

УДК 379.8

РАЗВИТИЕ ПРОЕКТА ВИРТУАЛЬНОГО ГОРОДА НА БАЗЕ ОТКРЫТЫХ API**Лёгенький Б.В., Аноприенко А. Я.**

Донецкий Национальный Технический Университет

Кафедра электронных вычислительных машин

E-mail: boris.legenkiy@gmail.com**Аннотация**

Лёгенький Б.В., Аноприенко А.Я. Развитие проекта виртуального города на базе открытых API. В этом докладе рассматривается исследование открытых WEB-API, разработка 3-d моделей зданий, их интеграция в “Google-Earth”, показан прототип портала, на котором будут размещаться модели «Виртуального города».

1.Открытые API. История и развитие. Постановка задачи.

Интерфейс программирования приложений (иногда **интерфейс прикладного программирования**) (англ. application programming interface, API [эй-пи-ай]) — набор готовых классов, процедур, функций, структур и констант, предоставляемых приложением (библиотекой, сервисом) для использования во внешних программных продуктах. Используется программистами для написания всевозможных приложений.

API определяет функциональность, которую предоставляет программа (модуль, библиотека), при этом API позволяет абстрагироваться от того, как именно эта функциональность реализована, а заостряют внимание на возможность получения конечного результата.

В зависимости от сферы применения API делятся на классы:

- API операционных систем.
- API графических интерфейсов
- API звуковых интерфейсов
- API аутентификационных систем
- Web API

Наибольший интерес в данной работе представляют WEB-API. WEB-API - это внешние мини-приложения, которые могут встраиваться в любой сайт. Обычно используют “HTML”, “XML”, “JSON”, “JavaScript” и “Flash” в различных комбинациях. Служат для повышения функциональности сайта. Данные приложения удобны и выгодны, т.к. встроить их довольно просто (необходимо предложенный производителем код интегрировать в нужное место на странице сайта) и при этом используются внешние ресурсы серверов, а не собственные.

Использование API крайне рекомендуется владельцам небольших сайтов, а так же сайтов с небольшой функциональностью. Блок мировых новостей, красивые часы, информер валют или погоды и многое другое может расширить функциональность сайта, украсить его, а значит, сделать более привлекательным для посетителей. Не следует пренебрегать возможностями API и крупным проектам. Например, встроенная проверка орфографии, переводчик, поисковик, интерактивная карта может помочь и большому проекту. На данный момент WEB-API имеет множество подразделений. Тенденция развития структуры данных приложений представлена на рис.1

В основе развития WEB-API была идея предоставления определенных возможностей крупных корпораций и фирм, что являлось рекламой на внешних порталах. Таким образом, однажды созданное «удачное» приложение начинало жить своей жизнью,

помогать в увеличении трафика своему родителю. Примеры крупных компаний, предоставляющих Web-API – это “Google”, “YouTube”, “Yandex” и другие.

Однако, API есть и закрытого типа. Обычно это приложения, которые необходимы для работы внутри компании, которые нет смысла опубликовать из-за конфиденциальности информации.

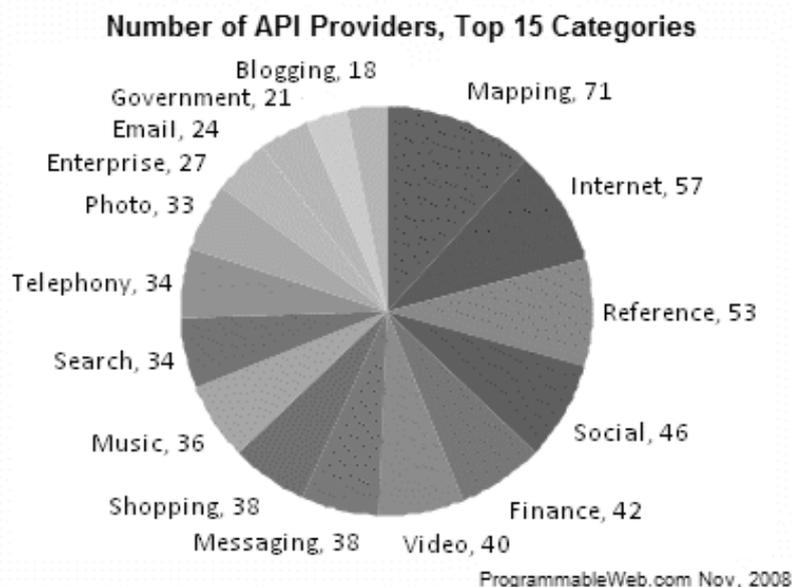


Рисунок 1 – Количество API провайдеров. 15 лучших категорий.

“Google”, “Yandex” имеет очень широкий спектр открытых приложений, которые в свою очередь имеют необходимую документацию и примеры подключений. Примерный список открытых API от “Google” представлен на рис.2.

Google code

например, "ajax api" или "open source"

Каталог сайта

<p>Продукты</p> <p>Все</p> <p>Объявления</p> <p>AJAX</p> <p>Браузер</p> <p>Географические приложения</p> <p>API продуктов</p> <p>Поиск</p> <p>Социальный</p> <p>Лаборатория Google</p> <hr/> <p>Ресурсы</p> <p>Открытый исходный код</p> <p>Образование</p> <p>События</p> <p>Скорость</p>	<p> Аутентификация аккаунтов Google Получайте доступ к приложениям для рабочего стола или мобильного устройства. Группа</p> <p> API Google AdSense Создайте возможность заработка для себя и своих посетителей, размещая объявления на веб-сайте. Документация - Блог - Группа</p> <p> API Google AdWords Автоматизируйте и оптимизируйте действия по управлению рекламными кампаниями. Документация - Блог - Группа</p> <p> AJAX API Google Реализуйте динамические веб-сайты с богатыми возможностями полностью на JavaScript и HTML. Документация - Блог - Группа</p> <p> AJAX API загрузки фидов Google Простое объединение общедоступных фидов с помощью JavaScript. Документация - Блог - Группа</p> <p> AJAX API для Google Переводчика С легкостью распознавайте множество языков и переводите с помощью JavaScript.</p>	<p> API Мастера сообществ Google (Лаборатория Google) API JavaScript и REST/RPC для Мастера сообществ Google Документация - Блог - Группа - Лаборатория Google</p> <p> API Гаджетов Создание мини-приложений, работающих на множестве сайтов, включая iGoogle, Google Desktop и любые веб-страницы. Документация - Блог - Группа</p> <p> Gears (Лаборатория Google) Сделайте возможной работу веб-приложений с локального компьютера или мобильного устройства без подключения к Интернету. Документация - Блог - Группа - Лаборатория Google</p> <p> API Google Health Управляйте личными медицинскими данными с помощью Google. Документация - Группа</p> <p> Главная страница разработчиков iGoogle (Лаборатория Google) Создавайте и тестируйте свой гаджет в новой тестовой программной среде для iGoogle. Документация - Блог - Группа - Лаборатория Google</p> <p> API тем iGoogle (Лаборатория Google) Разрабатывайте динамические темы для главной страницы iGoogle. Документация - Лаборатория Google</p>
--	--	---

Рисунок 2 – Список открытых API от “Google”

2. Среда разработки “Sketch-UP” + “Google Earth”

Следует отметить одну особенность всех WEB-API. При использовании данной технологии обычно передаются пакеты данных, упакованные в форматы “XML”, “JSON” и другие. Поэтому, для полного понимания сути подключения и работы механизма API, необходимо изучить основы. В этом случае можно будет с легкостью использовать любые внешние API, а также создавать свои API, которые в дальнейшем могут работать на благо собственного ресурса.

Тема данной работы «Развитие проекта виртуального города на базе открытых API». Для реализации этого проекта используются:

“Google Sketch-Up”

“Google Earth”

В “Google Earth” возможна интеграция 3-d моделей двумя способами:

1. из ресурса доступных мест “Google Builder”. При этом нет необходимости делать снимки и рисовать модель. Просто в “Google Builder” необходимо выделить доступное здание, открыть его в “Sketch-UP”, редактировать, интегрировать в “Google Earth”.
2. создать самостоятельно модель здания. Необходимо выбрать местность, сделать макет здания в “Sketch-UP” и наложить фотографии фасадов здания на модель. Далее необходимо интегрировать получившийся объект в “Google Earth”.

Первый вариант доступен для множества стран, однако для Украины пока нет такой возможности. Поэтому необходимо самостоятельно создавать проект в “Sketch-Up”. Для создания 3-d моделей также существует еще множество сред. Наиболее известные – это:

3-d MAX

Maya

ArchiCAD

Sweet Home 3D

и другие

Для реализации используется “Sketch-UP”, который хорошо взаимодействует с “Google Earth”. Но также можно использовать более мощный, более совершенный инструмент – “3-d MAX”.

Была поставлена задача – создать модель здания «Института Искусственного Интеллекта». Для этого в программе “Google Earth” получили местность, на которой будет происходить прорисовка здания, далее была получена объемная модель здания, на которую наложили текстуры. Вид здания изображен на рис.3

Далее 3d модель сохраняется в двух вариантах:

формат “.skp” – для дальнейшей работы в “Sketch-UP”;

формат “.dae” – для дальнейшей работы в “Google Earth”;

Для того, чтобы сохранить модель в формате “.dae”, необходимо выполнить File->Export-> 3d Model. На данном этапе завершается создание самой модели. Для дальнейшей интеграции в “Google Earth”, необходимо выполнить следующие действия:

в “Google Earth” выбрать Добавить->Модель;

выбрать ранее созданную модель;

подкорректировать координаты, чтобы точнее установить здание (если есть необходимость);

сохранить полученный результат, выполнив следующие команды Файл->Сохранить->Сохранить местоположение как;

получить файл с расширением “.kmz”

Именно файл с расширением “.kmz” можно активно использовать в “WEB-API” от “Google”.



Рис.3 – Модель здания в “Sketch-UP”

3. Портал «Виртуальный город»

Для демонстрации 3-d модели здания был создан прототип портала «Виртуального города ДонНТУ», который в дальнейшем будет основой для построения полноценного портала. На рис.4 изображен данный ресурс.



Рис.4 – Прототип портала с использованием “Google’s API”

Для реализации портала использовалось WEB-API, код которого представлен ниже

```
<script type="text/javascript"
src="http://www.google.com/jsapi?key=ABQIAAAAvJQ37IT9gerg2OrLfeqUhhSijads_sx4bEBiT
OLimmHLubfHAXt8PFnF1pVcW6pj32ehVxrGKOsdtg"></script>
<script type="text/javascript">
google.load("earth", "1");

var COUPUS_NUMBER =9;

var ge = null;
var kmlObjects = new Array(COUPUS_NUMBER);
var i;
var wooYayIntervalId = -1;

function init() {
  google.earth.createInstance("map3d", initCB, failureCB);
}

function initCB(object) {
  ge = object
  ge.getWindow().setVisibility(true);

  // add a navigation control
  ge.getNavigationControl().setVisibility(ge.VISIBILITY_AUTO);

  // add some layers
  ge.getLayerRoot().enableLayerById(ge.LAYER_BORDERS, true);
  ge.getLayerRoot().enableLayerById(ge.LAYER_ROADS, true);

  create3dModels();

  la = ge.getView().copyAsLookAt(ge.ALTITUDE_RELATIVE_TO_GROUND);
  la.setLatitude(47.994165768204);
  la.setLongitude(37.80888382229992);
  la.setRange(500);
  la.setTilt(45);
  la.setHeading(270);
  ge.getView().setAbstractView(la);
}

function failureCB(object) {
  alert('load failed');
}

function kmlFinishedLoading(kmlObject) {
  if (kmlObject) {
    var n=kmlObject.getName();
```

```
        ge.getFeatures().appendChild(kmlObject);
        kmlObjects[n]=kmlObject;
    }
    else
    {
        alert("Ошибка загрузки трехмерной модели!!!");
    }
}

function create3dModels() {
    google.earth.fetchKml(ge,'http://boristest.pp.ua/build_1.kmz', kmlFinishedLoading);
}

</script>
```

Особую важность имеют строки подключения, где необходимо указать ключ от “Google”, для этого необходимо пройти авторизацию, а также функция create3dModels(), в которой непосредственно указываются какие модели необходимо загружать. Стоит отметить, что функция google.earth.fetchKml(ge,url, kml) работает с url, а это значит, что свои модели надо размещать на каком-то сервере, создать свой домен.

Выводы

В этой работе были рассмотрены принципы построения модели, ее интеграции в “Google Earth”, был представлен прототип портала «Виртуального города ДонНТУ». Дальнейшие действия – создание большего количества моделей, улучшение ранее созданных моделей, улучшение портала, возможность использования полученных результатов другими пользователями и разработчиками.

Список литературы

1. Официальный сайт компании Google Inc. [Электронный ресурс] / Все о Google. – Режим доступа: <http://www.google.com/about.html>
2. Википедия – свободная энциклопедия [Электронный ресурс] / Google – Википедия. – Режим доступа: <http://ru.wikipedia.org/wiki/Google>
3. Google Code [Электронный ресурс] / Google – Google Code. – Режим доступа: <http://code.google.com/>
4. Google SketchUP [Электронный ресурс] / Google – Google SketchUP. – Режим доступа: <http://sketchup.ru/>
5. Неогеография и концепция электронного города [Электронный ресурс] / Аноприенко А.Я. – Режим доступа <http://ea.donntu.edu.ua:8080/jspui/bitstream/123456789/3543/1/2009-04-22-anopriyenko-bashkov-eremchenko-e-city.pdf>
6. Разработка трехмерной компьютерной модели корпусов ДонНТУ и особенности ее визуализации в Интернет [Электронный ресурс] / Аноприенко А.Я. – Режим доступа <http://ea.donntu.edu.ua:8080/jspui/bitstream/123456789/3143/1/2005-anopriyenko-babkov-donntu3d.pdf>
7. Как превратить карту в виртуальный город: brute force, уловки и роботы [Электронный ресурс] / Режим доступа <http://blogs.computerra.ru/13004>
8. Модели корпусов ДонНТУ [Электронный ресурс] / Ильин А.Н. – Режим доступа <http://googleapi.at.ua>