

Наши баги за 2 года разработки на Haskell

Катерина Галкина

GetShop.TV

02.12.2017



Обо мне

- GetShop.TV
- ВМК МГУ
- Пишу на Haskell с 2015
- Люблю ФП с 2014



fierce-katie



fierce_katie@xmpp.ru

Ошибки, которых у нас не было

- Непредвиденное создание нового поля структуры
- Использование неинициализированных значений
- Необработанные варианты значений параметров
- Непредсказуемое приведение типов

Откуда берутся ошибки?

- Человеческий фактор
- Сложная логика
- Фатальное недопонимание
- Raw SQL, скрипты на bash, python, и т.д.

База данных

Что случилось?

- Несоответствие типов
- Несоответствие схемы БД (primary key, foreign keys)
- Выкатили на production, “всё зависло” :(

Как починили?

```
newtype UserId
    = UserId UUID

data User = User
    { userName      :: Text
    , userSurname   :: Text
    , userBirthday  :: Maybe Date
    , userLastLogin :: Maybe UTCTime
    }
```

```
CREATE TABLE user (
    id uuid PRIMARY KEY
        DEFAULT uuid_gen(),
    name text NOT NULL,
    surname text NOT NULL,
    birthday date,
    last_login timestamp
);
```

Как починили?

```
newtype ContactId
    = ContactId UUID

data Contact = Contact
    { contactMail    :: Text
    , contactPhone   :: Int64
    }
```

```
CREATE TABLE contacts (
    id uuid PRIMARY KEY
        DEFAULT uuid_gen(),
    mail text NOT NULL,
    phone integer NOT NULL,
    user_id uuid NOT NULL
        REFERENCES users(id)
);
```


Как починили?

```
data TableSchema = TS Type Type [Type]
data ForeignKeyInfo = FKInfo Symbol Type
type MySchema =
  '[ TS User UserId '[ ]
    , TS Contact ContactId '[UserId]
  ]
type instance GetForeignKeyInfo UserId
  = FKInfo "user_id" User
```

Как починили?

- Тесты на соответствие типов
- Проверка соответствия ключей
- Заполнение БД перед тестами

Ошибки в логике

Что случилось?

- Ошибка, зависящая от времени старта сервера

```
pattern = [00:30, 02:30, 04:55]
startTimes = [00:00:30, 00:02:30, 00:04:55, ...
               23:55:30, 23:57:30, 23:59:55,
               00:00:30, ... ]

now :: TimeOfDay = serverStartTime + delta*n
until :: TimeOfDay = now + delta
events = takeWhile (< until) $
         dropWhile (< now) startTimes
```

Что случилось?

```
startTimes = [00:00:30, 00:02:30, 00:04:55, ... , 23:54:55,
              23:55:30, 23:57:30, 23:59:55,
              00:00:30, ... ]

events = takeWhile (< until) $ dropWhile (< now) startTimes
now = 23:54:58
until = 23:59:58

events = takeWhile (< 23:59:58)
    [23:55:30, 23:57:30, 23:59:55, 00:00:30, ... ]

--                               ^^^^^^^^

--  МАКСИМАЛЬНО ВОЗМОЖНОЕ ЗНАЧЕНИЕ
```

Как починили?

```
pattern = [00:30, 02:30, 04:55]

maxStartTime = 23:55:00 + (maximum pattern) -- 23:59:55

until :: TimeOfDay = min (now + delta) maxStartTime

events = take maxLength $
    takeWhile (< until) $
    dropWhile (< now) startTimes
```

Что случилось?

- Потеря данных из-за некритичных параметров в неверном формате

```
/stats?device_id=127001&engagement=3&timestamp=<invalid>
```

```
postStats :: Maybe DeviceId    -- Required
```

```
    -> Maybe Engagement -- Optional
```

```
    -> Maybe UTCTime      -- Optional
```

```
    -> Handler NoContent
```

```
postStats (Just deviceId) maybeEng maybeTime = ...
```

```
postStats _ _ _ = throwError err400
```

Как починили?

```
type StatsAPI = "stats"

  :> RequiredQueryParam "device_id" DeviceId
  :> LenientQueryParam  "engagement" Int
  :> LenientQueryParam  "timestamp" UTCTime
  :> PostNoContent

postStats :: DeviceId          -- Required
          -> Maybe Engagement -- Lenient
          -> Maybe UTCTime    -- Lenient
          -> Handler NoContent

postStats deviceId maybeEng maybeTime = ...
```


Уведомления об ошибках

Что случилось?

- Давно не было сообщений об ошибках: всё слишком хорошо? (нет)
- 5xx вместо предупреждений

Что случилось?

```
template.yaml.j2:
# Mail service settings
mail:
  sender:
    name: "{{ mail.sender.name }}"
    mail: "{{ mail.sender.mail }}"
    user_name:
      "{{ mail.sender.user_name }}"
    password:
      "{{ mail.sender.password }}"
  maintainers:
    {{ mail.maintainers }}
```

```
dev.yml:
  mail:
    sender:
      name: "Dev Server"
      mail: "support@host.com"
      ...
    maintainers: ['me@host.com',
                  'cto@host.com']
```

```
test.yml:

  mail:
    sender:
      name: "Test Server"
      mail: "support@host.com"
      ...
    maintainers: ['me@host.com']
```

Что случилось?

- CI server + jinja2 + template + vars = ???

```
settings_test.yaml (local):  
  
# Mail service settings  
mail:  
  sender:  
    name: "Test Server"  
    mail: "support@host.com"  
  maintainers: ['me@host.com']
```

```
settings_dev.yaml:  
  
# Mail service settings  
mail:  
  sender:  
    name: "Dev Server"  
    mail: "support@host.com"  
  maintainers: [ u'me@host.com'  
                 , u'cto@host.com' ]
```

Что случилось?

- “Caught exception on server at dev.ourserver.com: user error (Error in \$: key "notification_address" not present)”



fierce-katie commented on Jul 20



It's a feature.

Как починили?

- Генерируем конфиги с помощью ansible
- Уведомления о разных уровнях ошибок

Самокомментирующийся код

Что случилось?

```
sms_timeout: 1800  
duration: 600  
update_interval: 60  
validation_timeout: 300  
verification_timeout: 3000  
status_update_timeout: 300  
video_start_interval: 300
```


Что случилось?

```
sms_timeout: 1800
duration: 600
update_interval: 60
validation_timeout: 300
verification_timeout: 3000
status_update_timeout: 300
video_start_interval: 300
```

```
threadDelay :: Int -> IO ()
             ???
```

Как починили?

```
newtype SecondsInterval = SecondsInterval Int
newtype Microseconds = Microseconds Int
newtype Percentage = Percentage Float
checkSmsStatus :: SmsId -> SecondsInterval -> Handler ()
```

Критические ошибки

Ошибки на production

- Потеря информации
- Медленная работа БД
- Отсутствие обратной совместимости
- Отсутствие уведомлений
- “Duplicate key value violates unique constraint”

PROFIT???

GitHub issues

- Отмечены как bug: 72
- Имеют отношение к работе сервера: ~35
- Дошли до production: ~20
- Критические ошибки: 5-7

Спасибо за внимание!