

Django DB Mailer - простая и удобная батарейка, для отправки сообщений в ваших проектах

... и не только на Python

© Ruslan Askarov, 2015



<http://www.devconf.ru>

Давайте знакомиться

- Меня зовут Руслан.
- Работаю в компании LPgenerator.
- Пишу всяко разное на Python, Go, Lua and etc.
- Не только Web, но и GUI, и злых демонов с “зомбаками” ...
- Кручу-верчу непонятные Linux-ы/Макос-ы и ручки на форточках
- Всякие там Ansible, Vagrant, Docker, OpenVZ ...
- Ну и еще какие-то страшные и непонятные слова ...
- Излюбленное - это HighLoad и отказоустойчивость.
- И конечно же OpenSource.

LPGENERATOR - Что это и зачем? технологии привлечения клиентов

Облачная платформа для комплексной лидогенерации:

- WYSWYG редактор целевых страниц (лендингов);
- Центр оптимизации конверсии и сплит-тестирования;
- Удобная CRM-система управления лидами;
- Закупка контекстной рекламы внутри платформы;
- Огромный выбор шаблонов на разные темы

В конечном итоге:

- SaaS - Software as Service

В цифрах

- 200К+ пользователей
- 13М+ целей, из них 6М+ лидов
- 150Gb+ данных

В сутки:

- 5М+ запросов к бекенду
- 18М+ запросов к cdn
- 1К+ новых страниц
- 15К+ писем





Часть первая: История

История Django DB Mailer. С чего это? И когда это?

- Легаси код проекта 3-х летней давности
- Постоянные падения воркеров celery/крон команд (так родился django-celery-mon и django-mmс)
- Много дублирующегося и страшного кода (очень неприятно)
- Постоянные правки шаблонов на файловой системе
- Как итог: еще больше лапши и сумбура
- После коммиты, тесты, билды ...
- Деплой и рестарты uWSGI/Celery (аптайм нынче в цене)
- Для уведомления разработчиков костыли и хардкод
- Не было возможности активации и деактивации сообщений

Желания

- В нашей компании главное - это лиды. И потому этот момент является очень важным при взаимодействии с нашими клиентами
- В данный момент еще внедряется SPA (Single page application)
- С введением SPA необходимы уведомления в браузере
- Для мобильного приложения требуются PUSH-уведомления
- В целях безопасности необходим TTS (Text to call)
- Ну и не забывая о разработчиках/админах хорошо бы еще иметь в наличии VoxCar/Prowl/Pushover из коробки

* Кстати идею с бекендами предложил один из первых коммитеров на GitHub

Действительность - рождение Django DB Mailer





Часть вторая: Возможности

Django DB Mailer - для кого и для чего?

- Клиенты / Разработчики / Админы / Отделы
- Нет нужды в правках кода
- Нет рутины и дублирующегося кода
- Не нужно постоянно деплоить
- Вас не отвлекают от основных задач
- Сигналы для людей, а не только для разработчиков
- Отложенные уведомления
- и тд и тп ...

Возможности из коробки

- Простой и удобный функциональный интерфейс, для отправки писем
- Работа в указанной очереди
- Возможность переотправки сообщения в случае неудачи
- Логирование и дублирование писем ВСС
- Категории в админ панели
- Группы получателей (отделы, отдельные люди)
- Сигналы прямо в админке
- Отложенная отправка сообщений (акции и прочее)
- Встроенный браузер полей в моделях

Возможности из коробки

- Система приоритизации
- Редактирование шаблонов прямо в админке
- Шаблоны кешируются (нет обращений к БД)
- Визуальный редактор TinyMCE/CKEditor (dev позволяет больше)
- Версионирование изменений в шаблонах. Возможность откатиться (для модераторов самое оно)
- Локализация админки Русский/English
- Тестирование сообщений прямо из админки
- Из коробки тема django-grappelli
- Типы письма html и text

Возможности из коробки

- API — для совместной работы с приложениями на других языках/платформах
- Возможность задавать шаблону поле «От»
- SMTP настройки для адресатов
- Прикрепление стандартных файлов
- Мультиязычность шаблонов для локализованных сайтов
- Отключение/Включение отправки сообщений
- Отлов и сортировка по исключениям Python
- Поддержка смс/tts/push-уведомлений

Возможности из коробки

- Расширения возможностей, посредством использования собственных бекендов/провайдеров
- Обновление шаблонов в реальном времени

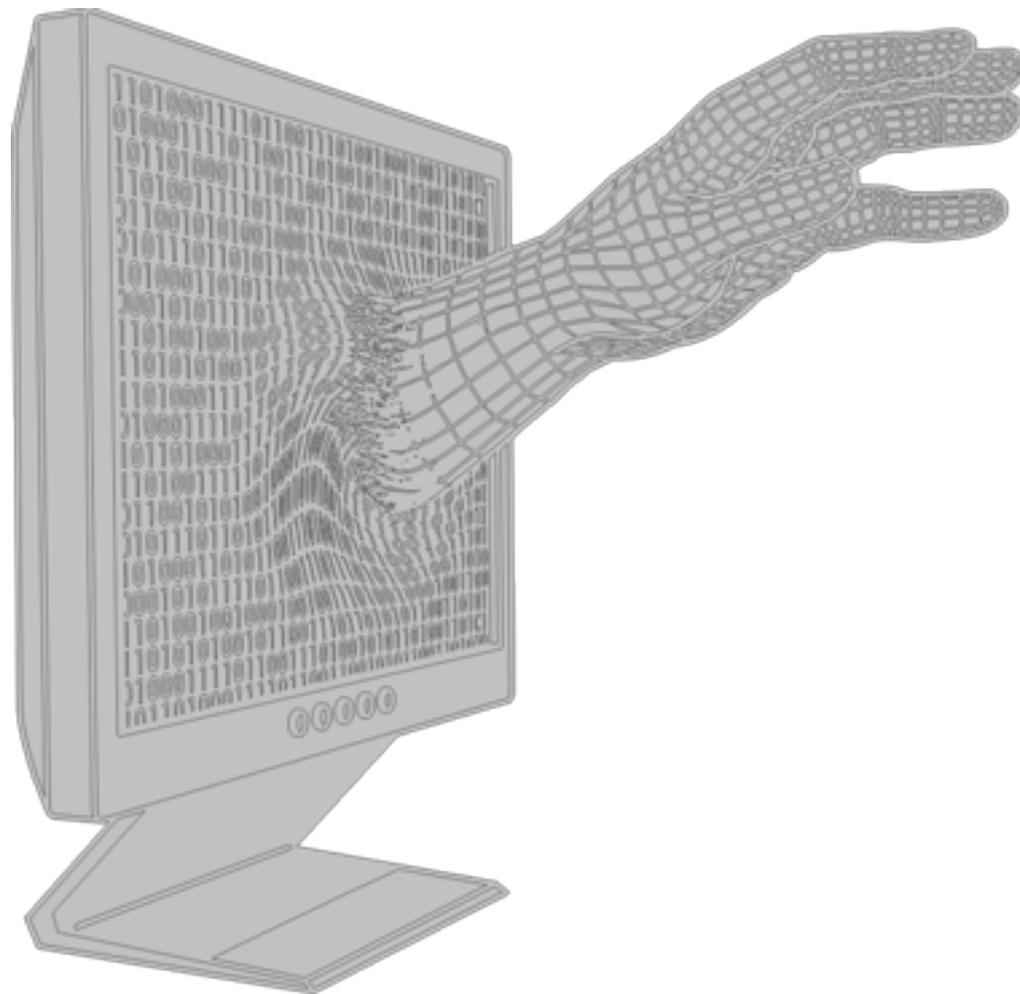
Более подробно в статье на Хабре: <http://habrahabr.ru/post/253445/>

Философия Unix?

- Пишите программы, которые делают что-то одно и делают это хорошо.
- Пишите программы, которые бы работали вместе.
- Пишите программы, которые бы поддерживали текстовые потоки, поскольку это универсальный интерфейс».

00011110 00011110 00011110 00011110 00011110 00011110 00011110 00011110

Решайте сами 😊 Скорее нет, чем да ;)



Часть третья: Подход и Технологии

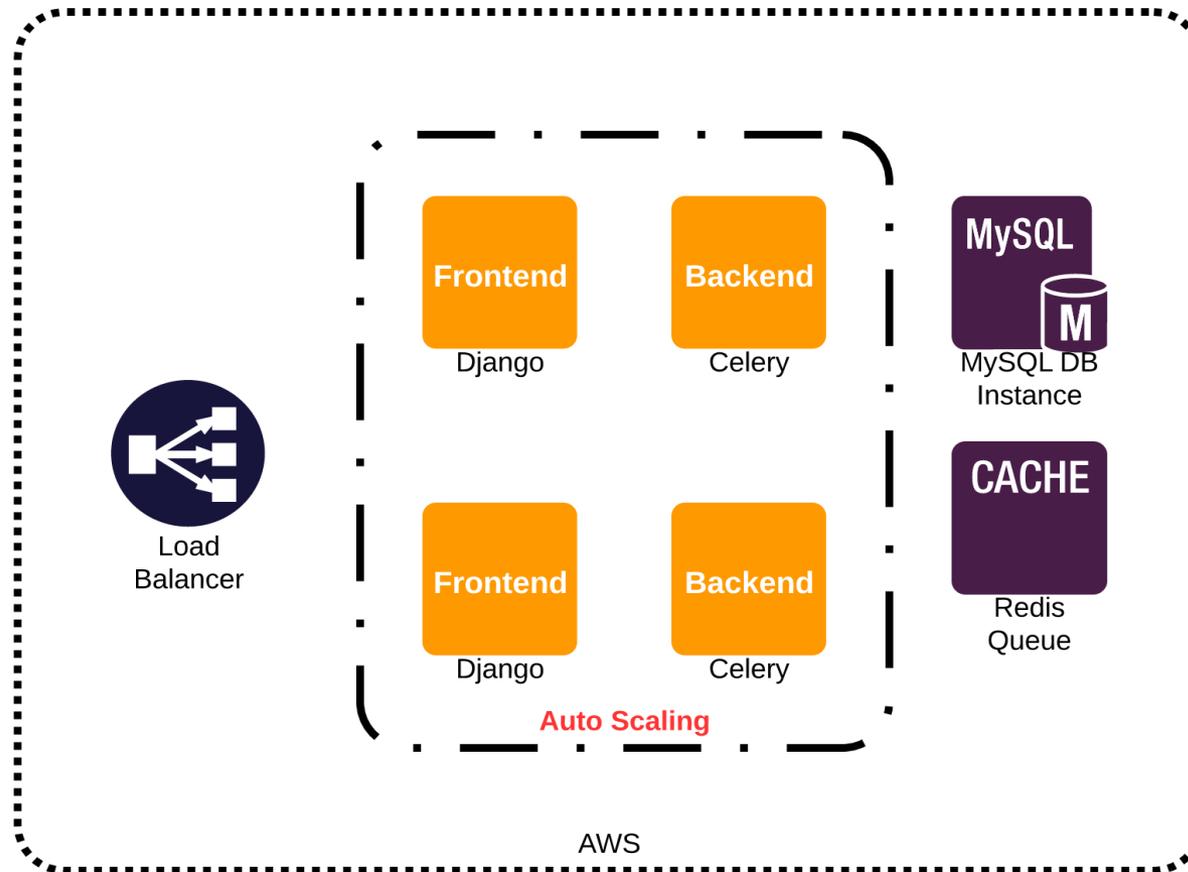
Подход на практике

- Использование как отдельного HTTP сервиса
- Без внешних зависимостей от основного проекта
- При необходимости встраиваем в админ панель основного проекта (используя multiple db)
- Единая и стандартная точка входа для всех уведомлений

Учитываем:

- Возможность переноса в другой ДЦ
- Аварийное переключение (отказоустойчивость)
- Горизонтальное/Вертикальное масштабирование

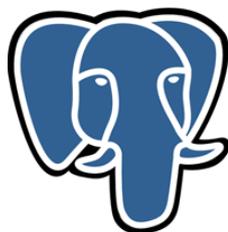
Горизонтальное масштабирование



Профит



Стек технологий Django DB Mailer

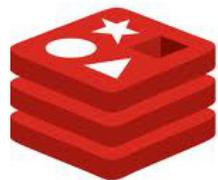


PostgreSQL

or



MySQL®



redis



Простой пример отправки сообщения

```
from dbmail import send_db_mail # send_db_sms, send_db_push ...
```

```
# данные контекста
```

```
data = {'path': request.path}
```

```
user = User.objects.get(pk=1)
```

```
settings = Settings.objects.get(user=user)
```

```
# отправка простого сообщения
```

```
send_db_mail('welcome-notification', user.email, settings, user, data)
```

Быстрый старт демки

```
$ git clone -b development https://github.com/LPgenerator/django-db-mailer.git db-mailer  
$ cd db-mailer
```

Vagrant:

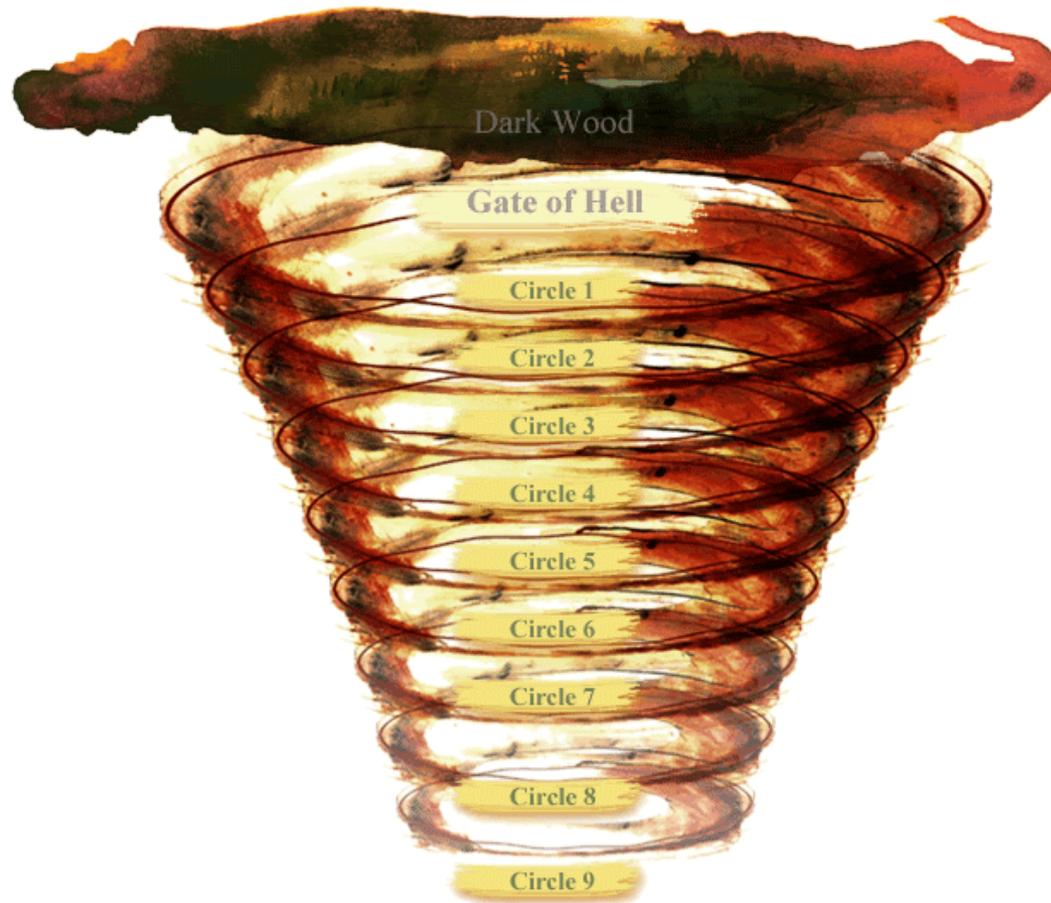
```
$ vagrant up --provider virtualbox  
$ vagrant ssh
```

Docker:

```
$ docker build -t dbmail .  
$ docker run -it -d -p 8000:8000 --name dbmail dbmail  
$ docker exec -i -t dbmail /bin/bash
```

```
$ cd /mailer/ && python manage.py shell_plus --print-sql  
$ URL=http://127.0.0.1:8000/admin/dbmail; xdg-open $URL || open $URL
```





Часть четвертая: Завершая круг

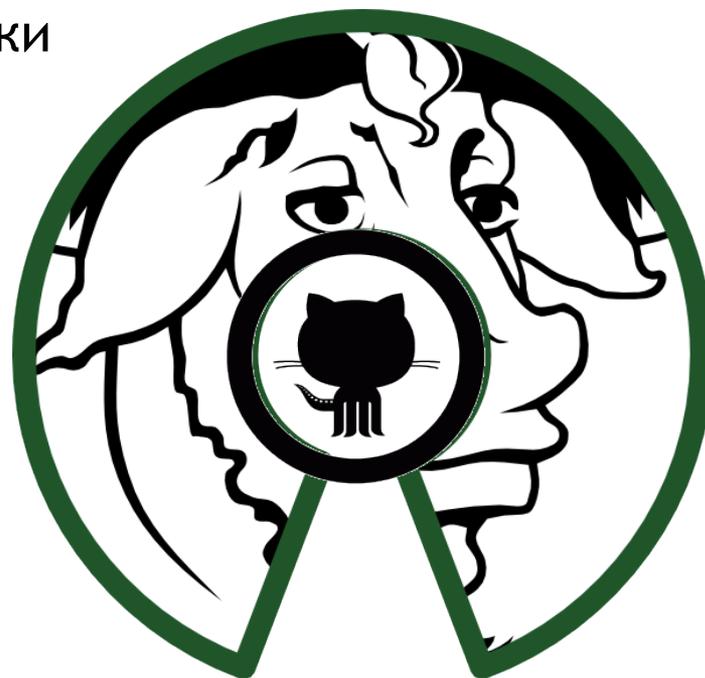
Кому еще полезно?

- Разработчикам iOS/Android приложений в связке с django-rest-framework
- Спамерам
- Как маленьким так и большим проектам
- Бородатым sys-админам и обычным разработчикам

Собственно языковых ограничений нет никаких. В документации есть примеры для: Go, Objective-C, Java, Python, PHP, Ruby, node.js, Bash

OpenSource Модель

- Социальность и публичность разработки
- Документация на ReadTheDocs
- Setup.py + PyPi
- Замороженный Public Api
- PEP8 + Landscape
- Тесты (Tox, Travis CI)
- Django 1.4-1.8
- Python 3.4+ уже в dev ветке
- Публичный репозиторий на GitHub
- Pull Request-ы приветствуются



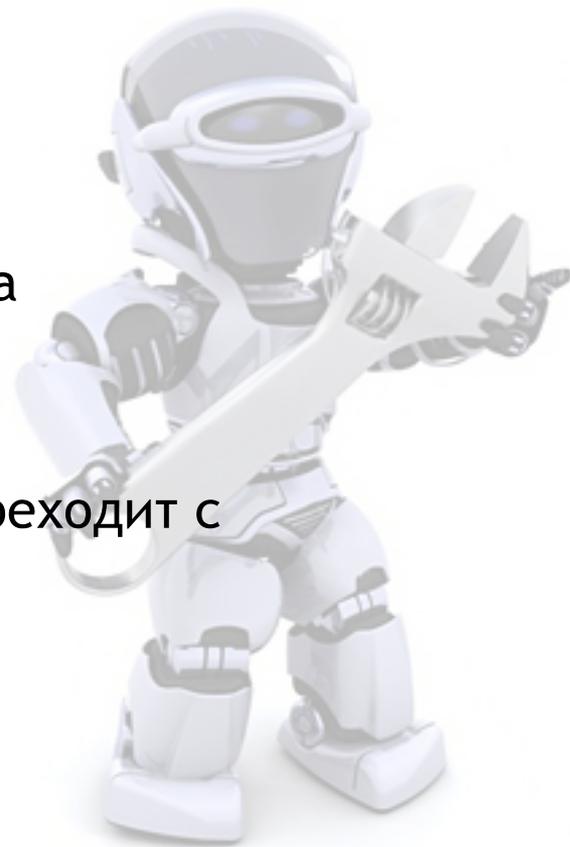
Чем хорош OpenSource?

- Дисциплинирует (код, документация)
- Учит взаимодействию (общение, координирование)
- Повышает опыт разработки
- Развивает в новом направлении
- Расширяет кругозор
- Опыт: Проектирования/Разработки/Тестирования
- Одна голова хорошо, а две лучше 😊
- Для компаний: реклама и меценатство



Поддержка

- Используется в 3-х больших проектах
- Поддержка со стороны проектов/сообщества
- Своевременные фиксы
- Master всегда крутится в продакшен
- В ближайшее время поддержка Python 3 переходит с development в master
- В обозримом будущем покрытие тестами



На птичьих правах OpenSource рекламы

- django-db-mailer (<https://github.com/LPgenerator/django-db-mailer/>)
- django-mmс (<https://github.com/LPgenerator/django-mmс>)
- django-celery-mon (<https://github.com/LPgenerator/django-celery-mon>)
- django-db-snapshots (<https://github.com/LPgenerator/django-db-snapshots>)
- django-secure-auth (<https://github.com/gotlium/django-secure-auth>)
- django-pinba (<https://github.com/gotlium/django-pinba>)
- django-geoip-redis (<https://github.com/gotlium/django-geoip-redis>)
- RedPilule (<http://redpilule.ru>)

Большое спасибо за внимание!

- Пожалуйста, ваши вопросы!
- С вами был Руслан Аскарлов,
- DevOps LPgenerator
- <https://github.com/gotlium>
- gotlium@gmail.com
- <http://lpgenerator.ru/>



Нам как раз нужны
Фронтенд и **Бекенд**
разработчики

