# Лабораторная работа по теме “Выбор хранилища данных”

## Задача:

Вы являетесь аналитиком данных в небольшой компании, которая решает перейти к современной системе управления данными. Вам поручено выбрать и обосновать выбор хранилища данных для компании. Ваша компания работает в сфере e-commerce и продает товары через онлайн-платформу. У вас есть следующие требования:

**Объем данных:** Ваша компания собирает миллионы записей о продажах, клиентах и инвентаре ежедневно.

**Аналитика**: Вам необходимо обеспечить возможность проведения сложных аналитических запросов к данным для выявления тенденций продаж, сегментации клиентов и определения эффективности маркетинговых кампаний.

**Масштабируемость**: Ваша компания растет, и вам нужна система, которая легко масштабируется с ростом объема данных и нагрузки.

**Реальное время:** Вам нужна возможность анализа данных в реальном времени для мониторинга заказов и складской информации.

## Задачи:

1. Выберите одно или несколько хранилищ данных (реляционная база данных, NoSQL, колоночное хранилище и т. д.), которые, по вашему мнению, наилучшим образом соответствуют требованиям компании.

2. Обоснуйте свой выбор, предоставив аргументы, почему данное хранилище данных является оптимальным для конкретных требований.

3. Опишите структуру данных, которые будут храниться в выбранном хранилище (таблицы, колонки и связи).

4. Предложите схему интеграции данных из разных источников в хранилище.

5. Опишите, как бы вы настроили систему для обработки данных в реальном времени.

Результат должен быть предоставлен в виде \*.docx файла со схемами

# Решение:

1. В качестве основой БД для данного проекта я бы выбрал реляционную PostgreSQL, и для хранения кеша NoSQL БД Redis.
2. И вот на каких основаниях:

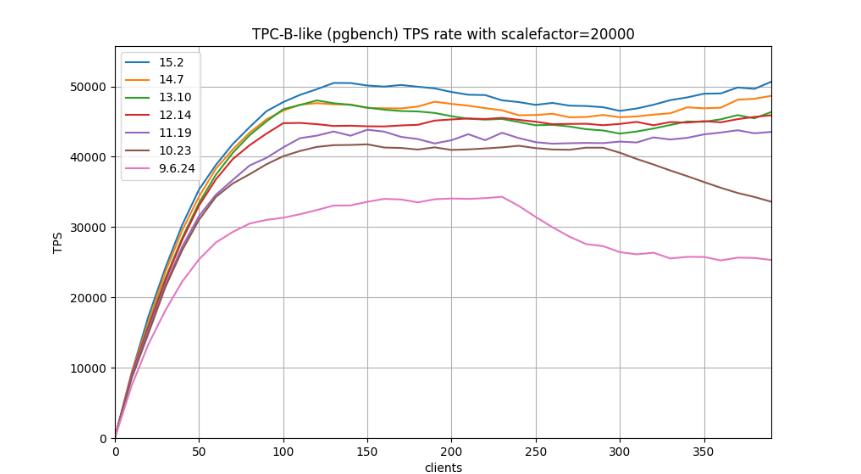
Т.к. компания занимается продажей товаров, то практический весь объем данных поддается структурированию, и в этом направлении деятельности необходимо в обязательном порядке обеспечить транзакционную целостность данных. Из этого следует, что лучшем решением будет реляционная БД.

Т.к. компания небольшая то топовые по цене решения в виде MS SQl или Oracle не подходят. Остается лучший вариант - PostgreSQL, т.к. он не проигрывает по функционалу.

Также в пользу бесплатной версии играет тот факт, что для реализации сложной аналитики, без существенного влияния на нагрузку системы стоит вынести ее на отдельную СУБД с настройкой репликации, а в этом случае не придется покупать две лицензии.

Рассмотрим PostgreSQL на требования по **объему данных и масштабируемости** – исходя из постановке по объему следует, что в день создается миллионы записей. Т.е. не десятки и не сотни миллионов (что логично т.к. мы помним что компания небольшая). Т.к. сфера e-commerce можно предположить, что в данные о продажах – это основная таблица по количеству записей, и если сложить со всеми остальными данными то выйдет не больше **20 Млн. записей в день**. По требованию масштабируемости планируется рост, предположим что за 10 лет, произойдет кратный рост, следовательно объем увеличится до **200 Млн. записей в день**

Согласно [текущим данным](https://www.enterprisedb.com/blog/performance-comparison-major-PostgreSQL-versions) PostgreSQL последней версии поддерживает около **50к TPS** (transaction per second)



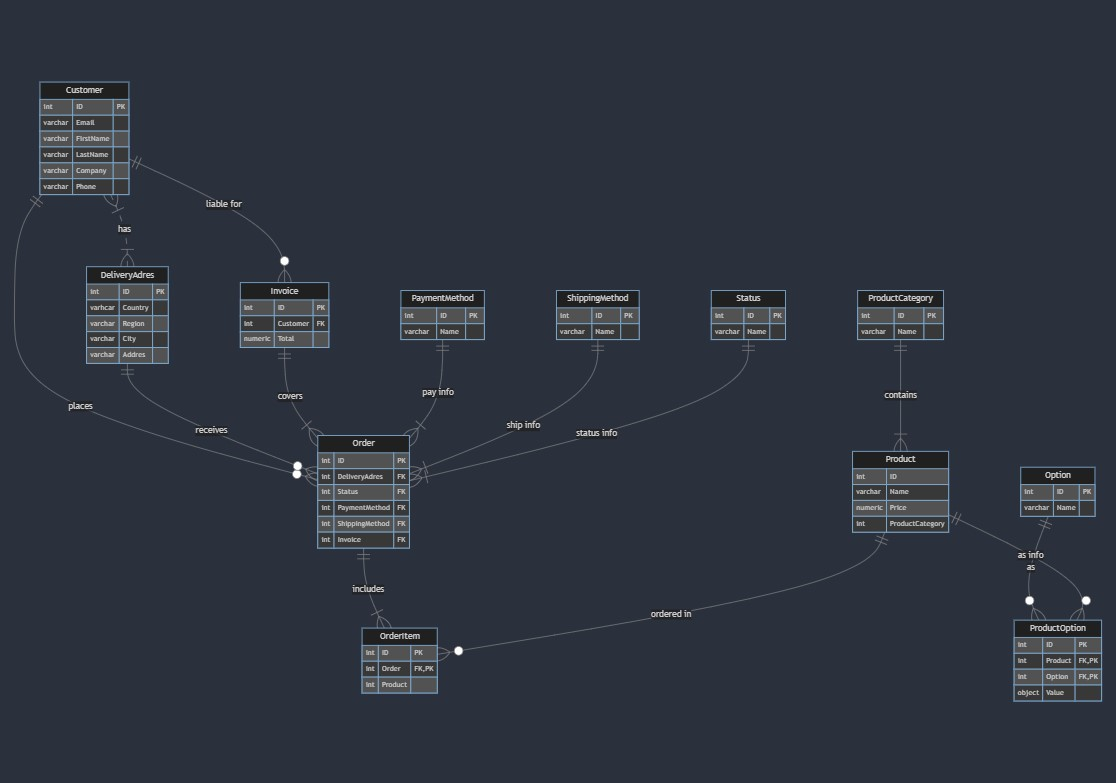
А это 50к\*86400 =~4.3 миллиарда записей в день уже сейчас (а как видно из графика от версии к версии идет прирост по TPS).

Что касается **объема** то если взять среднюю по размеру таблицу в 10 млн. записей, она будет занимать порядка 300 Мб. Текущие 20 Млн. записей в день будут занимать 300 \* 2 \* 365 =~**220 Гб** в год, и ~**2.5 Тб** в год через 10 лет, что является несущественным даже сейчас.

NoSQL БД Redis – можно использовать для хранения кеша, для уменьшения количество чтений в основную БД, т.к. как писал ранее в сфере e-commerce есть немало НСИ (нормативно-справочной информации) которая достаточно редко обновляется, и ее можно без особых проблем кешировать.

Что касается **требования по реальному времени** – применение специализированных БД реального времени (БДРВ) не целесообразно, т.к. основным критерием для их выбора является большое количество изменений уже существующих записей, что для текущей сферы не применимо, т.к. основное количество транзакций приходится на вставку. И с этой задачей отлично справляется классическая СУБД.

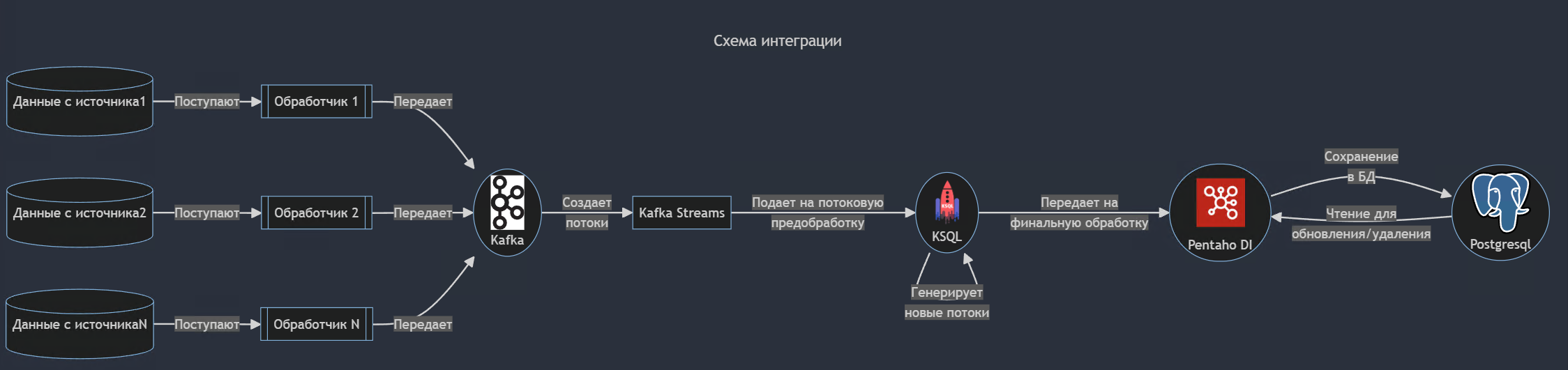
1. Примерная структура БД

[](https://mermaid.live/edit#pako:eNqdVcFuozAQ_RXL1ybV9ppblTRS1G0bbVd7WHFx7UnwFmxkTCQU-PcdwA4xkKYKByRm3rwZzzzGR8q1ALqgYFaS7Q1LI0XwWRa51SkYUlf399WRrCCRBzDlozCQkwWJWT4AVtV8ro_kzQj8WJAsYRwuYDbqoCUHREU0kewjAbLTJqIdOkw1oDXAAb2O2BM1oKoHcY3xDtLZzgEbCymCpOJJITzT1mhRcLtkFvbalD7AmVtOZZlUITyozvFGVDdfIDCDP9KWlSko-wI21oI8zOcPd6diI5qxErE77dHvscwyqfYX4Dm6Q7xltsjHuM58jnzLrNSqQf6482dwtiYCZ0rD811GBrynAR-77-aRypLNimyfe9OBGR4zQ55SJpOxeS1Nbl9ZCmPXT3bJs9RpxlQ5dmxjrRy-PpfC1yU2plCA64HXdXtoDkc8CgpHOnR7HXt7PSWb7_W271I9qaZbWbpT3xgddvQqSczbwRbKmonB_oI9ynBCCdJOoB9FkzOoxnf7uhJOuj4fmCrQIjn5rS1LxgJr18B16k6M6-fZ0OF-tlAI7ncc0X7V-_Nat_iCyTx-403lO23DG8fuFsaN0eHWud5R36SJnjqOgUd__APE_2FJ0WemM4otwwUl8FJsk0bUxoC10WbxCWY-m61XI44VVr-XitOFNQXMaJEJbJi7Rulix5IcrSCk1ealu2Xby3aGG1_91brHdJFPLdIZ6__Zs01m)

Ссылка на схему

<https://mermaid.live/edit#pako:eNqdVcFuozAQ_RXL1ybV9ppblTRS1G0bbVd7WHFx7UnwFmxkTCQU-PcdwA4xkKYKByRm3rwZzzzGR8q1ALqgYFaS7Q1LI0XwWRa51SkYUlf399WRrCCRBzDlozCQkwWJWT4AVtV8ro_kzQj8WJAsYRwuYDbqoCUHREU0kewjAbLTJqIdOkw1oDXAAb2O2BM1oKoHcY3xDtLZzgEbCymCpOJJITzT1mhRcLtkFvbalD7AmVtOZZlUITyozvFGVDdfIDCDP9KWlSko-wI21oI8zOcPd6diI5qxErE77dHvscwyqfYX4Dm6Q7xltsjHuM58jnzLrNSqQf6482dwtiYCZ0rD811GBrynAR-77-aRypLNimyfe9OBGR4zQ55SJpOxeS1Nbl9ZCmPXT3bJs9RpxlQ5dmxjrRy-PpfC1yU2plCA64HXdXtoDkc8CgpHOnR7HXt7PSWb7_W271I9qaZbWbpT3xgddvQqSczbwRbKmonB_oI9ynBCCdJOoB9FkzOoxnf7uhJOuj4fmCrQIjn5rS1LxgJr18B16k6M6-fZ0OF-tlAI7ncc0X7V-_Nat_iCyTx-403lO23DG8fuFsaN0eHWud5R36SJnjqOgUd__APE_2FJ0WemM4otwwUl8FJsk0bUxoC10WbxCWY-m61XI44VVr-XitOFNQXMaJEJbJi7Rulix5IcrSCk1ealu2Xby3aGG1_91brHdJFPLdIZ6__Zs01m>

1. Схема интеграции

[](https://mermaid.live/edit#pako:eNqllUFv0zAUx79KlEs3aYnTCCQoDAlRDhOjAnajqSYvcduoSRwSl2iaJm0dYwfQJsTOoJ24boOKqqPlK9jfiGcnbaHqtElckjzn_37vb_s52dFd6hG9ohuG4UTMZwGpaPxMHPI-_8XPNT7gI9GD4LvY4-fiPcQDJ1LiZkAzt40Tpq2_cqIqZrisGYbm6PwrH4t90RMH_DfkHIueo8ObR9oTGpXrdRB84RcSxy9A2BNHAB1qZUdvNHKOfTPHvpZjzzi1mzm1azm1nCM9g1hR-iDq8x8g7EuGRDzDzQ5WKvtWqtrNKnXLZWd8zH9ORU4E_sEgDA5hF2Y5GywhOEyL1CKaFBpP8jU-khs6Q4z5pTgQx8CF0cLO-O-V4ENxUJTZeLkOeLjm2M8gHqk5DMQerGthb6SYH3hfW-B0DjG3BModMMQ72XIwdiU-QuuBP-06Uy-qa04El-laiUMlGylzA96Xli41_omfFgk0Za2EpG8CyJs-q-xvqslVlgaWrsSJTIbC-Zz4Vf5WnCCYrbQ8HVBoZUUdgfoSP1UeRmodxD4cIdl84O5I8Yf8vLw8afTbqO2JunYbdU2qVR8s1R_6YUtLE3e11GYsTisIdeOAYs_M_I4fEs_HJk1aSEaxjJBLw5BGKbKQdRc9jrHbJpsdyTLTt62SlrLtgKw6-hZ2O62EdiPPcGlAk0rW9hl5AKcq8z3WXi3dsUoaKpqzsfxvX9bz_i6iRt4Si7z6lpnFJlhCWZaZoe8mNIXPDTF9irIYKkeMRKyYUopsq3wPlS0kccY6bVGjRjIzjsD3nCsQSFOwZ4vK4hb2A6i6lZhNPyCpmdHEi6FTUuXFtmwLlW0Ue_4i-AuwhNtUq66pEtMm-4_NsJF9H81ImyQgcRtHLN-TufJTmSyvr-ghSULse_CB34FjDh9D1iYhcfQKPHo46Ti6E-2CDncZ3diOXL3Cki5Z0buxhxmp-riV4FCvNHGQwij4YjR5nv8x1I9jRY9x9J)

Ссылка на схему

<https://mermaid.live/edit#pako:eNqllUFv0zAUx79KlEs3aYnTCCQoDAlRDhOjAnajqSYvcduoSRwSl2iaJm0dYwfQJsTOoJ24boOKqqPlK9jfiGcnbaHqtElckjzn_37vb_s52dFd6hG9ohuG4UTMZwGpaPxMHPI-_8XPNT7gI9GD4LvY4-fiPcQDJ1LiZkAzt40Tpq2_cqIqZrisGYbm6PwrH4t90RMH_DfkHIueo8ObR9oTGpXrdRB84RcSxy9A2BNHAB1qZUdvNHKOfTPHvpZjzzi1mzm1azm1nCM9g1hR-iDq8x8g7EuGRDzDzQ5WKvtWqtrNKnXLZWd8zH9ORU4E_sEgDA5hF2Y5GywhOEyL1CKaFBpP8jU-khs6Q4z5pTgQx8CF0cLO-O-V4ENxUJTZeLkOeLjm2M8gHqk5DMQerGthb6SYH3hfW-B0DjG3BModMMQ72XIwdiU-QuuBP-06Uy-qa04El-laiUMlGylzA96Xli41_omfFgk0Za2EpG8CyJs-q-xvqslVlgaWrsSJTIbC-Zz4Vf5WnCCYrbQ8HVBoZUUdgfoSP1UeRmodxD4cIdl84O5I8Yf8vLw8afTbqO2JunYbdU2qVR8s1R_6YUtLE3e11GYsTisIdeOAYs_M_I4fEs_HJk1aSEaxjJBLw5BGKbKQdRc9jrHbJpsdyTLTt62SlrLtgKw6-hZ2O62EdiPPcGlAk0rW9hl5AKcq8z3WXi3dsUoaKpqzsfxvX9bz_i6iRt4Si7z6lpnFJlhCWZaZoe8mNIXPDTF9irIYKkeMRKyYUopsq3wPlS0kccY6bVGjRjIzjsD3nCsQSFOwZ4vK4hb2A6i6lZhNPyCpmdHEi6FTUuXFtmwLlW0Ue_4i-AuwhNtUq66pEtMm-4_NsJF9H81ImyQgcRtHLN-TufJTmSyvr-ghSULse_CB34FjDh9D1iYhcfQKPHo46Ti6E-2CDncZ3diOXL3Cki5Z0buxhxmp-riV4FCvNHGQwij4YjR5nv8x1I9jRY9x9JrSmSbPfKqUxeDuH55R8m4>

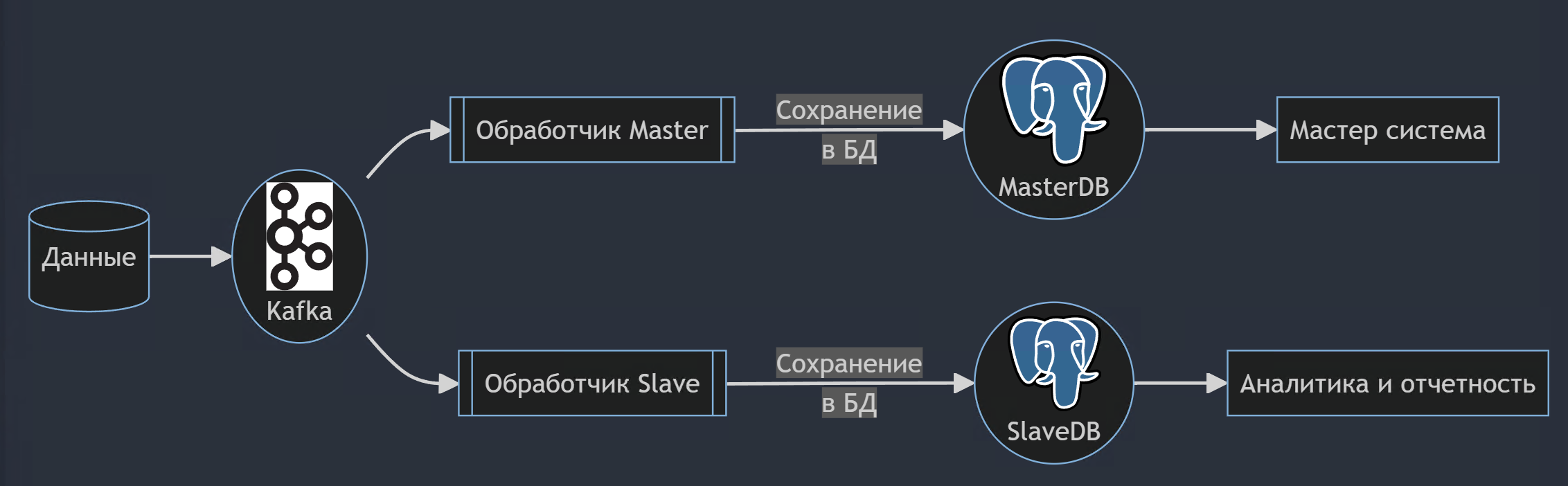
1. Настройка системы для обработки данных в реальном времени

В первую очередь, для максимального улучшения отклика системы, необходимо все задачи которые можно выполнять на отдельных инстансах БД – нужно туда выносить. К этому относится различная аналитика (в том числе сложная), не оперативная отчетность и т.п.

При этом можно использовать не только репликацию:



Но и дублирование потоков



Что касается реального времени, как было сказано выше – исходя из текучей/прогнозируемой нагрузки и объемов, это не является СРВ (система реального времени), тут достаточно требовательная по отклику система.