

Scrub (Ajax), Wash (SOAP) and REST

Use Google Checkout and AdWords APIs with PHP for fun and profit.



Thomas Steiner, Google intern, tsteiner@google.com
Intern AdWords API

Jacob Moon, Google, jacobm@google.com
Checkout API support

Patrick Chanezon, Google, chanezon@google.com
Checkout API Evangelist

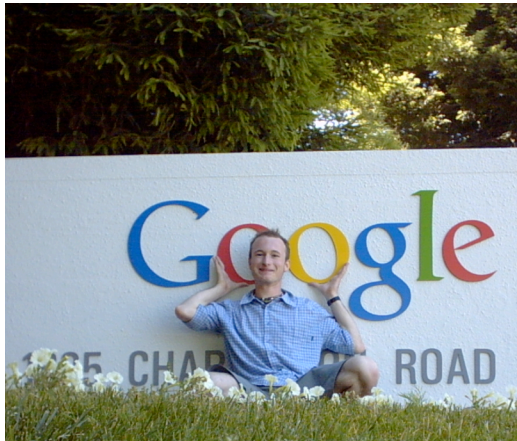
October 31 2006



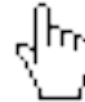
Who is Thomas Steiner?



Internship / Part time at Google



tomac@google.com



API Developer Relations (the better part time)

French/German CS student (the other part time)

<http://blog.tomayac.de>



Jacob Moon in a slide



API Support Engineer – Google Checkout



Maintainer of...

- Google Checkout API PHP Sample Code



- Google Checkout API Windows Classic ASP Sample Code



- Google Checkout module for osCommerce



- Google Checkout module for ZenCart

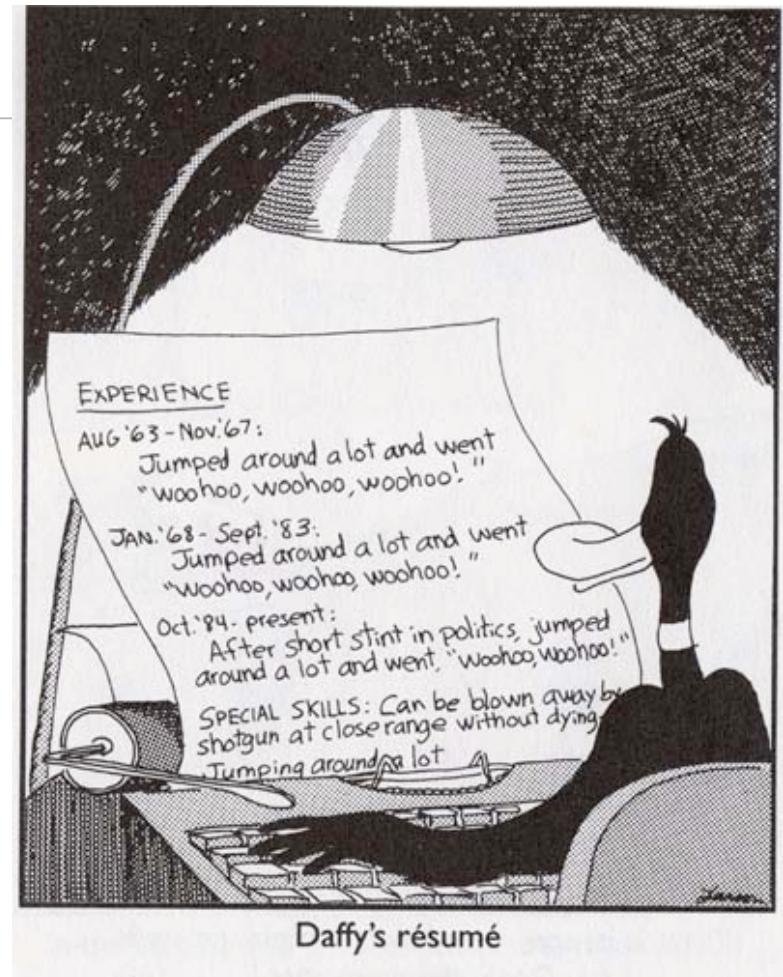


- Google Checkout Developer Community



P@ in a slide

- API Evangelist - Google Checkout
- Software plumber
- Family man: married, 3 kids
- Java geek... in scripting rehab: Ruby, JavaScript, PHP, Python
- Open Source: ROME, AdWords (java, C#, ruby)
- Sun: Blogs, Portals, eCommerce
- Netscape/AOL: LDAP, Calendar, App Servers, CMS, MyNetscape (RSS)
- More on my blog
<http://wordpress.chanezon.com>



Agenda



Happy Halloween!
Google AdWords API

- AdWords API
- Using the API in PHP: APllity
- Friendly Ajax UI in PHP: APllitax

Google Checkout API

- Google Checkout
- Checkout API in PHP

AdWords-Checkout Business Mashup Demo

AdWords API

- Lets you write software that interacts programmatically with the AdWords service
- Bid management, Reporting/Analytics, Creative generation

SOAP/WSDL web services API

- Standards based
- Tools available for these standards in most languages
- Doc/literal encoding

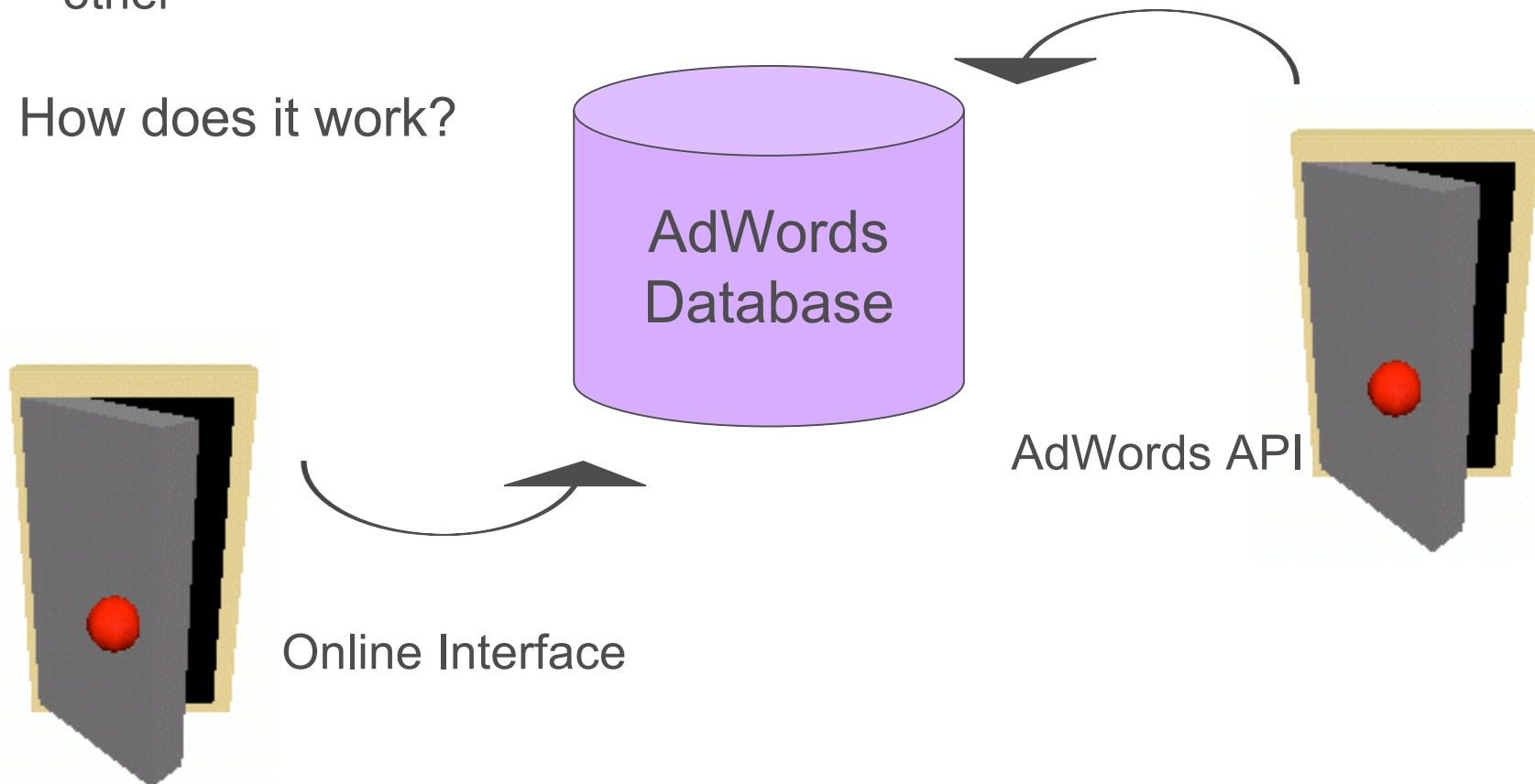
Product overview: AdWords API



What is it?

- A-P-I stands for Application Programming Interface
- It is a platform that permits two systems to talk efficiently to each other

How does it work?



The AdWords API enables anyone to integrate and extend the functionality of AdWords

Users can write programs and applications to perform functions in each of these five areas:

- **Account management** – change user preferences such as email address, password and more at the account level
- **Campaign management** – manage bids, update creative and other campaign operations
- **Reporting** – generate reports to integrate directly into local databases
- **Traffic estimation** – receive traffic estimates for individual keywords
- **Keyword generation** – generate keywords from a base keyword or a web page

Product overview: AdWords API Center



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 under "My Account" tab within My Client Center.

In the AdWords API Center, users can:

Generate, view, or reset their developer token

Check their quota allocation and quota units consumed

The AdWords API is designed to be a do-it-yourself program.* Using the Developer Website, developers can find valuable resources to help them create applications.

Google AdWords AdWords API

AdWords API Home

Overview
[Quota & Usage](#)
[Terms & Conditions](#)
[Featured Use Cases](#)
[Developer Registration](#)

Documentation
[Developer's Guide](#)
[Sample Code](#)

Support
[FAQ](#)
[Developer Forum](#)
[AdWords API Blog](#)

Toolbox
[SOAP Toolkits](#)

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

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>

* AdWords API is available globally, but sign-up and documentation is in **English only**

Resources: Developer Forum



The forum consists of a growing community of developers sharing ideas and helping others succeed. Excellent mechanism to obtain feedback on new business processes or implementation questions.

AdWords API Forum

Start a new topic - Manage group - Invite - Unsubscribe or change membership - About this group

Search this group

Viewing titles only View with message text

Sort by date of first message Sorted by date of most recent message

Thread Title	Author	New Messages	Total Messages	Authors	Date
Reading "operations" header using JAVA(axis)	by thedug	3 new of 5	5	2 authors	8:07am
NET Custom Report	by metrogy	3 new of 3	3	2 authors	Apr 1
KeywordReportJob	by dilip.ramji@gmail.com	3 new of 3	3	2 authors	Apr 1
Zero Report	by dilip.ramji@gmail.com	3 new of 3	3	2 authors	Apr 1
updating keywords...	by mindspore@gmail.com	12 new of 13	13	3 authors	Apr 1
Reports	by inasisi	2 new of 2	2	2 authors	Apr 1
ReportServices	by Wayne	1 new of 8	8	6 authors	Mar 31
Can the API only be accessed from a single Client Center, i.e. by a single agency?	by Akin	1 new of 1	1	1 author	Mar 31
Difference in stats	by googlegroups@peterdunham.me.uk	1 new of 3	3	2 authors	Mar 31
Where can I find my developer token and operations quota?	by Hippo	1 new of 3	3	2 authors	Mar 31
Updating keywords and ad-copy as a batch change - no. of operations?	by Yodey	1 message	1	1 author	Mar 31
Custom Report	by Whaler	1 message	1	1 author	Mar 30
Do reports impact the quota/operations?	by Mario Witte	10 messages	10	5 authors	Mar 30
Quota used up, will i get more next month - even if spend was the same?	by AdWordsAPIAdvisor	2 messages	2	2 authors	Mar 29
Adwords API VB.NET Port	by mike skramstad	7 messages	7	3 authors	Mar 29
why connection is closed incept er?	by KoEnemy	2 messages	2	1 author	Mar 29
Using ApiException c#	by jcheng	1 message	1	1 author	Mar 29
Unique ID	by Whaler	3 messages	3	2 authors	Mar 28
long	by Whaler	3 messages	3	2 authors	Mar 28
Having alot of problems with reportService	by inasisi	2 messages	2	2 authors	Mar 27
Creative Service Irregularities	by jacka@torreycommerce.com	2 messages	2	1 author	Mar 27
Polling Interval for Report Service	by inasisi	3 messages	3	3 authors	Mar 26

Participants:

Google

Advertisers

Commercial developers

Independent developers

Open Source client libraries and samples

PHP, Java, .NET, Ruby, Perl, OCAML (!)

Samples only: Python

See <http://www.google.com/apis/adwords/samplecode.html>

Quota Rate Sheet: What is it?



The rate sheet is like a restaurant menu; it lists all the available API operations, and their associated quota unit values.

Type of Operation	Quota Unit(s) consumed <small>(as of 8/1/05)</small>
Keyword status	1 unit
Change keyword bid <small>(individual or Ad Group level)</small>	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

Register for a developer token

- You will need to create a MCC account

Get your credentials ready

- developerToken: provided during registration, quota attached to it
- applicationToken: specific to an application
- User agent: a String you provide in your program
- Login email: of your MCC account
- Password: of your MCC account
- clientEmail: (optional) of your customers

Production

- Go to <http://www.google.com/apis/adwords/>
- Enter your account email address and password.
- Your “umbrella” MCC account will be created.
- Developer and application tokens will be generated. These tokens are used in each API call to Google.
- 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.
- You’re all set! You can access your AdWords account with the API immediately.

Sandbox

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

Client libraries have an option to dump the xml

- Can be useful in case of issues

```
<?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/v4">
      <keywordRequests>
        <maxCpc>50000</maxCpc>
        <text>flowers</text>
        <type>Broad</type>
      </keywordRequests>
    </estimateKeywordList>
  </env:Body>
</env:Envelope>
```


AdWords uses SOAP doc/literal encoding

- Namespace declaration for SOAP and XML schema

```
<?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">
```

AdWords API SOAP Request Headers



```
<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 ruby
</useragent>
    <password env:mustUnderstand="0">your_password
</password>
    <email env:mustUnderstand="0">toto@example.com
</email>
  </env:Header>
```

Contains AdWords API specific elements

- 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/v4">
  <keywordRequests>
    <maxCpc>50000</maxCpc>
    <text>flowers</text>
    <type>Broad</type>
  </keywordRequests>
</estimateKeywordList>
</env:Body>
```

Anatomy of an AdWords API SOAP Response



```
<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/v4">1999</responseTime>
    <operations soapenv:actor="http://schemas.xmlsoap.org/soap/actor/next"
soapenv:mustUnderstand="0" xmlns="https://adwords.google.com/api/adwords/v4">1</operations>
    <units soapenv:actor="http://schemas.xmlsoap.org/soap/actor/next"
soapenv:mustUnderstand="0" xmlns="https://adwords.google.com/api/adwords/v4">25</units>
    <requestId soapenv:actor="http://schemas.xmlsoap.org/soap/actor/next"
soapenv:mustUnderstand="0"
xmlns="https://adwords.google.com/api/adwords/v4">e701e5123f82cd5888f611eb97230579</requestId>
  </soapenv:Header>
  <soapenv:Body>
    <estimateKeywordListResponse xmlns="https://adwords.google.com/api/adwords/v4">
      <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>
```

Return informations about the call

- responseTime in ms
- Operations
- Units, quota cost
- requestId, useful for support issues

```
<soapenv:Header>
  <responseTime soapenv:actor="http://schemas.xmlsoap.org/soap/actor/next"
soapenv:mustUnderstand="0"
xmlns="https://adwords.google.com/api/adwords/v4">1999</responseTime>
  <operations soapenv:actor="http://schemas.xmlsoap.org/soap/actor/next"
soapenv:mustUnderstand="0" xmlns="https://adwords.google.com/api/adwords/v4">
1</operations>
  <units soapenv:actor="http://schemas.xmlsoap.org/soap/actor/next"
soapenv:mustUnderstand="0" xmlns="https://adwords.google.com/api/adwords/v4">
25</units>
  <requestId soapenv:actor="http://schemas.xmlsoap.org/soap/actor/next"
soapenv:mustUnderstand="0"
xmlns="https://adwords.google.com/api/adwords/v4">
e701e5123f82cd5888f611eb97230579</requestId>
</soapenv:Header>
```

AdWords API SOAP Response Body



Parsed by the client library into an object in the language used

- Typed data, based on XML Schema defined in the WSDL

```
<soapenv:Body>
  <estimateKeywordListResponse
xmlns="https://adwords.google.com/api/adwords/v4">
  <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>
```


SOAPFault elements: faultcode and faultstring, detail

- Include AdWords specific fields: code, message, trigger
- SOAPFaults need to be caught by client libraries
 - To be thrown as Typed Exceptions in the language used
 - With all the fields set, for error reporting
- List of error codes
http://www.google.com/apis/adwords/developer/adwords_api_error_codes.html

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

Use a library: they are maintained, avoid classic errors

Use a local store: API calls cost quota, store data locally in a database

80/20 rules for synchronization: keep most active keywords fresh

Use reports, they're cheap!

Do not use threads (at least not to many)

Log SOAP calls, or at least quota part, and when issues

Respect Terms and Conditions: implement all capabilities of API

Keep up with versions: 2 months

Add Creatives before Keywords

Java and .NET excellent: all code generated, interception structure to log and transform requests, mature

- Gotcha: `<valueType>Specified = true` necessary for value types in .NET

Ruby: soap4r doc/lit support used to be buggy but many improvements this year.

- Still difficult to map to our own package, but hack in `adwords4r`

PHP:

- PHP4 NuSoap: slow, manual xml, but deployed everywhere
- PHP5 Pear: native, fast, but PHP 5 only
- APllity uses NuSoap

Perl:

- waiting for new release of SOAP::Lite from Byrne Reese
- CPAN module, not complete yet

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, .NET, ruby, OCAML, Perl,...

This presentation is about APIlity [əpɪlɪtɪ]

- Like “ability” but with ‘p’, for **PHP** and the **ability** to do with the **API** whatever you want
- Most successful client library for AdWords API
 - We had to spin off a Forum just for it

Tell me more about APllity'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 APllity?

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

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



Object oriented design:

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

Principle of the least surprise → consistency

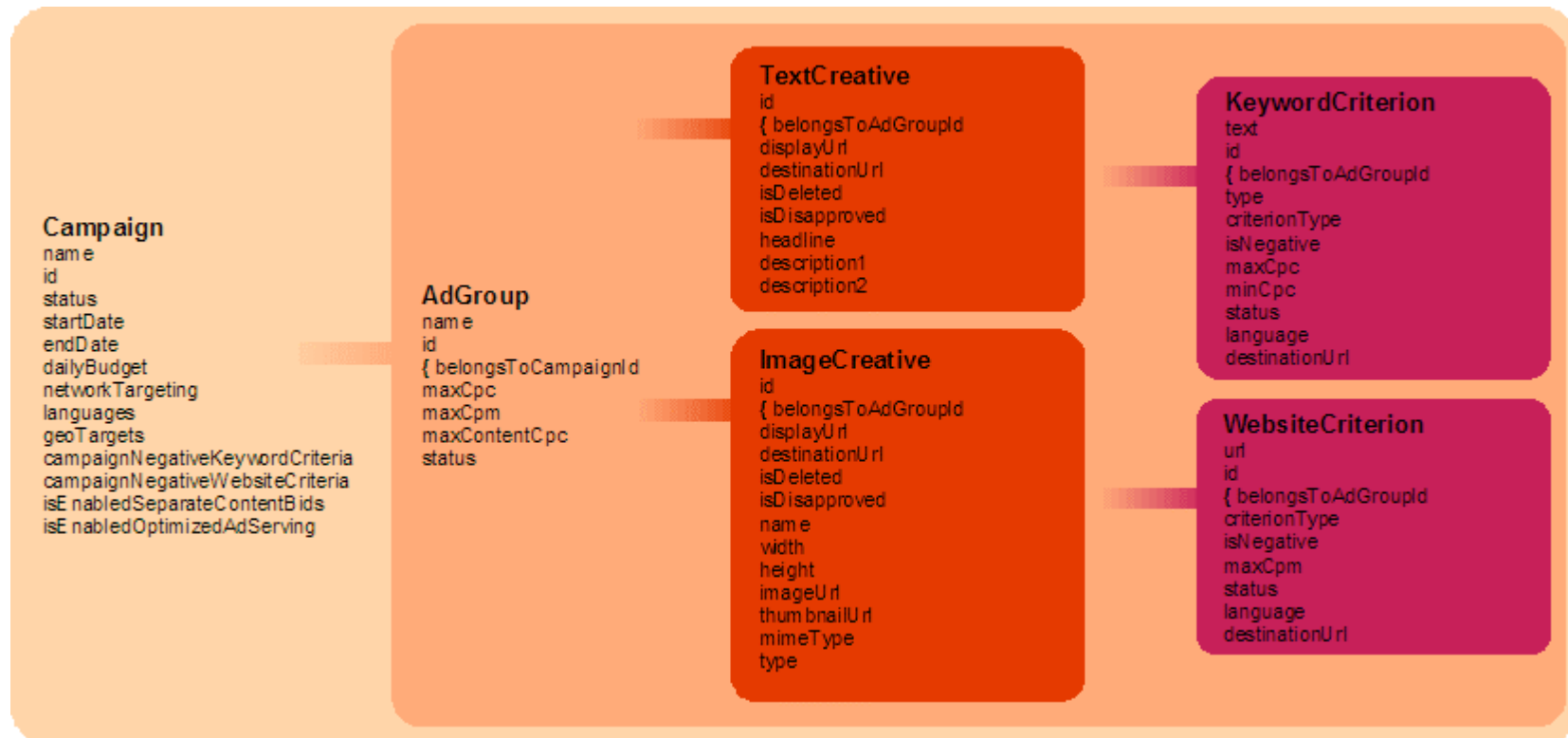
- Setting object statuses

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

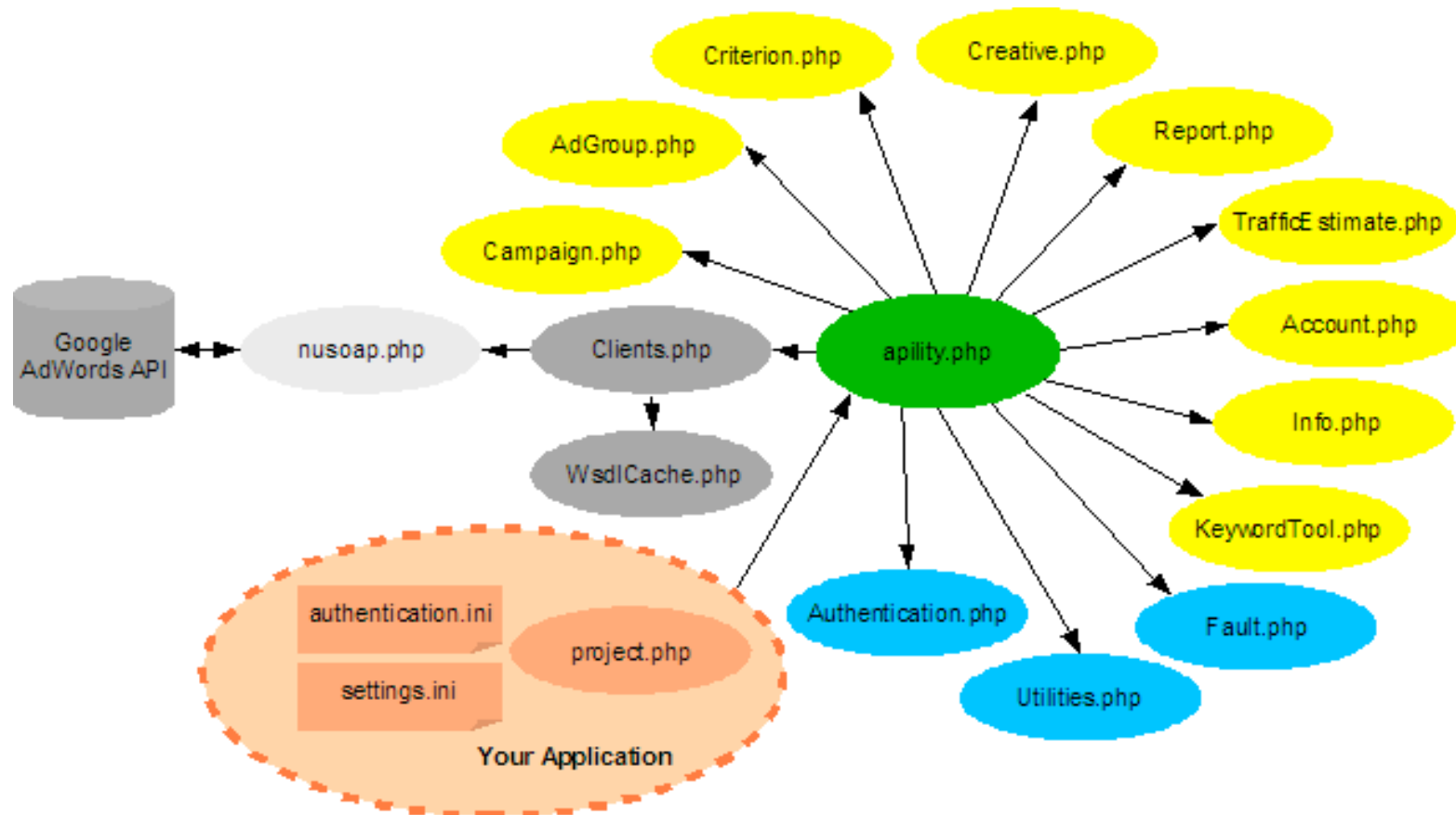
- Removing objects

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


APIity's object hierarchy



APility's class architecture



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

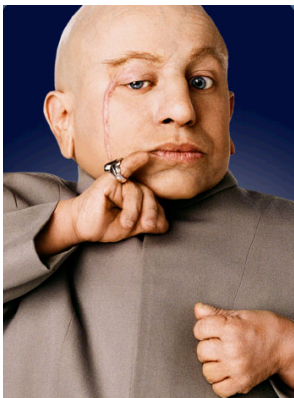
```
– $criterion = addKeywordCriterion("test",  
  123456789, "Exact", false, 0.65, "fr",  
  "http://www.google.com");
```

APllity allows data export in XML format

- XML? → Ajax

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

Idea

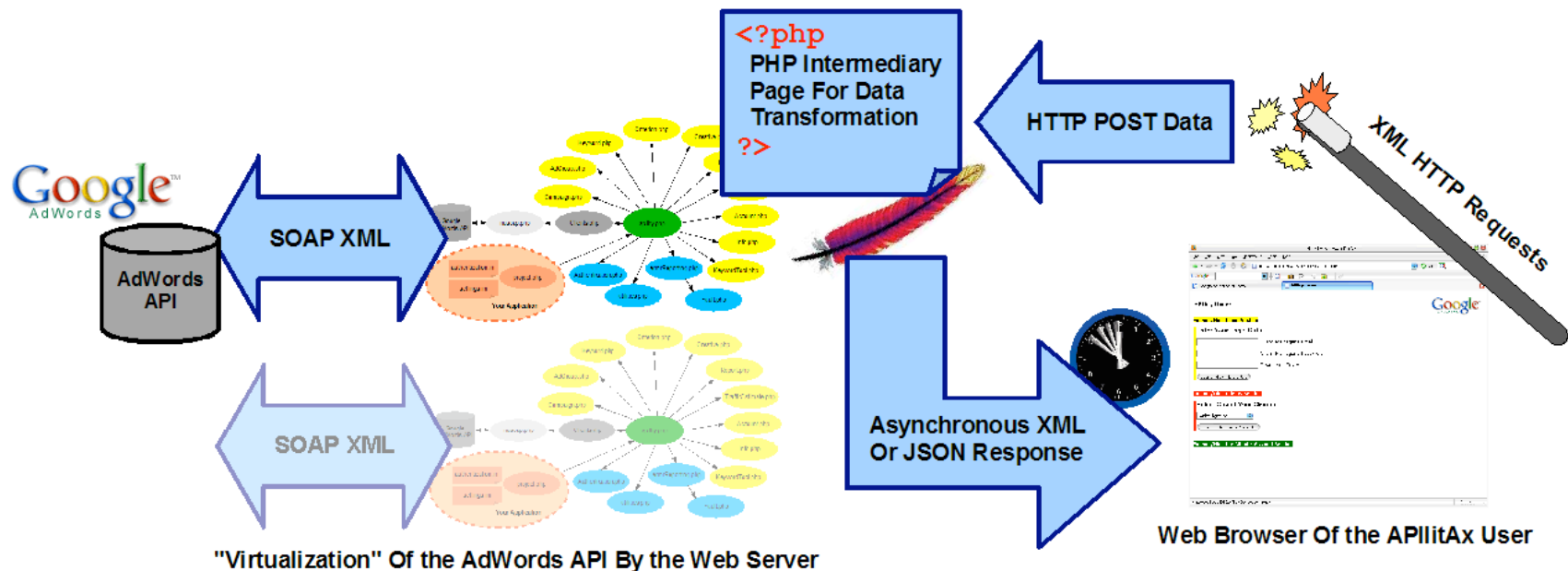


- Have a complex web application do the GUI stuff (JavaScript)
- Use APllity calls for the AdWords API communication (PHP)
- Use XML for the communication between the two layers

- Call the project APllity + Ajax = **APllitAx**

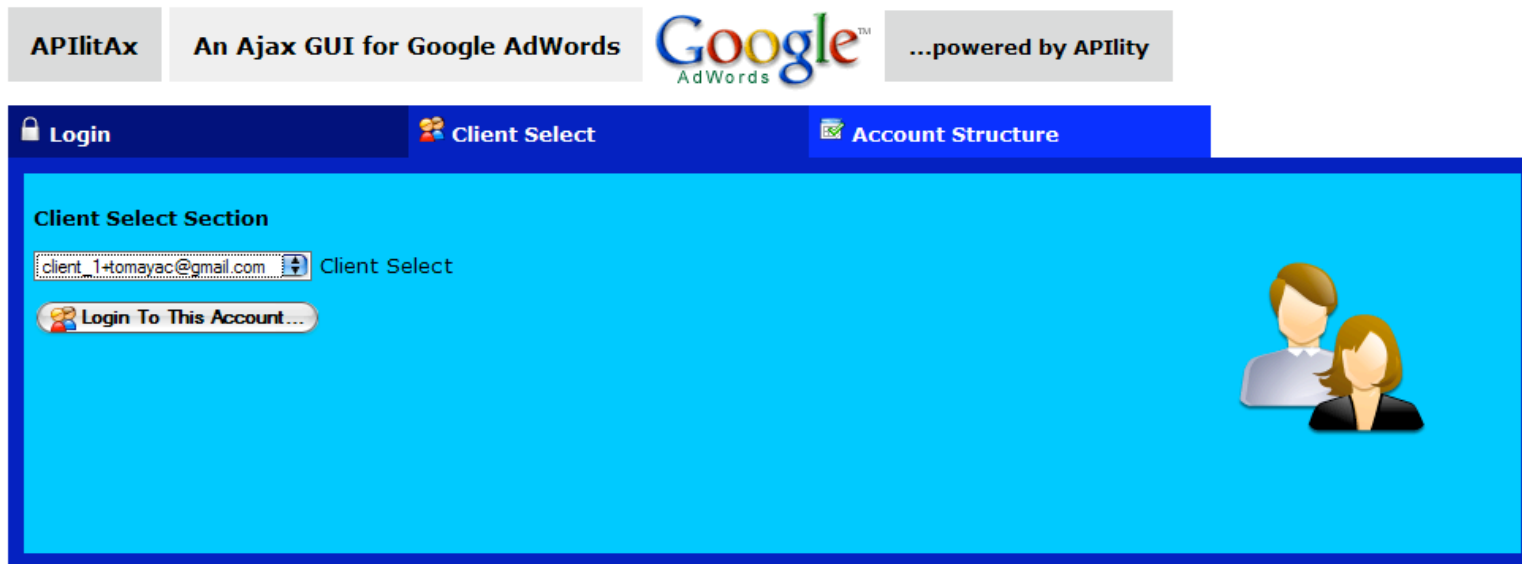
Principle

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

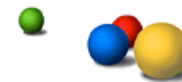


What does it look like?

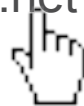
– Client Select section:



APIlitax 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](#).



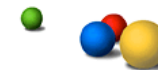
→ <http://apilitax.sourceforge.net>



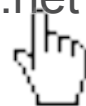
What does it look like?

- Account Structure section:

APIlitax 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>



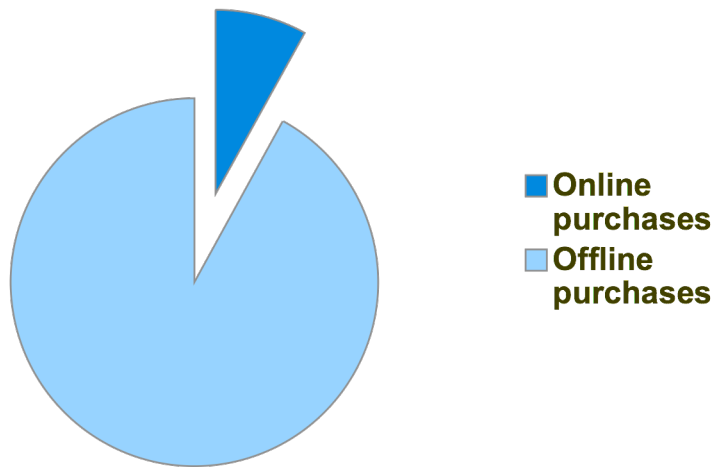


Fast, secure checkout across the Web



Only 8% of all retail purchases are expected to occur online in 2006*

2006 Consumer Purchases by Channel



* 2006 State of Retailing Online, Shop.org/Forrester

Problem: Discovery, Checkout

- Difficult to find trustworthy sellers
- Long, repetitive, checkout processes
- 63% of shopping carts are abandoned after beginning checkout*

* Online Customer Experience Survey, Allurent, February 2006

Opportunity: Search, Convenience

- 37% of online purchases start with search*
- 25% of search queries result in a purchase directly related to the query**
- Fast checkout can improve conversion

* 2005 Online Holiday Mood Study, Shop.org/BizRate Research

** The Role of Search in Consumer Buying, comScore/Google

Google Checkout is an alternative checkout process that provides a fast, secure way to buy and sell online.

Shoppers

- Easily identify Checkout merchants on Google with the Checkout badge
- Conveniently make purchases with a single login/password
- Review and manage purchases from various merchants in one central location

Merchants

- Acquire more leads by displaying the Checkout badge
- Convert more sales by offering a streamlined checkout process
- Lower costs with free transaction processing based on your AdWords spend

retailco

(fictitious company used for illustrative purposes)

Shoppers use Google to **search** for items to buy.



Shoppers look for the Google Checkout badge to find merchants to buy from.

The screenshot shows a Google search results page for the query "black handbags". The search bar at the top contains the text "black handbags" and a "Search" button. Below the search bar, the results are categorized under "Web" and show "Results 1 - 10 of about 18,500,000 for black handbags. (0.24 seconds)".

The search results include several entries, each with a title, a snippet of text, and a URL. A prominent feature is a large, semi-transparent box overlaid on the search results, containing the text "Find the Perfect Black Handbag Fabulous & Unusual Styles. Great Customer Service & Free Shipping!" and the URL "www.RetailCo.com". This box is positioned over the search results for "Handbags at Coach.com" and "Black Handbags".



The search results include:

- Handbags at Coach.com**: Find New & Stylish Coach Designs... Online Now at the Official Site
- Designer Handbags Online**: All Major Brands. Huge Selection. Free Shipping. Order Now! Zappos.Com
- Black Purses & Handbags from Sophie Bags**: 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
- Black Handbags**: My name is Stacey Leigh Black, I am 27 and live in Bredbo NSW. Like most girls, I have a deep obsession for handbags, which I change as often as my shoes. ... www.blackhandbags.com.au/ - 7k - Cached - Similar pages
- black : handbags shop by color: women's : JCPenney**: Women's, Handbags Shop By Color, Black. ... 'Call Me' Leather Handbag - St. John's Bay Woven Double Handle Shopper - St. John's Bay Woven Double Handle ... www.jcpenney.com/products/C033523.jsp - 46k - Cached - Similar pages
- Handbags Purses - Designer Handbags Discount**: Black Handbags, leather handbags, designer purses and designer wallets inspired by top designers like Louis Vuitton, Prada, Coach, Gucci and more! ... www.alternative-beautv.com/handbags/black.html - 30k - Cached - Similar pages
- 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

To **buy**, shoppers need only click Google Checkout and enter their username/password.

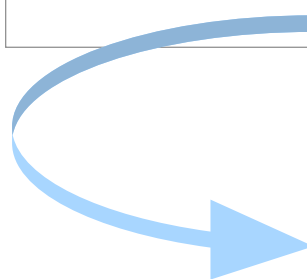
retailco myaccount | shoppingbag | help

Review Your Shopping Bag

Product Description	Price	Qty	Total	Remove
 Kate Landry Fur-Trim Footed Bag	\$29.50	1	\$29.50	

Tax: \$2.43
Subtotal: \$31.93

[Go back and continue shopping.](#) **Checkout** or **Google Checkout**
Fast checkout through Google

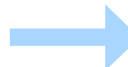


Or sign in
If you already have a **Google Account**

Email:

Password:

[Forgot your password?](#)



retailco Google Checkout

Review and place order

Order Details - RetailCo, Inc., 100 Las Vegas Blvd South, Las Vegas, NV 89109, 800-253-1000

Qty	Item	Price
1	Kate Landry Fur-Trim Footed Bag	\$29.95
	Shipping & handling (CA) <input type="text" value="Media.Mail (\$1.49)"/> :	\$1.49
	Tax (CA) :	\$2.47
Total:		\$33.46

Keep my email address confidential
Google will forward all email from Cooking.com to jemes@gmail.com. [Learn more](#)

I want to receive promotional email from retailco.com.

Ship to: **Todd Jackson - [Change](#)**
Google, Inc.
1600 Amphitheatre Pkwy
Mountain View, CA 94043

Pay with: **Visa xxx-1234 - [Change](#)**

[Edit order](#)

Google Checkout users can easily **manage** all of their purchases in a central location.

Purchase History

Default shipping address
Todd Jackson
Google Inc
1600 Amphitheatre Pkwy
Mountain View, CA 94043
[Edit shipping addresses](#)

Default payment method
Visa xxx-1234
Expires 04/07
Todd Jackson
1600 Amphitheatre Pkwy
Mountain View, CA 94043
[Edit payment methods](#)

[See all sellers I've rated](#)

Orders					1 - 20 of 35 orders	Next »
Order Date	Item	Seller	Status	Amount		
May 30, 2006	Kate Landry Fur-Trim Footed Bag	retailco.com	In progress	\$99.16		
May 18, 2005	Alice in Wonderland Book	Bill's Books	In progress	\$2.45		
Sep 14, 2005	Chocolate Chip cookies	Jane's Cookies	Ready for pickup	\$1.50		
Sep 2, 2005	Michael Jackson	Google Video	Credit card declined	\$27.34		
Sep 2, 2005	Tiger Woods highlights	Google Video	Completed - Rate this seller	\$5.95		
Dec 14, 2004	Tree landscape	Smith Photography	Cancelled Refund - Dec 15, 2003	\$15.34 (\$15.34)		
Dec 14, 2004	Fruit in bowl still life	Smith Photography	Shipped	\$47.50		

1 - 20 of 35 orders [Next »](#)

① Search

② Find

③ Buy

For Buyers...

Convenience

- Google Checkout badge helps identify merchants to buy from
- One place to track purchases

Speed

- Single log in for easy buying across the web

Confidence

- Credit card number concealment
- Fraud protection
- Email forwarding

For Merchants...

More Leads

- Google Checkout badge helps attract more customers

More Conversions

- Streamlined checkout process increases conversions

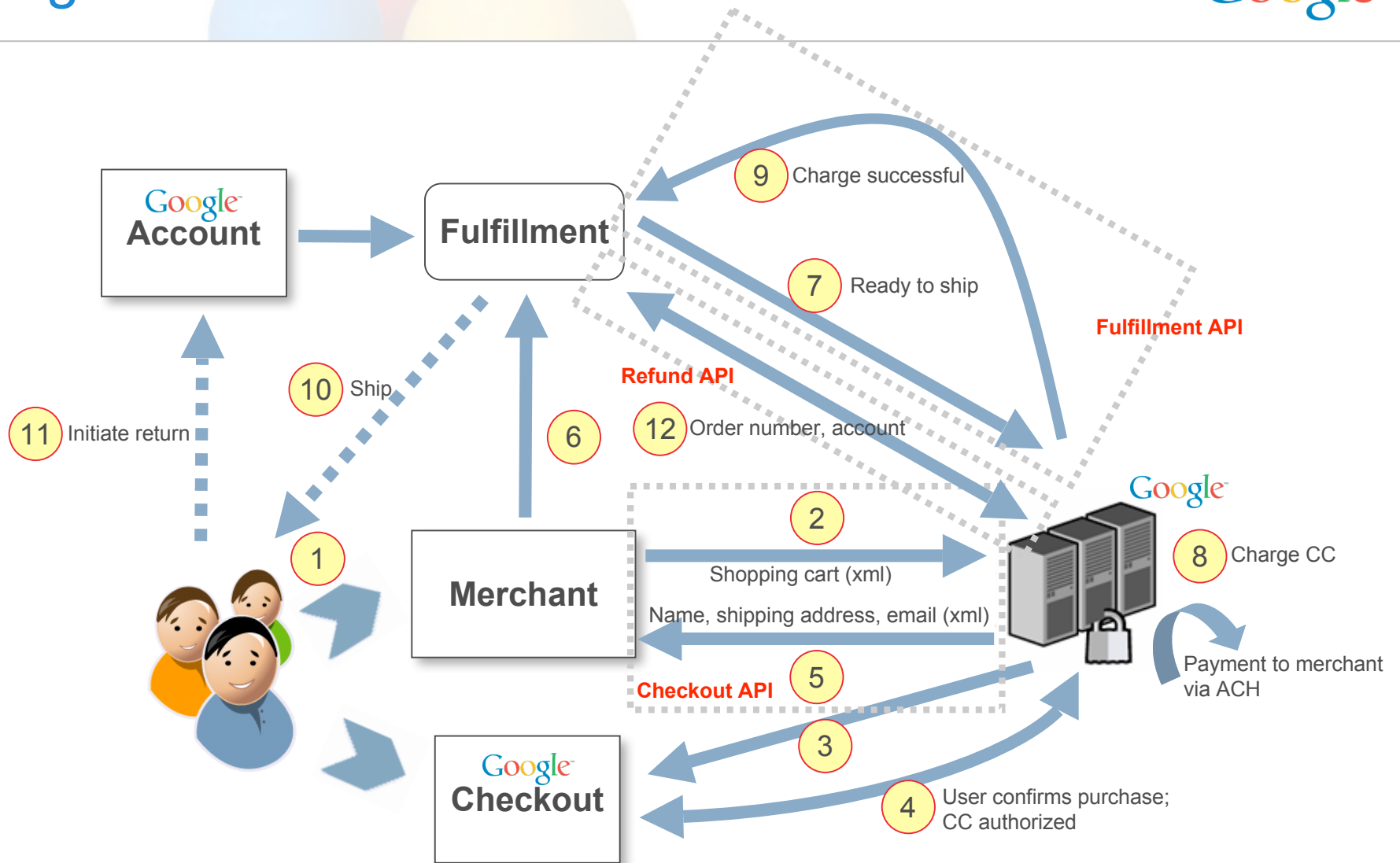
Lower Costs

- Free transaction processing for sales up to 10X AdWords spend
- Low 2% + \$0.20 per transaction fee for all other sales
- Fraud protection

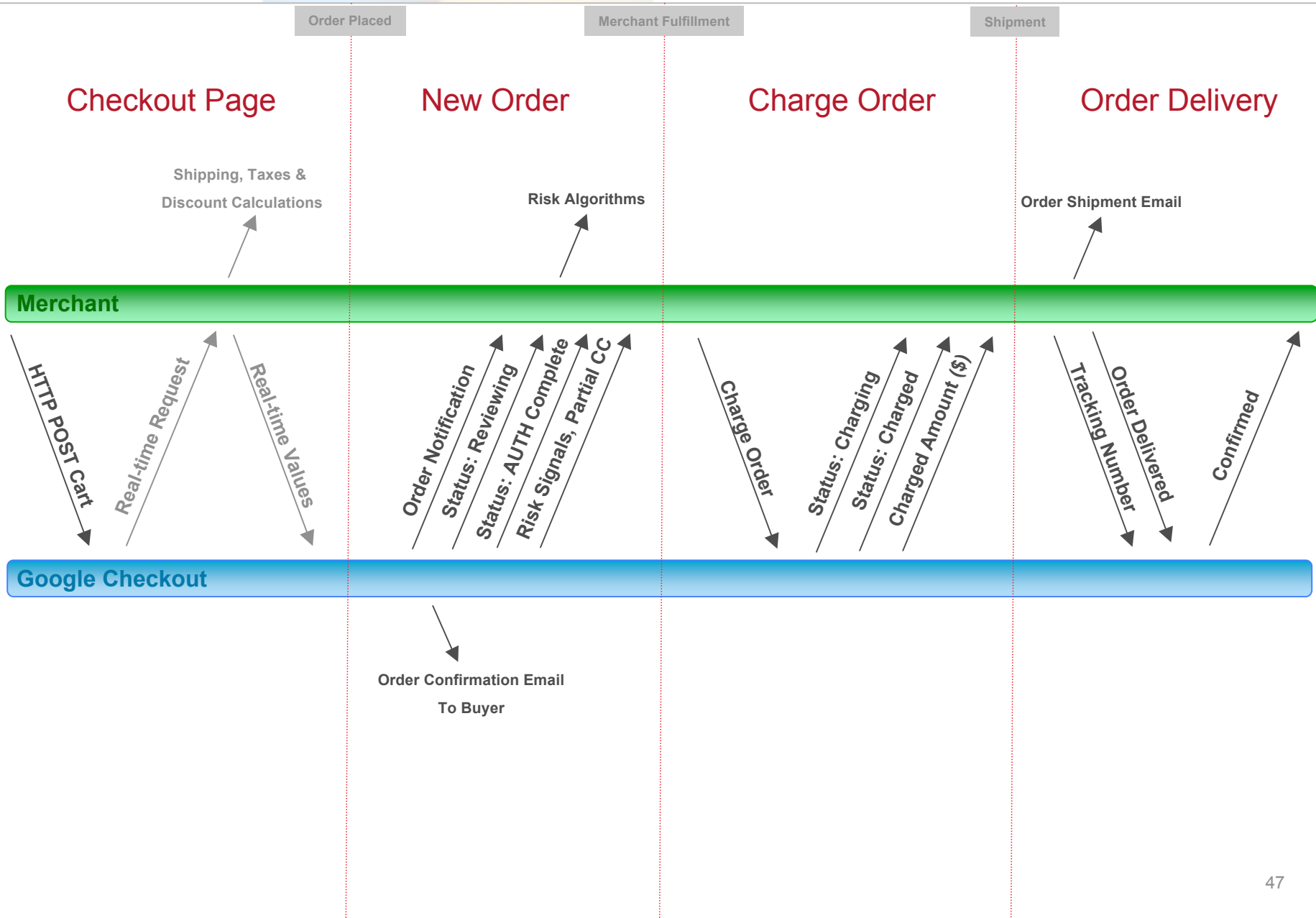
AdWords and Google Checkout work together to increase sales and lower costs.



Google Checkout Transaction Flow



Order Flows: Typical



Google Checkout Integration Options

- Buy Now buttons
- e-commerce partners
- Google Checkout API
- Systems Integrators specialized in Checkout

<http://checkout.google.com/seller/developers.html>

For Developers: bonuses for bringing Google Checkout to your merchants

https://services.google.com/inquiry/checkout_ecommerce

Google Checkout API

2 levels of integration

- Level 1: post shopping carts (taxes, shipping, coupons)
- Level 2: merchant calculations, order management, 2 ways API

REST API

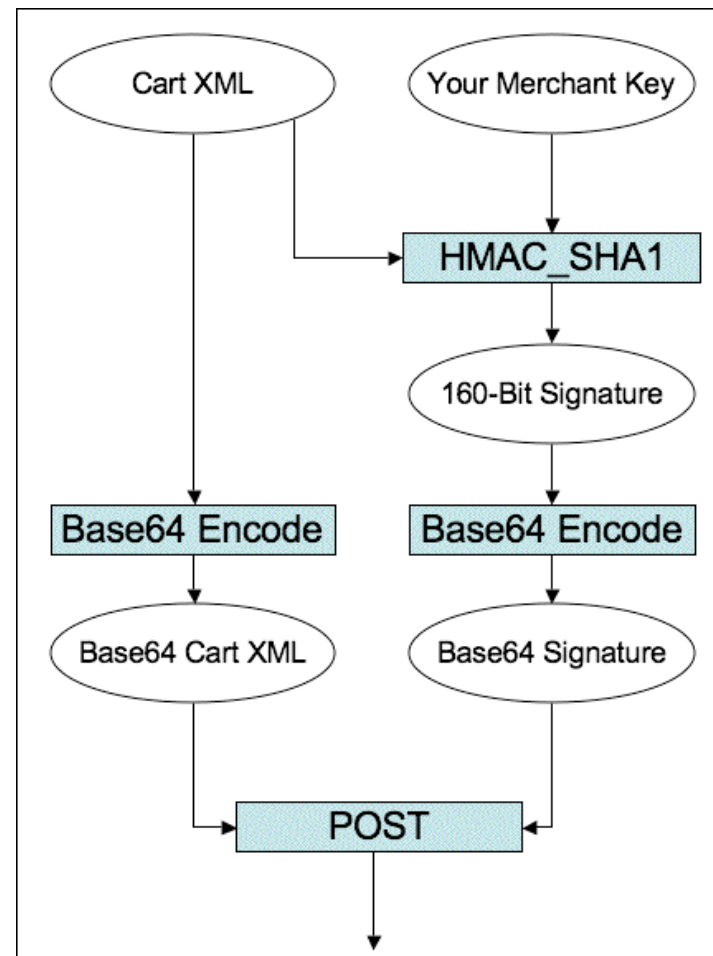
- XML over HTTP
- XML Schema
- Synchronous / Asynchronous (notifications)
- HMAC-SHA1 for signature, Basic auth over SSL
- Samples and Libraries in many environments
 - PHP, Java, Windows Classic ASP (COM), .NET
 - OsCommerce

Google Checkout API Level 1

Easy

Static Checkout Buttons

Post Cart XML + Signature



Samples at

- <http://code.google.com/p/google-checkout-php-sample-code/>

My PHP Google Checkout

- Byrne Reese, <http://code.google.com/p/myphpgooglecheckout/>
 - PHP5, basic server class for processing notifications, abstract classes, extensible, a MySQL implementation that logs and tracks payments in a database, a shopping cart framework for managing items in a cart for for composing your "Google Checkout" button. It is easy to extend, and is perfect for traditional OOP programmers.
 - does handle a basic purchase flow
 - “I wrote it to facilitate the purchasing and downloading of software - therefore I provide hooks in the process to verify a purchase and then send an email to the purchaser that contains a link to the downloadable Software”

Phpgcheckout, no code in it

OsCommerce, Zen Cart Integration

Old PHP Sample Code:

- Runs on **PHP4** only
- Requires **libxml**
- Problem: Hosted merchants can't install PHP extensions

New PHP Sample Code:

- Runs on both **PHP4** and **PHP5**
- Doesn't require **libxml**
- Fully **object oriented**
- **Open source** project - everyone is encouraged to modify, read or learn from the sources
- <http://code.google.com/p/google-checkout-php-sample-code>

Checkout API PHP Sample Code



Create a shopping cart XML and generate a button: cartdemo.php

```
//Create a new shopping cart object
$merchant_id = "912132643578668"; //Your Merchant ID
$merchant_key = "G1H1q4En076AerDy8dPzow"; //Your Merchant Key
$server_type = "sandbox";
$cart = new GoogleCart($merchant_id, $merchant_key, $server_type);

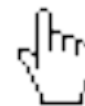
//Add items to the cart
$item1 = new GoogleItem("MegaSound 2GB MP3 Player",
    "Portable MP3 player - stores 500 songs", 1, 178);
$item2 = new GoogleItem("AA Rechargeable Battery Pack",
    "Battery pack containing four AA rechargeable batteries", 1 , 12);
$cart->AddItem($item1);
$cart->AddItem($item2);

//Add shipping options
$ship = new GoogleShipping("Ground", "flat-rate", 5);
$ship->SetAllowedCountryArea("CONTINENTAL_48");
$cart->AddShipping($ship);

$ship = new GoogleShipping("2nd Day", "flat-rate", 10);
$ship->SetAllowedCountryArea("FULL_50_STATES");
$cart->AddShipping($ship);

//Add tax options
$tax_rule = new GoogleTaxRule("default", 0.08);
$tax_rule->SetStateAreas( array("CA") );
$tax_table = new GoogleTaxTable("default");
$tax_table->AddTaxRules($tax_rule);
$cart->AddTaxTables($tax_table);

//Display Google Checkout button
echo $cart->CheckoutButtonCode("large");
```



- Main site: <http://checkout.google.com>
- Google Checkout Video / Demo:
<https://checkout.google.com/seller/demo.html>
- Places to Shop: <http://www.google.com/buy/m.html>
- Developers Center:
<https://checkout.google.com/seller/developers.html>
- Google Checkout API Center:
<http://code.google.com/apis/checkout/>
- Developer's Guide:
<http://code.google.com/apis/checkout/developer/index.html>

Demo

Fully Buzzword Compliant:-)

- PHP, SOAP, REST, Ajax, Business Mashup, Google

Sample: shows how to integrate AdWords and Checkout in your PHP application

Open Source soon

	+	-
SOAP	<p>Client toolkits: most code can be generated from WSDL</p> <p>Batch support: you batch inside of request</p>	<p>Doc/Lit support uneven</p>
REST	<p>url for resources</p> <p>Can use http libraries, no need SOAP client lib</p> <p>Caching</p> <p>Can load balance without parsing xml based on url components (like checkout API does)</p>	<p>Less code generation: xsd -> lang OK but still need to do http work</p> <p>No agreement on batch (cf Atom PaceBatch discussion)</p>



Questions...
