

Яндекс

Архитектура высоконагруженного сервиса на примере бэкенда Яндекс.Store

Андрей Урусов, Алексей Васильев

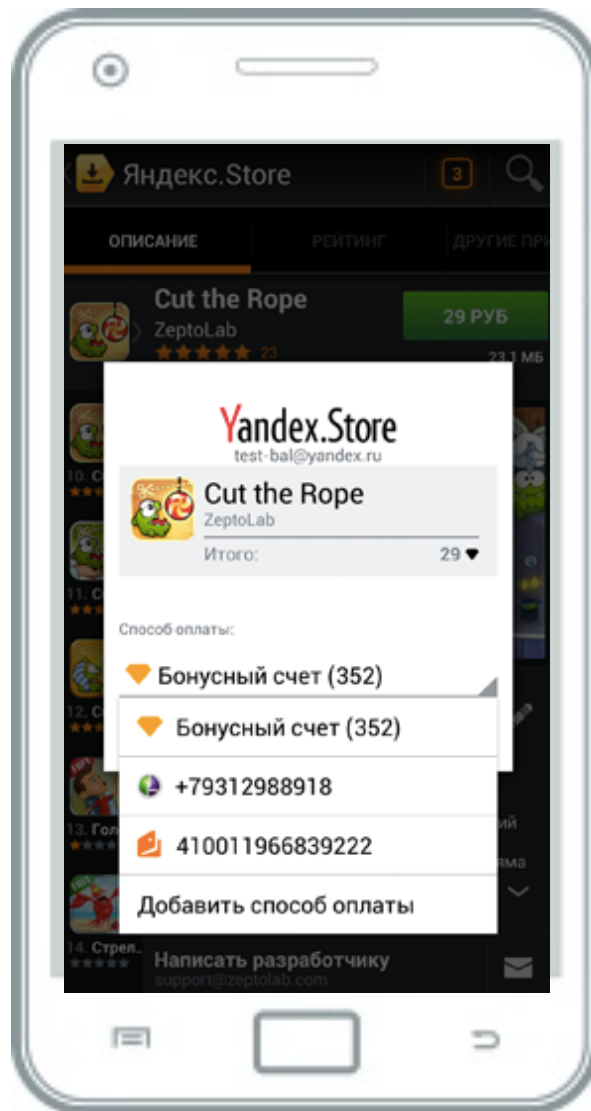
Магазин приложений для Android



Магазин приложений для Android



Магазин приложений для Android



Процесс разработки

› Непрерывная интеграция

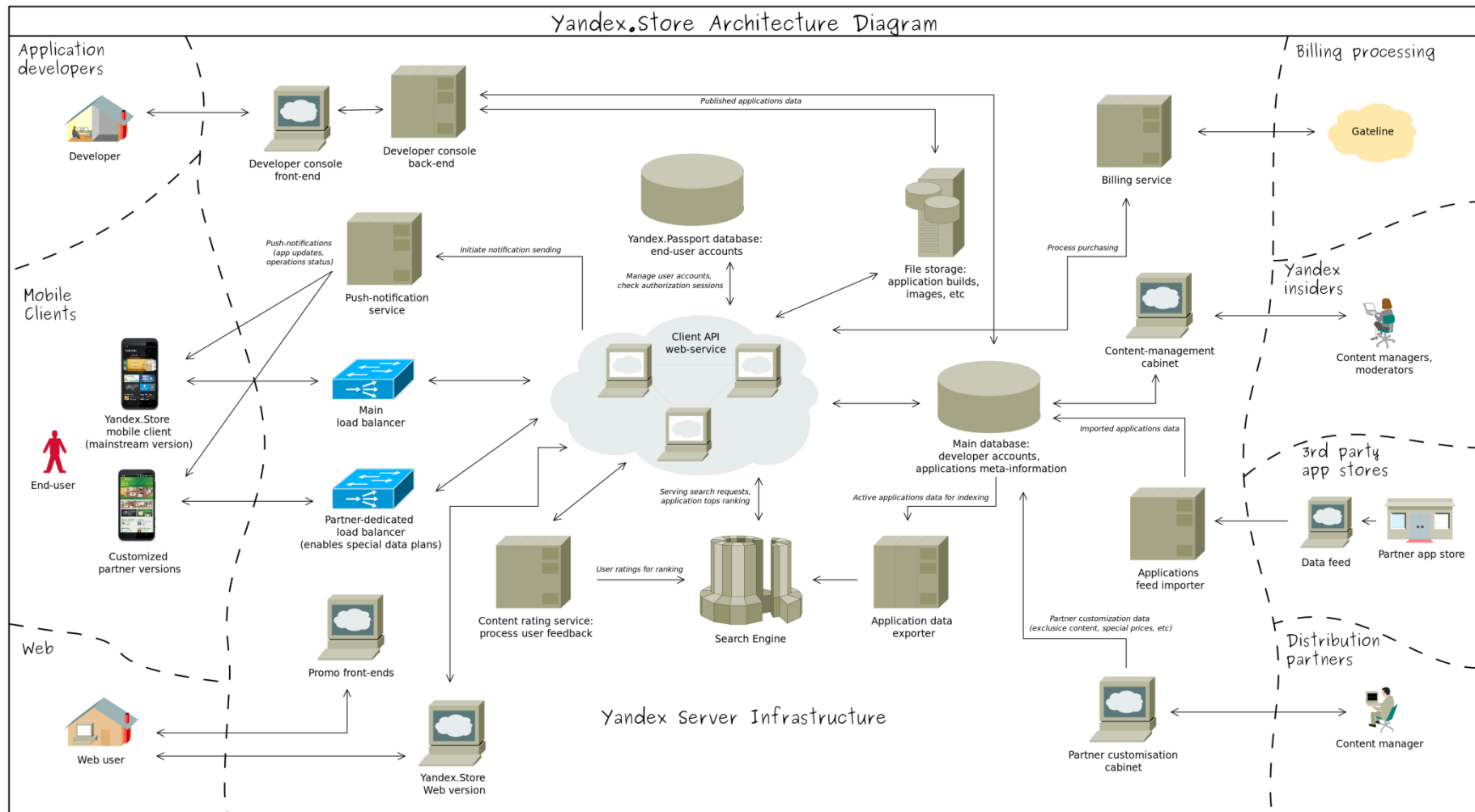
- Система контроля версий — git
- Автосборка — TeamCity
- Код-ревью — Stash
- Автотесты — Boost Unit Test Framework

› Нагрузочное тестирование — Яндекс.Танк

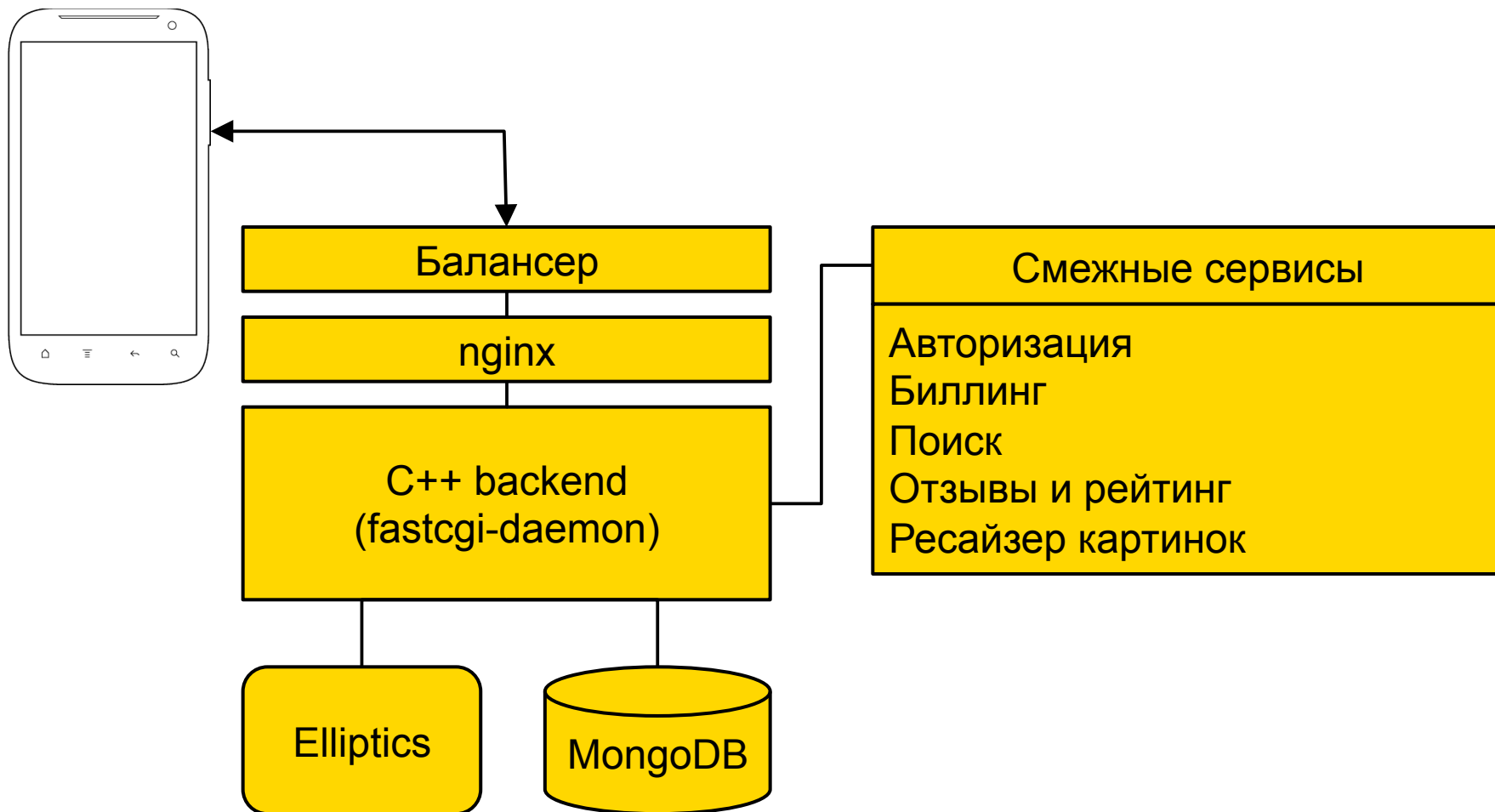
› Разработка в облаке — X11/ssh

› Пятничные холивары

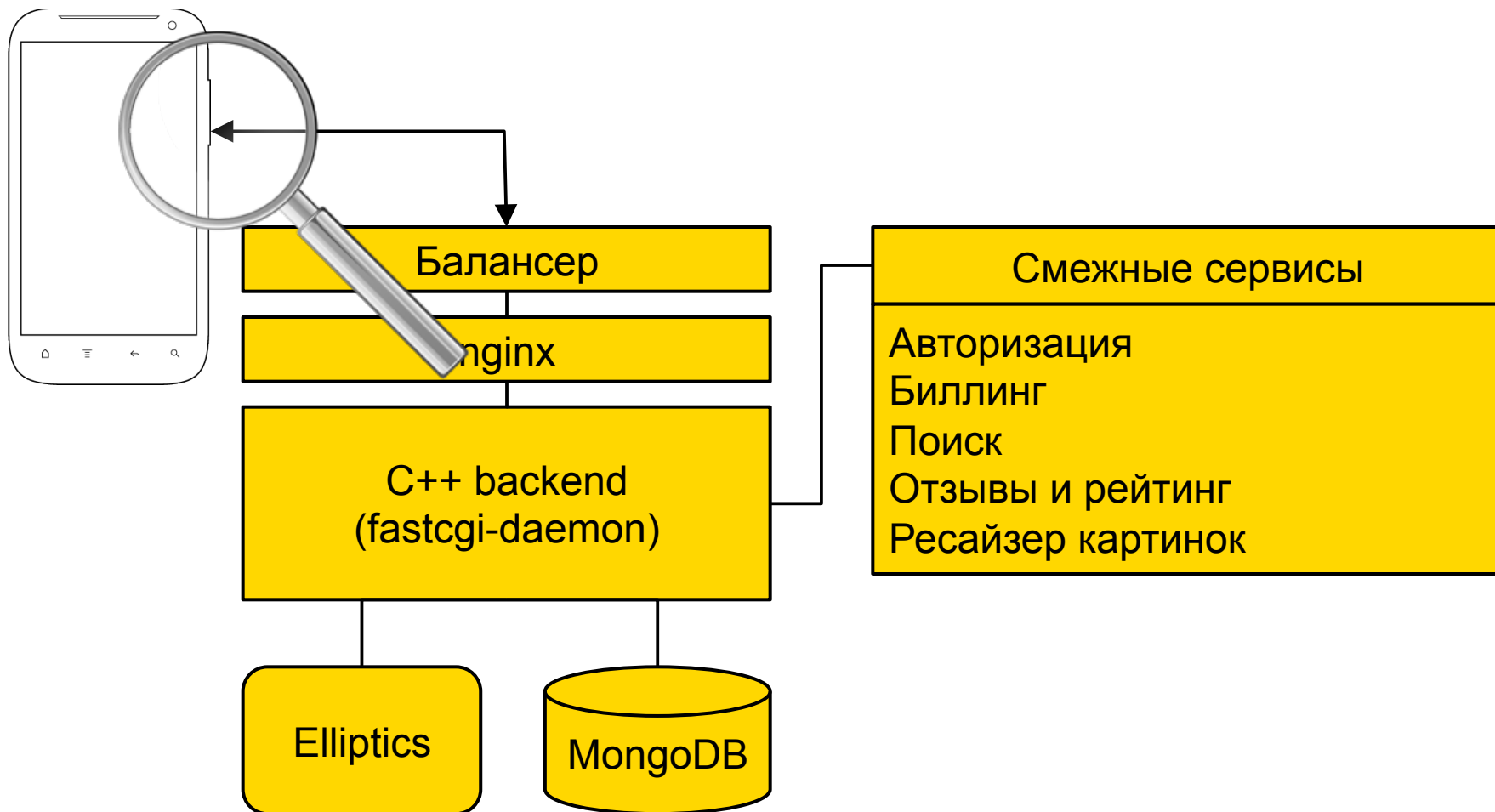
Архитектура системы



Архитектура API



Архитектура API



Protobuf

```
message Date
{
    required uint32 day = 1;
    required uint32 month = 2;
    required uint32 year = 3;
}

message Event
{
    required Date date = 2;
    required string description = 3;
}

message Log
{
    repeated Event events = 1;
}
```

```
std::unique_ptr<Log> log (new Log);

Event* event = log->add_events();

event->mutable_date()->set_day(20);
event->mutable_date()->set_month(9);
event->mutable_date()->set_year(2014);

event->set_description("Я.Субботник");

log->SerializeToOstream(&std::cout);
```

Обратная совместимость

```
message Product
{
    enum ImageSize
    {
        DEFAULT = 0;
        SMALL = 1;
        MEDIUM = 2;
        LARGE = 3;
    }
    optional ImageSize image_size = 1
    [default = DEFAULT];

    required bytes image = 2;
}
```

```
message Product
{
    enum ImageSize
    {
        DEFAULT = 0;
        SMALL = 1;
        MEDIUM = 2;
        LARGE = 3;
        XLARGE = 4;
    }
    optional ImageSize image_size = 1
    [default = DEFAULT];

    required bytes image_landscape = 2;
    required bytes image_portrait = 3;
}
```

Наследование

```
message Application
{
    required uint32 version = 1;
}

message Subscription
{
    required Time period = 1;
}

message Product
{
    required string title = 1;
    required Price price = 2;

    optional Application application = 3;
    optional Subscription subscription = 4;
}
```

Наследование

```
message Product
{
    required string title = 1;
    required Price price = 2;

    enum Type
    {
        APPLICATION = 0;
        SUBSCRIPTION = 1;
    }
    required Type type = 3;

    extensions 100 to max;
}
```

```
message Application
{
    extend Product
    {
        optional uint32 version = 100;
    }
}

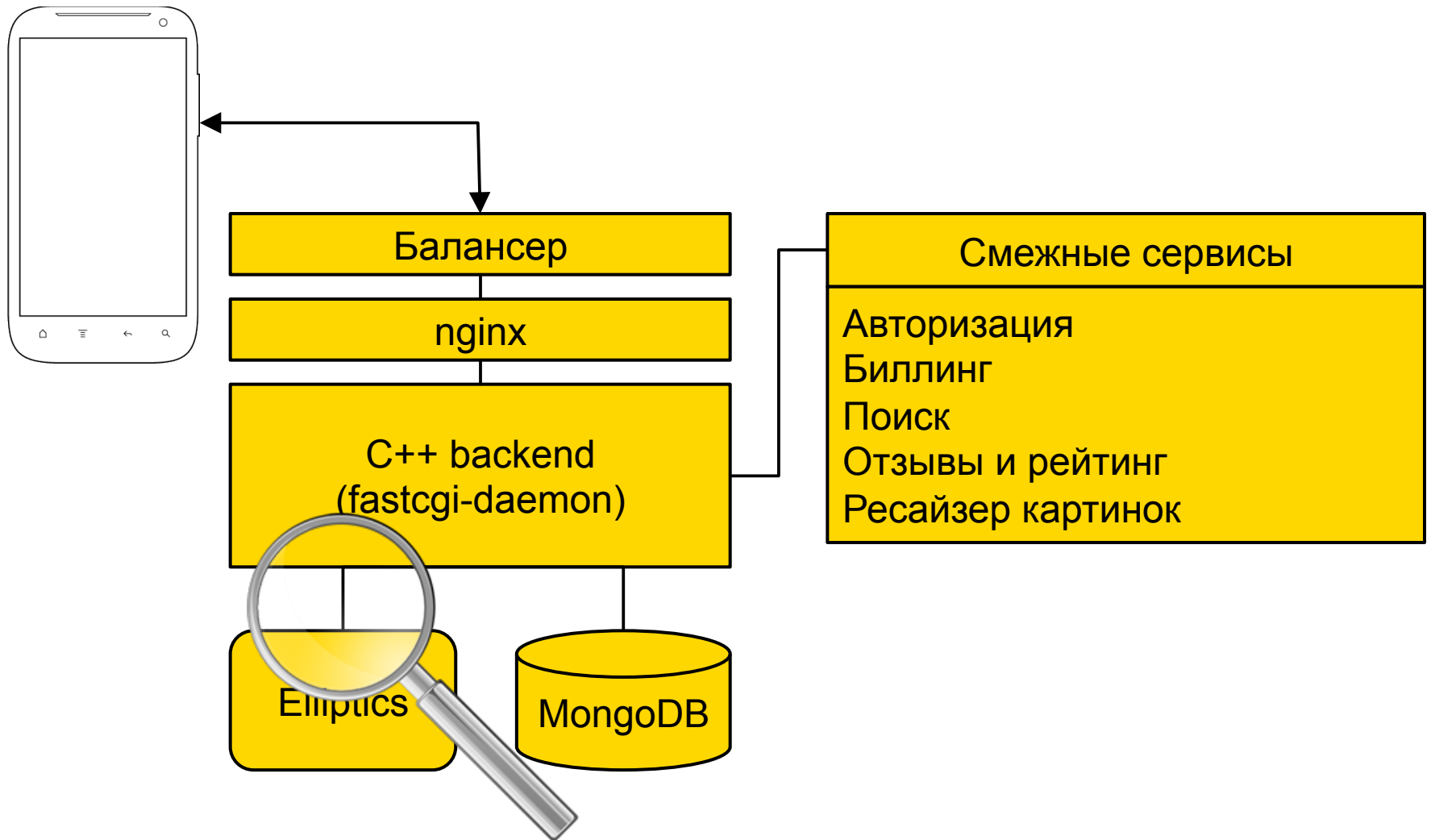
message Subscription
{
    extend Product
    {
        optional Time period = 200;
    }
}
```

Наследование

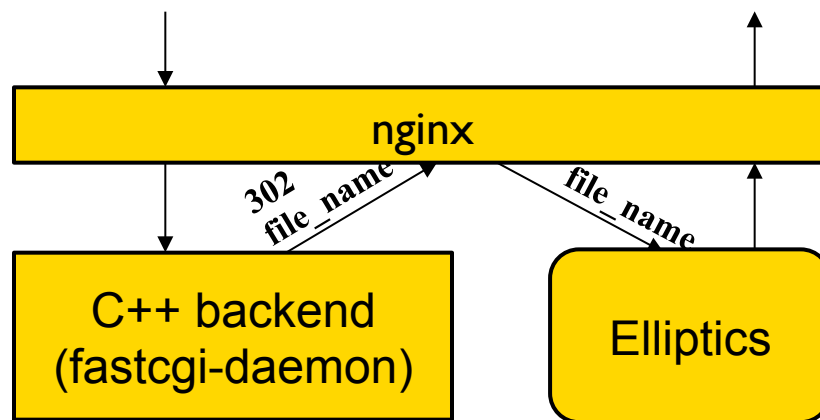
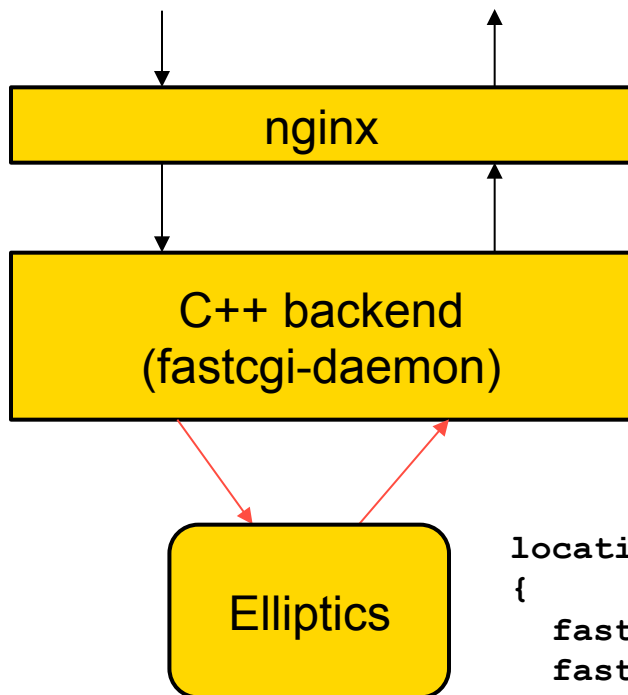
<http://www.indelible.org/ink/protobuf-polymorphism/>

<https://clck.ru/9Jknx>

Архитектура API



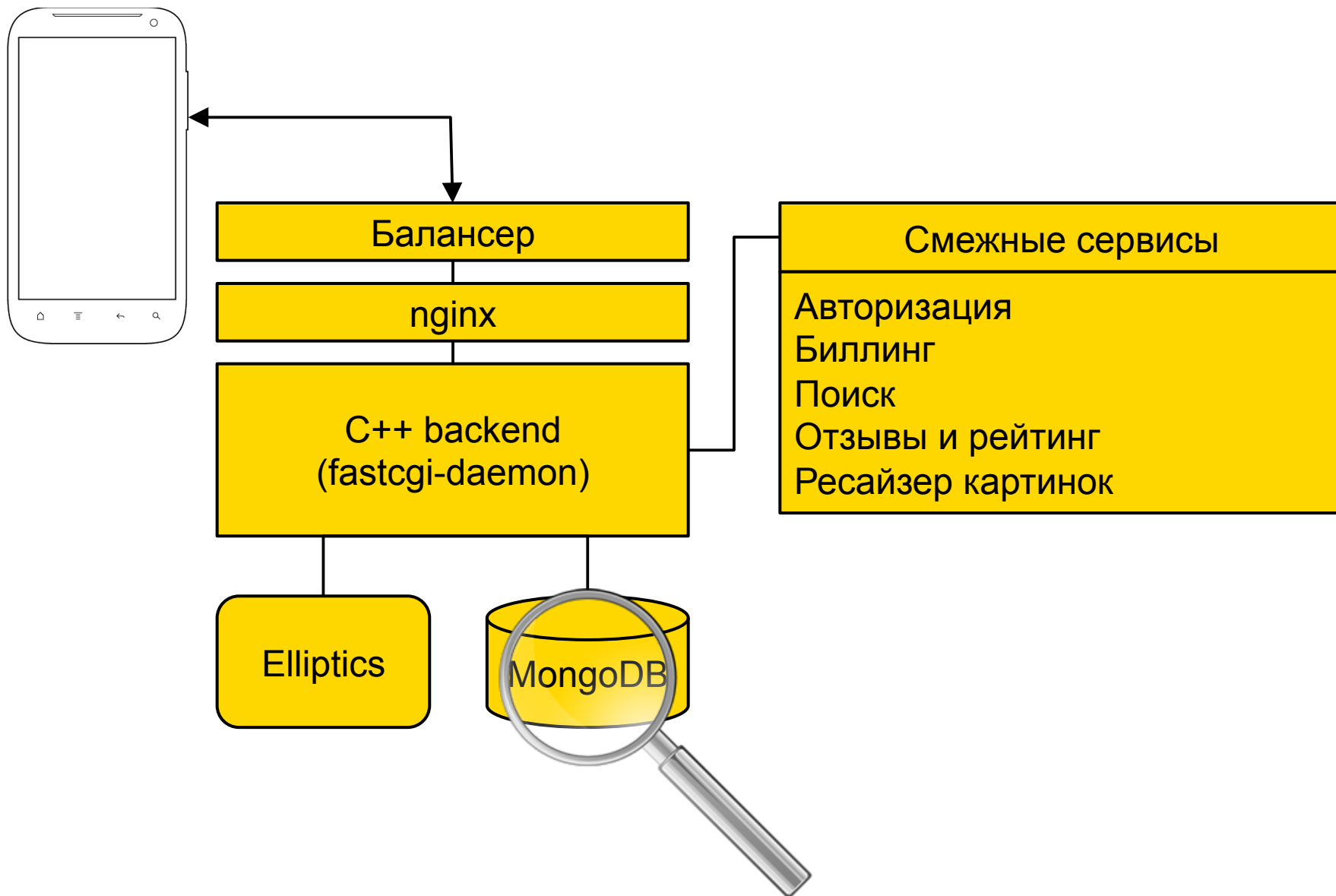
Прокси с авторизацией



```
location /download
{
    fastcgi_pass your_download_authORIZER;
    fastcgi_intercept_errors on;
    error_page 302 = @storage;
}

location @storage
{
    rewrite ^ /get?file_name=$upstream_http_file_name break;
    proxy_pass http://your_storage;
}
```


Архитектура API



Драйвер MongoDB

```
mongo::DBClientConnection connection;  
connection.connect(config->GetDbAddress());  
connection.auth(  
    config->GetDbName(),  
    config->GetDbUser(),  
    config->GetDbPassword(),  
    ...);  
  
// hardcoded  
connection.findOne("db_name.collection_name", ...);  
  
// inconvenient  
connection.findOne(config->GetDbName() + "collection_name", ...);
```

Драйвер MongoDB

Assertion: 13111:
wrong type for field (title) 2 != 16



Алексей Васильев
loudhorr@yandex-team.ru



Андрей Урусов
drobus@yandex-team.ru