

Состояние моделей и инструментов управления данными с учетом многозадачности среды предприятий

Марк Ривкин

Начальник отдела технического консалтинга,
Postgres Professional

Вступление

- Теоретический и практический подход к моделям данных. Архитекторы больше работают с моделями данных, у Вас более глубокий теоретический подход
- Архитекторы vs практиков (я иду от практики, возможно это не строго)
- Типы данных/модели данных/типы нагрузок – все взаимосвязано
- Поэтому будем говорить обо всем как о модели и о типах СУБД

Почему существует множество моделей, типов данных, СУБД

- Вначале были иерархическая, сетевая, реляционная модели и одномодельные СУБД
- Появились неструктурированные данные
- Универсальные СУБД
- Одна модель и ее типы данных сужают круг решаемых задач
- Бизнес требует все больше новых данных, нагрузок, парадигм разработки

Примеры задач с одновременным использованием разных типов данных

- Инвалиды в затопляемых домах – МЧС
- ЦКАД – инфо об объектах, координаты, карты и планы, изображения, документы
- Строительство – видео с беспилотника, аудио совещаний и т д (снижаем риски)
- Движок для разных ГИС

Предположим, консультант-проектировщик БД приходит на предприятие и ...

Обнаруживает, что все данные предприятия хранятся в трёх частных БД, условно с моделями данных и приложениями **JSON**, **Geo** и **текстовые документы**

В этих БД-1, БД-2 и БД-3 существуют коллизии, которые приводят к появлению **несогласованностей** в данных (нет «однократного ввода», есть **избыточность**, приводящая к появлению разных значений одного поля и др)

Он должен трансформировать ситуацию так, чтобы **без переписывания приложений** интегрировать три частных БД в одну БД0 так, чтобы

- исключить появление указанных выше противоречий в данных
- пользователь мог видеть все данные в одном представлении (как если бы они были в физически одной корректной интегрированной БД)

ВОПРОС: Какой может быть схема архитектуры информации и данных в предположении, что корректное интегрированное описание БД0 делается на едином ЯОД конвергентной СУБД и проектировщику доступны разумные модули расширений языков, адаптеры движков, и др

Какие бывают СУБД: Модели данных

- **Модели данных:**
 - Сетевые, Иерархические → Реляционные (Кодд 1971)
 - Ключ-значение (Redis, Oracle NoSQL, BerkeleyDB, etcd, Amazon DynamoDB, InterSystems, Cache, Tarantool)
 - BigData, Hadoop
 - Поколоночные (Vertica, SAP Hana, Click House, HBase)
 - MOLAP
 - Документно-ориентированные/Json (MongoDB, CouchDB, Google Cloud Firestore, Енисей, YTsaurus)
 - Объектно-ориентированные базы данных
 - NoSQL
 - Базы данных временных рядов (Prometheus, Graphite, TimescaleDB)
 - Графовые базы данных
 - GEO/GIS (геоинформация)
 - RDF (Resource Description Framework) – триплеты объект – предикат – субъект
 - Native XML СУБД
 - Блокчейн БД
 - IoT
 - In-memory СУБД (TimesTen, SolidDB)

Примеры специфических СУБД: Модели данных

- **Модели данных новые:**
 - Поисковые базы данных (Search Engines) (Elasticsearch, Splunk) документ преобразуется в набор закодированных эл-тов индексации
 - Event СУБД (баз данных переходов состояний)
 - Векторные СУБД для нейронных сетей
 - Навигационные (Navigational) СУБД (иерархическая+сетевая)
 - Базы данных массивов
 - Lakehouse – помойка файлов Data Lake с управлялкой (SQL, ACID, статистика)
 - Распределенные шардированные СУБД
 - Семейство столбцов-column-family (BigTable) – Google
 - И т д

Какие бывают СУБД: нагрузки

- **Нагрузки:**
 - OLTP
 - Аналитическая
 - DW
 - IoT
 - Hadoop
 - Поточковая обработка
 - Микросервисы !!!
 - Векторные операции
 - Смешанные нагрузки

Какие бывают СУБД: архитектуры, типы

- **Архитектуры:**
 - Централизованная,
 - RAC
 - Мастер-реплики,
 - Шардинг,
 - Распределенная
 - Data Mesh
- **Типы СУБД:**
 - Специализированные,
 - Нишевые,
 - Коммерческие/универсальные

Проблема использования нескольких специализированных СУБД на одном предприятии

- Выбор СУБД, оптимальной для конкретного типа задач, часто приводит к использованию на одном предприятии нескольких несогласованных инструментов и баз данных
- Это в свою очередь
 - либо приводит к «лоскутной» организации данных предприятия, снижает качества данных и удорожает их администрирование,
 - либо требует дорогостоящей и сложной работы по интеграции таких «лоскутов»
- В качестве альтернативы для обеспечения информационной архитектуры предприятия предлагается в качестве её основной платформы ставить мультимодельную и конвергентную СУБД

Что значит мультимодельная и конвергентная БД ?

- Раньше говорили про универсальная СУБД, т к поддерживает различные типы данных – реляционные, видео, аудио, текст, пространственные, XML и т д
- Теперь кол-во типов данных и нагрузок резко возросло, Oracle Database – **мультимодельная** СУБД
- Мультимодельная СУБД поддерживает:
 - различные форматы и модели данных,
 - различные методы доступа и методы индексирования,
 - различные алгоритмические языки, необходимые для доступа к данным (SQL, XQUERY, SPARQL, DICOM, JSON и т д)
 -
- Поддерживает поколочную in-memory модель и шардинг, работу с документами, пространственной информацией и графами
- Обеспечивает единый механизм администрирования, единую политику безопасности и надежности, транзакционность и согласованность (подробности ниже).

Конвергентная СУБД

Мультимодельная + различные нагрузки (Multi-Workload)

Конвергентная СУБД

Много моделей

**Множество типов данных
(модели и семантика)**

Relational, Document, JSON, XML,
OLAP, Spatial, Graph, Object-
Oriented, Text, etc.



Много нагрузок

**Множество типов приложений
(нагрузка и технологии)**

Operational, Analytics,
Transactional, IoT, ML, In-Memory,
Block-Chain, HTAP, etc.

Конвергентность

Сближение, интеграция

Телефон,
Пейджер,
Фотоаппарат,
Календарь,
Музыка,
Навигатор
Записная книжка
Калькулятор
Фонарь
Часы
...



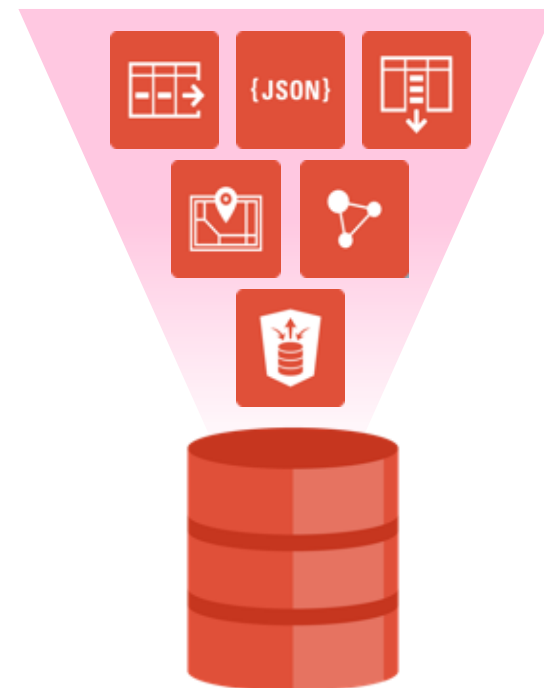
Специализированная vs. Многоцелевая

Вместо

Телефон,
Пейджер,
Фотоаппарат,
Календарь,
Музыка,
Навигатор
Записная
книжка
Калькулятор
Фонарь
Часы
...



Смартфон



Вместо

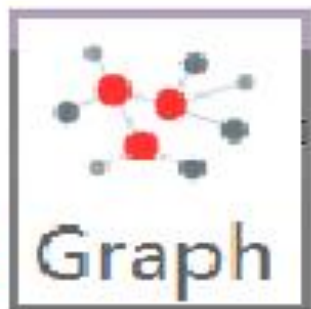
Relational, No-SQL,
JSON, XML, OLTP,
Analytics, DW, In-
Memory, Key-Value,
IoT, ML, Blockchain,
Spatial, Sharding...

Конвергентная СУБД

Разные языки и расширения SQL

- **SQL**
- PL/SQL
- Java
- XQuery – XML
- SPARQL – RDF
- PGQL – графы
- SQL/JSON
- DICOM
- REST
- JSON
- SODA
- DOM
- R

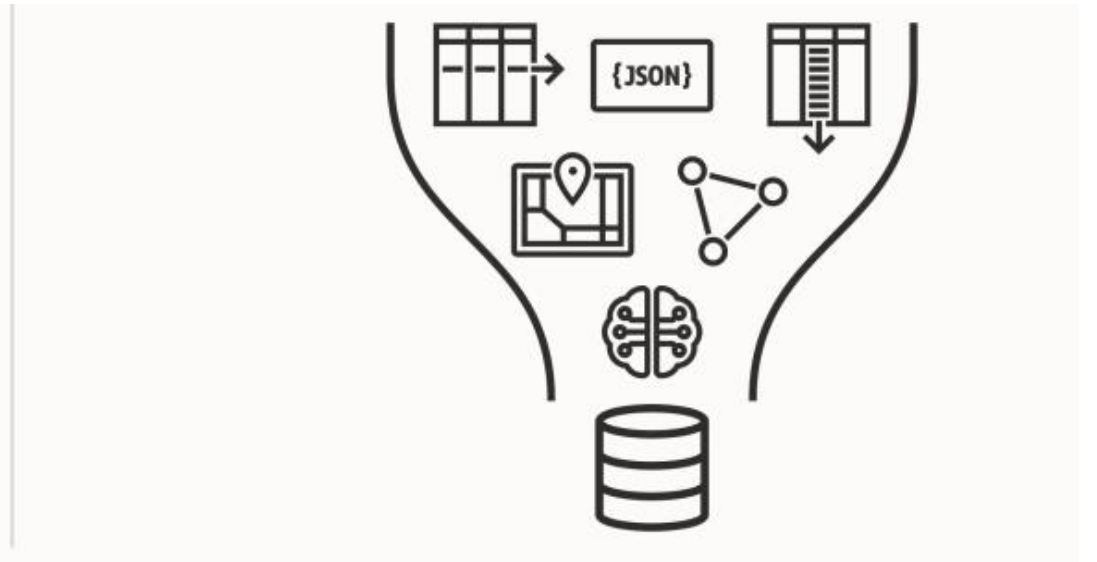
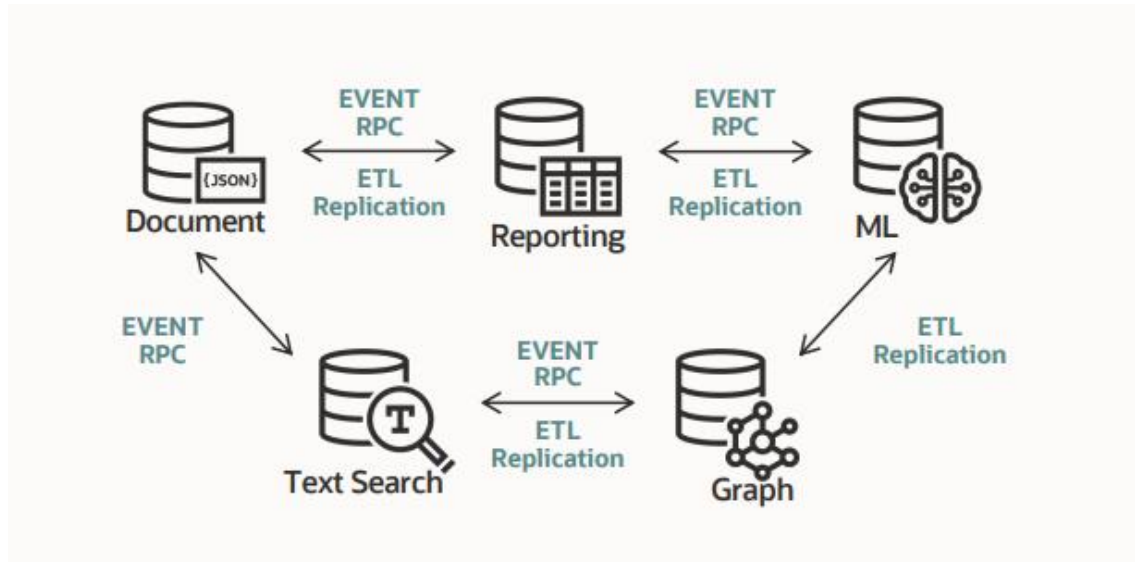
- Graal



Распределенная или конвергентная архитектура ?

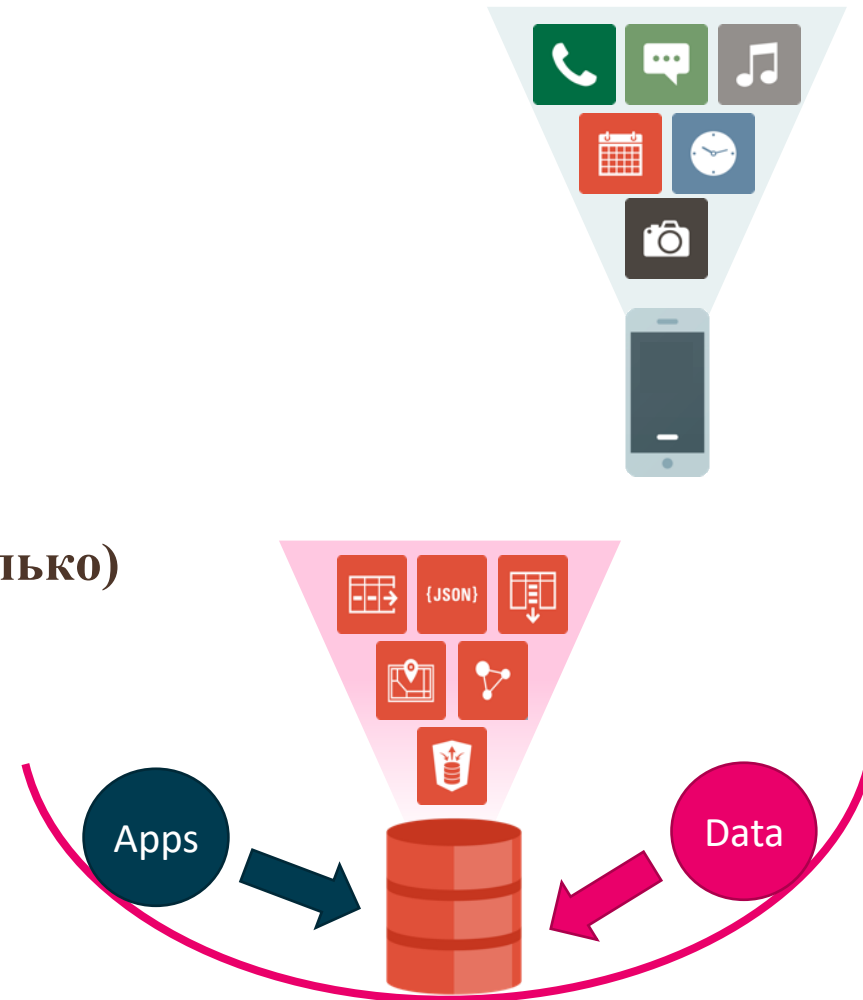
Фокус на интеграцию

Фокус на инновации – просто выполни SQL



Чем хороша мультимodelьная и конвергентная БД?

- **Единый механизм администрирования**
- **Единая политика безопасности**
- **Единая политика надежности**
- **SQL, Транзакционность и согласованность**
- **Облегчение изучения и сопровождения (1 СУБД а не несколько)**
- **Создание смешанных приложений**
- **Снижение будущих рисков (что еще понадобится ?)**
- **Снимается проблема интеграции**
- **In-memory, sharding, Exa, Cloud и т д для всех типов данных**



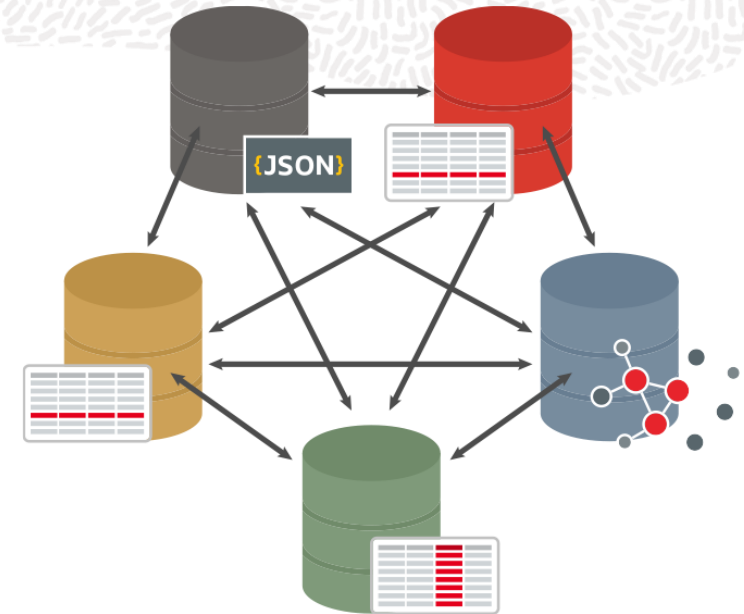
Суммарный эффект применения мультимодельной и конвергентной БД

Позволяет обеспечить необходимый уровень качества данных для всего предприятия

- от выбора для решения каждой прикладной задачи оптимального способа хранения данных и доступа к ним
- до обеспечения семантической корректности и согласованности деловой информации для всех её потребителей в разных задачах

Конвергентные vs специализированные

- **Каждая** специализированная БД хранит **фрагмент** данных компании, Требуется перемещение и трансформация данных
- **Интеграция** фрагментированных данных делает разработку и анализ данных очень **сложным и дорогим процессом**
- **Управление** множеством уникальных БД - **сложно, дорого, рискованно**
- Но иногда специализированная СУБД делает больше и лучше, чем конвергентная
- В России выбор ограничен !!!



versus



Отчет ЦСР 2022

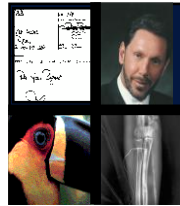


Таблица 3 Детализация продаж СУБД российских вендоров

Продажи СУБД	Доля, %	Объем, млрд руб.
СУБД общего назначения	47,8%	10,00
Аналитические СУБД	32%	6,70
Резидентные СУБД (in-memory)	14,3%	3,00
Платформа HADOOP и СУБД на ней	4,5%	0,95
Графовые СУБД	0,8%	0,16
Документные базы данных	0,5%	0,10
Мультимодельные базы данных	0,1%	0,02

Типы данных – можно создать свои

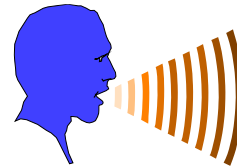
IMAGE



CONTEXT



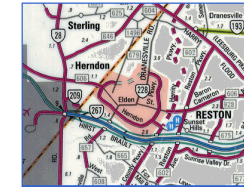
AUDIO



VIDEO



LOCATOR

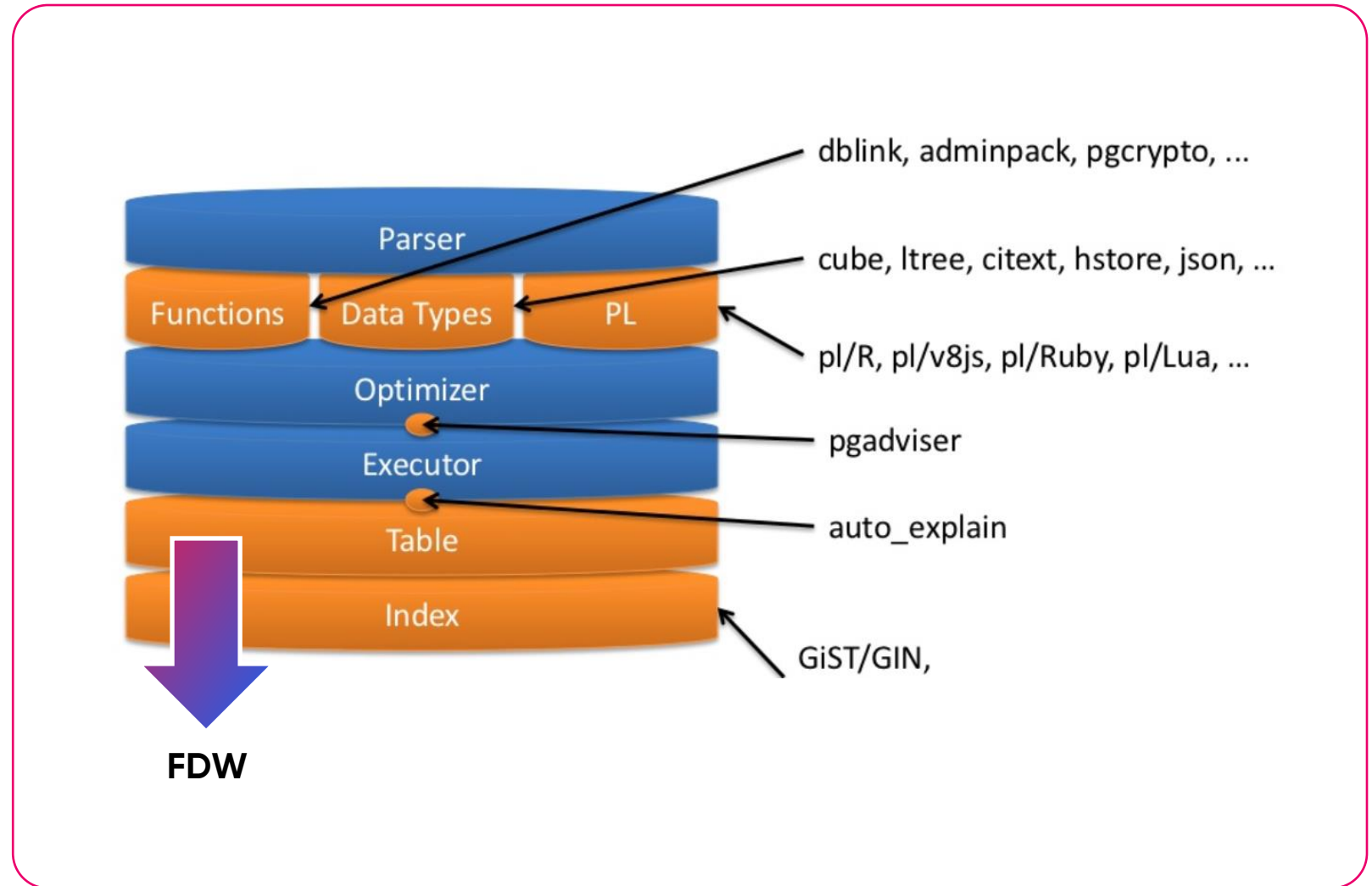


- Реляционные
- Текстовые (текст, JSON, XML)
- Пространственные
- Графы, property graph
- RDF, семантические сети
- RFID
- BLOB, Security Files
- | | |
|--|--|
| | |
| | |

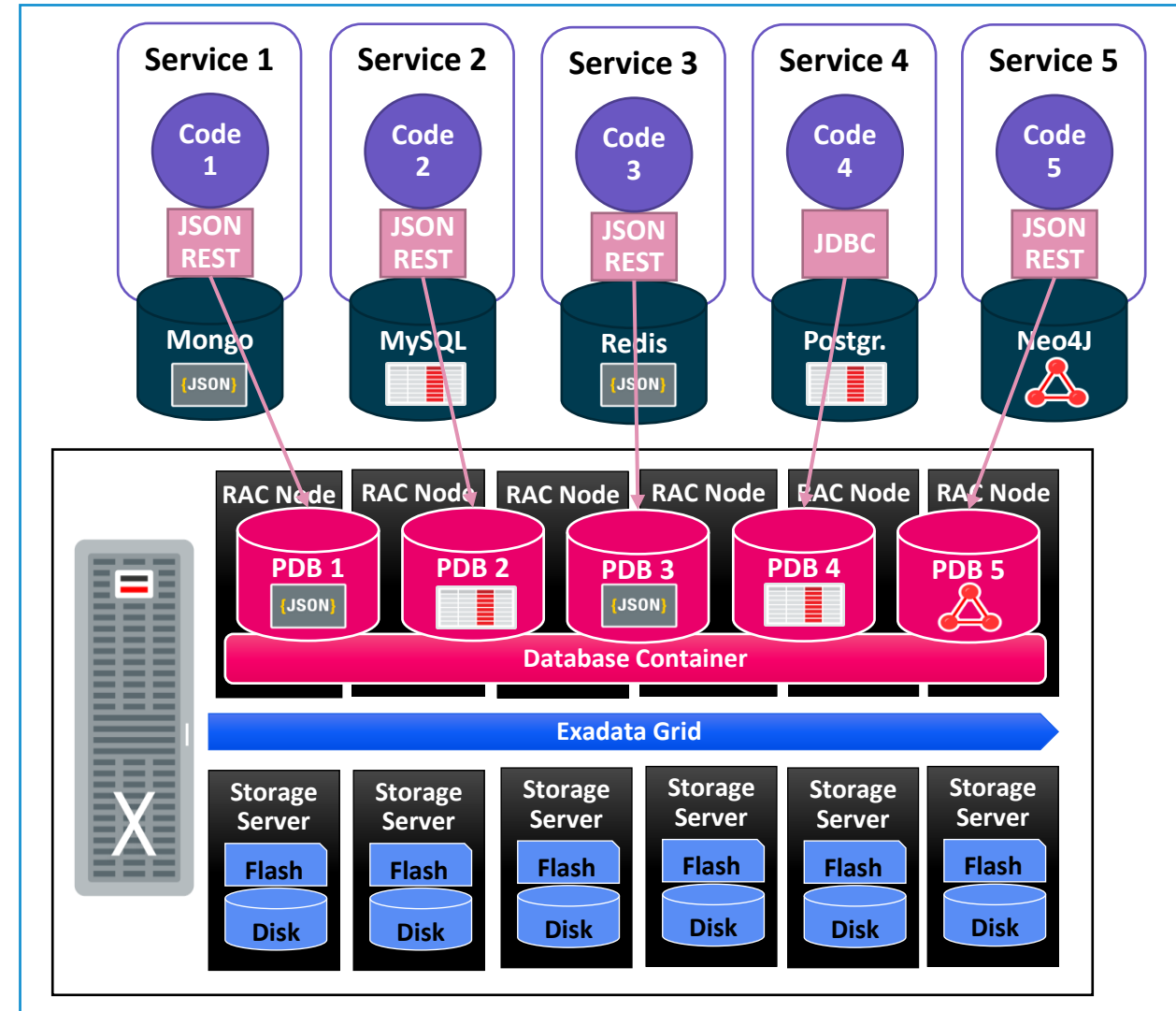
 Аудио
- Видео
- Изображения, фото (JPEG, MPEG, GIF)
- Объекты
- Медицина (DICOM)
- Многомерные кубы
- Big data, Hadoop

Расширяемость PostgreSQL

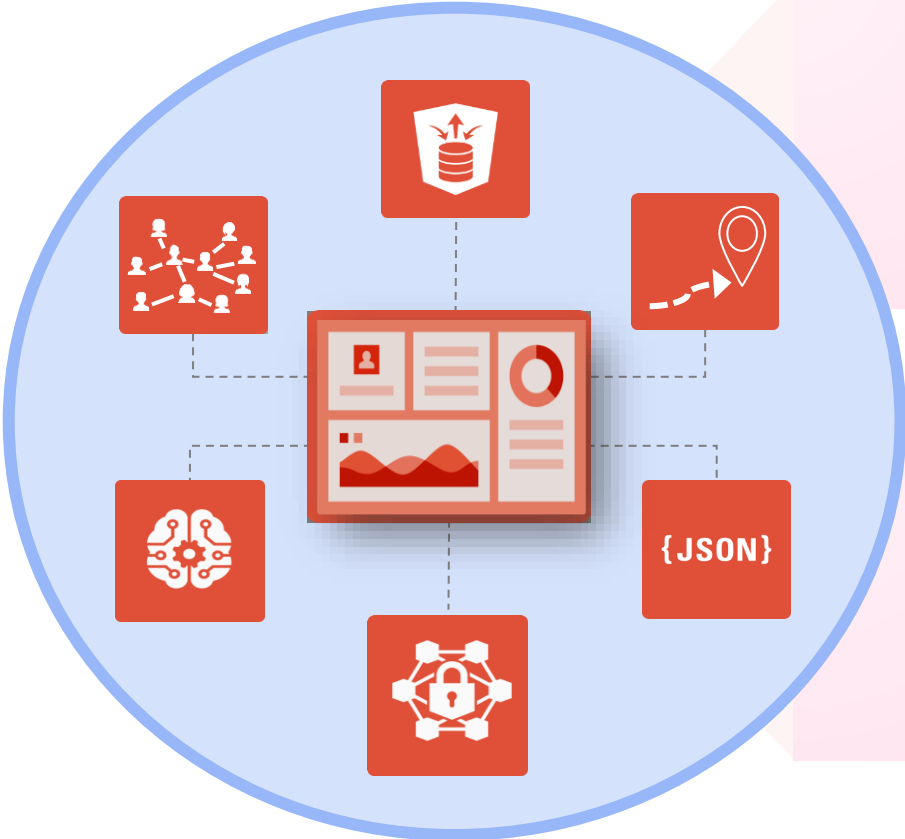
- **Extension** – фактически plugin для СУБД
- **Может добавлять новые типы данных, менять поведение**
- **Примеры:**
 - pg_partman (partitioning)
 - pg_variables (глобальные переменные)
 - pg_transfer (онлайн перемещение таблиц)



Multi Tenant БД, Микросервисы и мультимодельность



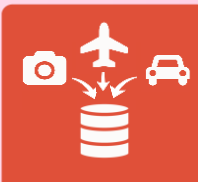
Data Driven приложения: Новые способы работы с данными (технологии)



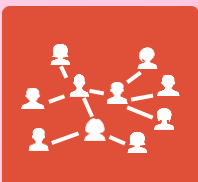
Machine Learning



Real-Time Analytics



IoT



Social Graph Analysis



Documents



Text Search

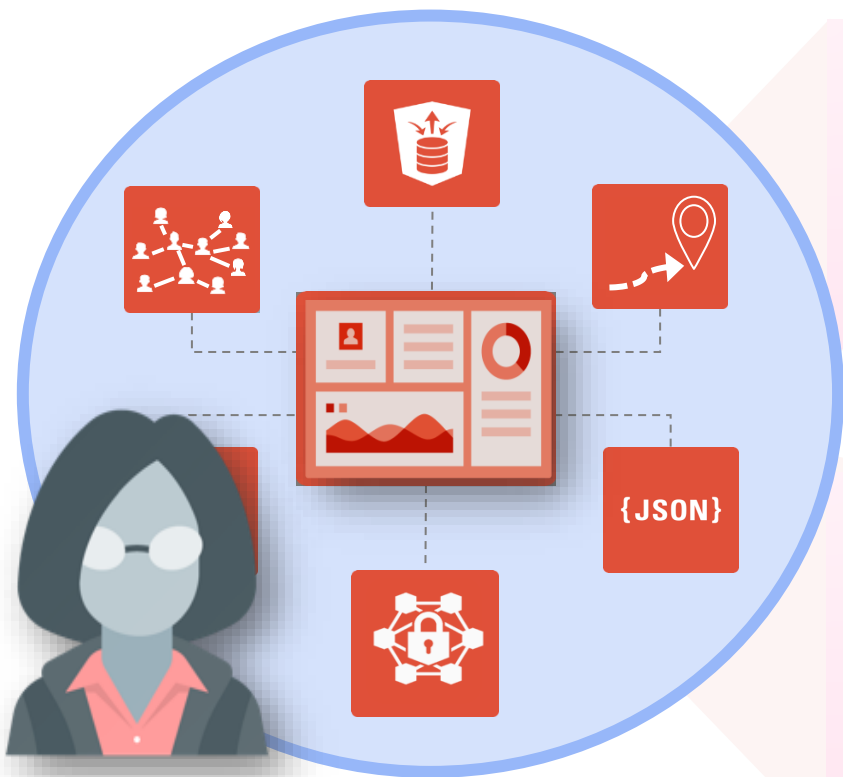


Spatial Processing

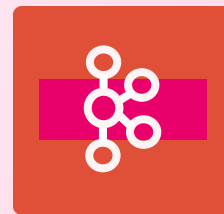


Blockchain

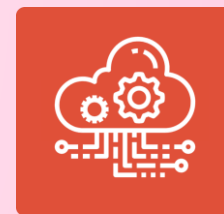
Data Driven приложения: Современные парадигмы разработки



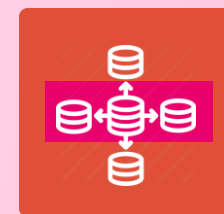
Microservices



Events



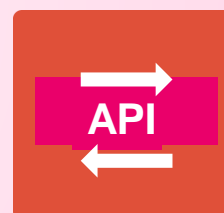
SaaS



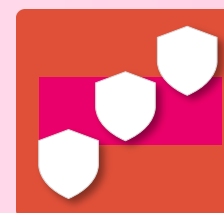
Distributed Data



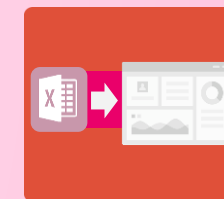
CI/CD



API Driven
Development



Defense
in Depth



Low Code

В конвергентной БД вы можете применять **совместно**:

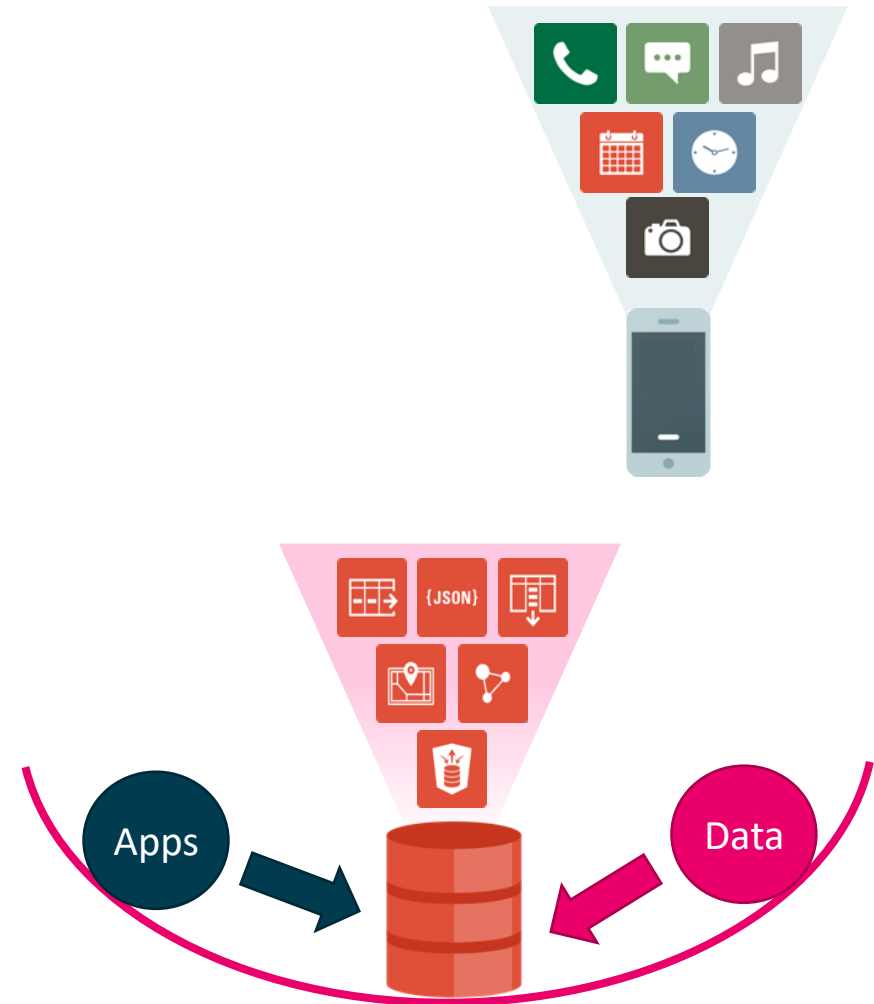
- **Различные виды нагрузок**
- **Различные типы данных, модели данных и спец алгоритмы**
- **Различные типы приложений (технологии)**
- **Различные парадигмы разработки**

Механизмы интеграции БД

- доступ в другие БД (свои и чужие)
 - DBLink
 - FDW
 - Gateway
- GoldenGate
- Репликация
- external table – хранение части данных в файлах ОС
- DW - интеграция в 1 БД
- etc

Data Gravity vs Data Mesh

- Data Gravity – разные приложения к одной БД
- Data Mesh – данные как сервис под каждое приложение
 - Подмножество данных
 - Синхронизация данных
 - Простое создание подмножества



Российский рынок, импортозамещение

- Западные СУБД ушли (Oracle, MS, IBM, Terradata ...)
- Российский реестр + ФСТЭК + реляционная модель
- К 2025 году перейти на российское ПО
- Выбор не велик – RedBase, форки PostgreSQL
- Страусиная политика – остаться и ждать
- Миграция – долго, сложно, дорого, куда?
- Многих механизмов нет (RAC, Exadata, VLDB, пакеты, RAC, ASM, шардинг, EM, LOB ...)
- Доработка Postgres Pro Enterprise и инструментов миграции
- Автоматической миграции НЕТ

Форки PostgreSQL и родимые пятна

- Postgres Pro
- Tantor
- Pangolin
- Diasoft Q-Database
- Jatoba
- Квант Гибрид
- Аренадата (PostgreSQL + GreenPlum)

Что такое Postgres Pro Enterprise

- **Отечественная коммерческая объектно-реляционная СУБД** для высоконагруженных систем крупных предприятий
- **Богатая функциональность** (SQL, таблицы, индексы, MV, view, последовательности, функции, процедуры, триггеры, ACID, роли/привилегии, ограничения целостности, BLOB, оконные функции и т. д.) + **ПАКЕТЫ**
- **Разработчики и техническая поддержка в России**
- **Сертификат ФСТЭК Г1, уровень доверия УД 4** (защита конфиденциальной информации, у Oracle и MS – УД ниже и = 6)
- **В реестре** российских программных продуктов
- **Регулярное слияние** с ванильной СУБД PostgreSQL
- **Расширяемый код**

Тенденции развития в России

- ◆ Доработка СУБД под требования Enterprise
 - ◆ Адаптация инструментов миграции
 - ◆ Создание единого окна
 - ◆ Развитие облачной модели
 - ◆ Конвергентные БД
 - ◆ Направление автономных баз данных
 - ◆ Создание программно-аппаратного комплекса
 - ◆ Внедрение ИИ
- А также шардирование, In-memory-технологии и поддержка множества различных языков.

Q & A

