

RedMadRobot

Билайн®



Johnson & Johnson



Ренессанс®
страхование

TELE2

M.video

победа●●●



xerox 



ABBYY®



Коммерсантъ®

@mail.ru®



P&G




Pernod Ricard


STADA

LAND-ROVER



Ford



Агенда

- Persistence
- Идеология DAO
- Demo
- Недостатки

Персистентность ЗАДАЧИ

- Сохранить данные
- Обновить данные
- Читать сохранённые данные
- Удалить сохранённые данные
- Синхронизировать данные с сервером

Персистентность

Подходы к хранению

- Каждый раз запрашиваем все
- Запрашиваем то, чего не хватает
- Сложная БЛ хранения и запросов

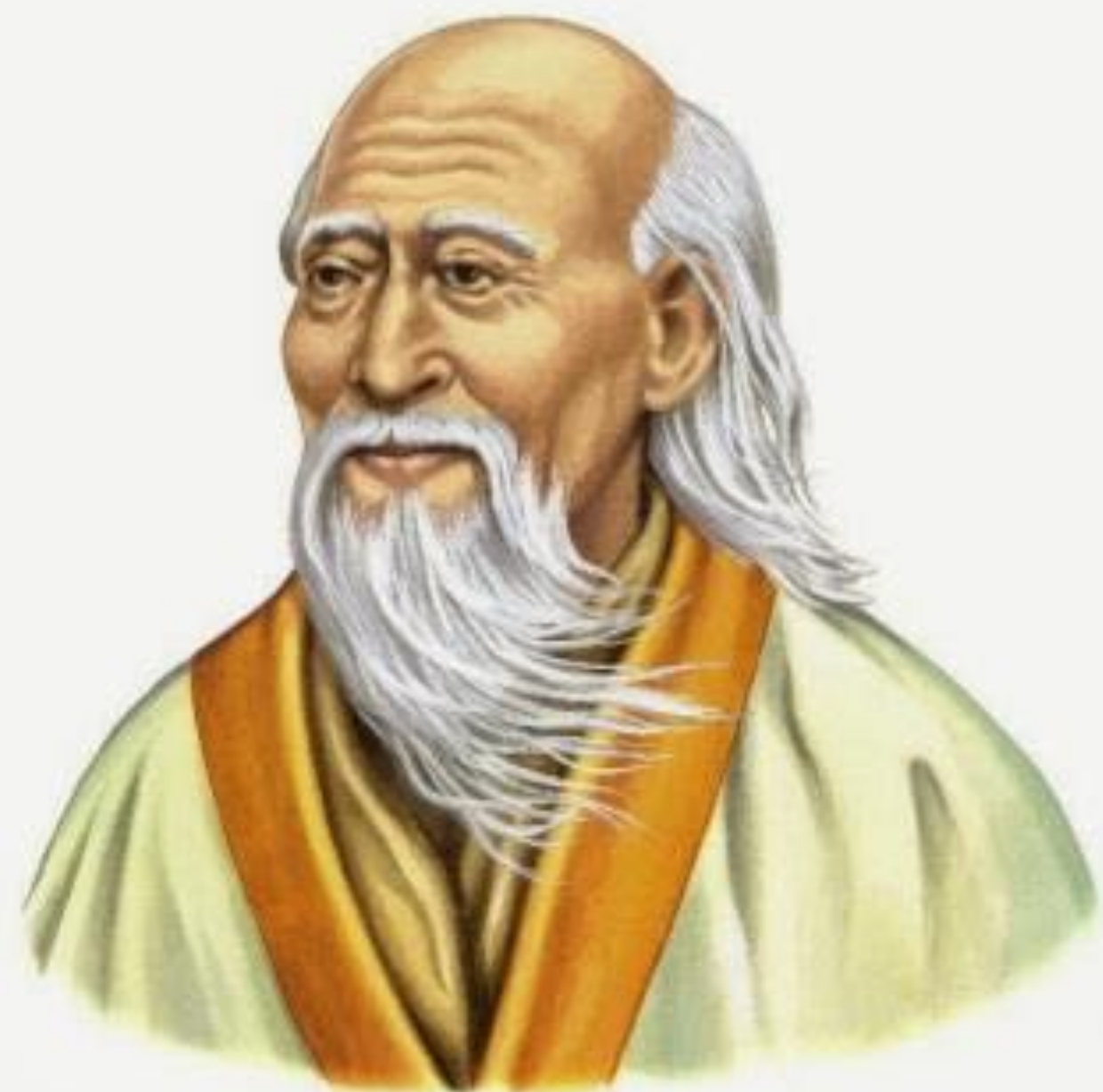
Синхронизация данных с сервером

ПРИМИТИВНЫЙ СЛУЧАЙ

- GET /entities -> { headers } { body }
- HEAD /entities -> { headers } Etag
- GET /entities If-Modified-Since: 1454590970 -> 304 Not Modified

Основные термины

- **DAO**
- **Entity**
- **Entry**
- **Translator**



<http://bit.ly/2nXspsh>

06

REDMUDROBOT

DAO

ИДЕОЛОГИЯ

- **Взаимодействие с DAO посредством легковесных объектов (структур)**
- **Инжектируемость**
- **“Соединение” с БД**
- **Потокобезопасность**
- **Транзакционность**

DAO ИНТЕРФЕЙС

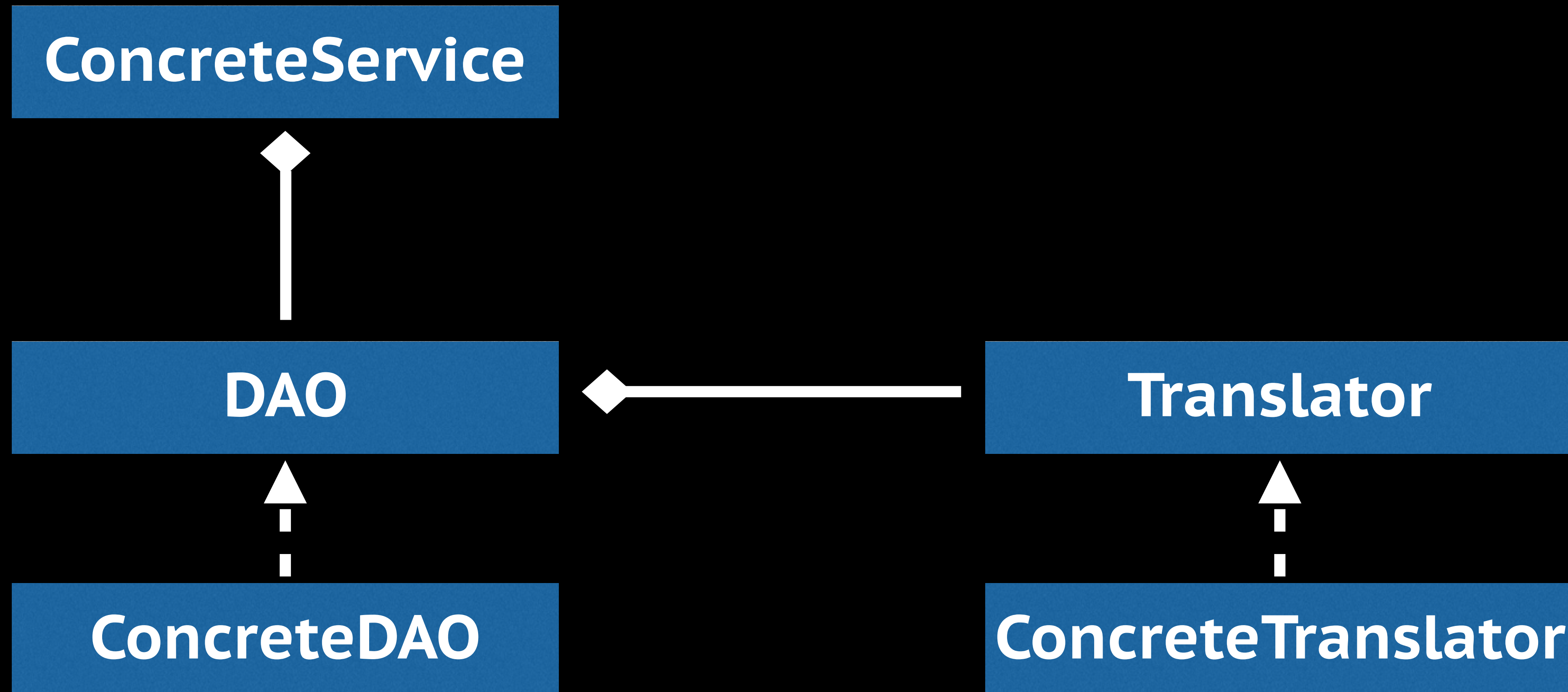
- `persist(Entity) -> Bool`
- `persist([Entity]) -> Bool`
- `read(ID) -> Entity`
- `read() -> [Entity]`
- `erase(ID) -> Bool`
- `erase() -> Bool`

Translator

ПРИМЕР ИНТЕРФЕЙСА.

- toEntry(*Entity*) -> *Entry*
- toEntries([*Entity*]) -> [*Entry*]
- toEntity(*Entry*) -> *Entity*
- toEntities([*Entry*]) -> [*Entity*]

DAO



Demo

<http://bit.ly/2oDXroN>

Android

<http://bit.ly/2naxAI2>

Недостатки

- Невозможность использовать особенности конкретной БД
- Производительность на большом количестве объектов
- Дублирование кода
- Несколько соединений к БД

ИТОГИ

- Вход/выход из фонового потока координируется сервисами
- Изолированная база данных = безопасная база данных
- Может не подойти под конкретный проект

Иван Вавилов
iv@redmadrobot.com

REDMADROBOT