** 8](#_Toc193673817)

[**1.2.2.2 MapReduce** 8](#_Toc193673818)

[**1.2.2.3 SQL и NoSQL-подходы** 8](#_Toc193673819)

[**1.2.2.4 Индексы и материализованные представления** 8](#_Toc193673820)

[**1.3 Исследование особенностей работы оперативной аналитической обработки** 9](#_Toc193673821)

[**1.3.1 Определение оперативной аналитической обработки (OLAP)** 9](#_Toc193673822)

[**1.3.2 Основные принципы OLAP** 9](#_Toc193673823)

[**1.3.3 Технологии оперативной аналитической обработки** 9](#_Toc193673824)

[**1.3.4 Преимущества и недостатки OLAP** 9](#_Toc193673825)

# ТЕРМИНЫ И ОПРЕДЕЛЕНИЯ

В настоящем отчете применяют следующие термины, сокращения и определения с соответствующими определениями.

|  |  |
| --- | --- |
| OLAP |  |
| API |  |
| ETL |  |
| BI-система |  |
| БД |  |
| NoSQL-хранилищами |  |
| batch processing |  |
| stream processing |  |
| OLAP |  |
| MOLAP |  |
| ROLAP |  |
| HOLAP |  |
| **MapReduce** |  |
| **SQL** |  |
| PostgreSQL |  |
| MySQL |  |
| Oracle |  |
| СУБД |  |
|  |  |
|  |  |
|  |  |

**ВВЕДЕНИЕ**

В современном мире финансовая аналитика требует обработки больших объемов данных, поступающих из различных источников. Для эффективного анализа и принятия решений необходимо агрегировать данные, сводя их к структурированным наборам, удобным для последующей обработки. В данной главе рассматриваются особенности систем агрегирования данных, их архитектурные принципы и функциональные возможности.

1. **Исследование предметной области**

В современных финансовых организациях и крупных компаниях система мотивации сотрудников часто включает премирование на основе KPI, объемов продаж, выполнения планов и других показателей. Ручной расчет таких премий трудоемок, подвержен ошибкам и не позволяет оперативно анализировать данные в различных разрезах (по отделам, регионам, временным периодам).

На протяжении последних лет различные исследователи и компании предлагали решения в области агрегирования данных и аналитической обработки. Однако существующие подходы либо ориентированы на обработку данных в пакетном режиме, что замедляет расчет премий в реальном времени, либо требуют значительных вычислительных ресурсов. Исследование направлено на разработку системы, способной эффективно обрабатывать финансовые данные в режиме оперативной аналитики, сочетая высокую скорость обработки, точность расчетов и адаптивность к изменяющимся условиям.

* 1. **Исследование особенностей систем агрегирования данных**
     1. **Определение и назначение систем агрегирования данных**

Системы агрегирования данных представляют собой программные и аппаратные комплексы, предназначенные для сбора, обработки, обобщения и хранения информации из разнородных источников. Эти системы применяются для консолидации данных и их подготовки к аналитической обработке.

Основные задачи агрегирования данных:

* Объединение данных из различных источников (базы данных, API, файловые хранилища);
* Очистка и трансформация данных;
* Поддержка процессов ETL (Extract, Transform, Load);
* Обеспечение оперативного доступа к агрегированным данным;
* Подготовка данных для аналитических и BI-систем.

Кому и зачем это нужно?

HR-отделы и руководство компаний – для автоматизации и ускорения расчета премий.

Финансовые аналитики – для гибкого анализа эффективности сотрудников.

IT-департаменты – для построения масштабируемой системы обработки данных.

Проблема:

Существующие решения (Excel, реляционные БД, ERP-системы) не справляются с:

большими объемами данных (миллионы транзакций);

сложными аналитическими запросами (агрегация по нескольким измерениям);

реальным временем (необходимостью мгновенного пересчета при изменении данных).

* + 1. **Классификация систем агрегирования данных**
       1. **По типу источников данных**

Реляционные базы данных (SQL-based) – традиционные базы данных, такие как PostgreSQL, MySQL, Oracle, которые хорошо подходят для структурированных данных и обеспечивают мощные средства запросов и агрегирования.

NoSQL-хранилища – базы данных вроде MongoDB, Cassandra, предназначенные для обработки неструктурированных и полуструктурированных данных, часто применяются для потоковой аналитики и масштабируемых решений.

Потоковые данные (Stream Data) – данные, поступающие в реальном времени из сенсоров, логов, API-интерфейсов (Kafka, Apache Flink). Используются для построения предсказательных моделей и аналитики в реальном времени.

* + - 1. **По способу обработки данных**

Пакетная обработка (Batch Processing) – применяется, когда анализ проводится на больших объемах данных, но в режиме периодического обновления. Подходит для исторического анализа, но не всегда удовлетворяет требованиям оперативной аналитики.

Потоковая обработка (Stream Processing) – позволяет анализировать данные в реальном времени. Применяется для финансовой аналитики, где требуется оперативное реагирование.

Гибридные подходы – сочетают пакетную и потоковую обработку, позволяя обрабатывать данные как в реальном времени, так и ретроспективно. Этот вариант предпочтителен в задачах финансовой аналитики, так как он обеспечивает баланс между производительностью и точностью.

* + - 1. **По архитектуре**

Централизованные системы – все данные собираются и обрабатываются в одном хранилище. Такой подход удобен для небольших объемов данных, но плохо масштабируется.

Распределенные системы – данные обрабатываются на множестве узлов, что повышает отказоустойчивость и масштабируемость. Такие системы, как Hadoop, Spark и ClickHouse, позволяют работать с большими объемами данных и обеспечивают высокую производительность.

Выбор конкретного класса системы агрегирования данных зависит от требований к обработке данных, скорости аналитики и потребностей бизнеса. В рамках исследования предпочтение отдается распределенной системе с гибридной моделью обработки, так как она позволяет анализировать финансовые данные как в реальном времени, так и с учетом исторических трендов.

* + 1. **Архитектурные принципы систем агрегирования данных**

Современные системы агрегирования данных строятся на основе следующих архитектурных подходов:

* Многоуровневые архитектуры, включающие уровни сбора, обработки и хранения данных.
* Микросервисный подход, обеспечивающий гибкость и масштабируемость.
* Использование облачных технологий, позволяющее динамически изменять ресурсы под нагрузку.
  1. **Исследование методов выполнения аналитических запросов**
     1. **Введение в аналитические запросы**

Аналитические запросы предназначены для обработки больших объемов данных с целью выявления закономерностей, трендов и аномалий. Они широко используются в финансовой аналитике для расчета премий, оценки рисков и прогнозирования.

* + 1. **Основные методы выполнения аналитических запросов**
       1. **OLAP (Online Analytical Processing)**

OLAP-технология предназначена для многомерного анализа данных и позволяет выполнять сложные аналитические запросы. Основные типы OLAP:

* MOLAP (Multidimensional OLAP) — хранение данных в многомерных кубах.
* ROLAP (Relational OLAP) — хранение данных в реляционных таблицах, обработка с помощью SQL-запросов.
* HOLAP (Hybrid OLAP) — гибридный подход, сочетающий MOLAP и ROLAP.
  + - 1. **MapReduce**

Метод MapReduce позволяет обрабатывать большие объемы данных параллельно на распределенных системах. Этот метод эффективен для работы с неструктурированными и полуструктурированными данными.

* + - 1. **SQL и NoSQL-подходы**
* SQL-методы: Используются в традиционных реляционных БД (PostgreSQL, MySQL, Oracle) для аналитических запросов с агрегацией (SUM, AVG, COUNT, GROUP BY).
* NoSQL-методы: Используются в документоориентированных, графовых и других БД (MongoDB, Cassandra) для обработки данных в реальном времени.
  + - 1. **Индексы и материализованные представления**

Для ускорения аналитических запросов применяются индексы (B-деревья, Bitmap-индексы) и материализованные представления, хранящие предварительно рассчитанные результаты запросов.

* 1. **Исследование особенностей работы оперативной аналитической обработки**
     1. **Определение оперативной аналитической обработки (OLAP)**

Оперативная аналитическая обработка (OLAP) представляет собой технологию, обеспечивающую быстрый доступ к агрегированным данным в многомерных структурах. OLAP используется в системах финансовой аналитики для мгновенного расчета показателей, таких как премии, рентабельность и финансовые риски.

* + 1. **Основные принципы OLAP**
* Многомерность данных: Данные организованы в виде кубов с различными измерениями (время, категория, география).
* Агрегация: Данные сводятся к обобщенным показателям, что снижает объем вычислений.
* Оптимизация хранения: Используются специальные структуры данных для быстрого доступа.
  + 1. **Технологии оперативной аналитической обработки**

Среди наиболее распространенных технологий OLAP можно выделить:

Apache Druid: Высокопроизводительная аналитическая база данных для работы с потоковыми и историческими данными.

ClickHouse: Колонночная СУБД с высокой скоростью выполнения аналитических запросов.

Microsoft Analysis Services: Инструмент для работы с OLAP-кубами в экосистеме Microsoft.

* + 1. **Преимущества и недостатки OLAP**

Преимущества:

* Высокая скорость выполнения запросов за счет предварительной агрегации данных.
* Возможность многомерного анализа данных.
* Поддержка сложных аналитических вычислений.

Недостатки:

* Высокие затраты на вычислительные ресурсы.
* Ограниченная гибкость по сравнению с транзакционными базами данных.
* Сложность настройки и поддержки.
  + 1. **Системы, использующие OLAP**

Oracle OLAP – интегрирован в Oracle Database, обеспечивает мощные аналитические функции для обработки многомерных данных.

Минусы: высокая стоимость лицензирования, сложность настройки и администрирования, требует значительных вычислительных ресурсов.

Microsoft Analysis Services (SSAS) – мощный инструмент для построения OLAP-кубов и выполнения аналитических запросов.

Минусы: ограниченная поддержка неструктурированных данных, зависимость от экосистемы Microsoft, требует профессиональной настройки.

IBM Cognos TM1 – высокопроизводительная платформа для финансового моделирования и планирования.

Минусы: высокая стоимость, сложность интеграции с внешними системами, требует квалифицированных специалистов.

SAP BW/4HANA – корпоративное хранилище данных с OLAP-функциями, работает в in-memory режиме.

Минусы: высокая стоимость лицензий и инфраструктуры, требует больших объемов оперативной памяти, сложность администрирования.

Amazon Redshift – облачное хранилище данных с поддержкой OLAP-запросов и масштабируемой аналитикой.

Минусы: высокая стоимость при больших объемах данных, зависимость от AWS-инфраструктуры, задержки при обработке сложных многомерных запросов.

Каждая из этих систем обладает своими характеристиками и подходами. Например, Oracle OLAP интегрирован в реляционную СУБД и поддерживает сложные аналитические запросы, а Amazon Redshift предлагает горизонтальное масштабирование и облачную доступность. Microsoft SSAS ориентирован на удобство работы с Excel и Power BI, а SAP BW/4HANA обеспечивает высокую скорость обработки за счет технологии in-memory.