

# Яндекс

организация асинхронной  
обработки задач

# Зачем

Не сломаться под нагрузкой

Справиться с ошибками

Я

# Тестирование

Многопоточность

Производительность

Ошибки

Все задачи выполняются  
ровно один раз

Я

# Архитектура

Записываем задачи в storage

Скорость обработки  
фиксирована

Пишем изо всех оставшихся  
сил

Я

# Storage

ID

Когда выполнить

Сколько попыток было

Задача

Я

# Обработка

Заблокировали

Увеличили время

Разблокировали

Выполнили

Удалили

Я

# Запись: масштабирование

Сколько insert'ов в секунду  
выдержит ваша база?

Несколько независимых баз  
+ Load Balancer!

# Ошибки

Увеличиваем время  
ожидания обработки задачи  
в два раза после каждой  
ошибки

Я

# Задержки

После ошибки

После пустой очереди

После выполненной задачи

# Мониторинг

Сколько задач

Сколько ошибок

Количество попыток

Я

# Скорость реакции

Нужно обрабатывать в  
несколько потоков

Я

# Oracle AQ

Очень быстро

Очень дорого

Очень сложно

Я

# Oracle

```
declare
  rid NUMBER := 0;
  rtask VARCHAR2(4000) := "";
  cursor c is select id, task from tasks where
    process_time <= sysdate for update skip locked;
begin
  open c;
  fetch c into rid, rtask;
  if (c%NOTFOUND) then
    rid := 0;
  end if;
  close c;
  if (rid > 0) then
    update tasks set process_time = sysdate
+ (power(2, attempt))/1440, attempt = attempt + 1 where id = rid;
  end if;
  ? := rid;
  ? := rtask;
end;
```

Яков Сироткин

Я