УТВЕРЖДАЮ

Директор ООО «Центр-инвест ИТ»

_____ Дорошкевич Е. В.

«_1_» ____апреля____ 2026 г.

ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ «ЦИТ.ДатаЛаборатория»

Руководство администратора/пользователя

Описание функциональных характеристик и информация, необходимая для установки и эксплуатации ПО «ЦИТ.ДатаЛаборатория»

Оглавление

1. Термины

ПО «ЦИТ.ДатаЛаборатория» — это полностью контейнеризированная среда для локальной разработки и тестирования Корпоративного Хранилища Данных (КХД).

Greenplum — это MPP-база данных для аналитических нагрузок, используемая как основное хранилище данных.

Apache Airflow — это платформа оркестрации workflow для управления ETL/ELT пайплайнами.

MinIO — это S3-совместимое объектное хранилище для работы с неструктурированными данными.

DAG (Directed Acyclic Graph) — это направленный ациклический граф, представляющий набор задач и их зависимостей в Apache Airflow.

ETL (Extract, Transform, Load) — это процесс извлечения, преобразования и загрузки данных из источников в целевое хранилище.

ELT (Extract, Load, Transform) — это процесс извлечения, загрузки и последующего преобразования данных непосредственно в целевом хранилище.

MPP (Massively Parallel Processing) — это архитектура параллельной обработки данных, используемая в Greenplum для распределения вычислений по нескольким сегментам.

Профиль — это набор опциональных сервисов, которые могут быть запущены вместе с базовым окружением.

Makefile — это единый интерфейс управления всеми операциями в проекте.

Docker Compose конфигурация — это инструмент, который описывает все сервисы, их зависимости и сетевые взаимодействия.

Сегмент Greenplum — это узел хранения и обработки данных в распределённой архитектуре Greenplum, часть MPP-кластера.

Координатор (Master) — это центральный узел в кластере Greenplum, обрабатывающий клиентские запросы и распределяющий работу между сегментами.

PXF (Platform Extension Framework) — это фреймворк в Greenplum для доступа к внешним источникам данных (HDFS, S3, JDBC и др.).

S3 API — это программный интерфейс для работы с объектными хранилищами, совместимый с Amazon S3.

Dev-контур — это тестовое/разработочное окружение, с которым возможна синхронизация конфигураций локального окружения.

CeleryExecutor — это механизм выполнения задач в Apache Airflow, использующий Celery для распределённой обработки.

Volume Docker — это механизм для сохранения данных, созданных и используемых контейнерами, независимо от жизненного цикла контейнеров.

2. Описание ПО «ЦИТ.ДатаЛаборатория»

ПО **«ЦИТ.ДатаЛаборатория»** представляет собой комплексное инфраструктурное решение, которое предоставляет разработчикам и аналитикам данных готовую среду для работы с современным стеком технологий обработки и хранения данных. Решение объединяет ключевые компоненты современного data stack'a.

Примеры использования:

- Использование для локальной разработки и отладки ETL/ELT пайплайнов;
- Использование для тестирования DAG'ов Apache Airflow перед выгрузкой в продакшен;
- Сопоставление результатов трансформаций данных с ожидаемыми результатами в изолированной среде;
- Использование для обучения новых сотрудников работе с компонентами КХД;
- Прототипирование новых аналитических запросов и отчетов на локальных данных.

ПО «**ЦИТ.ДатаЛаборатория»** состоит из следующих элементов:

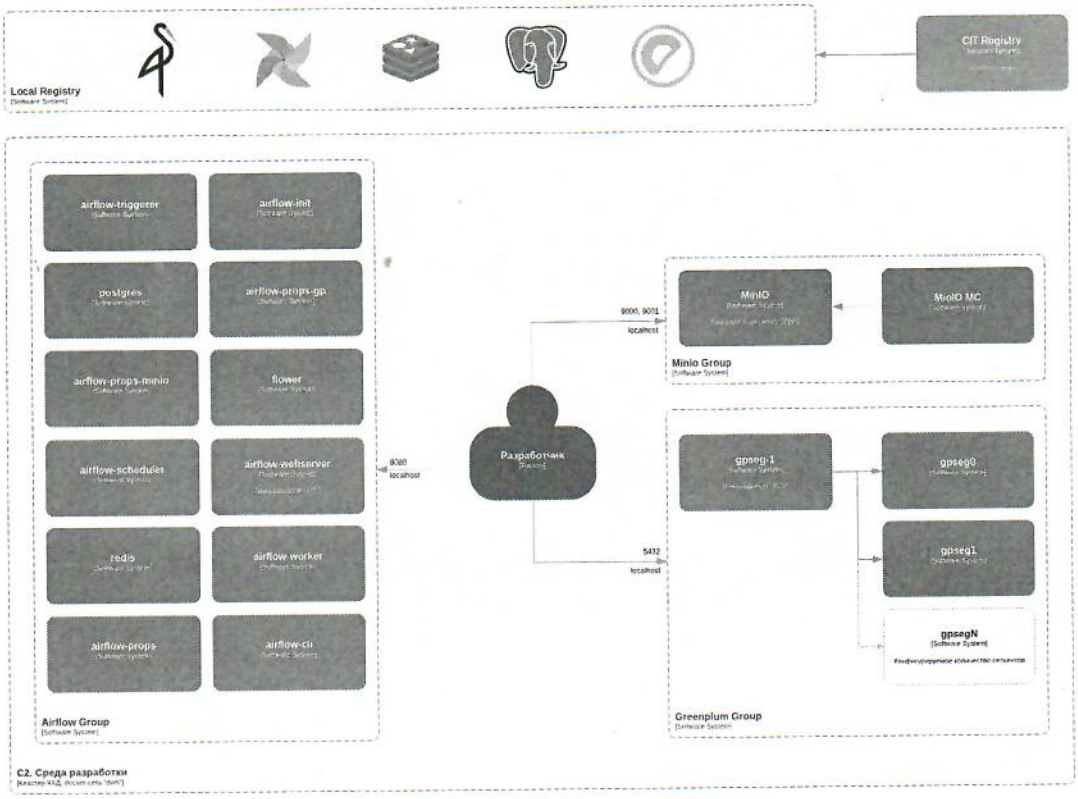
- Greenplum Database;
- Apache Airflow;
- MinIO;
- Docker Compose конфигурация;
- Makefile.

В разработке ПО «**ЦИТ.ДатаЛаборатория»** использовались:

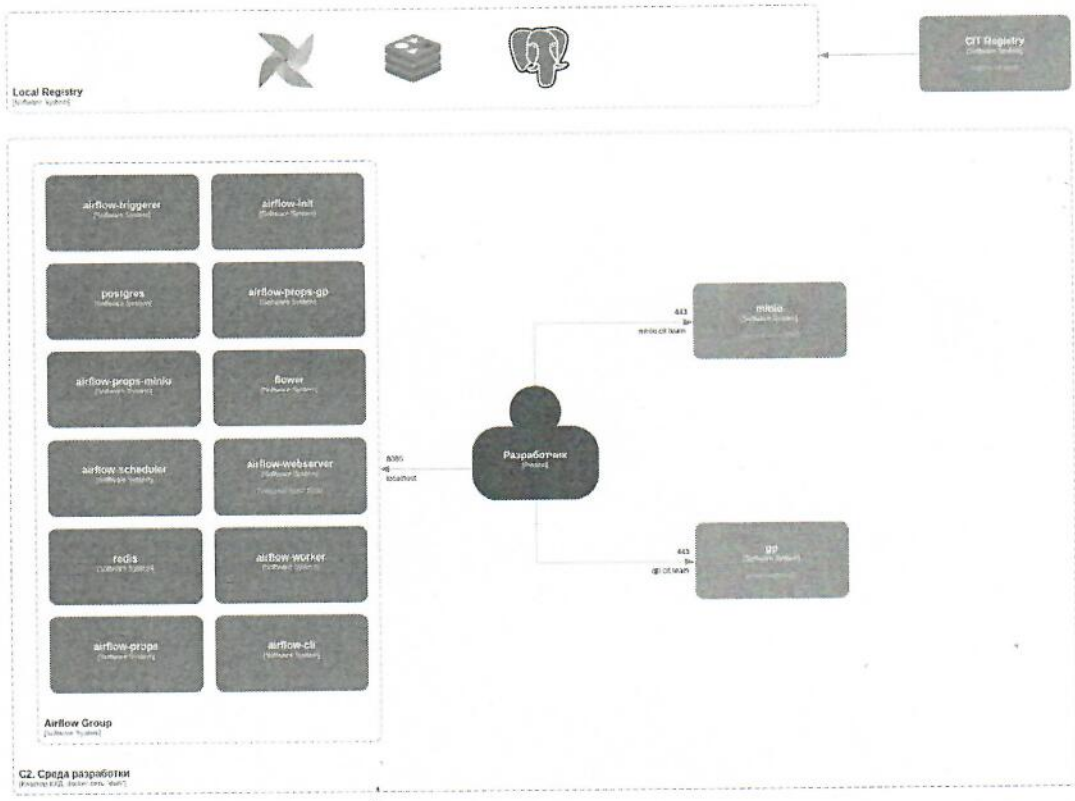
- технологии контейнеризации: Docker, Docker Compose;
- системы управления базами данных: Greenplum (на базе PostgreSQL);
- платформы оркестрации: Apache Airflow с CeleryExecutor;
- объектные хранилища: MinIO с S3-совместимым API.

Варианты развёртывания

- Полностью локальный режим.



- Режим с интеграцией внешних источников.



3. Программные и аппаратные требования к системе

Для корректной работы с ПО «**ЦИТ.ДатаЛаборатория»** необходима следующая конфигурация рабочего места администратора.

№ n/n	Параметр	Минимально допустимые значения	Рекомендуемые значения
1.	Процессор (CPU)	2 ядра	i9
2.	Оперативная память (RAM)	4 ГБ	32 ГБ
3.	Свободное место на диске	10 ГБ	40 ГБ

4. Руководство администратора/пользователя

4.1. Установка

Установка экземпляра:

Получите установочный архив. Распакуйте архив в удобную директорию на рабочем компьютере. Перейдите в корневую папку распакованного дистрибутива.

```bash
cd /path/to/cit.datalab
```

Инициализация окружения:

```bash
chmod +x install.sh
./install.sh          # загрузка образов из bin/*.tar (если они есть в архиве)
make init             # автоматическая генерация .env файла из ..env
make mkdirs           # создание необходимых директорий
```

Запуск полного окружения:
Выполните команды из Makefile:

```bash
# Только базовые компоненты
make all
# Полный локальный контур (Airflow + MinIO + Greenplum)
make all COMPOSE_PROFILES=minio,gp,flower
```

4.2. Основные компоненты

Apache Airflow:

Web-интерфейс: доступен по адресу http://localhost:8080

Учетные данные по умолчанию: Логин: airflow, Пароль: airflow.

Директории на хосте:

src/airflow/

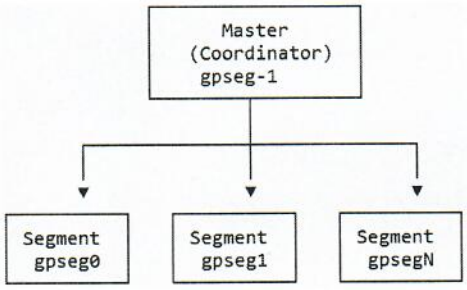
- dags/ — DAG-файлы
- logs/ — Логи выполнения
- config/ — Конфигурационные файлы
- plugins/— Пользовательские плагины

Greenplum Database:

Подключение:

- Host: localhost
- Port: 5432
- User: gpadmin
- Database: postgres

Архитектура кластера:



Предустановленные компоненты:

Компонент	Версия	Описание
Greenplum	6.25.1	MPP-база данных
PXF	6.10.0	Platform Extension Framework для доступа к внешним данным

Компонент	Версия	Описание
Java	11	Для работы PXF

MinIO (опционально, запускается с профилем minio):

Web Console: http://localhost:9000

S3 API: порт 9001

Учетные данные по умолчанию: Access Key: minioadmin, Secret Key: minioadmin

Директории на хосте:

run/minio/

- config/ — Конфигурация MC
- data/ — Локальные файлы для зеркалирования
- meta/ — Данные MinIO

Конфигурация (определяется в файле .env):

```env
# User isolation
UID=1000
AIRFLOW_UID=1000

# Airflow
AIRFLOW_WEBSERVER_PORT=8080
_AIRFLOW_WWW_USER_USERNAME= airflow
_AIRFLOW_WWW_USER_PASSWORD= airflow

# MinIO
MINIO_USER=minio
MINIO_PASS=minio123
```

На хосте после клонирования репозитория и запуска системы создаются следующие каталоги и файлы:

```
.
├── ..env
├── Makefile
├── docker-compose.yml
├── cluster.yml
├── install.sh
├── assets/
│   ├── about.txt
│   ├── cit.datalab.\*.jpg
│   └── scripts/
└── src/airflow/   # dags, logs, config, plugins, dbt
```

4.3. Команды управления

Основные команды Makefile:

make all — полная инициализация и запуск

make init — инициализация переменных окружения

make build — сборка Docker-образов

make up — запуск контейнеров

make mkdirs — создание необходимых директорий

make up — Запуск контейнеров

make down — Остановка контейнеров

make clear — Полная очистка (остановка + удаление ./run/)

make scan — Пересканирование DAG-файлов

4.4. Работа с профилями

Docker Compose поддерживает профили для опционального запуска дополнительных сервисов:

Профиль	Сервисы	Описание
`minio`	MinIO, MC	S3-хранилище с клиентом для управления бакетами
`flower`	Flower	Веб-интерфейс для мониторинга Celery-воркеров
`gp`	Greenplum (gpseg-1 + сегменты)	MPP-кластер Greenplum
`debug`	Airflow CLI	Интерактивная консоль Airflow для отладки

Примеры использования:

```bash
# С MinIO
make up COMPOSE_ARGS="--profile minio"

# С MinIO + Greenplum + Flower
make up COMPOSE_PROFILES=minio,gp,flower

# Полная сборка с профилями
make all COMPOSE_ARGS="--profile minio --profile gp"
```

4.5. Устранение неисправностей

Проблемы с правами доступа

Greenplum запускается от имени пользователя gpadmin:gpadmin – это ограничение самой системы Greenplum 6. При использовании bind mount томов для сегментов Greenplum, директория ./run/gp может стать недоступной для редактирования пользователем. Для прямой работы с этой директорией (удалить, переместить) может потребоваться root-доступ.

Контейнеры не запускаются

1. Проверьте доступные ресурсы: RAM ≥ 4 GB, свободное место ≥ 10 GB
2. Убедитесь, что порты не заняты (8080, 5432, 9000/9001)
3. Проверьте логи: docker compose logs -f

DAG'и не появляются в Airflow

```bash
make scan
```

Общие критические проблемы

```bash
make clear && make all
```

Проверка состояния контейнеров

```bash
docker compose ps
```

5. Ограничения

Количество сегментов Greenplum ограничено ресурсами хоста. Каждый дополнительный сегмент увеличивает потребление памяти, CPU и дискового пространства.

6. Авторские права

Данное ПО защищено авторскими правами, запрещается копирование, модификация и распространение данного ПО без согласия правообладателя.