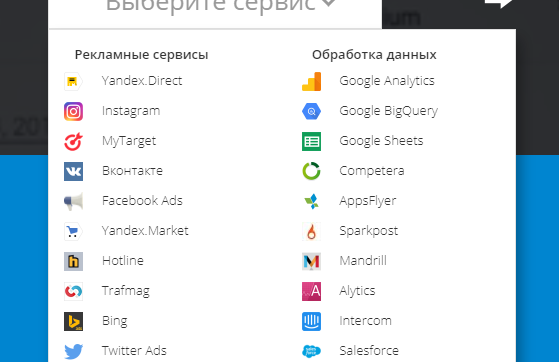
Сервис сбора статистики

Вся разработка делится на два этапа:

1. Сбор статистики из рекламных кабинетов: Вконтакте, Facebook, Яндекс Директ, myTarget.
2. Передача статистики в систему аналитики Google Analytics или базу данных SQL

Пример зарубежного сервиса решающего данную задачу: <https://www.owox.ru/products/bi/pipeline/>



# Этап 1

## Описание задачи

Требуется реализовать систему сбора статистических данных из рекламных кабинетов по следующим параметрам: показы, клики, расходы, название рекламной кампании, utm метки

## Процесс решения

Коннектор под каждую рекламную площадку состоять из 2 модулей:

1. Авторизация в системе, запрос прав на доступ к данным, получение токкена, сохранение связки токен/аккаунт пользователя.
2. Получение статистики из рекламного кабинета за 1 день и сохранение ее в базу данных.

В зависимости от выбора системы на которой будет происходить разработка данные могут передаваться в требуемом системой виде, но по возможности, входные/выходные данные должны передаваться в виде ассоциативного массива (если требуется многомерного), название элементов массива должны соответствовать названиям полей, которые запрашивает/отдает рекламная система, по этой же логике следует строить базу данных.

Для обеспечения целостности данных от разных систем, для каждого модуля желательно создавать свои таблицы для хранения токкена и данных.

В дальнейшем в базе желательно учесть связку учетной записи пользователя/токкенна сервиса загружаемых данных/проект.

## API рекламных систем

**Вконтакте**

* <https://vk.com/dev/first_guide>
* <https://vk.com/dev/ads>
* <https://vk.com/dev/ads.getStatistics>
* <https://vk.com/dev/ads.getAdsLayout>

**Facebook**

* <https://developers.facebook.com/docs/marketing-api/insights-api/getting-started#query-for-statistics>
* <https://developers.facebook.com/docs/graph-api/overview?locale=ru_RU>

**Яндекс директ**

* https://tech.yandex.ru/direct/doc/dg/concepts/about-docpage/
* <https://tech.yandex.ru/direct/doc/reports/reports-docpage/>

**myTarget**

* <https://target.my.com/adv/api-marketing/doc/>
* https://target.my.com/doc/api/detailed/#resource\_statistics

# Пояснения на примере ВК

## Получение токкена VK

**Справка:**

* <https://vk.com/dev/first_guide>
* <https://vk.com/dev/authcode_flow_user>

**Входные данные в модуль:**

* ID приложения/client\_id
* Cрок действия токкена/scope: **offline,ads**
* display: **popup**
* response\_type: **code**
* URL Редиректа/redirect\_uri
* Версия API/v

**Выходные данные:**

* **access\_token**
* **user\_id**

Реализация авторизации, получения токена, в случае подключения к агентскому аккаунту, следует делать запрос идентификаторов клиентов, предоставлять пользователю выбор нужного для импорта данных и сохранение полученных данных в базу.

## Получение данных

Подключение при помощи полученного токена и ID клиента (в случае подключения к агентскому аккаунту) и загрузка данных за указанный день по следующим показателям (в дальнейшем следует оставить возможность расширения полей): дата, utm метки (medium, source, campaign), количество показов, кликов, расход, url на который ведет объявление.

**Справка:**

* <https://vk.com/dev/ads>
* <https://vk.com/dev/ads.getStatistics>
* <https://vk.com/dev/ads.getAdsLayout>

**Входные данные в модуль:**

* access\_token
* account\_id
* client\_id – для рекламных агенств
* include\_deleted = 1 - собираем данные даже с тех РК, которые ушли в архив.
* День, за который собираем данные
* **Префикс импорта рекламных кампаний** – задается в случае если на аккаунте рекламируется несколько сайтов и требуется передавать статистику только по определенным, префикс – это часть названия рекламной кампании, по задумке это название рекламируемого бренда/сайта/продукта. Например если префикс ЛСР, а в рекламном аккаунте есть следующие кампании ЛСР\_Загород, элитка\_\_ЛСР, другая РК, то импортируются расходы только по первым двум.

Подробное описание метода получения ID кампаний, ID Объявлений и UTM

* <https://vk.com/dev/ads.getAdsLayout>

### Этапы получения данных

**Первый шаг**

Модуль запрашивает данные по активным и архивным кампаниям и объявлениям. После этого из поля «link\_url» извлекаются utm метки.

**Второй шаг**

Подробное описание метода получения расходов, кликов, показов и прочего.

* <https://vk.com/dev/ads.getStatistics>

На основании данных полученных на предыдущем шаге, следует сформировать запрос для получения статистики по расходам, кликам, показам в разрезе объявлений и кампаний.

# Передача данных в систему Google Analyticts

**Справка:**

https://developers.google.com/analytics/devguides/config/mgmt/v3/mgmtDataImport?hl=ru

# Работа по Crone

**Шаг 1**

Каждый коннектор должен запускаться раз в сутки и собирать все данные, за предыдущий день, из всех привязанных аккаунтов и сохранять их в базу с привязкой к проекту и аккаунту в разрабатываемой системе. Также следует реализовать возможность загрузки данных по дням, за предыдущие периоды (неделя, месяц, произвольные даты).

После каждой загрузки следует записывать успешность импорта данных в таблицу отчета, примерный набор полей отчета:

* ID проекта
* ID Агента – это идентификатор потока рекламной системы, т.е. поток для загрузки данных из вк для такого то проекта
* Дата (день, месяц, год) – за который успешно были загружены данные.
* Дата и время успешного импорта данных

При запуске модуля по крону, модуль проверяет таблицу отчета, за какой день были успешно загружены данные, удаляет данные за те дни, которые были загружены частично и как следствие строчки в отчете об успешности загрузки нет. И заново начинает сбор с самой ранней даты по вчерашний день.

**Шаг 2**

После загрузки данных в систему, модуль передачи данных в GA начинает передавать загруженные данные в привязанные счетчики GA.

Аналогично предыдущему шагу, успешность загрузки следует логировать в БД.

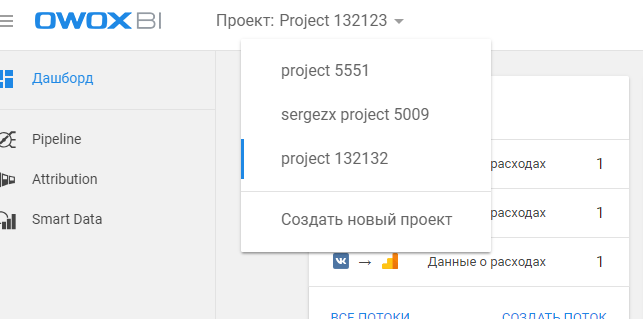
**Шаг 3**

Для экономии места в БД, возможно удалять успешно загруженные данные и локальной базы данных на сервере. Но это пока под вопросом.

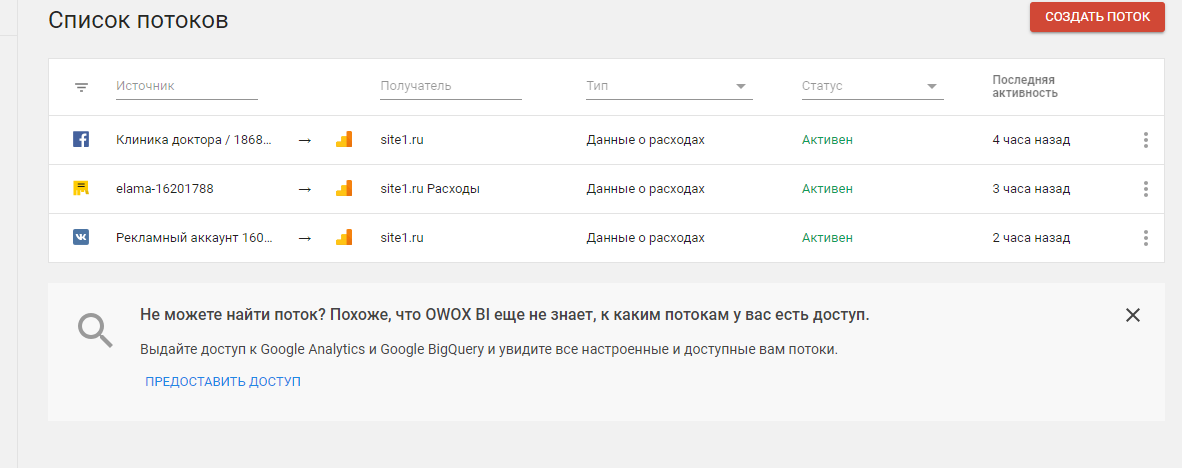
# Пользовательский интерфейс

За пример пользовательского интерфейса взят сервис owox.ru

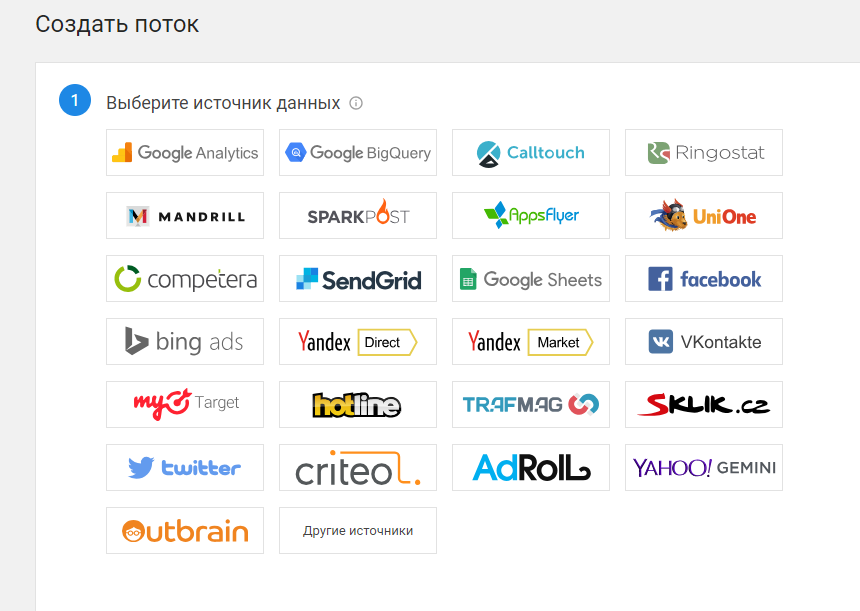
**Выбор и создание проекта**



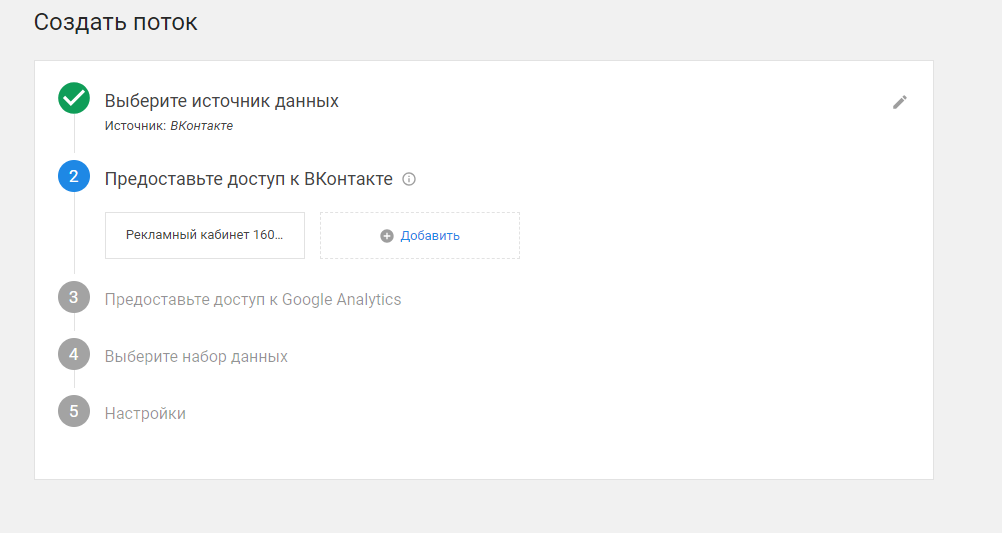
**Список созданных потоков**



**Создание нового потока: выбор источника получения данных**



**Факт успешного получения токкена**



Выбор представления для экспорта данных

