

Базы данных

А2. Принципы управления БД



Московский государственный технический университет
имени Н.Э. Баумана

Факультет ИБМ

Фев 2025 года

Москва

Артемьев Валерий Иванович © 2025

Основные принципы управления БД

- **Независимость данных**

помогает изменить схему БД на одном уровне системы базы данных без необходимости изменения схемы на более высоком уровне.

- **Эффективное выполнение операций с данными**

обеспечивает быстрый поиск, чтение, запись и удаление данных

- **Поддержка целостности данных**

чтобы данные и их изменения не нарушали работу с БД, она должна содержать сведения о правилах, а СУБД обеспечивать контроль их выполнения.

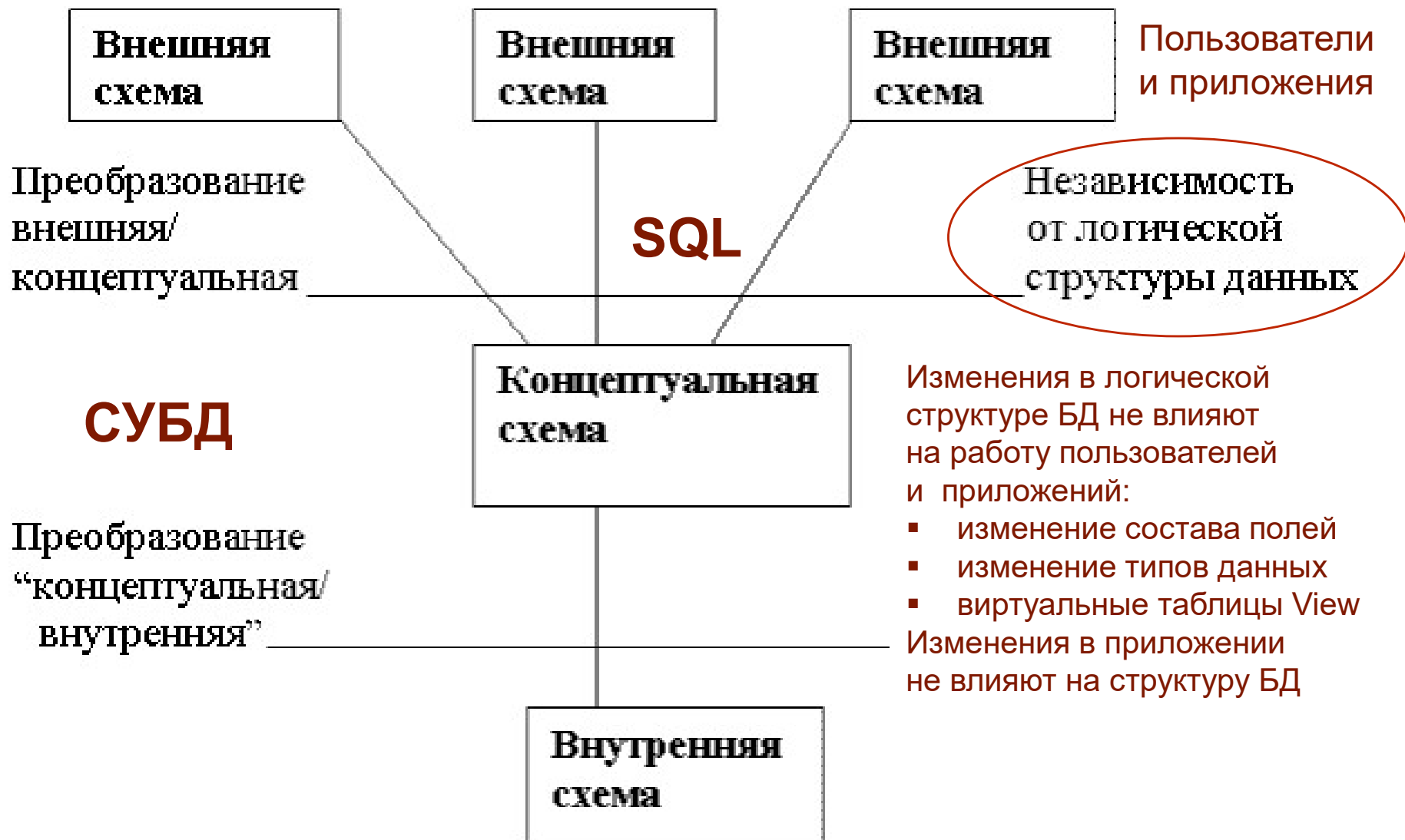
- **Стандартизация**

база данных не должна менять свою структуру или свойства при обновлении приложения или СУБД.

Независимость данных в архитектуре ANSI/SPARC для БД



Логическая независимость данных



Физическая независимость данных



CRUD – основные операции с БД



Create

Создание новой записи с заполнением атрибутов в существующей таблице БД. Создание БД, её объектов и элементов данных отражается в словаре-справочнике данных. СУБД должна выполнять и контролировать правильность всех операций создания.

Read Retrieval

Поиск и чтение данных и метаданных данных. БД должна быть определённым образом структурирована и записи уникально идентифицированы. СУБД должна обеспечивать выборочное чтение данных и метаданных.

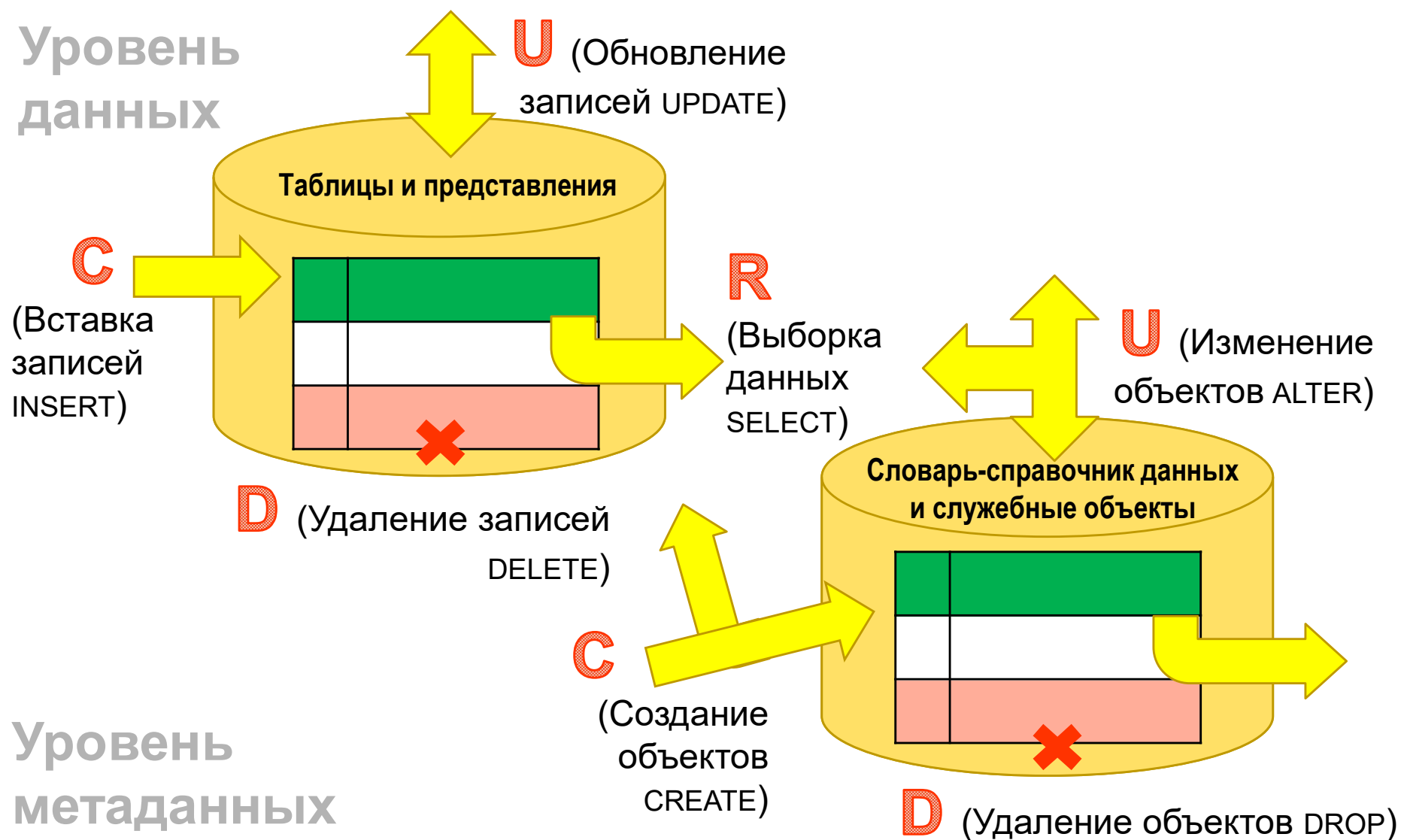
Update

Обновление данных в БД и изменение структуры, правил целостности, соответствующее обновление метаданных в словаре-справочнике данных. СУБД должна обеспечивать выборочное обновление данных и метаданных.

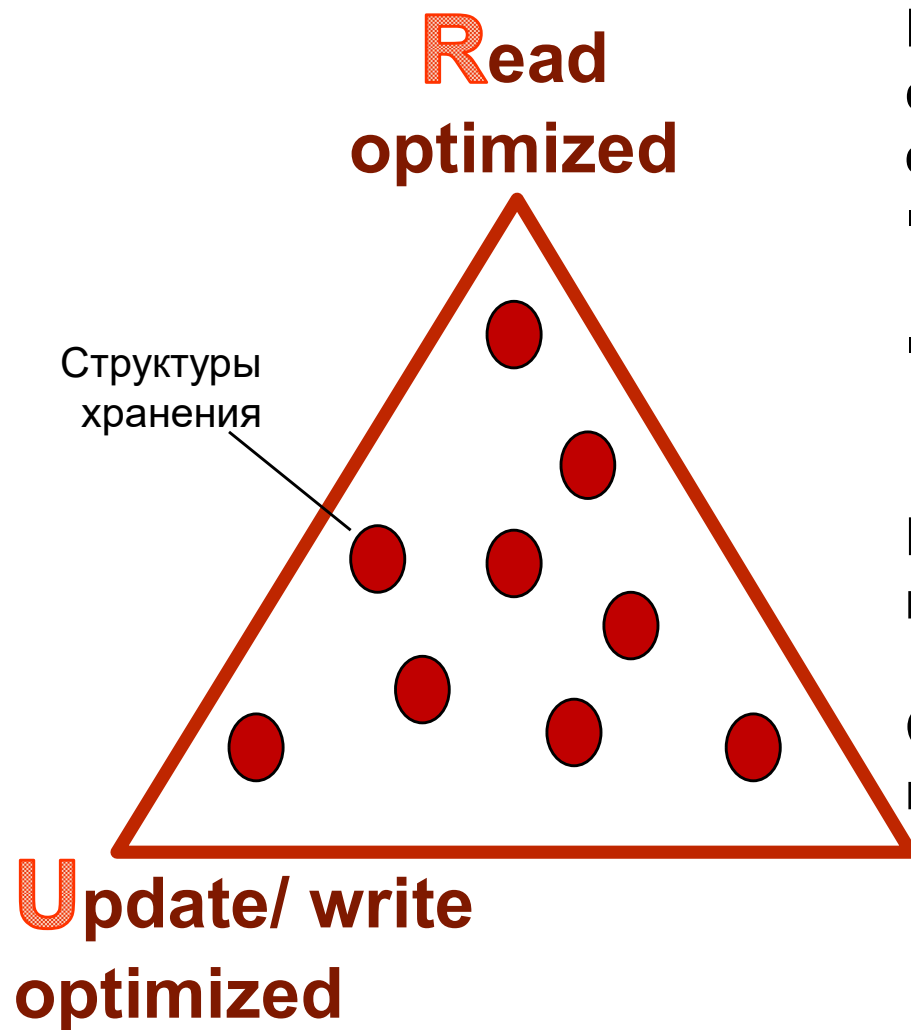
Delete

Удаление выбранных записей из БД, возможно каскадное удаление записей из зависимых таблиц, а также удаление БД, объектов и элементов данных. СУБД должна обеспечивать выборочное удаление данных и метаданных.

Операции с данными/ метаданными



Компромисс эффективности операций RUM conjecture



На практике пока не удалось создать структуру хранения данных, чтобы одновременно:

- эффективно выполнялись чтение и запись (обновление),
- данные занимали пространство, только необходимое для их хранения.

Поэтому СУБД предоставляют несколько структур хранения данных.

Структуры хранения оцениваются мультипликаторами сложности.

Использование дисковой и основной памяти

Изменяемые структуры хранения

Все традиционные реляционные СУБД

Страничная организация доступа к диску (2-64 Кбайта)
«Сырое» блочное устройство (**Oracle, IBM Db2**)
или файловая система (**MS SQL Server**).

Уменьшают время чтения/записи при увеличении памяти:

- кэширование страниц или блоков
- резерв для расширений
- фрагментация блока, страницы и экстента
- метаданные для управления страницами.

Увеличивают время операций:

- чтение с диска страницами
- применение файлов вместо «сырых» блочных устройств.

Уменьшают время поиска и увеличивают расход памяти индексы.

Увеличение памяти и времени операций записи при журналировании.

Балансировать эффективность использования памяти и выполнения операций позволяет сжатие и распределение данных по частям.

Неизменяемые структуры хранения

Нереляционные СУБД NoSQL

Как правило, файлы (**MongoDB, Cassandra, Riak KV** и др.).

На первый взгляд эффективная запись – записывают ровно столько, сколько занимают изменённые объекты.

Изменчивость *увеличивает использование памяти* за счёт копий данных.

Увеличивают время операций

- применение файловой системы
- фоновые процессы очистки неактуальных копий.

Уменьшают время операций и увеличивают расход памяти

- применение индексов
- дублирование и параллельная обработка данных.

Увеличение памяти и времени операций записи при журналировании.

Балансировать эффективность использования памяти и выполнения операций позволяет выбор подходящего формата и распределение данных по частям.

Ограничения целостности БД

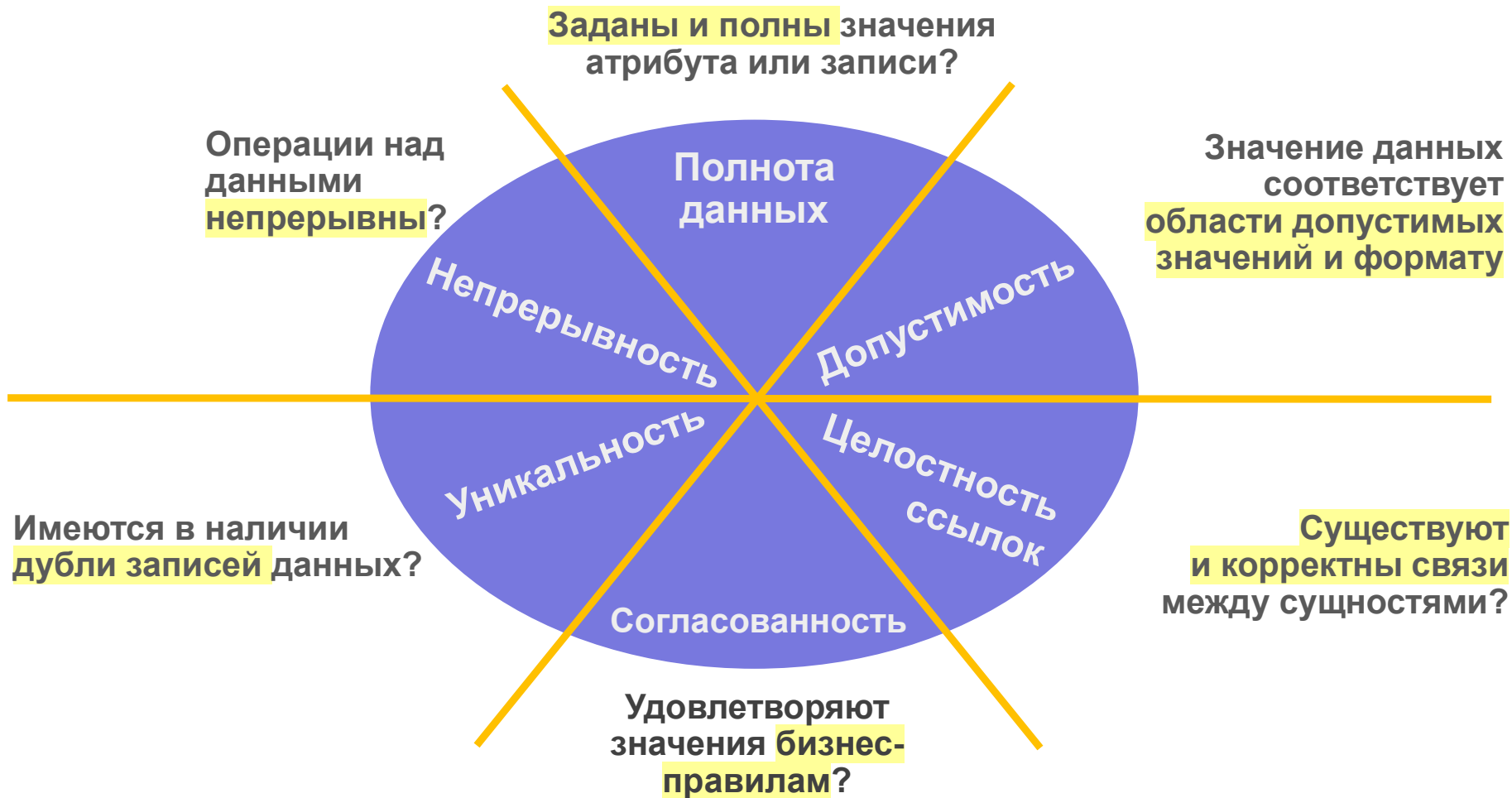


Накладываемые на элементы данных условия, ложные значения которых говорят о нарушении целостности состояния базы данных и обрабатываются определённым образом.

Ограничения целостности определяются

- *содержанием данных,*
- *структурой данных,*
- *смыслом данных и бизнес-правилами.*

Требования к целостности БД



Непрерывность операций



Обеспечение надежности оборудования

- Резервирование серверов БД
 - Резервирование каналов связи
 - Резервирование и восстановление дисков в СХД
-

Обеспечение непрерывности в СУБД

- Журналирование операций
 - Обеспечение целостности локальных и распределённых транзакций*
 - Создание контрольных точек
 - Теневое копирование данных (снимки данных)
 - Распределение (дублирование) данных
-

Организационно-технические мероприятия

Резервное копирование и восстановление данных*

Хранение резервных копий на удалённой площадке

Репликация данных на резервную площадку

Репликация данных в облако

Аварийное развёртывание резервной площадки

Обеспечение целостности транзакций

Транзакция – неделимая последовательность операторов манипулирования данными (чтения, удаления, вставки, модификации), приводящая к одному из двух возможных результатов:

- либо последовательность выполняется, если все операторы правильные и не было сбоя, и фиксируются результаты **COMMIT**
- либо вся транзакция откатывается **ROLLBACK**, если хотя бы один оператор не может быть успешно выполнен.

Явно определяются границы транзакции.

Возможно создание контрольных точек и откат к последней контрольной точке.

ACID – поддержка транзакций в СУБД

Atomicity

Атомарность транзакций – операции транзакции должны быть выполнены все или ни одна из них.

Consistency

Согласованность – транзакция гарантирует, что БД до и после транзакции находится в согласованном состоянии, т.е. не нарушены ограничения целостности данных.

Isolation

Изолированность – транзакция изолирует детали её операций от всех, кроме инициатора. Транзакции выполняются изолированно и не мешают друг другу.

Durability

Долговечность – результаты транзакции не исчезнут после её совершения. При отключении электричества результаты транзакции после перезагрузки системы БД сохраняются.

Изоляция транзакций

В *многопользовательском режиме при параллельном выполнении транзакций* СУБД создаёт у каждой транзакции иллюзию, что она единственная в этот момент времени.

Степень иллюзий называется **уровнем изоляции** (isolation level).

Наивысший уровень изоляции – *сериализация* – соответствует эффекту *последовательного выполнения*, наиболее затратный режим для СУБД.

Существуют *более слабые уровни изоляции*, при которых могут возникать *аномалии*.

Стандартом **ANSI SQL 92** определены 4 уровня изоляции:

- read uncommitted
- read committed
- repeatable read
- serializable

Аномалии стандартных уровней изоляции

Уровень изоляции	Черновое чтение	Неповторяемое чтение	Фантомы
Read Uncommitted – чтение незавершённых транзакций	да	да	да
Read Committed – чтение завершённых транзакций	нет	да	да
Repeatable Read – повторяемое чтение	нет	нет	да
Serializable – последовательное чтение	нет	нет	нет

Грязное (черновое) чтение – транзакция читает данные, которых никогда не было в БД.

Неповторяемое чтение – транзакция повторно читает одни и те же строки, получая разные результаты.

Фантомное чтение (фантом) – транзакция при одном и том же запросе получает разные наборы строк.

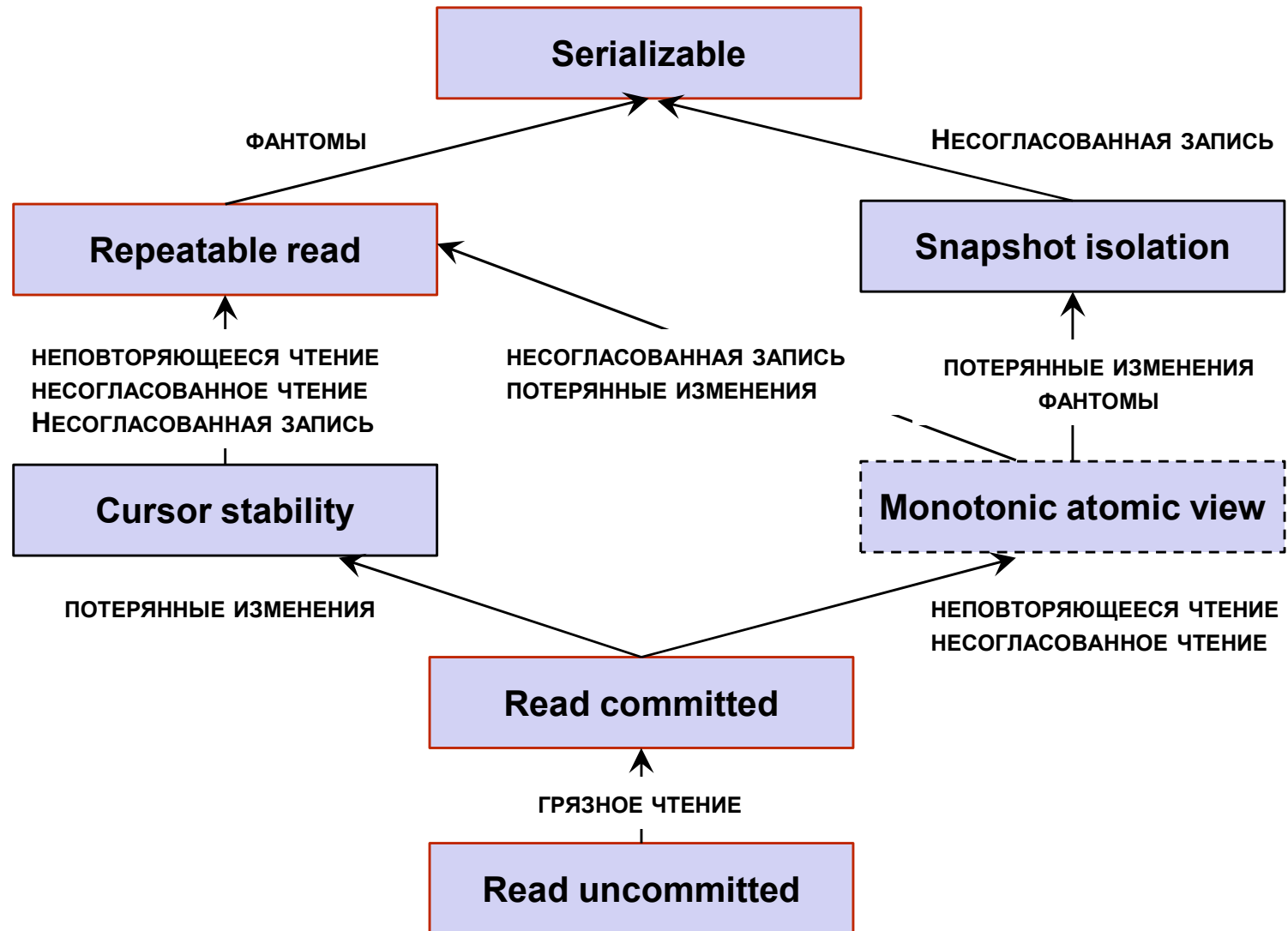
Уровни изоляции, поддерживаемые известными СУБД

СУБД	Read uncommitted	Read committed	Repeatable read	Serialized
MS SQL, PostgreSQL, MySQL	+	+	+	+
IBM Db2	+		+	+
Oracle		+		+

Реализуются с помощью *блокировок данных* и *меток версий* данных.

Не все СУБД поддерживают рекомендованные стандартные уровни изоляции.

Иерархия уровней изоляции транзакций в СУБД



Средства реляционных СУБД для контроля целостности базы данных

1. *Декларативные ограничения целостности* для атрибутов, ключей и записей, а также *утверждения*.
2. *Механизм транзакций*
3. *Триггеры* – процедуры событийной проверки данных.
4. *Хранимые процедуры* обеспечивают сложные алгоритмы контроля данных.

Организационно-технические мероприятия обеспечения непрерывности бизнеса



Резервное копирование и восстановление данных*
Хранение резервных копий на удалённой площадке

Холодный и горячий резерв

Репликация данных на резервную площадку

Репликация данных в облако

Аварийное развёртывание резервной площадки

Резервирование и архивирование данных

ОБЯЗАТЕЛЬНО

ТЕСТИРОВАНИЕ
ДОСТУПНОСТИ И
ВОССТАНОВЛЕНИЯ



РЕЗЕРВИРОВАНИЕ
СИСТЕМЫ



Точки восстановления

ЖЕЛАТЕЛЬНО



Образы системы

РЕЗЕРВИРОВАНИЕ
ДАННЫХ



Еженедельное полное резервирование + ежедневное дифференциальное или инкрементное



Наиболее ценные данные — дополнительно в «облако»

АРХИВЫ
ДАННЫХ

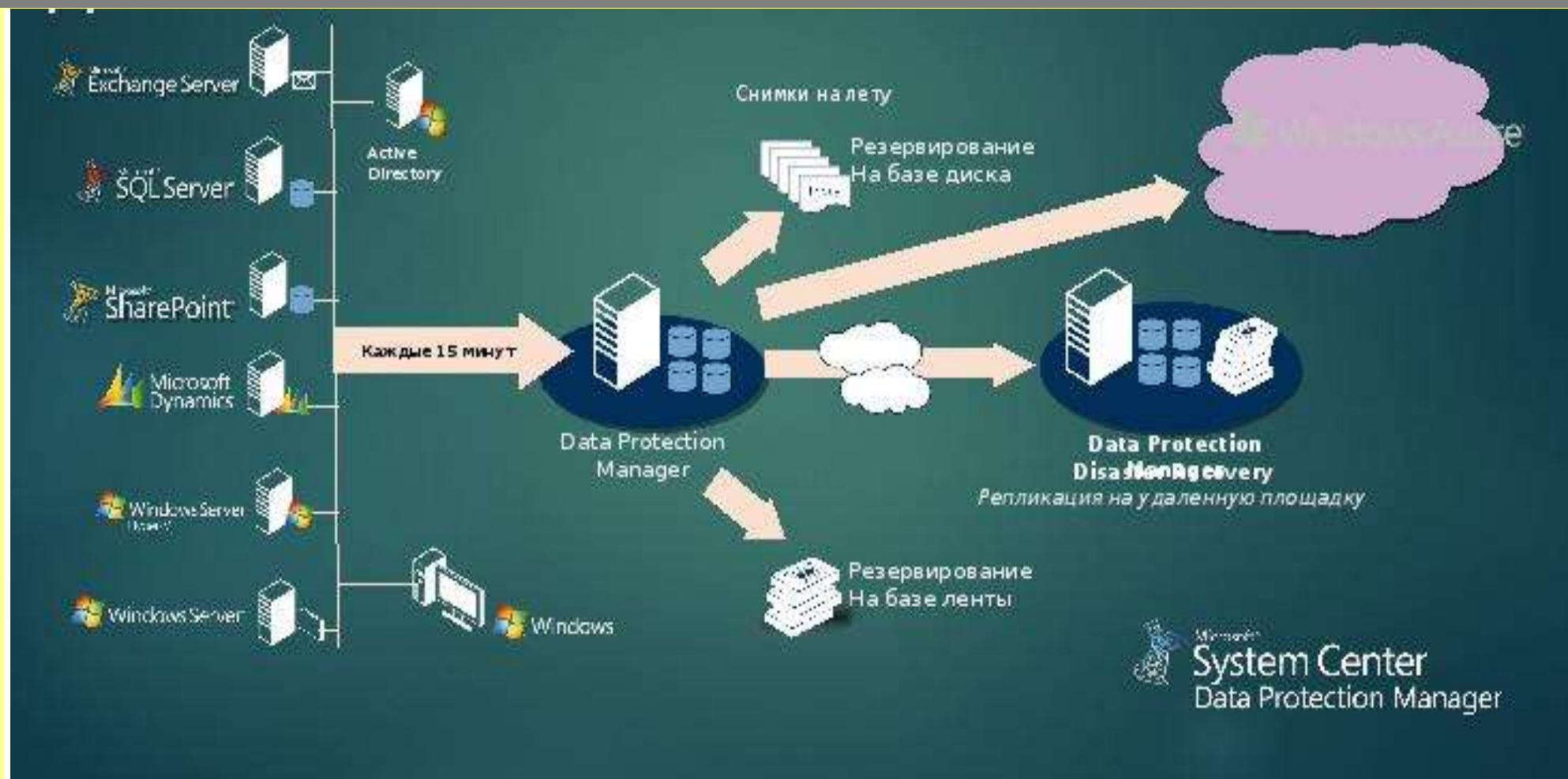


Копирование на качественные DVD-носители, перенос на новые раз в пять лет



Наиболее ценные данные — дополнительно в «облако»

Регулярное резервирование данных





Проблемы масштабируемости систем БД

С ростом БД и количества её пользователей *увеличивается время реакции* на запросы пользователей.

Вертикальное масштабирование – увеличение мощности технических средств сервера БД. Нелинейно растёт стоимость оборудования.

Горизонтальное масштабирование – распределение данных в сети и организация параллельной обработки.

- *репликация* – дублирование данных на серверах
- *шардинг* – распределение данных между серверами

Теорема CAP для распределённых СУБД (Э. Брюер)

Система управления *распределёнными данными* может обеспечить надёжно только два из следующих условий:

Consistency

Согласованность – при каждом считывании данных выводятся последние записанные данные или сообщение об ошибке.

Availability

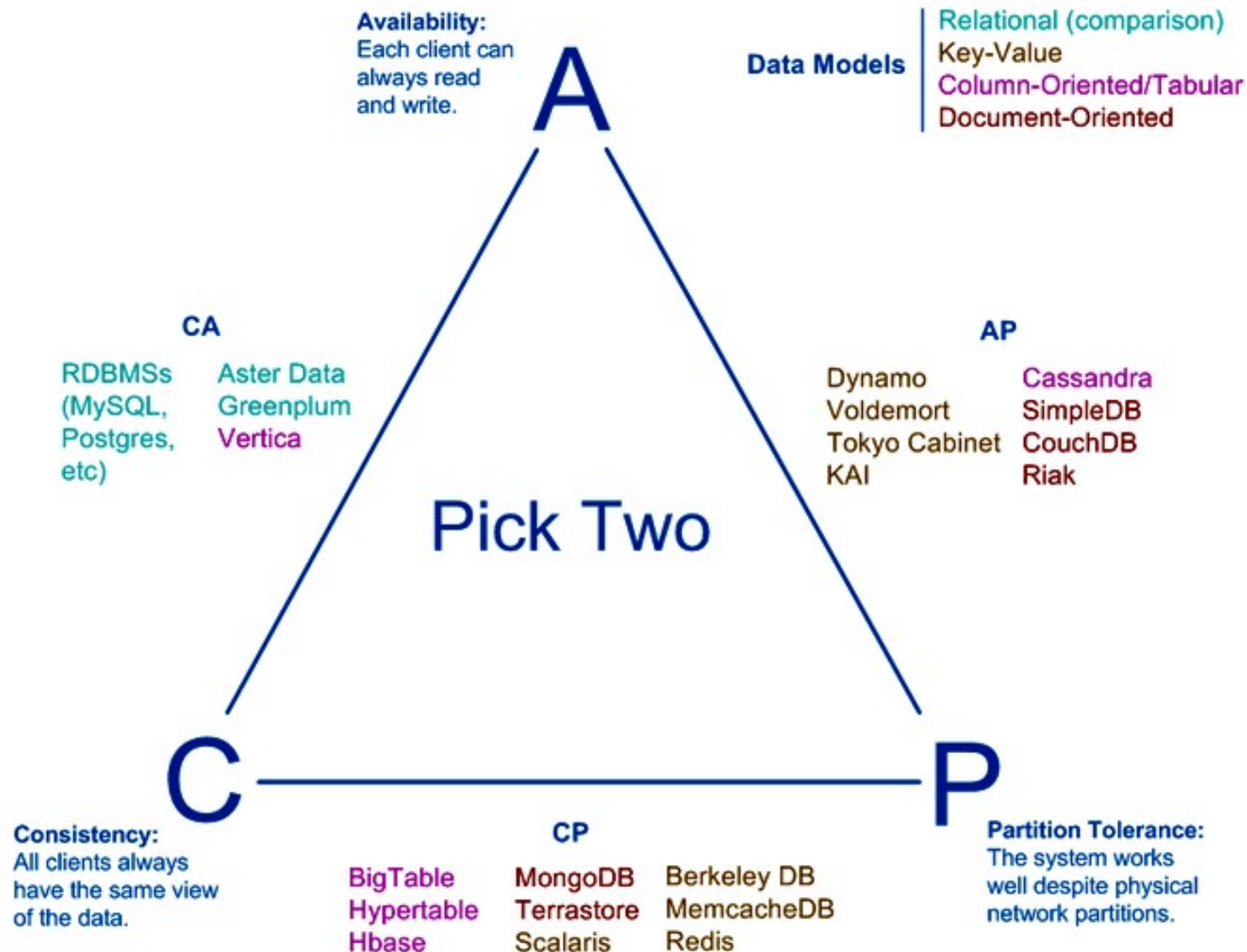
Доступность – при считывании данных выводится сообщение об ошибке, если не гарантируется выдача последних записанных данных.

Partition tolerance

Устойчивость к разделению – СУБД продолжает работать, даже если сообщения отбрасываются или задерживаются сетью при передаче между разделами (узлами).

При централизованной СУБД можно добиться условий CA

Распределённые СУБД



BASE – свойства СУБД NoSQL

Basically **A**vailable

Базовая доступность – данные при распределённом хранении могут быть не согласованы сразу, но на любой запрос должен быть результат или отказ.

Soft state

Неустойчивое состояние – нужно время для прохождения изменений по всей системе. Эти СУБД не обеспечивают согласованность, полагаются на разработчика.

Eventually **c**onsistent

Согласованность в конечном итоге – базы данные в конце концов станут согласованы, но это обязательно произойдёт до следующего чтения данных.

Безопасность баз данных



Физическое разграничение секретных систем

Ограничение размещения и распространения персональных и конфиденциальных данных

Аутентификация по паролю, аппаратному ключу

Авторизация – определение прав доступа к данным и выполнения операций:

- Ролевые (групповые) права доступа RBAC
- Создание представлений View
- Метки безопасности Security Level
- Права доступа к атрибутам ABAC

Шифрование данных

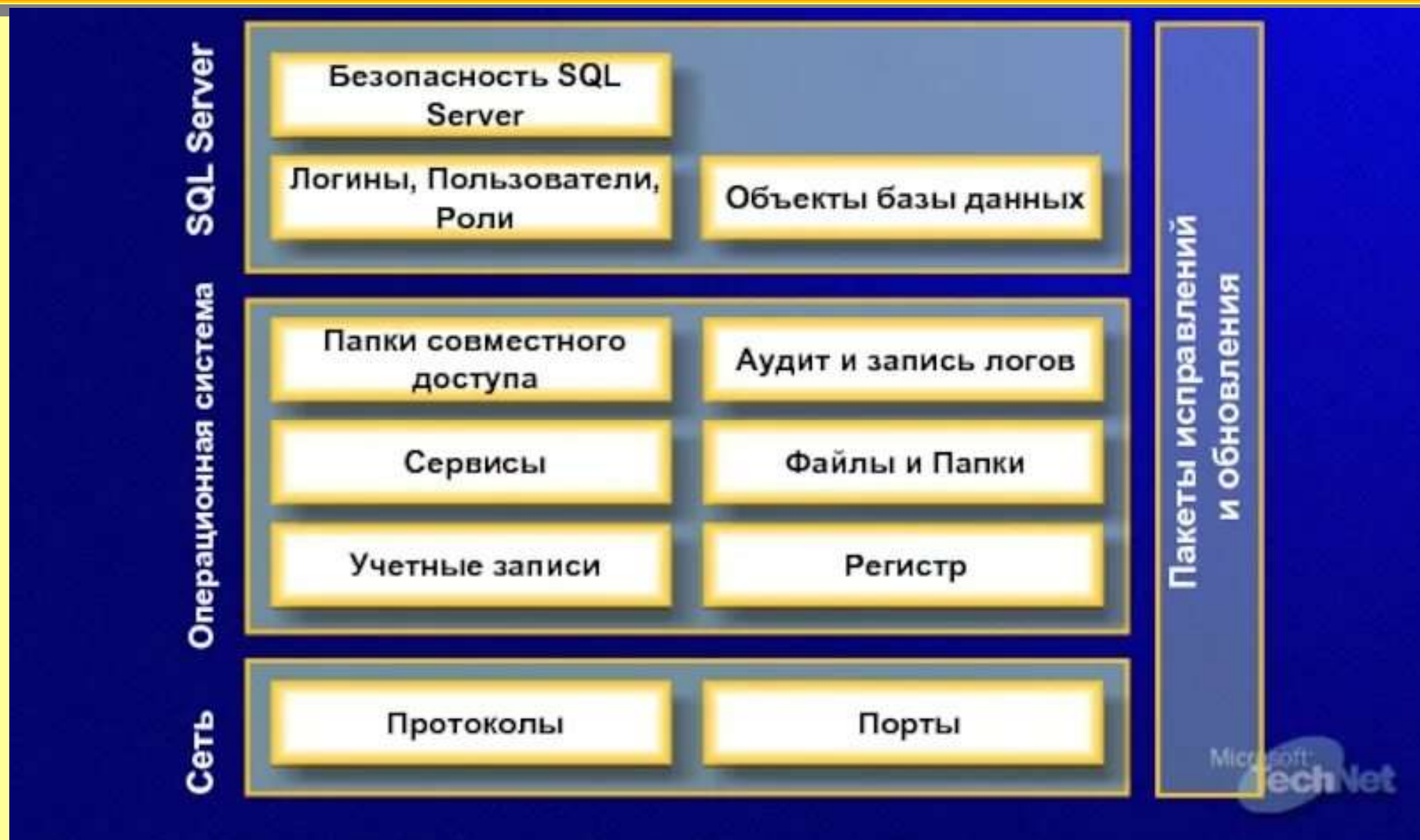
Мониторинг атак

Контроль уязвимостей и управление рисками

Мониторинг активностей

Маскирование и анонимизация данных

Объекты защиты системы баз данных



**Терпения и удачи всем, кто
связан с базами данных**

Спасибо за внимание!

Валерий Иванович Артемьев

МГТУ имени Н.Э. Баумана, кафедра ИУ-5

Банк России

Департамент данных, проектов и процессов

Тел.: +7(495) 753-96-25

e-mail: viart@bmstu.ru