

ХАКАТОН 2.0

Резервное копирование БД

la espada

Наша команда

Старцев Захар - Teamlead

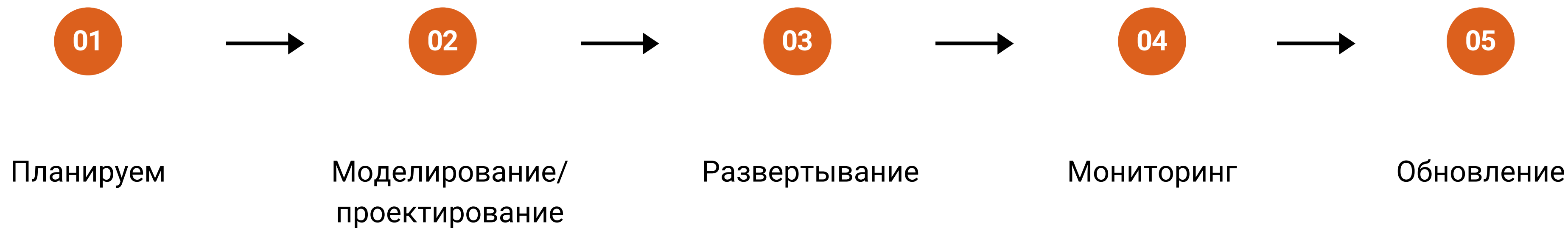
Маланин Дмитрий Project manager

Пушкарёва Алина - Search master

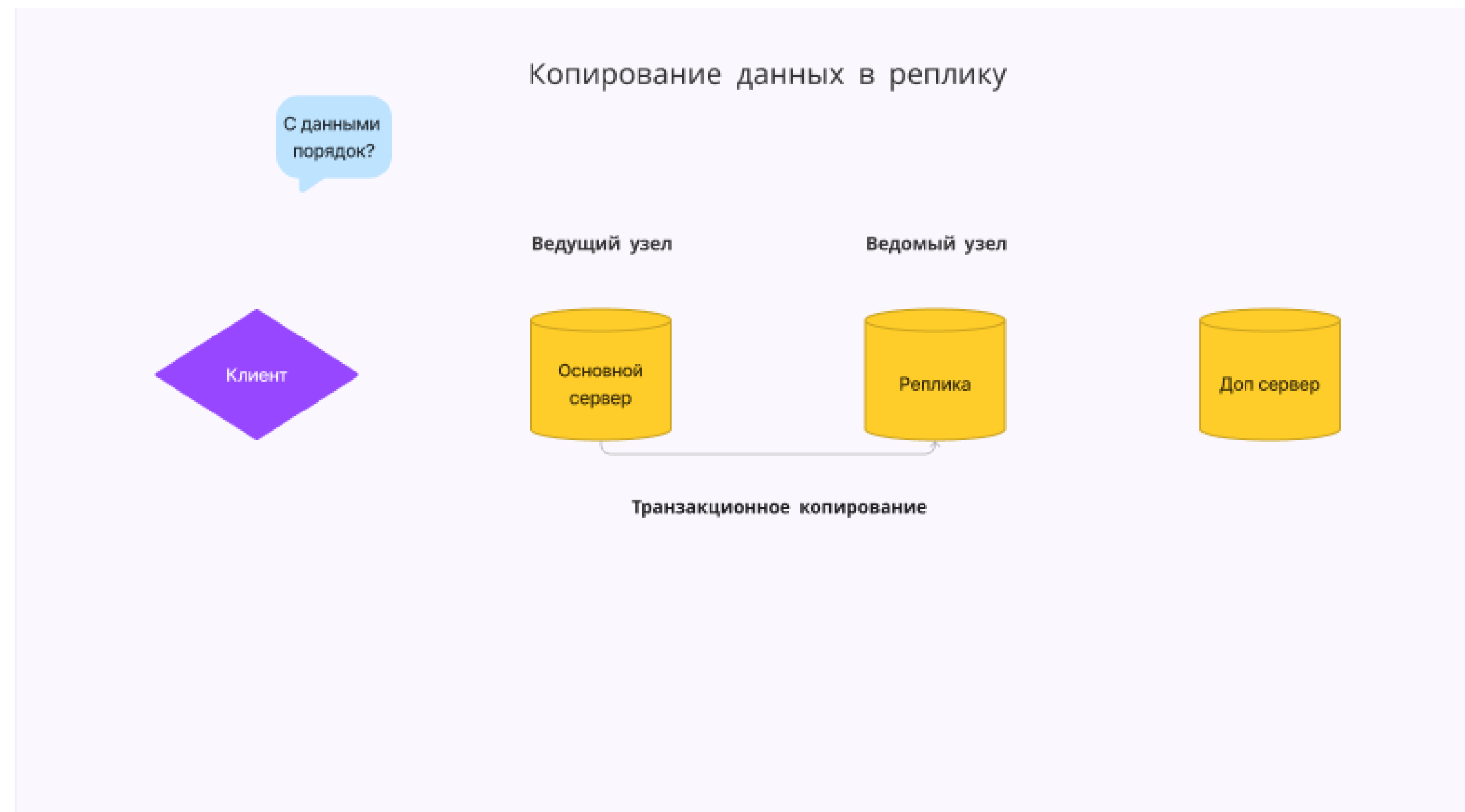
Кузьмишин Вениамин - Scrum master

Алиев Шукри - Search master

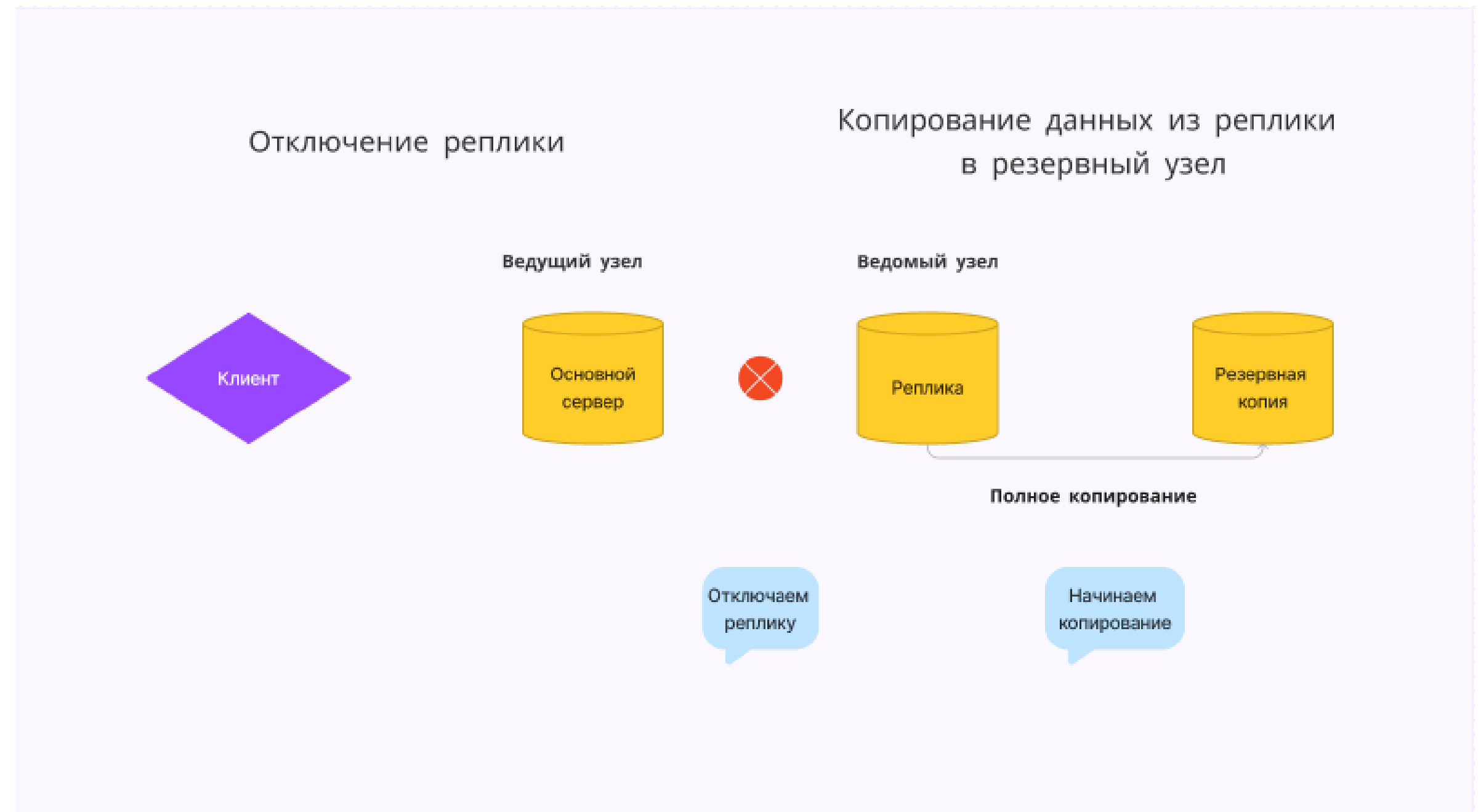
Жизненный цикл



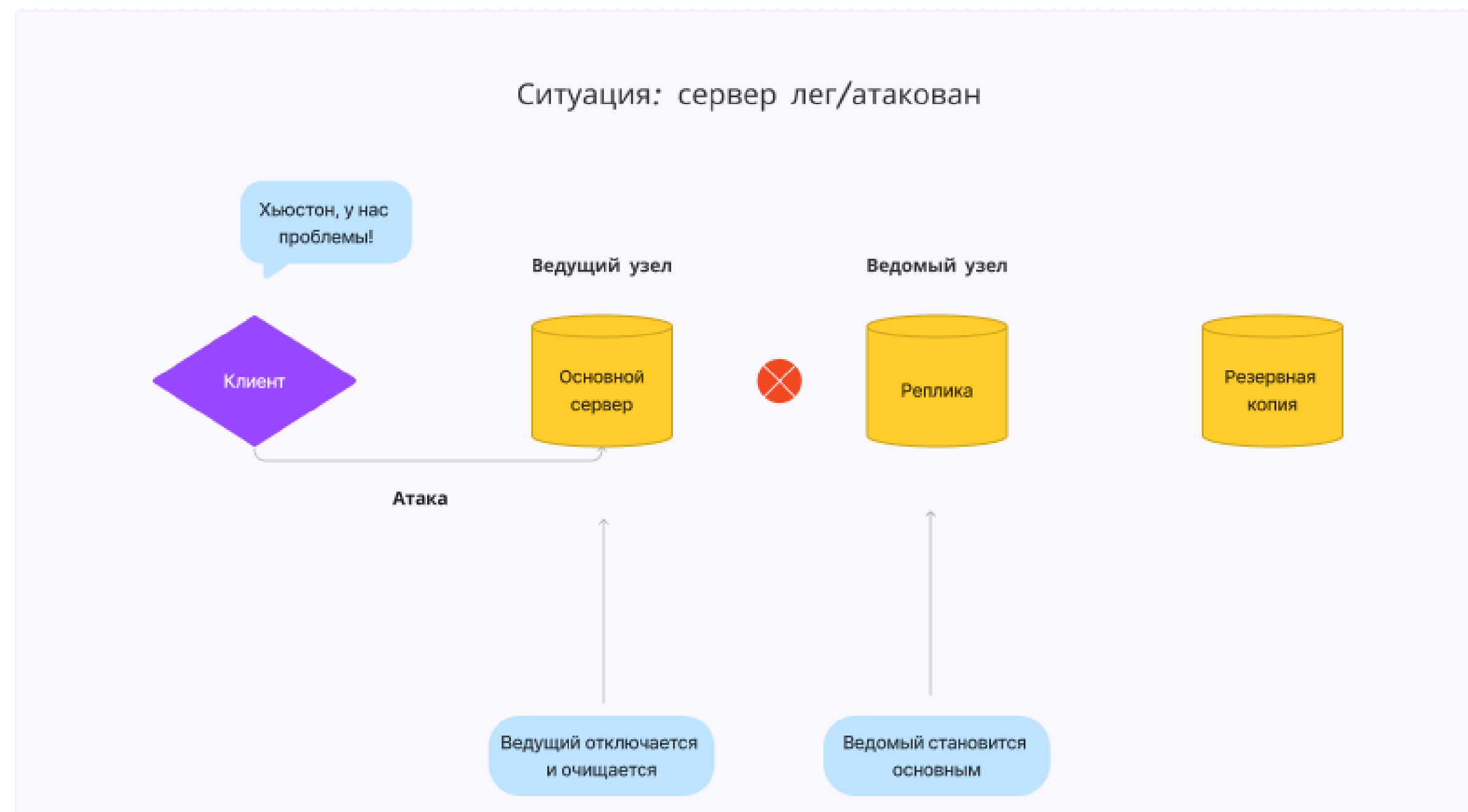
Архитектура



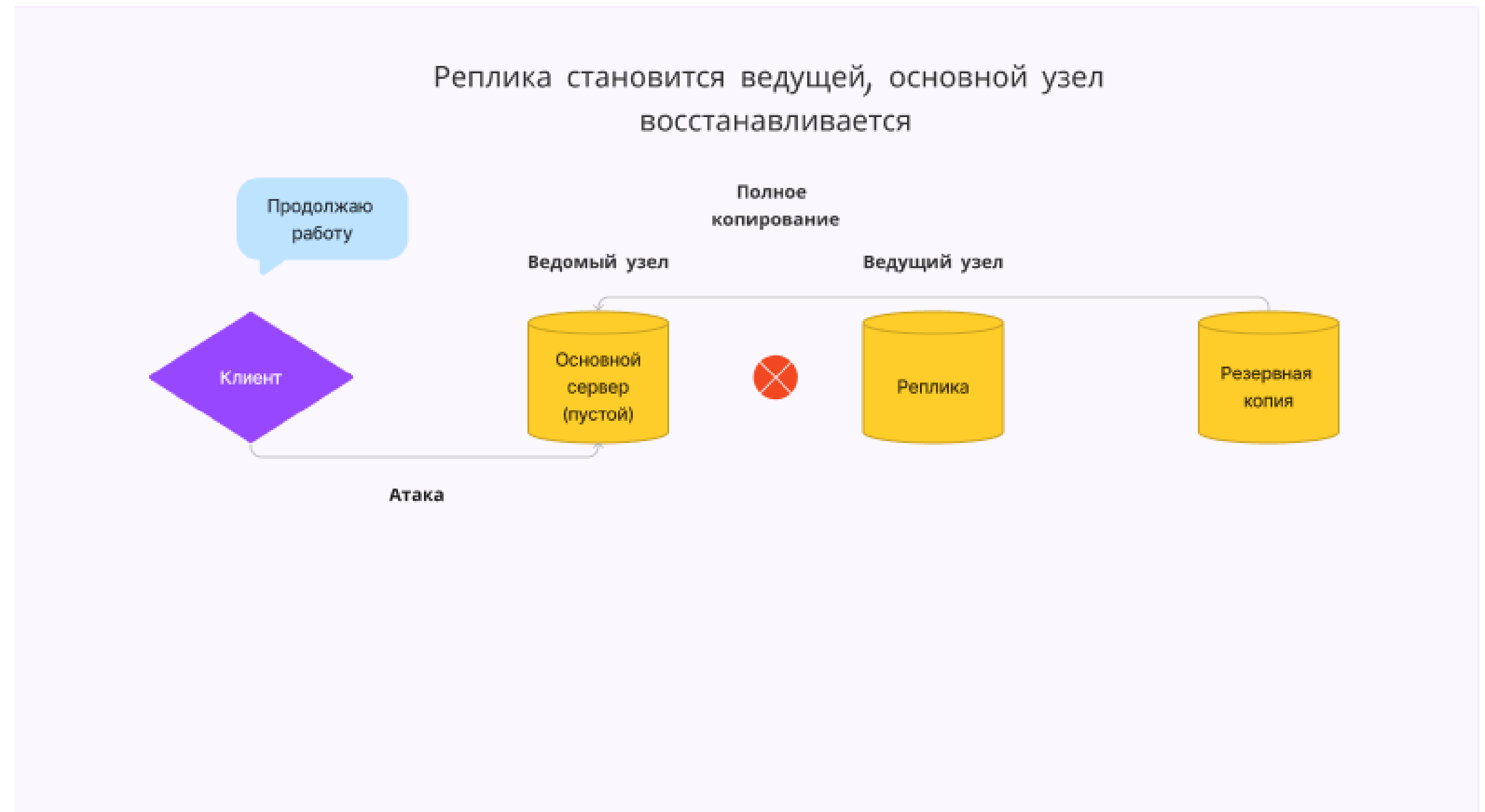
Архитектура



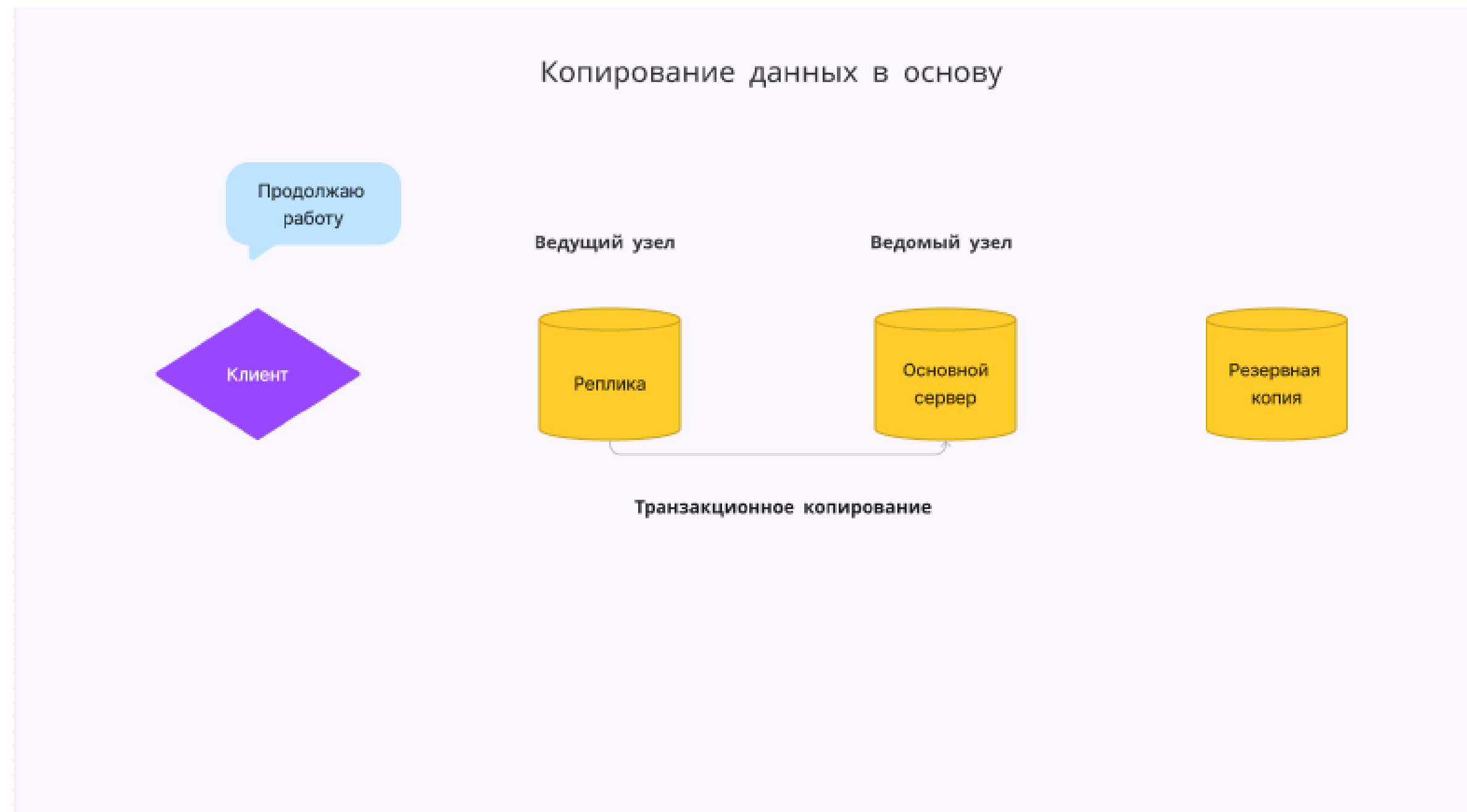
Архитектура



Архитектура



Архитектура





СЕРВЕР

DELL R760 24SFF

2x Intel Xeon Gold 6458Q (32C 60M Cache 3.10 GHz) / 512 (8x 64GB) / H755 / Samsung 990 pro HDD 2.5" 1.8TB SAS 10K + салазки / 2x1400W



HDD 24 BAY24 × HDD 2.5" 1.8TB SAS 10K + салазки



это один из самых популярных и широко используемых дистрибутивов Linux. Он имеет большое сообщество разработчиков и пользователей, что обеспечивает поддержку и обновления ОС.

RPO, RTO

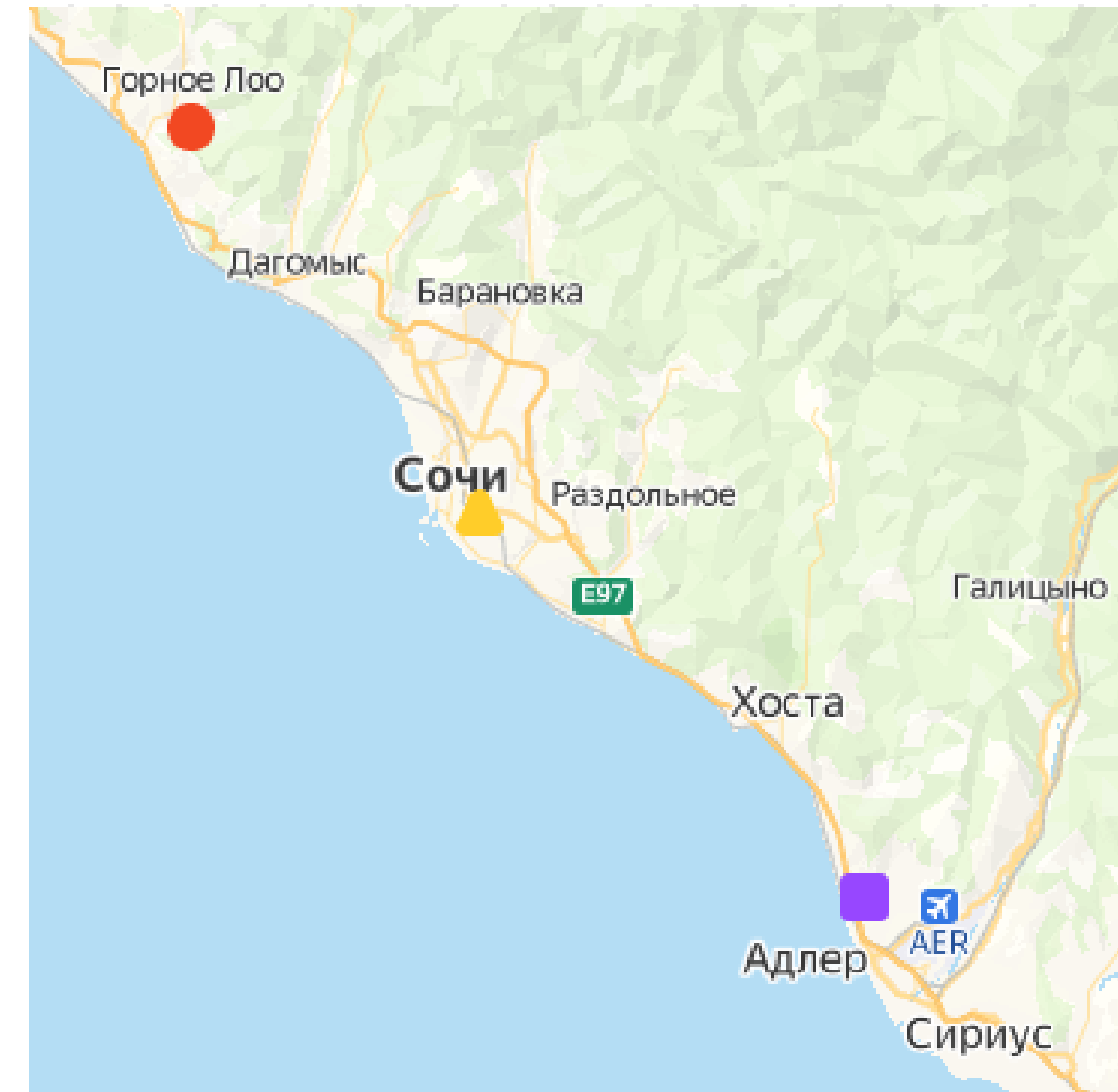
в идеальных условиях SSD пишет 19800 мб/с.
у нас 6 кабелей по 10 гигабит и 6 кабелей по 1
гигабиту. значит у нас потолок передачи данных
 $66 \text{ гбит/с} = 8.25 \text{ гигабайта/с}$

Таким образом, база данных полностью
скопируется за $5500 / 8.25 / 60 = 11.4$ минуты

В случае с записью мы также упираемся в лимит
кабелей

Связь между серверами

Основной сервер ставим в Адлере, в центре резерв, в Лоо репликант. Между ними провести высокую производительности можно рассмотреть использование оптической сети связи на основе технологии DWDM (Dense Wavelength Division Multiplexing).



План бекапа в случае сбоя системы:

01 Реплика назначается ведущим узлом (автоматически)

02 Очищаем данные из основной базы данных

03 Из дополнительного сервера, на котором лежит резервная копия данные передаем в первоначальную бд до полного восстановления.
Теперь узел, который был ведомым становится репликой

≈39,686,600

Полные затраты

Взаимодействие клиента с сервером



Приложение будет использовать API для взаимодействия с сервером и получения данных из базы данных.



Для наилучшей работы лучше использовать RESTful API, и для получения данных из базы данных отправлять HTTP запросы (GET, POST, PUT, DELETE) на определенные эндпоинты сервера.



Для обработки и передачи данных между клиентом и сервером может использоваться JSON формат.



Если система переключается на реплику базы данных при сбое, приложение может продолжать работать с репликой, используя тот же API и методы взаимодействия.

