

Tengri Data platform:

Аналитическая платформа для эры AI



Что нужно организациям для внедрения AI?

Единая Платформа AGI

Интегрированная среда работы и обучения AI агентов

Развернутое описание данных и бизнес-контекста, метаданные

Возможность трансформировать и анализировать данные за секунды

Возможность загружать и хранить весь объем данных: тексты, медиа, таблицы

Tengri Data Platform - все, что нужно для внедрения AI - собрано в единой интегрированной платформе

1. Загрузить любые данные

2. Хранить сотни TB данных

3. Трансформировать данные SQL/Python



Tengri Data Platform

4. Визуализировать данные

5. Анализировать данные AI моделями.

6. Делегировать задачи AI Агентам.

01 Интегрированная платформа

Tengri устанавливается одним пакетом: не нужно интегрировать и донастраивать компоненты.

02 Удобство аналитика

Полная поддержка любых операций с данными на языках SQL и Python: не ищите редких специалистов, нанятый аналитик сразу принесет пользу.

03 Контроль и безопасность

Все операции с данными покрываются системой прав и разрешений, все действия логируются, все зависимости доступны.

04 Минимальное администрирование

Операции над данными максимально эффективны и изолированы за счёт использования современных технологий.

05 Полный цикл работы с данными

В единой платформе собраны инструменты для загрузки, трансформации данных, оркестрации расчетов, визуализации, машинного обучения и AI.

06 Разделение Compute&Storage

Ресурсы вычисления и хранения данных независимо эластично масштабируются, On-premise, в облаках или гибридно.

07 Хранение: OpenLakeHouse в S3

Данные хранятся в S3, в формате Iceberg, доступном для стороннего использования через Iceberg REST API.

08 Вычисление: эластичный кластер

Вычисление выполняется на эластичном кластере под управлением K8S или под собственным контролем Tengri для небольших инсталляций.

Тенгри: платформа SQL вычислений

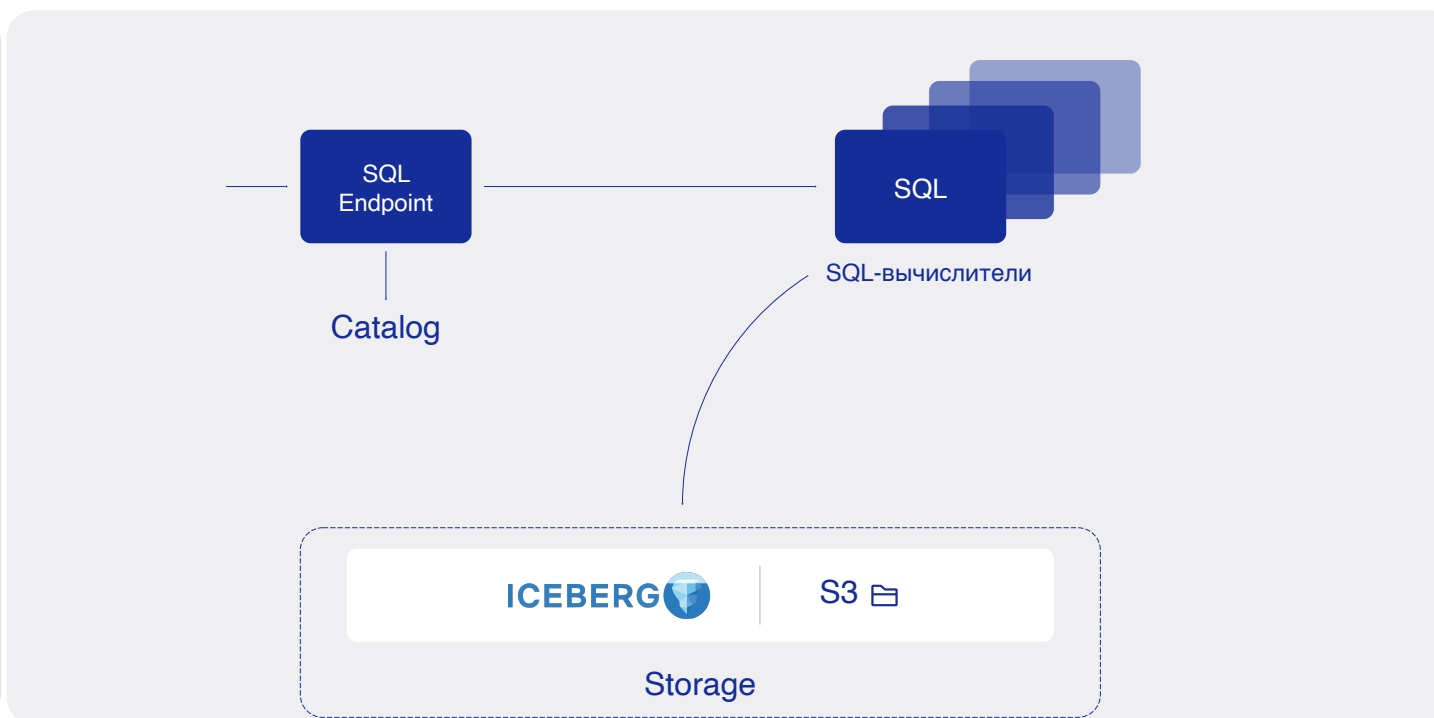
Главное: индивидуальные сессионные вычислители

Данные таблиц хранятся в S3
в открытом формате Apache Iceberg

SQL Endpoint принимает SQL запросы
диалекта PostgreSQL

Каждая сессия получает свой
вычислитель - аналог Warehouse в
Snowflake

Процессы в вычислителях соблюдают
ACID - гарантируется целостность,
изоляция.



Преимущество Tengri над традиционными OLAP/MPP - изоляция, масштабирование

Бенчмарк TPC-H

sf=100, статический тест

Параметры: случайный запрос 1-10, случайные
параметры, 1 час, 1-5-10-20 потоков

Tengri

100 vCPU, 200GB
RAM

Greenplum

1 координатор + 3 ноды,
по 30 vCPU, 64GB RAM

cloudberry 1.6.0
(based on postgres 14,
originally forked from GP 7)

Tengri линейно масштабируется
при увеличении количества
аналитиков, эффективно утилизируя
вычислительные ресурсы

Tengri и Greenplum: количество запросов в час в зависимости от количества потоков



Тенгри: аналитическая платформа

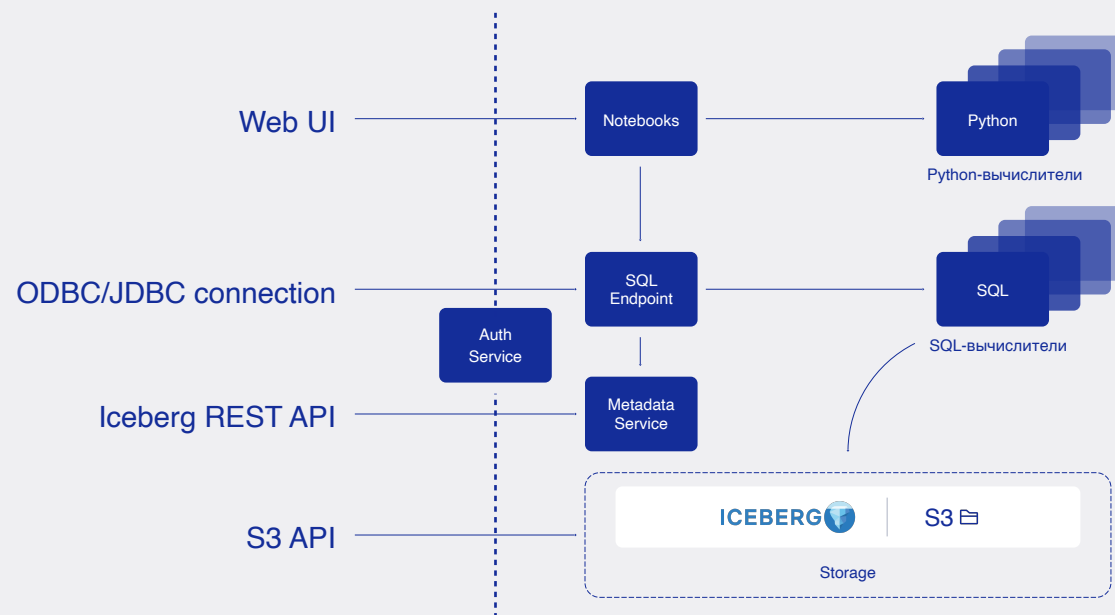
Главное: ноутбук для аналитиков с поддержкой Python

Аналитики работают с данными создавая ноутбук

Ноутбук выполняются SQL- и Python-вычислителями

Ноутбуки можно редактировать, публиковать, ставить на расписание

Все виды доступа к данным идут через единый сервис авторизации, единую ролевую модель доступа, RBAC.



Тенгри: аналитическая платформа

Главное: ноутбук для аналитиков с поддержкой Python

The screenshot shows the TNGRI Playground interface. On the left, there's a sidebar with 'Add notebook' and a list of default notebooks like 'Ingest data from API', 'Create chart (matplotlib)', and 'Create forecast (prophet)'. The main area displays a Python notebook with code for data analysis and forecasting. The code includes:


```

y_col = 'petroleum'
df = cell_output[[x_col, y_col]]
df = df.rename(columns={x_col: 'ds', y_col: 'y'})

# Initialize and fit the Prophet model
model = Prophet(interval_width=0.95)
model.fit(df)

# Create a dataframe for future dates
future = model.make_future_dataframe(periods=months, freq='ME')

# Generate predictions
forecast = model.predict(future)

fig = model.plot(forecast, uncertainty=False, plot_cap=False)
fig
    
```

 Below the code, there's an 'Output' section showing a time-series plot of petroleum data from 1970 to 2020, with a blue line representing the forecast and black dots for historical data.

Визуализация и прогнозирование

The screenshot shows the TNGRI Catalog interface. On the left, there's a 'Catalog' sidebar with a search bar and a tree view of catalogs including 'chembl_35', 'fineweb', 'fw_13times_sent_filter', 'fw_sample', and 'fw_sample_90'. The main area displays the metadata for the column 'language_score' in the 'fineweb.fw_sample_90' catalog. The metadata table is:

column	value
typename	DOUBLE
min_value	0.6500000357627869
max_value	0.9998643398284912
count	94315564
null_count	0
distinct_count	5082738

 Below the table, there's a histogram for 'language_score' showing the distribution of values, with a peak around 0.95.

Каталог и федерализация

The screenshot shows the TNGRI Schedule interface. At the top, there's a search bar for notebooks and a filter for the time range (Week, Day, Hours). Below, there's a table of scheduled notebook runs:

Notebook	User	Environment	Cron	Status
Example: Create chart (matplotlib)	asfilatov	shared	00/10 00/1 * * MON, TUE, WED, THU, FRI, SAT, SUN	active
Blank notebook	naorlov	shared	00 00/1 * * MON, FRI, SUN, THU, WED	disabl...

 The 'Status' column shows a visual representation of the schedule with vertical bars indicating active periods.

Расписание и оркестрация

Тенгри: платформа AI агентов

Главное: AI Агенты



Чат - единое пространство общения пользователей

AI агенты - особые пользователи платформы, управляемые LLM

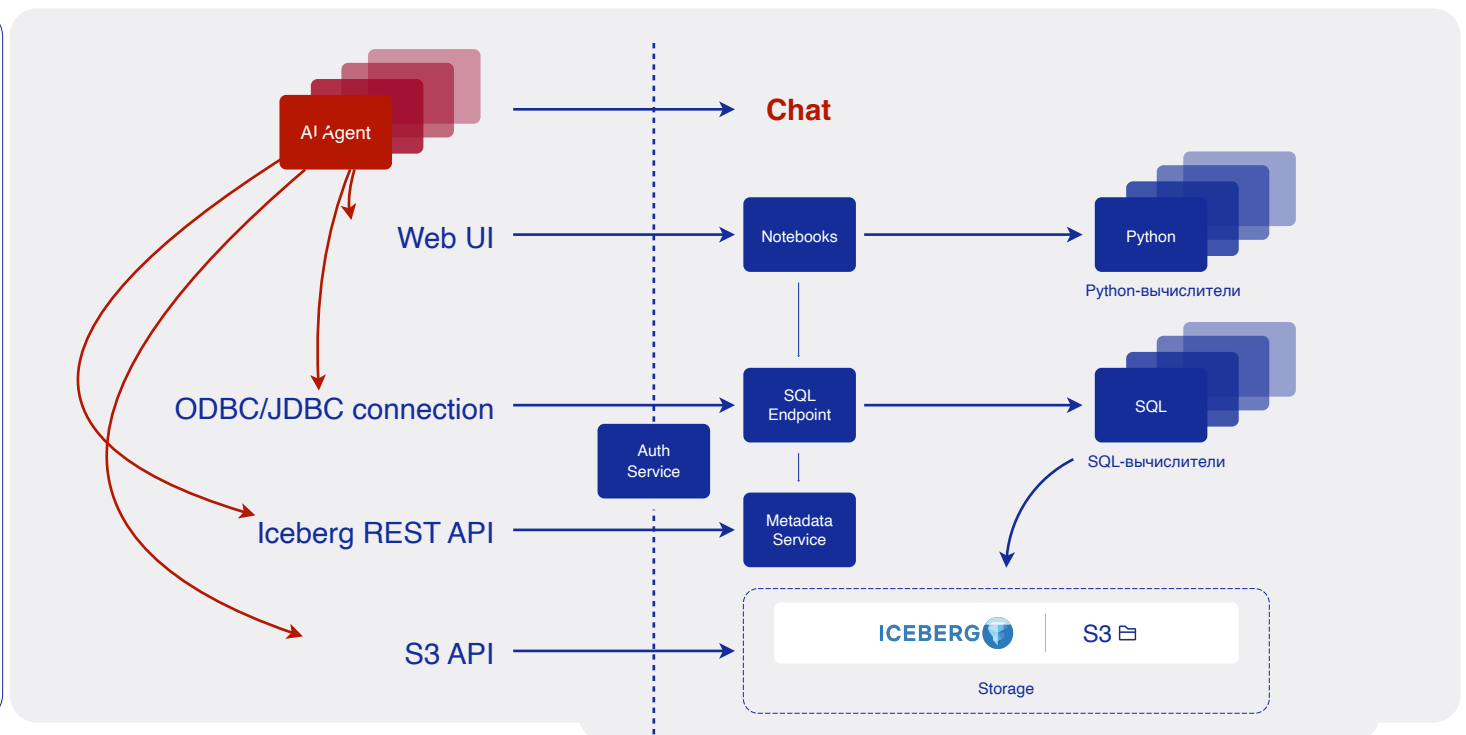
AI агенты работают в диалоге с живыми и неживыми пользователями



AI агенты обладают краткосрочной и долгосрочной памятью для развития навыков



AI агенты обладают набором инструментов - SQL, каталог, поиск в интернете, память, создание и запуск кода python/javascript



Тенгри: платформа AI агентов

Главное: AI агенты

Выведи последние координаты трека Alps 2017 Day 1

13:24

Infosos

В базе данных есть несколько таблиц, связанных с геоданными и треками. Для поиска координат трека "Alps 2017 Day 1" подходят таблицы:

- demo_geo_points
- demo_geo_points_for_map
- demo_geo_points_silver
- demo_geo_tracks

нужно уточнить, какую именно таблицу использовать для анализа, так как все они содержат геоданные. Пожалуйста, укажите, какую таблицу вы хотите использовать: demo_geo_points, demo_geo_points_for_map, demo_geo_points_silver или demo_geo_tracks?

13:24

А в каких таблицах есть колонка track_name?

13:26

Infosos

В базе данных есть две таблицы с колонкой track_name:

- demo_geo_tracks
- demo_geo_tracks_silver

Обе таблицы содержат:

- track_id — идентификатор трека
- track_name — название трека (например, "Alps 2017 Day 1")
- Дополнительные геоданные (широта, долгота, длина и т.д.)

Чтобы вывести последние координаты трека "Alps 2017 Day 1", нужно выбрать одну из этих таблиц. Пожалуйста, укажите, какую использовать: demo_geo_tracks или demo_geo_tracks_silver?

13:27

Возьми demo_geo_tracks_silver

13:27

Infosos

Результаты анализа

Таблица данных

latitude	longitude	time_stamp
45.812387	9.074852	2017-07-24T19:03:11

Выводы

Последние координаты трека "Alps 2017 Day 1":

- Широта: 45.812387
- Долгота: 9.074852
- Время: 24 июля 2017, 19:03:11 (UTC)

Координаты соответствуют району **Алп**, близко к итальянской границе (возможно, область Ломбардия или Валь-д'Аоста).

Данные получены из таблицы demo_geo_points_silver, которая содержит детальные точечные данные трека, связанные с demo_geo_tracks_silver по track_id.

Запрос выполнен корректно. Последняя точка трека найдена.

13:28

askpostgres

Подсчет количества прилетов в аэропорт Домодедово

Выполню анализ количества рейсов, прибывающих в аэропорт Домодедово (IATA-код: DME) на основе данных о рейсах из таблицы flights_db_bookings_flights.

```
-- Подсчет количества прилетов в аэропорт Домодедово (DME)
SELECT COUNT(*) AS arrival_count
FROM raw.flights_db_bookings_flights
WHERE arrival_airport = 'DME';
```

114 ms

Run Output py sql

arrival_count
6378

1 row

Результаты анализа

Количество прилетов в аэропорт Домодедово

| № | Количество прилетов | |-----| | 1 | 6,378 |

Анализ

Ключевые наблюдения:

- В базе данных зафиксировано **6,378 прилетов** в аэропорт Домодедово (DME)
- Данные относятся к рейсам из таблицы raw.flights_db_bookings_flights, которая содержит полную историю полетов
- Код аэропорта Домодедово в базе — **DME**, что соответствует стандартному IATA-коду

Вывод: Аэропорт Домодедово является одним из ключевых узлов воздушного транспорта в России, что подтверждается значительным количеством прилетов — более 6 тысяч рейсов в выборке. Этот объем свидетельствует о высокой загруженности аэропорта и его важной роли в национальной и международной авиасети.

16:04

Готовлю итоговый дашборд в виде готового React-приложения, которое можно сразу запустить — через deploy_js_code.

✓ **Пример использования:**

Пользователь: «Покажи дашборд по продажам из таблицы web_sales»

Я:

- Узнаю структуру таблицы
- Определяю даты и метрики (сумма продаж, средняя цена, количество)
- Генерирую SQL-запросы
- Строю график продаж и трендов
- Проверяю код на ошибки
- Отдаю готовый дашборд как рабочее React-приложение

Готов начать — просто скажите, **какую таблицу** вы хотите визуализировать, и я сделаю всё остальное!

02.11.2025, 09:25

визуализируй cbrf_money_columns

02.11.2025, 09:28

dasha

Log Output

CBRF Money Columns Dashboard

Total Observation Value (Bar Chart)

300 000
250 000
200 000
150 000
100 000
50 000
0

Total Observation Value

1993-01-01T00:00:00
1994-06-01T00:00:00
1995-11-01T00:00:00
1997-04-01T00:00:00
1998-09-01T00:00:00
2000-02-01T00:00:00
2001-07-01T00:00:00
2002-12-01T00:00:00
2004-05-01T00:00:00
2005-10-01T00:00:00
2007-03-01T00:00:00
2008-08-01T00:00:00
2010-01-01T00:00:00
2011-06-01T00:00:00
2012-11-01T00:00:00
2014-04-01T00:00:00
2015-09-01T00:00:00
2017-02-01T00:00:00
2018-07-01T00:00:00
2019-12-01T00:00:00
2021-05-01T00:00:00
2022-10-01T00:00:00
2024-03-01T00:00:00
2025-08-01T00:00:00

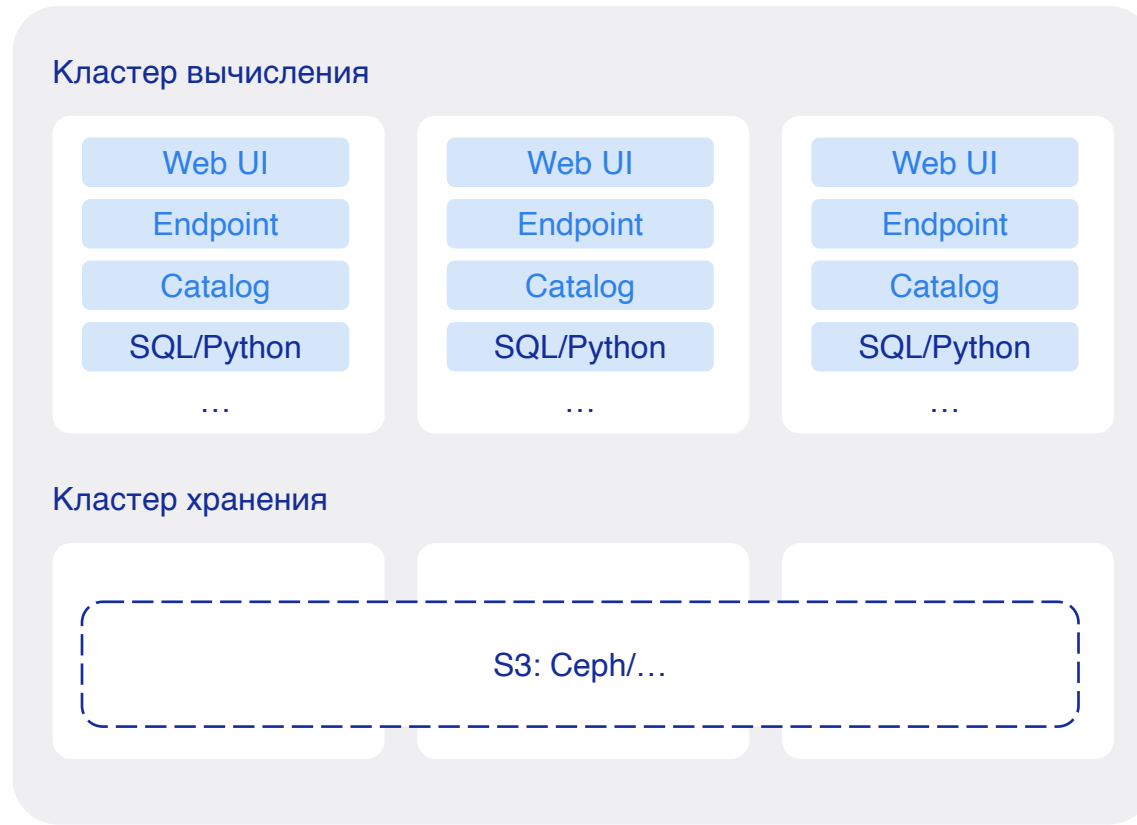
Варианты развертывания Tengri

Быстрый старт:
единственный сервер
-идеально для пилота



Высокопроизводительный
on-premise кластер

Гибридный формат



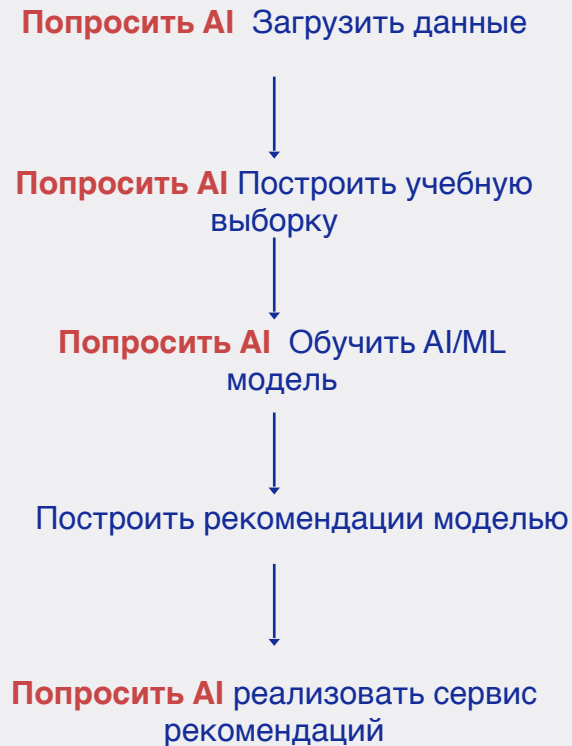
Для всех вариантов развёртывания цена лицензии Tengri = X руб. × количество vCPU серверов вычисления в год

Сценарии использования Tengri с AI

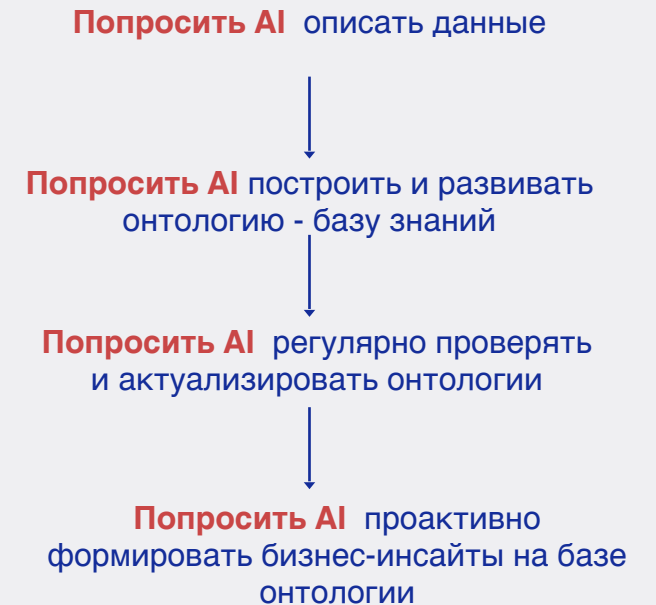
Регулярно формировать отчёт руководству



Формировать для продукта AI/ML рекомендации



Описать данные и породить инвайты



Преимущество Tengri над классическими OLTP базами

	Tengri	Oracle	MySQL	MS SQL
ANSI SQL, ACID, транзакции	+	+	+	+
Горизонтальное масштабирование	+	-	-	-
Высокопроизводительные OLAP запросы	+	+/-	-	-
Интегрированные AI агенты	+	-	-	-

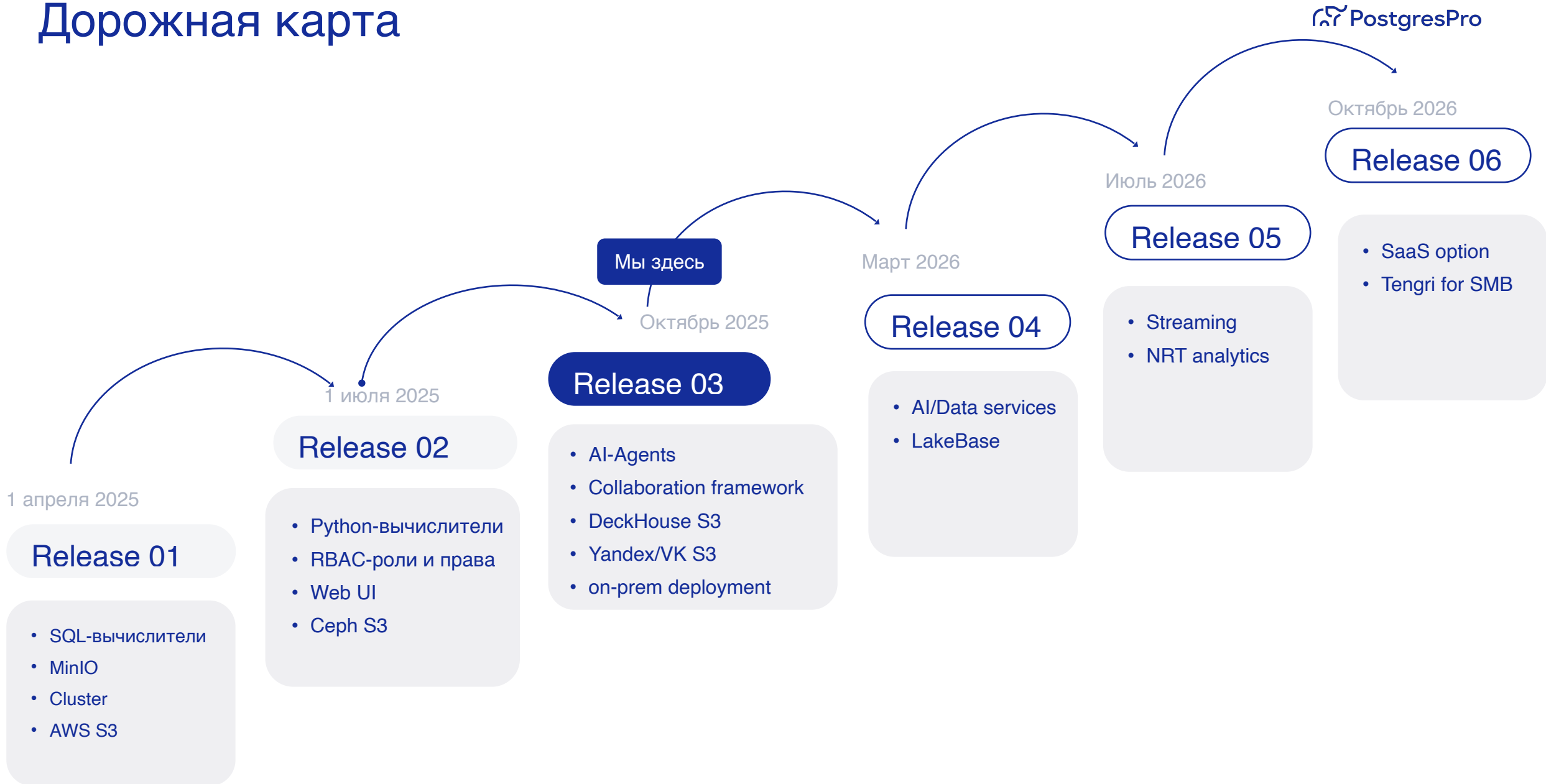
Преимущество Tengri над OLAP/MPP базами

	Tengri	Greenplum	Clickhouse
Горизонтальное масштабирование	+	+	+
ANSI SQL, ACID, транзакции	+	+	+/-
Полная поддержка Python, любые пакеты	+	+/- (хранимки)	+/- (хранимки)
Интегрированные AI агенты	+	-	-

Преимущество Tengri над open-source LakeHouse решениями

	Tengri	Trino+	Spark+
Быстрые ACID OLAP запросы	+	+	+
Транзакциональная целостность записи в Iceberg	+	+/- (требуется настроить)	+/- (требуется настроить)
Сквозная ролевая модель безопасности данных (RBAC)	+	+/- (требуется интегрировать)	+/- (требуется интегрировать)
Развертывание одним пакетом без интеграции	+	-	-

Дорожная карта





Николай Голов

Директор по продукту
Tengri Data Platform

Почта

n.golov@postgrespro.ru

Телеграм

[@Azathot_mail](https://t.me/Azathot_mail)

Михаил Тюрин

Директор департамента
аналитической платформы
Tengri

Почта

m.tyurin@postgrespro.ru

Телеграм

[@mikhailtyurin](https://t.me/mikhailtyurin)

Станислав Гришин

Руководитель продаж
Tengri

Почта

s.grishin@postgrespro.ru