







Ajax and Web Services in PHP

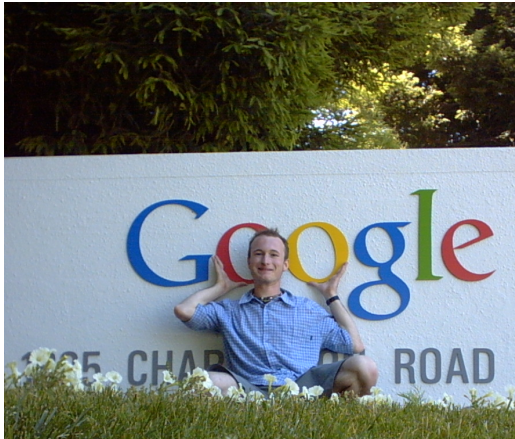
Google AdWords API with APIility

Forum PHP 2006, Paris
November 9, 2006

Thomas Steiner Steiner.Thomas@gmail.com
in cooperation with
Patrick Chanezon chanezon@google.com
Jacob Moon jacobm@google.com



-  Presentation of the speaker and introduction
-  About Google AdWords and the AdWords API
-  Introduction of the Google APIity Library (live demo)
-  AdWords goes Ajax: APIity "ajaxified" (live demo)
-  Mashing up Google Checkout and AdWords (live demo)
-  Questions



Maintainer of the **APIity Library**
and the **APIlitAx project**

Steiner.Thomas@gmail.com
<http://blog.tomayac.de>

Google engineering intern in summer 2006

French/German double degree MS CS grad student

– ENSIMAG (Grenoble, France)



– University of Karlsruhe (Germany)



How does AdWords work?

- **You create your ads**

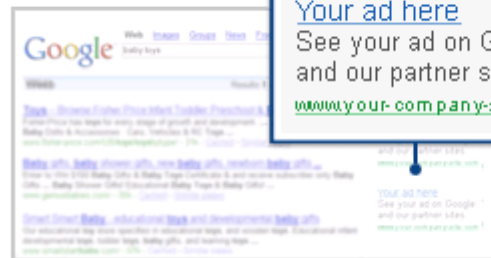
You create ads and choose keywords, which are words or phrases related to your business

- **Your ads appear on Google**

When people search on Google using one of your keywords, your ad may appear next to the search results

- **People can simply click** your ad to make a purchase or learn more about you

Your ads appear beside related search results...



People click your ads...

...And connect to your business

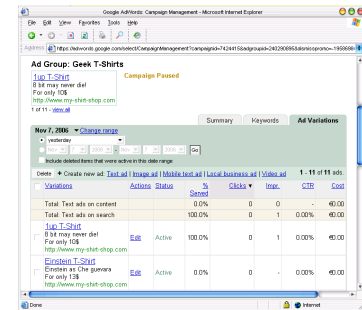


Why does AdWords require an API?

API = **A**pplication **P**rogramming **I**nterface, permits two or more systems to talk efficiently to each other



AdWords API



Web Front End

The API defines a programmatic way to access AdWords

What is the AdWords API good for?

- Lets developers design computer programs that interact directly with the AdWords platform.
- Allows automation of common tasks such as:
 - Generating automatic keyword, ad text, and custom **reports**
 - **Integrating AdWords** with databases or inventory systems
 - Developing **additional applications**

Who can use the AdWords API?

- All advertizers can use the API
- Simply sign up for a **My Client Center** account



How is API usage administrated?

The screenshot shows the Google AdWords API Center interface. At the top, there is a navigation bar with the Google AdWords logo, a "My Client Center" link, and a "Jump to client" dropdown menu. Below this is a secondary navigation bar with tabs for "My Client Center", "Client Reports", and "My Account", and a row of links for "Billing Preferences", "Account Preferences", "Access", and "AdWords API Center".

The main content area is titled "AdWords API Center" and contains several sections:

- API Usage:** Shows usage for October 2006. Quota units used: 0. Purchasable quota units remaining: 250,000. A note states "Usage data is not in real time (updated daily)."
- Rate and Budget:** Rate: USD \$0 per 1000 quota units. Monthly budget: \$1.00.
- Your Developer Token:** Shows a token: abcdefghijklmnopqrstuvwxyz. A "reset" link is available.
- Your Application Tokens:** Shows a token: abcdefghijklmnopqrstuvwxyz. Links for "add" and "label" are present. A "Details" link is also available.
- Developer Details:** Shows company information: Independent software developer, Google, http://google.com, 501+ employees, chanezon@gmail.com.

- Each API user is assigned a unique *developer token* which is linked to the user's My Client Center account
- Quota costs $\$0.25/1000$ quota units, billed to a developer token
- *Application token* is specific to the application you develop
- Users can access information about their *API usage* via the AdWords API Center and
 - Generate, view, or reset their developer token
 - Check their quota allocation and quota units consumed

Where can I get started?

- Using the Developer Website, developers can find valuable resources to help them create applications
- Designed to be a do-it-yourself program

Google AdWords API

Introduction
Google's free AdWords API service lets developers engineer computer programs that interact directly with the AdWords server. With the applications created, advertisers can more efficiently - and creatively - manage their large AdWords accounts and campaigns.

Limitless Functionality
What can you do with the AdWords API? This all depends on your programming genius and clients' advertising needs. Some possibilities might include:

- Generating automatic keyword, ad text, URL, and custom reports
- Integrating AdWords data with databases, such as inventory systems
- Developing additional tools and applications to help you manage AdWords accounts

Multi-Language Compatibility
To make your programming environment even easier, the AdWords API supports a growing number of languages - including Java, PHP, Perl, Java, C#, and XML. The API abides by SOAP and WSDL standards.

Simple and Speedy Signup
All you need to get started is an active [My Client Center](#) account. If you don't currently have a My Client Center, you can create one in the next step. After signup, you'll be given a [developer token](#) and [operations quota](#) so you can begin writing programs immediately.

Ready to sign up? [Click here](#)

Because AdWords API is a free beta, the online documentation is available in English only. However, all API services are compatible with any language and currency.

[Sign in to your AdWords API Center](#)

Sample Code
Learn from sample code, available in five programming languages:
[Java](#) · [.NET](#) · [Perl](#) · [PHP](#) · [XML](#)

Support
[AdWords API FAQ](#)
Search or browse the FAQ for answers to common questions.
[Ask Other Developers](#)
Find answers, or post questions for the Developer Forum

©2005 Google - [AdWords Home](#)

Key components:

- Developer's Guide
- Developer Forum
- Frequently Asked Questions
- Sample Code
- Blog

<http://www.google.com/apis/adwords>

What does API usage cost?

- Usage is measured in quota
- Each unit costs \$0.25

Type of Operation	Quota Unit(s)
Keyword status	1 unit
Change keyword bid	10 units
Traffic estimator	25 units
Add keyword	50 units
Add creative	250 units
Pull report	1000 units

- This is a simplified version of the actual rate sheet, which lists all available API services
- However, the numbers reflected here are accurate and represent all the changes that are being made

How do I sign up for API usage?

Production



- Go to <http://www.google.com/apis/adwords>
- Create your **MCC (My Client Center) account** using your email address and password
- Developer and application tokens will be generated
- Set a limit for API quota: you pay for quota
- Visit the API website to view more information including documentation, sample code, and the Developer Forum

Sandbox



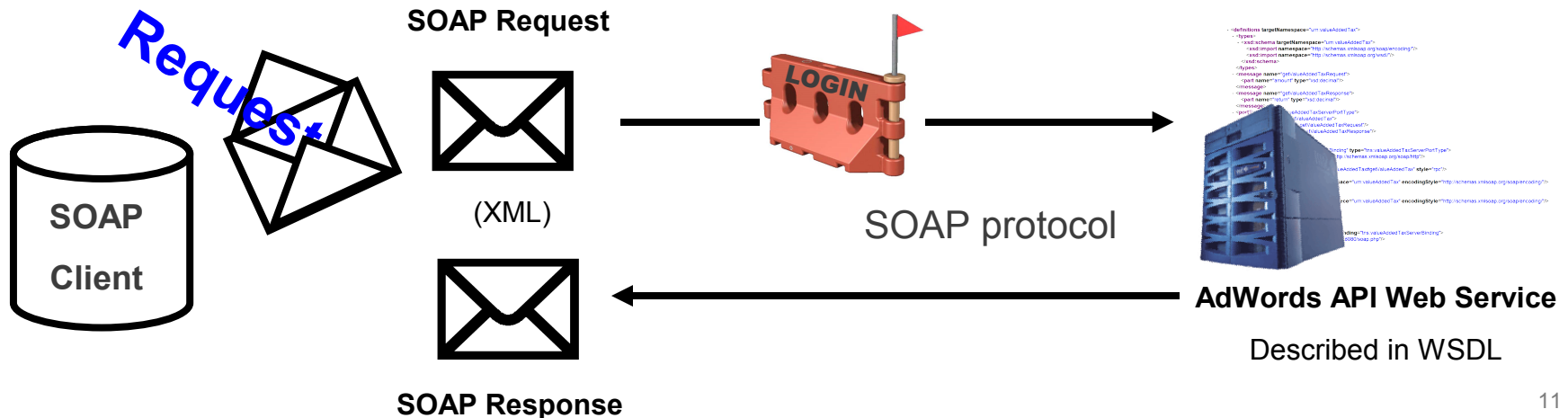
- For development create a sandbox account
- **No quota cost** there

What do I need in order to use the AdWords API?

- **SOAP-based API** → multi-language compatibility
- Supported languages: Java, .NET, Perl, PHP, Python, OCaml, Ruby, XML, and more...

What is SOAP?

- Protocol for exchanging **XML-based messages** over a computer network, normally using HTTP



What goes actually on the wire?

```
<?xml version="1.0" encoding="utf-8" ?>
<env:Envelope xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:env="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <env:Header>
    <developerToken env:mustUnderstand="0">your_dev_token
    </developerToken >
    <applicationToken env:mustUnderstand="0">your_app_token
    </applicationToken >
    <useragent env:mustUnderstand="0">P@ playing with the API from PHP
    </useragent>
    <password env:mustUnderstand="0">your_password
    </password>
    <email env:mustUnderstand="0">toto@example.com
    </email>
  </env:Header>
  <env:Body>
    <estimateKeywordList xmlns="https://adwords.google.com/api/adwords/v6">
      <keywordRequests>
        <maxCpc>50000</maxCpc>
        <text>flowers</text>
        <type>Broad</type>
      </keywordRequests>
    </estimateKeywordList>
  </env:Body>
</env:Envelope>
```

What's in the SOAP header?

```
<env:Header>  
  <developerToken env:mustUnderstand="0">your_dev_token  
  </developerToken >  
  <applicationToken env:mustUnderstand="0">your_app_token  
  </applicationToken >  
  <useragent env:mustUnderstand="0">P@ playing with the API  
  </useragent>  
  <password env:mustUnderstand="0">your_password  
  </password>  
  <email env:mustUnderstand="0">toto@example.com  
  </email>  
</env:Header>
```

How about the SOAP body?

- Schema defined in the WSDL file for the service
- Namespaced with version specific URL

`https://adwords.google.com/api/adwords/vN`

```
<env:Body>
  <estimateKeywordList
    xmlns="https://adwords.google.com/api/adwords/v6">
    <keywordRequests>
      <maxCpc>50000</maxCpc>
      <text>flowers</text>
      <type>Broad</type>
    </keywordRequests>
  </estimateKeywordList>
</env:Body>
```

AdWords SOAP details (cont'd)



```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <soapenv:Header>
    <responseTime soapenv:actor="http://schemas.xmlsoap.org/soap/actor/next"
      soapenv:mustUnderstand="0"
      xmlns="https://adwords.google.com/api/adwords/v6">1999</responseTime>
    <operations soapenv:actor="http://schemas.xmlsoap.org/soap/actor/next"
      soapenv:mustUnderstand="0"
      xmlns="https://adwords.google.com/api/adwords/v6">1</operations>
    <units soapenv:actor="http://schemas.xmlsoap.org/soap/actor/next"
      soapenv:mustUnderstand="0"
      xmlns="https://adwords.google.com/api/adwords/v6">25</units>
    <requestId soapenv:actor="http://schemas.xmlsoap.org/soap/actor/next"
      soapenv:mustUnderstand="0"
      xmlns="https://adwords.google.com/api/adwords/v6">e701e5123f[...]</requestId>
  </soapenv:Header>
  <soapenv:Body>
    <estimateKeywordListResponse xmlns="https://adwords.google.com/api/adwords/v6">
      <estimateKeywordListReturn>
        <id>-1</id>
        <lowerAvgPosition>7.0</lowerAvgPosition>
        <lowerClicksPerDay>111.112045</lowerClicksPerDay>
        <lowerCpc>50000</lowerCpc>
        <upperAvgPosition>10.0</upperAvgPosition>
        <upperClicksPerDay>167.89006</upperClicksPerDay>
        <upperCpc>50000</upperCpc>
      </estimateKeywordListReturn>
    </estimateKeywordListResponse>
  </soapenv:Body>
</soapenv:Envelope>
```

What's in the SOAP response body?

```
<soapenv:Body>
  <estimateKeywordListResponse
    xmlns="https://adwords.google.com/api/adwords/v6">
    <estimateKeywordListReturn>
      <id>-1</id>
      <lowerAvgPosition>7.0</lowerAvgPosition>
      <lowerClicksPerDay>111.112045</lowerClicksPerDay>
      <lowerCpc>50000</lowerCpc>
      <upperAvgPosition>10.0</upperAvgPosition>
      <upperClicksPerDay>167.89006</upperClicksPerDay>
      <upperCpc>50000</upperCpc>
    </estimateKeywordListReturn>
  </estimateKeywordListResponse>
</soapenv:Body>
```


How about error handling?

- SOAP fault elements
- Include AdWords specific fields: code, message, trigger

```
<soapenv:Fault>
  <faultcode>soapenv:Server.userException</faultcode>
  <faultstring>Invalid client email specified.</faultstring>
  <detail>
    <ns1:code xmlns:ns1="https://.../v6">9</ns1:code>
    <ns2:message xmlns:ns2="https://.../v6">
      Invalid client email specified.</ns2:message>
    <ns3:trigger xmlns:ns3="https://.../v4">toto</ns3:trigger>
  </detail>
</soapenv:Fault>
```

I don't care about SOAP. I speak {Java|PHP|Perl|...}



No, thanks!

- External developers and Google offer **client libraries** that provide an abstraction from SOAP
 - Java Client Library (Java)
 - Adwords4r (Ruby)
 - Merjis AdWords Toolkit (OCaml)
 - ...

This presentation is about APiIity [əpɪlɪtɪ]

- Like "ability" but with 'p', for **PHP** and the **ability** to do whatever you want with the **API**.

Tell me more about APiIity's features!

- Fully **object oriented**
- Runs in both **PHP4**, and **PHP5**
- **Consistent** in usage, all objects have adequate methods
- Released under the very business-friendly **BSD licence**
- **Open source** project - everyone is encouraged to modify, read or learn from the sources



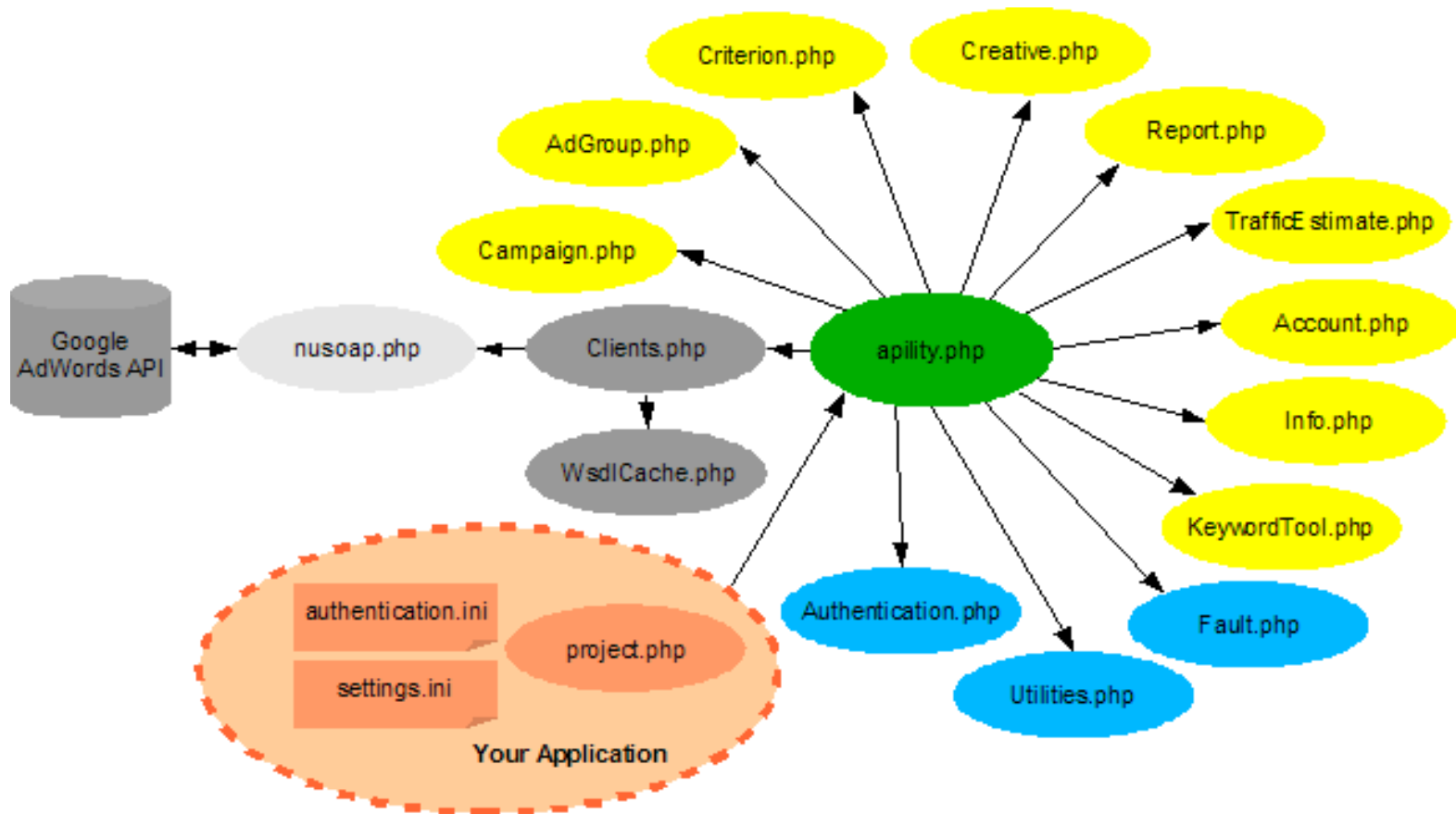
Which credo has guided the development of APiIity?

- Larry Wall: *"Easy things should be easy and hard things should be possible"*

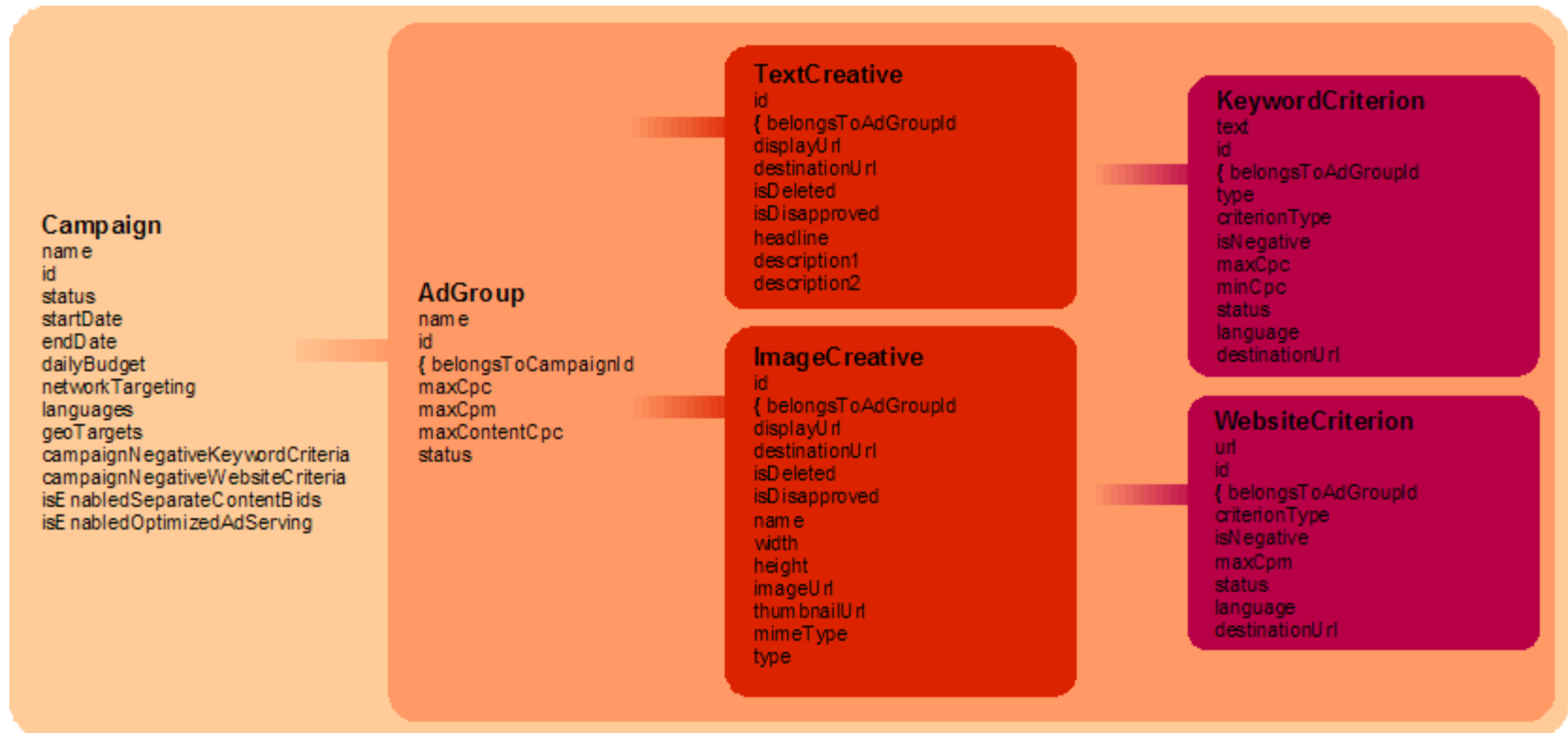
→ <http://google-apility.sourceforge.net>



APIity's class architecture



APIity's object hierarchy



Object oriented design:

```
- $adGroupObject = createAdGroupObject(123456789) ;  
  $adGroupObject->setName("newName") ;  
  $adGroupObject->setMaxCpc(0.65) ;  
  echo "MaxCPC: " . $adGroupObject->getMaxCpc()
```

Principle of the least surprise → consistency

- Setting object statuses

```
- $adGroupObject->setStatus("Paused") ;  
- $campaignObject->setStatus("Paused") ;
```

- Removing objects

```
- removeAdGroup($adGroupObject) ;  
- removeCampaign($campaignObject) ;  
- removeCriterion($criterionObject) ;
```

Programming samples:

- Hello World

```
- require_once('apility.php');  
  print_r( getAllCampaigns() );
```

- Adding a new keyword: `addKeywordCriterion()`

- Parameters:

- String text
- Integer belongsToAdGroupId
- String type
- Boolean isNegative
- Integer maxCpc
- String language
- String destinationUrl

```
- $criterion = addKeywordCriterion("Forum PHP",  
  123456789, "Exact", false, 0.65, "fr",  
  "http://www.google.fr");
```

APIIity live demo

APIlity allows data export in XML format

```
- $campaignObject = createCampaignObject(123456789) ;  
  echo ( $campaignObject->toXml() );
```

- XML? → Ajax

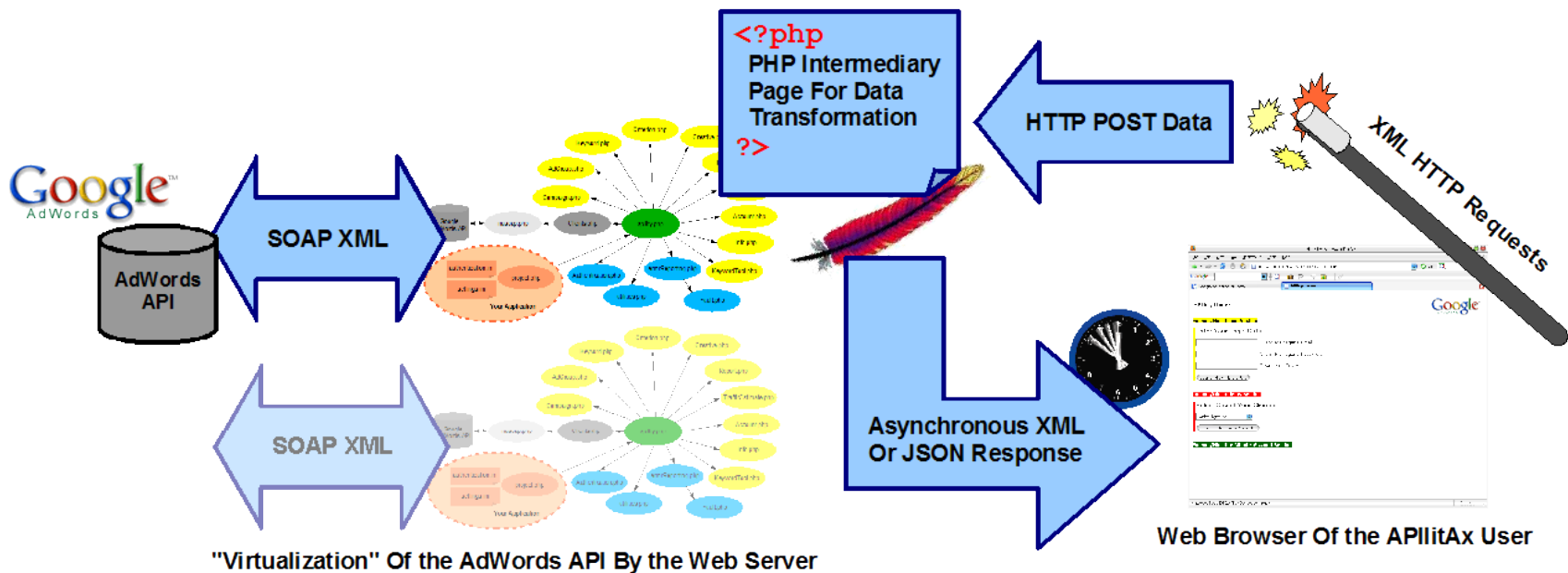
Idea



- Have a complex web application do the **GUI** stuff (JavaScript)
- Use APIlity calls for the AdWords API communication (PHP)
- Use XML for the communication between the two layers
- Call the project APIlity + Ajax = **APIlitAx**

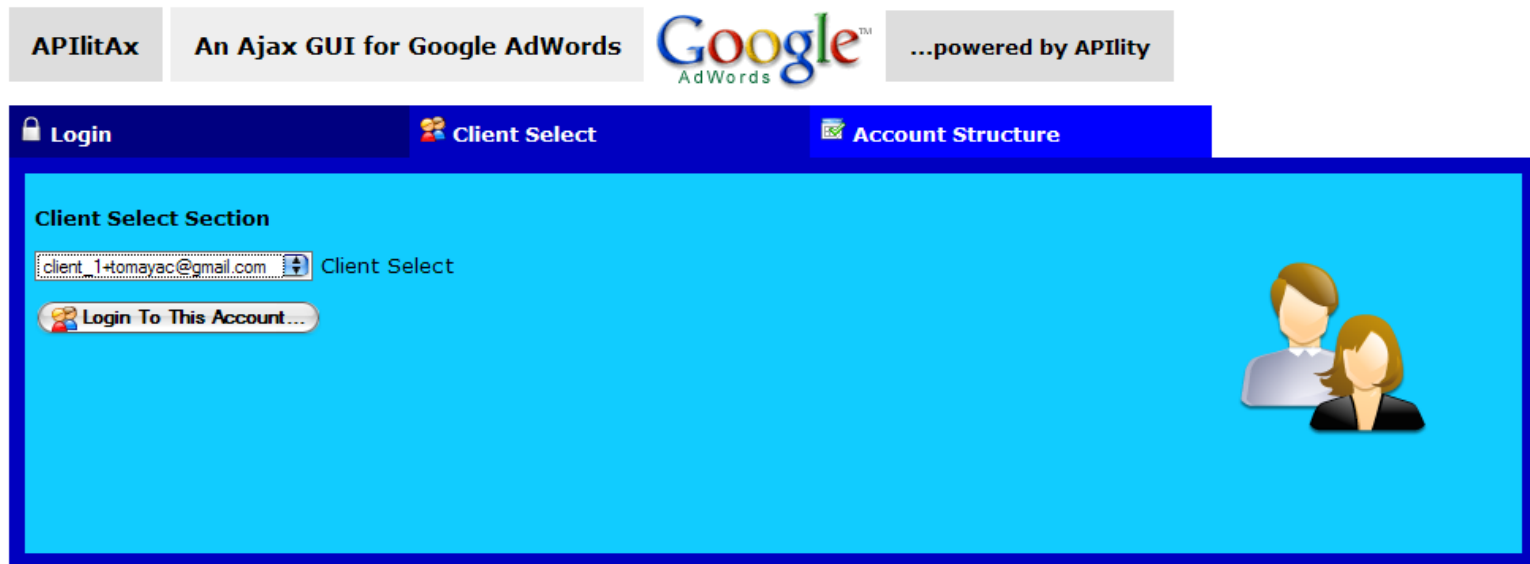
Principle

- JavaScript XML HTTP requests from a web application access intermediate local PHP files communicating via HTTP POST
- PHP files make the APIity API calls, running on the local server
- Asynchronous responses are translated back into JavaScript



What does it look like?

- Client Select section:



APiIity is an open source project released under the [BSD license](#). It is hosted on [SourceForge](#) and provided "as is" without any warranty. Graphics based on Everaldo's [Crystal icon theme](#).



→ <http://apilitax.sourceforge.net>

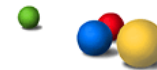


What does it look like?

- Account Structure section:

The screenshot shows the APiIity interface for the 'Account Structure' section. At the top, there are navigation tabs for 'Login', 'Client Select', and 'Account Structure'. Below the navigation, there's a header for 'Account Structure Section' with options to 'Reload Campaigns' and 'Add Campaign', and a 'Selected Client' field containing 'client_1+tomayac@gmail.com'. The main content area displays two campaign cards. Each card includes a 'Campaign Name' field, a 'DailyBudget' input, 'StartDate' and 'EndDate' pickers, 'NetworkTargeting' checkboxes for 'ContentNetwork', 'GoogleSearch', and 'SearchNetwork', 'GeoTargets' and 'Languages' sections with 'Modify...' buttons, and a 'Status' dropdown set to 'Active'. The 'Cou' card also shows 'Countries' (FR, DE, +1 other) and 'SeparateContentBids' (Enabled). The 'RegMetCit' card shows 'Regions', 'Metros', and 'Cities' (US-CA, US-HI, +1 other; 527, 548, +2 others) and 'Languages' (de, fr). Both cards have 'Update', 'Delete', 'Details', and 'Detach' actions. At the bottom of each card is an 'AdGroups' section with an 'Add AdGroup' button.

APiIity is an open source project released under the [BSD license](#). It is hosted on [SourceForge](#) and provided "as is" without any warranty. Graphics based on Everaldo's [Crystal icon theme](#).



➔ <http://apilitax.sourceforge.net>



APIlitAx live demo



What is Google Checkout?

- Many shopping carts are cancelled during checkout process
- ➔ Google Checkout is an alternative checkout process that provides a **fast, secure way to buy and sell online**

The screenshot shows a Google search for "black handbags". The search bar contains the text "black handbags" and the search button is labeled "Search". Below the search bar, the results are displayed under the heading "Web". The results include several sponsored links and organic search results. A callout box highlights the text "Find the Perfect Black Handbag" and "Fabulous & Unusual Styles. Great Customer Service & Free Shipping!" from the organic result for "www.RetailCo.com".

Google Web Images Groups News Froogle Local more »
black handbags Search Advanced Search Preferences

Web Results 1 - 10 of about 18,500,000 for **black handbags**. (0.24 seconds)

Sponsored Links
Handbags at Coach.com
www.Coach.com Find New & Stylish Coach Designs - Online Now at the Official Site

All the HandBags
Handbags.
Find the Perfect Black Handbag
Fabulous & Unusual Styles. Great Customer Service & Free Shipping!
www.RetailCo.com

Sponsored Links
Designer Handbags Online
All Major Brands. Huge Selection. Free Shipping. Order Now!
Zappos.Com

Black Purses
Compare Prices on a Wide Variety of Apparel and Save Money.
PriceGrabber.com

Find the Perfect Black Handbag
Fabulous & Unusual Styles. Great Customer Service & Free Shipping!
www.RetailCo.com

Women's Handbags
Spring Sale: Up To 50% Off Select Styles and \$5 Standard Shipping.
www.AnnTaylorLoft.com

Find the Perfect Handbag
Fabulous & Unusual Styles. Great Customer Service & Free Shipping!
www.eBags.com

AdWords and Google Checkout work together to

- increase sales
- lower costs.

2 Get **more traffic** with the Google Checkout badge

1 Promote your business with AdWords

3 Get **higher sales conversions** by offering a streamlined checkout process

4 Lower your costs with free transaction processing



Checkout APIItAx...

What does it look like? – Vendor Side

Checkout APIItAx Google Checkout in Combination with Google AdWords

Google Checkout & Google AdWords ...powered by APIItAx
Fast checkout through Google

Vendor Side (Your Online Store Back-End)

Online Store Inventory as retrieved from Database:

Image	Item Name	Item Description	Unit Price	In Stock	ID	DB Access	AdWords Access
	Tup T-Shirt	8 bit may never die!	10\$	50	1	Update Item Delete Item	 Create AdWords
	Einstein T-Shirt	Einstein as Che	13\$	130	2	Update Item Delete Item	 Create AdWords
	Major League Gamer	For hardcore players.	13\$	500	3	Update Item Delete Item	 Create AdWords
	The ESC T-Shirt	Hit and then run.	12\$	40	4	Update Item Delete Item	 Create AdWords
	WASD T-Shirt	Gamers know what this m	12\$	40	5	Update Item Delete Item	 Create AdWords
	Perl T-Shirt	Obfuscated Perl Code.	10\$	10	6	Update Item Delete Item	 Create AdWords

[Add Item](#)

<http://www.my-shirt-shop.com>

Add Creative

Headline

Description 1

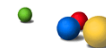
Description 2

Display Url

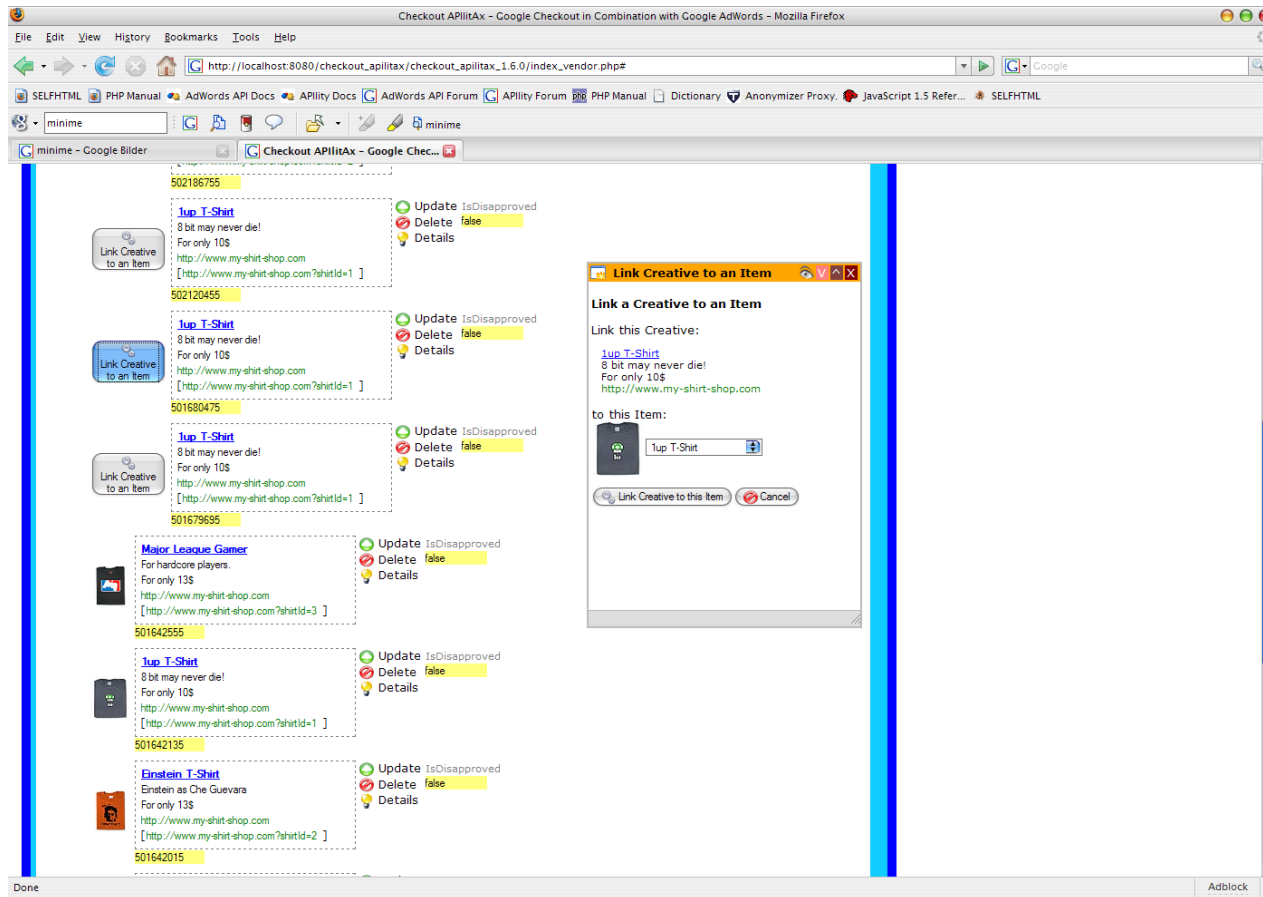
Destination Url

[Add Creative](#) [Cancel](#)

Add a Creative to the current AdGroup



What does it look like? – Vendor Side



What does it look like? – Buyer side

Checkout APIItAx Google Checkout in Combination with Google AdWords

Google Checkout & Google AdWords ...powered by APIITY
Fast checkout through Google

Buyer Side (Your Online Store Front-End)

Select Items and Number to purchase:

Image	Item Name	Item Description	Unit Price	Quantity
	1up T-Shirt	8 bit may never die!	10\$	0
	Einstein T-Shirt	Einstein as Che	13\$	0
	Major League Gamer	For hardcore players.	13\$	0
	The ESC T-Shirt	Hit and then run.	12\$	2
	WASD T-Shirt	Gamers know what this means	12\$	3
	Perl T-Shirt	Obfuscated Perl Code.	10\$	0
Overall Price:			60\$	5 Items

Hidden Shopping Cart Section

Shopping Cart

HTTP POST Data (Plaintext)
\$0_itemName=The%20ESC%20T-Shirt&0_itemDescription=Hit%20and%20then%20run.&

Shopping Cart (XML)

```
<?xml version="1.0" encoding="UTF-8"?>
<checkout-shopping-cart xmlns="http://checkout.google.com/schema/2">
  <shopping-cart>
    <items>
      <item>
        <item-name>The ESC T-Shirt</item-name>
        <item-description>Hit and then run.</item-description>
        <unit-price currency="USD">12</unit-price>
        <quantity>2</quantity>
      </item>
      <item>
        <item-name>WASD T-Shirt</item-name>
        <item-description>Gamers know what this means</item-description>
      </item>
    </items>
  </shopping-cart>
</checkout-shopping-cart>
```

Shopping Cart (Base64-encoded XML)
FD94bWgdmVyc2lvcj0iMS4wI1B1bmlvZGluZz0iVVRGLTgPz4KPGNoZW50b3V0LXNoY3EwaW5nLWlnb3R0

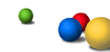
Shopping Cart Signature (Base64-encoded HMAC_SHA1 Signature)
vpd1g07N6Y1od6phAfx9dqxFq4=

This Part is Normally Invisible

Prepare Shopping Cart

Google Checkout
Fast checkout through Google

Show/Hide Debugging Info



Checkout APIlitAx live demo

Questions



Right here, right now

or

Steiner.Thomas@gmail.com