

# Документация по экспорту данных автосалонов из системы MaxPoster

Экспорт данных автосалонов .....	2
Назначение .....	2
Соглашение .....	2
Получение доступа к экспортным данным .....	2
Порядок формирования URL для получения данных .....	2
Версии формата экспортных данных .....	3
Актуальность данных .....	3
Форматы экспортных данных .....	3
Марки и модели транспортных средств (marks) .....	3
Форма поиска транспортного средства (search_form) .....	4
Список транспортных средств (vehicles) .....	5
Описание транспортного средства (vehicle) .....	6
Описание всех транспортных средств (full_vehicles) .....	8
Ошибка (error) .....	8
Библиотека для получения экспортных данных .....	8
Назначение .....	8
Требования к среде .....	8
API .....	8
maxException .....	8
maxOption .....	9
maxXmlClient .....	9
maxCacheXmlClient .....	11
maxCacheHtmlClient .....	11
Установка и настройка .....	12
Получение кода Клиента .....	12
Пример использования .....	12
Контакты .....	13

# Экспорт данных автосалонов

## Назначение

Предоставление автосалонам данных о продаваемых автомобилях.

## Соглашение

Доступ к экспортным данным предоставляется пользователям сервиса MaxPoster (<http://maxposter.ru/>).

Данные предоставляются в формате XML.

Структура XML документов может изменяться MaxPoster в одностороннем порядке путем выпуска новых версий экспорта.

С момента выпуска новой версии старая версия поддерживается в течение 1 месяца. По истечении месяца запросы к устаревшей версии экспорта перестают обслуживаться.

Информация о выпуске новой версии экспорта публикуется на сайте, а так же рассылается по списку рассылки зарегистрированным пользователям.

Экспортные данные обновляются с частотой раз в час. Период обновления данных может быть изменен MaxPoster в одностороннем порядке без дополнительного уведомления пользователей.

Полученные экспортные данные (как XML так и фотографии) должны кэшироваться на стороне автосалона до момента истечения их актуальности, при этом срок актуальности данных передается сервисом в заголовке ответа Expires. Кэширование обеспечивает использование клиентских скриптов (см. далее), применение которых при запросах крайне желательно. За чрезмерное количество идентичных запросов, вызванное отсутствием кэширования данных на стороне автосалона, доступ к экспортным данным может быть приостановлен.

## Получение доступа к экспортным данным

Интернет-сервис размещен на двух зеркалах, расположенных по адресам:

- <http://export1.maxposter.ru>
- <http://export2.maxposter.ru>

Доступ к Интернет-сервису защищен с помощью логина и пароля.

В качестве логина выступает код автосалона в системе (либо составной код автосалонов при запросе данных для нескольких автосалонов за один раз).

Для получения пароля необходимо написать запрос на [info@maxposter.ru](mailto:info@maxposter.ru) с email пользователя, зарегистрированного в системе MaxPoster с темой «Доступ к экспортным данным автосалона». В ответ на запрос будет отправлено письмо содержащее:

- код автосалона в системе MaxPoster;
- пароль для доступа;
- номер текущей версии API.

## Порядок формирования URL для получения данных

Интернет-сервис построен на базе архитектуры REST, и использует запросы типа GET и POST.

URL формируется следующим образом:

1. Адрес зеркала (<http://export1.maxposter.ru> или <http://export2.maxposter.ru>);
2. Версия API (например: /api1 для первой версии, /api2 для второй версии и т.п.);
3. Код автосалона (например: /5);
4. идентификатор запрашиваемых данных (например: /marks.xml).

Пример URL: <http://export1.maxposter.ru/api1/5/marks.xml>

## **Версии формата экспортных данных**

В случае внесения существенных изменений в формат экспортных данных выпускается новая версия сервиса (далее версия API).

Зарегистрированные в листе рассылки пользователи оповещаются о выходе новой версии по электронной почте. Так же информация о выходе новой версии публикуется в настоящем разделе.

Предыдущая версия API поддерживается в работоспособном состоянии в течение 1 месяца с момента выпуска новой версии.

Версия API фиксирует:

1. перечень допустимых запросов;
2. структуру XML, возвращаемых в ответ на запросы (перечень нод, перечень параметров нод, иерархию нод).

В рамках одной версии допустимо изменение API за счет:

1. добавления новых нод и параметров нод;
2. изменения (как добавления так и удаления) вариантов значений нод;
3. расширение параметров запросов к Интернет-сервису.

## **Актуальность данных**

Ответы Интернет-сервиса сопровождаются заголовками:

- Last-Modified;
- Expires.

В заголовке Last-Modified указывается время обновления данных.

В заголовке Expires указывается время, до которого данные можно помещать в кэш.

## **Данные в формате XML**

Все XML, возвращаемые по автосалону в конкретный момент времени имеют одинаковое значение в заголовке Last-Modified. Если у автосалона есть изменения хотя бы в одном объявлении о продаже автомобиля, то при обновлении данных, значения Last-Modified устанавливается по всем объявлениям автосалона так, чтобы на любой запрос выдавалось одно и тоже новое значение.

После обновления экспортных данных, время, указанное в заголовке Expires, равно времени обновления данных, указанному в Last-Modified, плюс период обновления данных ( $Expires = Last-Modified + 1 \text{ час}$ ).

При очередном обновлении экспортных данных система проверит наличие изменений в данных автосалона. Если данные автосалона не изменились, то значение заголовка Last-Modified останется без изменений, но при этом заголовок Expires увеличится так, чтобы указывать на время следующего обновления (т.е.  $Expires = Last-Modified + 2 \text{ часа}$ ).

## **Фотографии**

В рамках сервиса maxposter.ru файлы с фотографиями не изменяются: фотографии один раз загружаются, получают уникальное имя, и хранятся. Таким образом фотографии можно кэшировать до конца периода действия объявления.

## **Форматы экспортных данных**

### **Марки и модели транспортных средств (marks)**

#### **Содержание**

XML содержит список марок и моделей автомобилей, продаваемых в настоящий момент автосалоном.

#### **Способ получения**

XML возвращается при запросе URL:

[http://export1.maxposter.ru/\[Код API\]/\[Код автосалона\]/marks.xml](http://export1.maxposter.ru/[Код API]/[Код автосалона]/marks.xml)

## Описание XML

```
<?xml version="1.0" encoding="UTF-8" ?>
<response id="marks">
  <marks>
    <mark mark_id="[Код марки]">
      <name>[Название марки]</name>
      <count>[Количество активных авто]</count>
      <models>
        <model model_id="[Код модели]">
          <name>[Название модели]</name>
          <count>[Количество активных авто]</count>
        </model>
      </models>
    </mark>
  </marks>
</response>
```

Данные отсортированы в следующем порядке:

- Марки отсортированы по алфавиту;
- Модели отсортированы по алфавиту.

## Форма поиска транспортного средства (search\_form)

### Содержание

XML содержит данные, для формирования формы поиска по транспортным средствам.

### Способ получения

XML возвращается при запросе URL

[http://export1.maxposter.ru/\[Код API\]/ \[Код автосалона\]/search\\_form.xml](http://export1.maxposter.ru/[Код API]/ [Код автосалона]/search_form.xml).

Данные формы установлены в значения по умолчанию.

## Описание XML

```
<?xml version="1.0" encoding="UTF-8" ?>
<response id="search_form">
  <search_form>
    <list name="search[mark_id]" error="[Сообщение об ошибке]">
      <option value="[Код марки]" selected="1">[Название марки]</option>
    </list>
    <list name="search[model_id]" error="[Сообщение об ошибке]">
      <optgroup mark_id="[Код марки]" label="[Название марки]">
        <option value="[Код модели]" selected="1">[Название модели]</option>
      </optgroup>
    </list>
    <field name="search[year][from]" value="[Значение]" size="4" error="[Сообщение об ошибке]"/>
    <field name="search[year][to]" value="[Значение]" size="4" />
    <list name="search[steering_wheel]" error="[Сообщение об ошибке]">
      <option value="[Код расположения руля]" selected="1">[Название способа расположения руля]</option>
    </list>
    <list name="search[engine_type]" error="[Сообщение об ошибке]">
      <option value="[Код типа двигателя]" selected="1">[Название типа двигателя]</option>
    </list>
    <list name="search[drive_type]" error="[Сообщение об ошибке]">
      <option value="[Код типа привода]" selected="1">[Название типа привода]</option>
    </list>
    <list name="search[gearbox_type]" error="[Сообщение об ошибке]">
      <option value="[Код типа КПП]" selected="1">[Название типа КПП]</option>
    </list>
    <list name="search[body_type]" error="[Сообщение об ошибке]">
      <option value="[Код типа кузова]" selected="1">[Название типа кузова]</option>
    </list>
    <field name="search[price][from]" value="[Значение]" size="[Макс. количество знаков]" error="[Сообщение об ошибке]" />
    <field name="search[price][to]" value="[Значение]" size="[Макс. количество знаков]" />
    <list name="search[price][unit]">
      <option value="[Код валюты]" selected="1">[Название валюты]</option>
```

```

</list>
<list name="search[availability]" error="[Сообщение об ошибке]">
  <option value="[Код наличия авто]" selected="1">[Наличие авто]</option>
</list>
</search_form>
</response>

```

## Список транспортных средств (vehicles)

### Содержание

XML содержит:

3. критерии, по которым сформирован список транспортных средств;
4. перечень транспортных средств, соответствующих заданным критериям.

Может содержать как все транспортные средства автосалона (в случае, если критерии формирования списка не заданы, либо если применение критериев не привело к усечению списка), так и усеченный список транспортных средств.

### Способ получения

XML с полным списком транспортных средств возвращается при запросе URL:

[http://export1.maxposter.ru/\[Код API\]/\[Код автосалона\]/vehicles.xml](http://export1.maxposter.ru/[Код API]/[Код автосалона]/vehicles.xml)

Если в запросе методом POST передать параметры для фильтрации списка, то полученный в ответ XML будет содержать отфильтрованный список транспортных средств. Перечень допустимых параметров и их значения определяется XML [Форма поиска транспортного средства](#).

### Описание XML

```

<?xml version="1.0" encoding="UTF-8" ?>
<response id="vehicles">
  <search_form>
    <!--
      См. search_form. Для нод вместо значений по умолчанию расставлены
      значения запроса.
    -->
  </search_form>
  <vehicles>
    <vehicle vehicle_id="[Код объявления]" dealer_id="[Код автосалона]"
type="[Тип тр. средства]">
      <mark mark_id="[Код марки]">[Название марки]</mark>
      <model model_id="[Код модели]">[Название модели]</model>
      <year>[Год выпуска]</year>
      <steering_wheel>
        <place steering_wheel_place_id="[Код расположения]">[Расположение
руля]</place>
      </steering_wheel>
      <engine>
        <type engine_type_id="[Код типа двигателя]">[Название типа
двигателя]</type>
        <volume unit="cm3">[Объем двигателя в см3]</volume>
        <power unit="hp">[Мощность двигателя в л.с.]</power>
      </engine>
      <drive>
        <type drive_type_id="[Код типа привода]">[Название типа
привода]</type>
      </drive>
      <gearbox>
        <type gearbox_type_id="[Код типа КПП]">[Название типа КПП]</type>
      </gearbox>
      <body>
        <type body_type_id="[Код типа кузова]">[Название типа
кузова]</type>
        <color body_color_id="[Код цвета]" metallic="[Признак цвета
металлик]" >[Название цвета]</color>
      </body>
      <mileage>
        <value unit="[Единица измерения пробега]">[Пробег]</value>
        <without_rus_mileage /><!-- Без пробега по России -->
      </mileage>
      <condition condition_id="[Код состояния автомобиля]">[Название
состояния автомобиля]</condition>
      <price>
        <value unit="[Код валюты]" rub_price="[Стоимость в
рублях]">[Стоимость тр. средства]</value>

```

```

    <without_customs /><!-- Цена указана без учета таможенных сборов -->
    <bargaining /><!-- По цене возможен торг -->
  </price>
  <availability availability_id="[Код наличия]">[Название наличия
тр. средства]</availability>
  <photo photo_id="[Уникальный код фото]" file_name="[Уникальное имя файла]">
    <thumbnail>[Ссылка на уменьшенное изображение главной фотографии
авто]</thumbnail>
  </photo>
  <dates>
    <created_at>[Дата создания объявления в формате ГГГГ-ММ-
ДД]</created_at>
    <updated_at>[Дата изменения объявления в формате ГГГГ-ММ-
ДД]</updated_at>
    <expires_at>[Срок актуальности объявления в формате ГГГГ-ММ-
ДД]</expires_at>
  </dates>
</vehicle>

</vehicles>
</response>

```

Данные отсортированы в обратном хронологическом порядке, т.е. новые записи расположены сверху.

Жирным шрифтом выделены ноды, наличие которых в XML обязательно.

## Описание транспортного средства (vehicle)

### Содержание

XML содержит полное описание транспортного средства.

### Способ получения

XML возвращается при запросе URL: [http://export1.maxposter.ru/\[Код API\]/\[Код автосалона\]/\[Код объявления\].xml](http://export1.maxposter.ru/[Код API]/[Код автосалона]/[Код объявления].xml)

### Описание XML

```

<?xml version="1.0" encoding="UTF-8" ?>
<response id="vehicle">
  <vehicle vehicle_id="[Код объявления]" dealer_id="[Код автосалона]"
тр. средства">
    <mark mark_id="[Код марки]">[Название марки]</mark>
    <model model_id="[Код модели]">[Название модели]</model>
    <year>[Год выпуска]</year>
    <steering_wheel>
      <place steering_wheel_place_id="[Код расположения]">[Расположение
руля]</place>
      <power steering_wheel_power_id="[Код типа]">[Название типа усилителя
рулевого управления]</power>
      <adjustment steering_wheel_adjustment_id="[Код способа]">[Название
способа регулировка руля]</adjustment>
      <heater /><!-- Обогрев руля -->
    </steering_wheel>
    <engine>
      <type engine_type_id="[Код типа двигателя]">[Название типа
двигателя]</type>
      <volume>[Объем двигателя в см3]</volume>
      <power>[Мощность двигателя в л.с.]</power>
      <gas_equipmnet /><!-- Установлено газоболонное оборудование -->
    </engine>
    <drive>
      <type drive_type_id="[Код типа привода]">[Название типа
привода]</type>
    </drive>
    <gearbox>
      <type gearbox_type_id="[Код типа КПП]">[Название типа КПП]</type>
    </gearbox>
    <body>
      <type body_type_id="[Код типа кузова]">[Название типа кузова]</type>
      <color body_color_id="[Код цвета]" metallic="[Признак цвета
металлик]">[Название цвета]</color>
    </body>
    <mileage>
      <value unit="[Единица измерения пробега]">[Пробег]</value>
      <without_rus_mileage /><!-- Без пробега по России -->
    </mileage>
    <condition condition_id="[Код состояния автомобиля]">[Название
состояния автомобиля]</condition>
    <vin>[VIN]</vin>

```

```

<price>
  <value unit="[Код валюты]" rub_price="[Стоимость в
рублях]">[Стоимость тр.средства]</value>
  <without_customs /><!-- Цена указана без учета таможенных сборов -->
  <bargaining /><!-- По цене возможен торг -->
</price>
<pts_owner_count>[Количества владельцев по ПТС]</pts_owner_count>
<availability availability_id="[Код наличия]">[Название наличия
тр.средства]</availability>
<abs /><!-- Есть антиблокировочная система -->
<asr /><!-- Есть антипробуксовочная система -->
<cruise_control /><!-- Есть круиз-контроль -->
<trip_computer /><!-- Есть бортовой компьютер -->
<parktronic /><!-- Есть парктроник -->
<esp /><!-- Есть система курсовой стабилизации -->
<light_alloy_wheels /><!-- Есть литые диски -->
<sensors>
  <rain /><!-- Есть датчи дождя -->
  <light /><!-- Есть датчик света -->
</sensors>
<headlights>
  <xenon /><!-- Есть ксеноновые (или биксеноновые) фары-->
  <washer /><!-- Есть омыватель фар-->
</headlights>
<mirrors>
  <power /><!-- Есть электрозеркала -->
  <defroster /><!-- Есть обогрев зеркал -->
</mirrors>
<windows>
  <power window_power_id="[Код электропривода стекол]">[Название электропривода
стекол]</power>
  <tinted /><!-- Есть тонированные стекла -->
</windows>
<seats>
  <heater /><!-- Есть обогрев сидений-->
  <driver_adjustment>[Название способа регулировки сидения]</driver_adjustment>
  <passanger_adjustment>[Название способа регулировки
сидения]</passanger_adjustment>
</seats>
<alarm_system /><!-- Есть охранная сигнализация -->
<central_lock /><!-- Есть центральный замок -->
<airbag>[Количество подушек безопасности]</airbag>
<climate_control>[Название способа управления климатом]</climate_control>
<multimedia >[Название пакета мультимедиа]</multimedia >
<compartment>
  <decoration>[Название отделки салона]</decoration>
  <color>[Оттенок салона]</color>
</compartment>
<hatch /><!-- Есть люк -->
<description>[Дополнительная информация]</description>
<inspection_place>[Место, где можно осмотреть автомобиль]</inspection_place>
<contact contact_id="[Код контакта]">
  <name>[Имя контакта]</name>
  <phone from="[Звонить с]" to="[Звонить по]">[Телефон]</phone>
  <phone from="[Звонить с]" to="[Звонить по]">[Телефон]</phone>
  <phone from="[Звонить с]" to="[Звонить по]">[Телефон]</phone>
</contact>
<uin>[Код автомобиля в учетной системе рекламодателя]</uin>
<photos>
  <photo photo_id="[Уникальный код фото]" file_name="[Уникальное имя файла]">
    <thumbnail>[Ссылка на уменьшенное изображение размером 120x90
пикселей]</thumbnail>
    <original>[Ссылка на большое изображение размером 640x400 пикселей]</original>
  </photo>
  <!-- На 24.08.09 допустимо до 16 фотографий для одного тр.средства -->
  ...
</photos>
<dates>
  <created_at>[Дата создания объявления в формате ГГГГ-ММ-
ДД]</created_at>
  <updated_at>[Дата изменения объявления в формате ГГГГ-ММ-
ДД]</updated_at>
  <expires_at>[Срок актуальности объявления в формате ГГГГ-ММ-
ДД]</expires_at>
</dates>
</vehicle>
</response>

```

Жирным шрифтом выделены ноды, наличие которых в XML обязательно.

## Описание всех транспортных средств (full\_vehicles)

### Содержание

XML содержит перечень транспортных средств с полным их описанием.

### Способ получения

XML возвращается при запросе URL: `http://export1.maxposter.ru/[Код API]/[Код автосалона]/full_vehicles.xml`

### Описание XML

```
<?xml version="1.0" encoding="UTF-8" ?>
<response id="full_vehicles">
  <vehicles>
    <vehicle vehicle_id="[код объявления]" dealer_id="[код автосалона]" type="[Тип тр. средства]">
      <!-- Содержание аналогично ноде из описания транспортного средства -->
    </vehicle>
  </vehicles>
</response>
```

### Ошибка (error)

#### Содержание

XML содержит сообщение о возникшей ошибке.

#### Способ получения

Возвращается при возникновении сбоя при запросе к сервису.

#### Описание XML

```
<?xml version="1.0" encoding="UTF-8" ?>
<response id="error">
  <error error_id="[код ошибки]">[описание ошибки]</error>
</response>
```

## Библиотека для получения экспортных данных

### Назначение

Библиотека (далее Клиент) предназначена для:

1. получения экспортных данных от Интернет-сервиса;
2. кэширования полученных данных на стороне Клиента;
3. преобразования данных из формата XML в HTML.

### Требования к среде

Для корректной работы Клиента необходим PHP версии 5.2 с поддержкой:

- curl;
- dom;
- xml;
- xsl.

## API

### maxException

#### Описание

Класс используется для генерации исключений при использовании Клиента. Содержит перечень возможных исключений: их коды и названия.

#### Статические публичные методы

***getException(\$\_code, \$\_param = null)***

назначение

Генерация исключения по коду.



#### входящие параметры

`$_code` код генерируемого исключения;

`$_param` перечень данных, которые должны быть включены в сообщение исключения

#### возвращаемое значение

Экземпляр класса `maxException` с установленным кодом и сообщением.

## maxOption

### Описание

Абстрактный класс обеспечивающий работу с опциями

### Публичные методы

***\_\_construct(array \$\_options = array())***

#### назначение

Конструктор

#### входящие параметры

`$_options` ассоциированный массив параметров класса.

***getOption(\$\_name)***

#### назначение

Возвращает значение параметра по названию.

#### входящие параметры

`$_name` название параметра

#### возвращаемое значение

Значение параметра.

***getOptions()***

#### назначение

Возвращает все параметры класса

#### возвращаемое значение

Массив, содержащий все параметры класса.

***setOption(\$\_name, \$\_value)***

#### назначение

Установка параметра класса.

#### входящие параметры

`$_name` название параметра

`$_value` значение параметра

***setOptions(array \$\_options)***

#### назначение

Добавление/установка нескольких параметров класса

#### входящие параметры

`$_options` ассоциированный массив с добавляемыми / устанавливаемыми параметрами и их значениями

## maxXmlClient

### Описание

Класс для запросов экспортных данных автосалонов от Интернет-сервиса MaxPoster.

### Наследуемые публичные методы

Методы наследуются от классов:

- [maxOption](#)

### Допустимые параметры конструктора

Параметр	Значение	Пример значения
<code>mirror1</code>	Основная точка доступа к сервису	<code>http://export1.maxposter.ru/</code>
<code>mirror2</code>	Резервная точка доступа к сервису	<code>http://export2.maxposter.ru/</code>

api_version	Номер версии API Интернет-Сервиса	1
dealer_id	Код автосалона в системе MaxPoster.ru	5
password	Пароль для http-авторизации	555
allowed_request_themes	Список допустимых тем запросов (кроме перечисленных тем допустим запрос с кодом автообъявления)	array('marks', 'search_form', 'vehicles')

## Публичные методы

### *setRequestThemeName(\$\_requestThemeName)*

#### назначение

Установка темы запроса к Интернет-сервису.

#### входящие параметры

*\$\_requestThemeName* может принимать значение:

Значение	Тип запрашиваемых данных
marks	запрос XML « <a href="#">Марки и модели транспортных средств</a> »
search_form	запрос XML « <a href="#">Форма поиска транспортного средства</a> »
vehicles	запрос XML « <a href="#">Список транспортных средств</a> »
целое число с кодом объявления	запрос XML « <a href="#">Описание транспортного средства</a> »
full_vehicles	запрос XML « <a href="#">Описание всех транспортных средств</a> »

### *getRequestThemeName()*

#### назначение

Получение темы запроса.

#### возвращаемое значение

Тема запроса.

### *setRequestParams(array \$\_params)*

#### назначение

Установка параметров запроса.

Применяется для темы vehicles, принимающих POST-параметры, для задания параметров фильтрации списка объявлений.

#### входящие параметры

*\$\_params* ассоциированный массив с параметрами запроса.

### *getRequestParams()*

#### назначение

Получение параметров запроса.

#### возвращаемое значение

Ассоциированный массив с параметрами запроса.

### *getXml()*

#### назначение

Запрос XML от Интернет-сервиса.

#### возвращаемое значение

XML в формате DOMDocument.

### *getResponseThemeName()*

#### назначение

Получение темы ответа. Тема ответа отражает тип XML, полученного на запрос.

#### возвращаемое значение

Название темы ответа. Название темы ответа соответствует значению параметра id ноды response в XML, полученном в ответ на запрос.

Допустимы следующие значения:

- marks
- search\_form
- vehicles
- vehicle
- full\_vehicles
- error

## maxCacheXmlClient

### Описание

В расширение возможностей класса maxXmlClient добавлено кэширование данных на стороне Клиента.

Перед запросом к Интернет-сервису проверяется наличие актуальных данных в кэше. Если в кэше данные есть, то данные берутся из кэша и запрос к Интернет-сервису не производится. Если актуальных данных в кэше нет, то после получения данных они кэшируются для повторных обращений.

### Допустимые параметры конструктора

В дополнение [параметров класса maxXmlClient](#), допустимы параметры:

Параметр	Значение	Пример значения
cache_dir	Путь к каталогу для кэширования. У процесса, выполняющего скрипт должны быть права gwx на каталог и его файлы. Путь должен заканчиваться слэшем.	cache/
cache_actual_file	Название файла, в котором хранятся данные об актуальности кэша	actual_point.txt
cache_xml_dir	Название каталога для хранения xml-кэша. Каталог будет создан внутри каталога cache_dir	xml/
cached_xml_themes	Название тем, которые должны кэшироваться в виде XML (т.е. до XSLT-преобразования). Кэшировать XML обосновано для XML, на основании которых формируется несколько страниц, например vehicles.	array('marks', 'vehicles', 'vehicle')
error_theme	Название темы с описанием возникшей ошибки	error

### Наследуемые публичные методы

Методы наследуются от классов:

- [maxOption](#)
- [maxXmlClient](#)

## maxCacheHtmlClient

### Описание

В расширение возможностей класса maxCacheXmlClient добавлено:

- преобразование XML в HTML посредством XSLT;
- кэширование полученного HTML.

### Допустимые параметры конструктора

В дополнение [параметров класса maxCacheXmlClient](#), допустимы параметры:

Параметр	Значение	Пример значения
xslt_dir	Путь к каталогу, содержащему XSLT. Путь должен заканчиваться слэшем.	source/xsl/
cache_html_dir	Название каталога для хранения xml-кэша. Каталог будет создан внутри каталога cache_dir. Должен заканчиваться слэшем.	html/
cached_html_themes	Название тем, которые должны кэшироваться с виде HTML	array('marks', 'vehicles', 'vehicle')

### Наследуемые публичные методы

- [maxOption](#)
- [maxXmlClient](#)
- [maxCacheXmlClient](#)

### Публичные методы

#### *setXsltTemplate(\$\_xslt)*

##### назначение

Установка специального XSLT-шаблона для преобразования XML в HTML.

По умолчанию Клиент пытается выполнить преобразование XML с помощью шаблона, имя которого соответствует названию темы ответа.

Специальный шаблон имеет смысл задавать только если необходимо заменить шаблон по умолчанию.

##### входящие параметры

Строка с названием специального XSLT-шаблона.

#### *getXsltTemplate()*

##### назначение

Получение названия специального XSLT-шаблона.

##### возвращаемое значение

Строка с названием специального XSLT-шаблона.

#### *getHtml()*

##### назначение

Запрос данных и преобразование ответа в HTML. Данные сперва проверяются в КЭШе в следующей очередности:

- если есть данные в HTML кэше – возвращаются они;
- если нет данных в HTML кэше, но есть данные в XML-кэше, то берутся данные из XML-кэша и преобразуются в HTML;
- если нет данных в XML-кэше, то происходит запрос к Интернет-сервису MaxPoster.

##### возвращаемое значение

HTML с ответом.

## Установка и настройка

### Получение кода Клиента

Скрипты Клиента можно получить из репозитория на github:

<https://github.com/maxposter/maxDealerAPI>

### Пример использования

Скачайте тестовый раздел <https://github.com/maxposter/maxDealerSample>.

Разде имеет структуру каталогов:

lib/maxAPI	- каталог для скриптов клиента
lib/xslt	- каталог с XSLT-шаблонами
lib/php	- кастомизация скриптов Клиента

cache	- каталог для кэша
web	- document root для интернет-сервера

На каталог с данными кэша (cache) важно установить права 777 (rwx) для процесса, исполняющего скрипт.

В каталоге lib/xslt размещены XSLT-шаблоны для каждого типа ответов.

## **Контакты**

<http://www.maxposter.ru/>

info@maxposter.ru