

APIs! Getcher APIs!!!



Speed, scale, agility



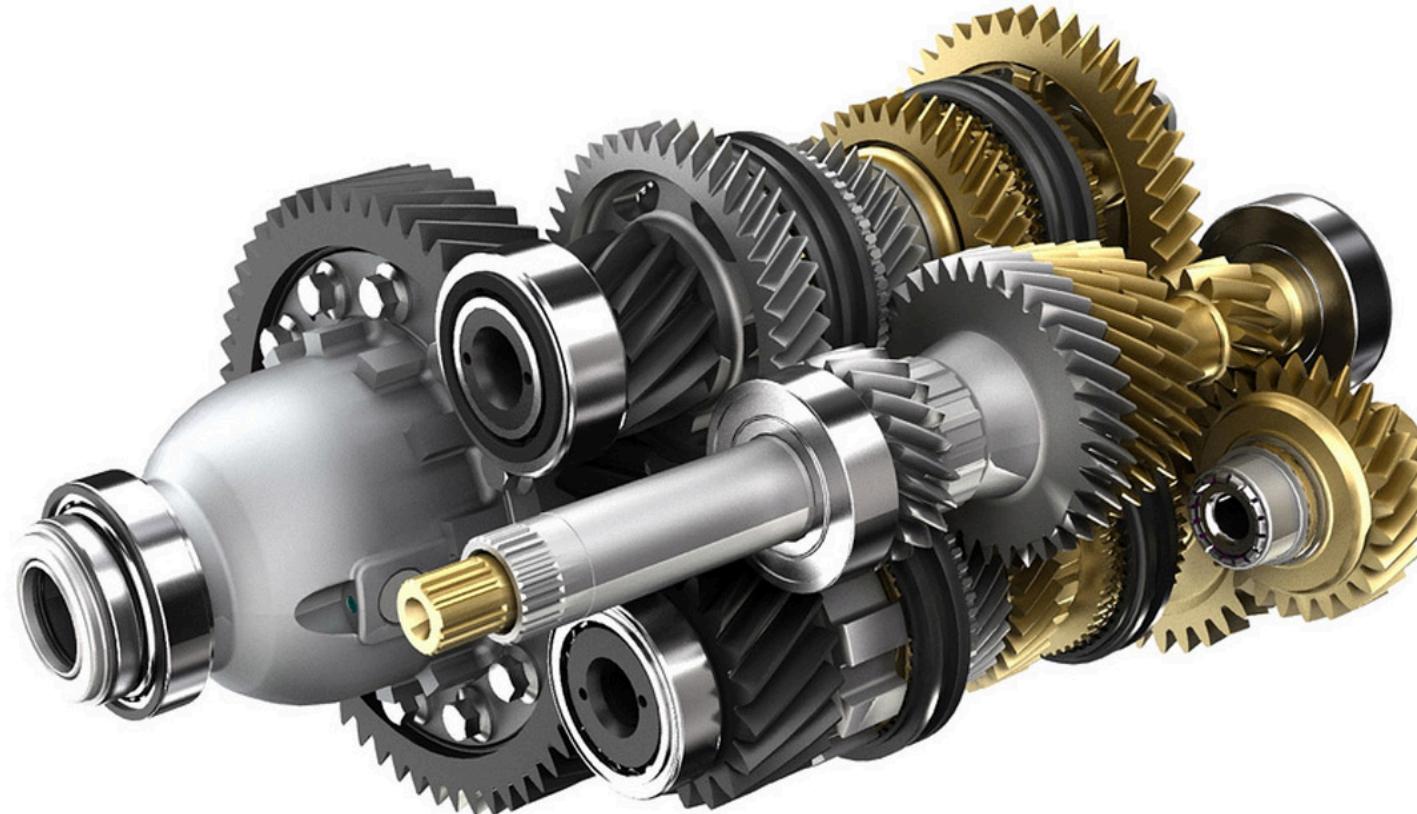
Secret sauce doesn't change often. Or fast. Or easily.



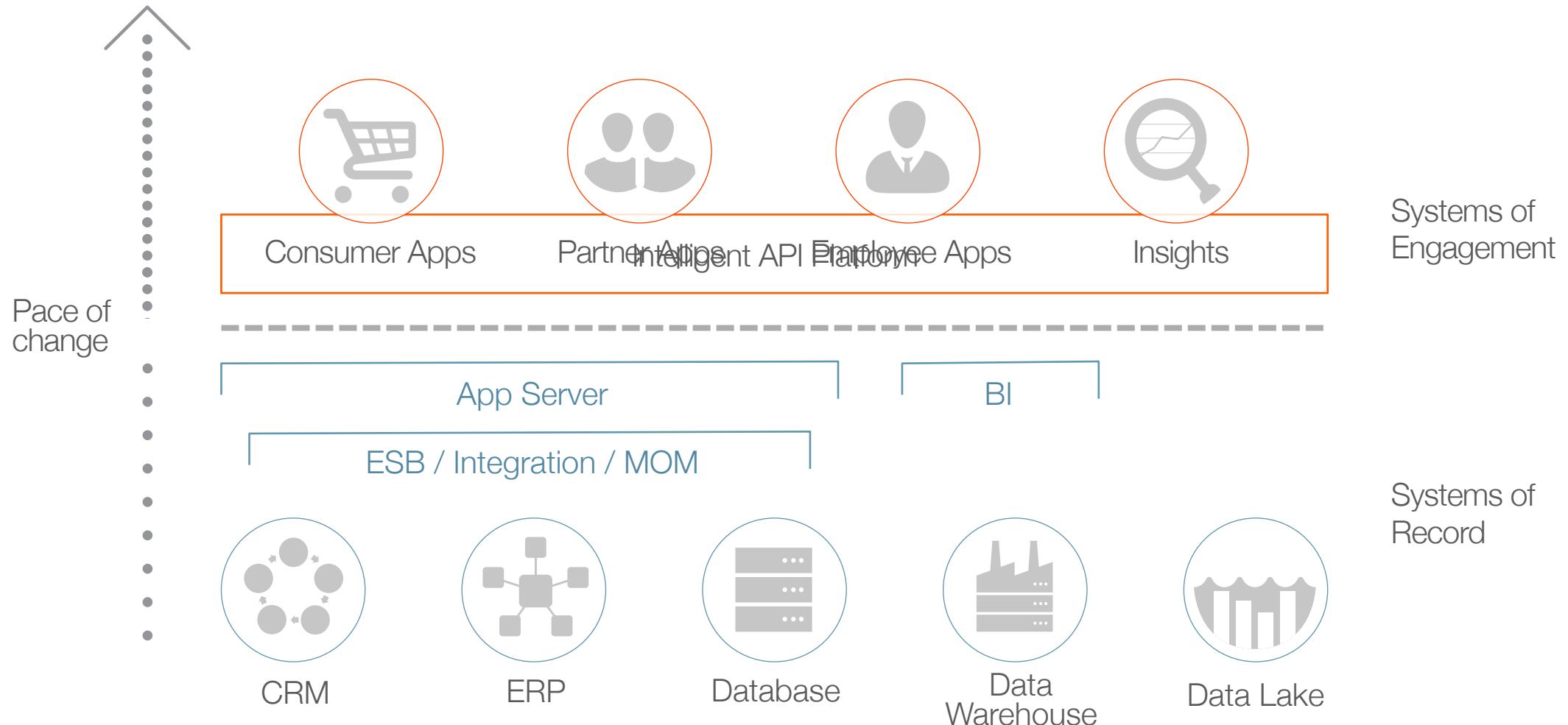
Your users & the world of technology move at a different pace



Need a way to manage the disparate expectations of the people
inside the vault vs those of your customers



Two-Speed IT rationalizes two IT realities



The Aspect-Oriented Enterprise

- One interface for all your secret sauce, no matter which channel
- Apply any security, logging, decisioning in the same place
- Full 360° view of the enterprise
- Change once, change everywhere

Developer Engagement

API Developer Program Success Requires:



Product: The foundation for API adoption

- Appeal: How ‘cool’ is your product?
 - Innovative
 - Useful
 - Using the latest technology
- You solve a developer problem
 - Developer considers your API because they think its less pain to come to you than solve it on their own
- The value of your API
- Treat APIs as products, don’t minimize the investment
- Get feedback early and iterate quickly
- Early and consistent vision, strategy, and metrics



[Home](#) [Profile](#) [Find People](#) [Settings](#) [Help](#) [Sign out](#)

Deep wisdom from [@raffi](#): Goals of the
[@twitterapi](#): 1) to be ridiculously simple,
2) to be obvious, 3) to be self-describing.
^MG

43 minutes ago via CoTweet

[Reply](#) [Retweet](#)



“

The real issue is about design: designing things that have the power required for the job while maintaining understandability, the feeling of control, and the pleasure of accomplishment.

-Donald Norman, academic



Source: <http://stylebizz.com/design-quotes/>

“

Always design a thing by considering it in its next larger context - a chair in a room, a room in a house, a house in an environment, an environment in a city plan.

-Eliel Saarinen, architect

HOW TO BUILD A MINIMUM VIABLE PRODUCT

NOT LIKE THIS



1

2



3



4

LIKE THIS



1



2



3

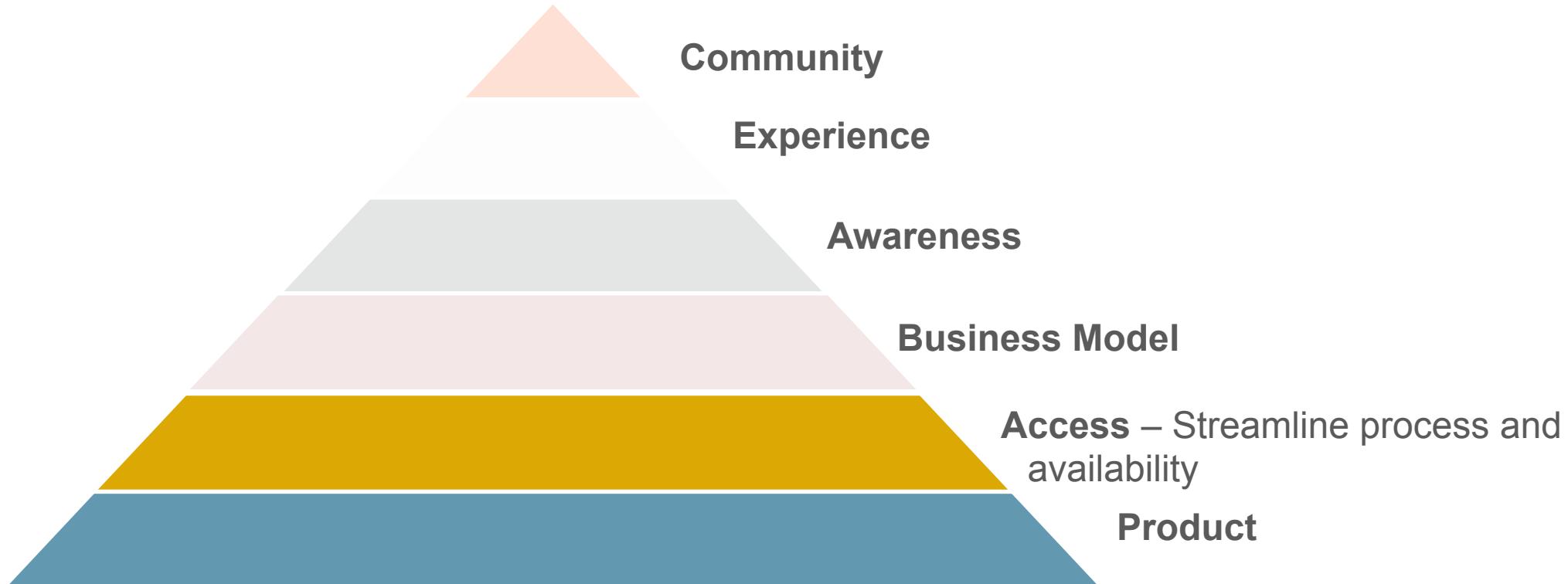


4



5

API Developer Program Success

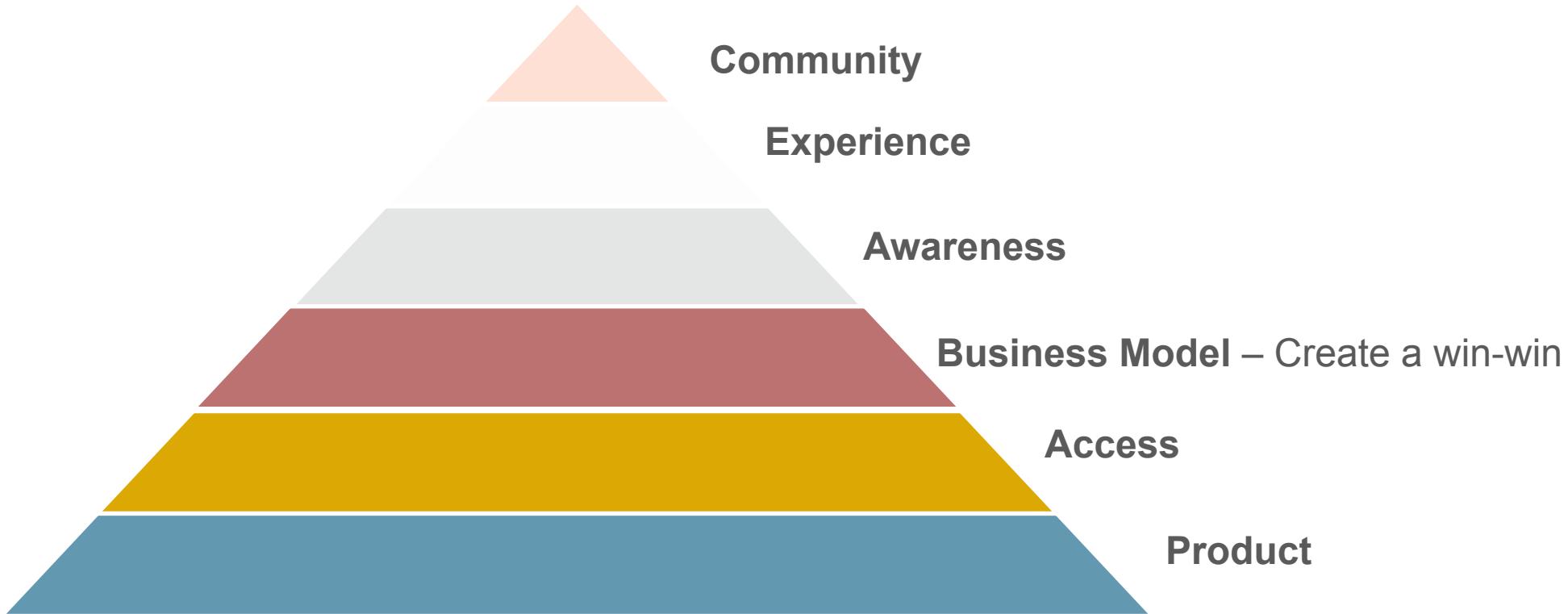


Access: Streamline process and availability

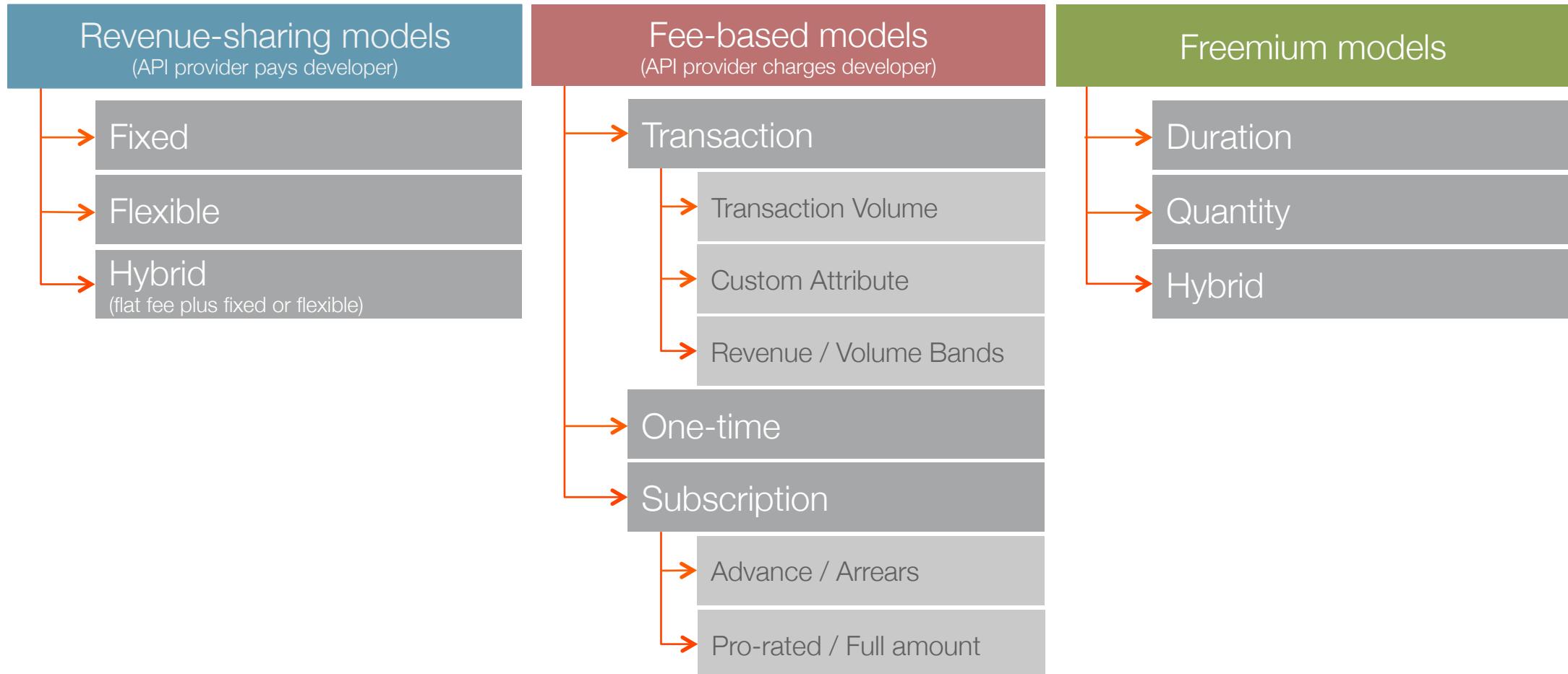
- Developers have a short attention span, if they can't self-service register they will go elsewhere
- The 'Time to First Response' is critical!
- 'Free' access encourages taking the next step
- Sample code makes it easy to experience success
- Example Apps show functionality in context
- 'Learn by trying' environment simplifies learning
- Available docs answer any corner cases
- Access to support eliminates obstacles



API Developer Program Success



Business models for API monetization

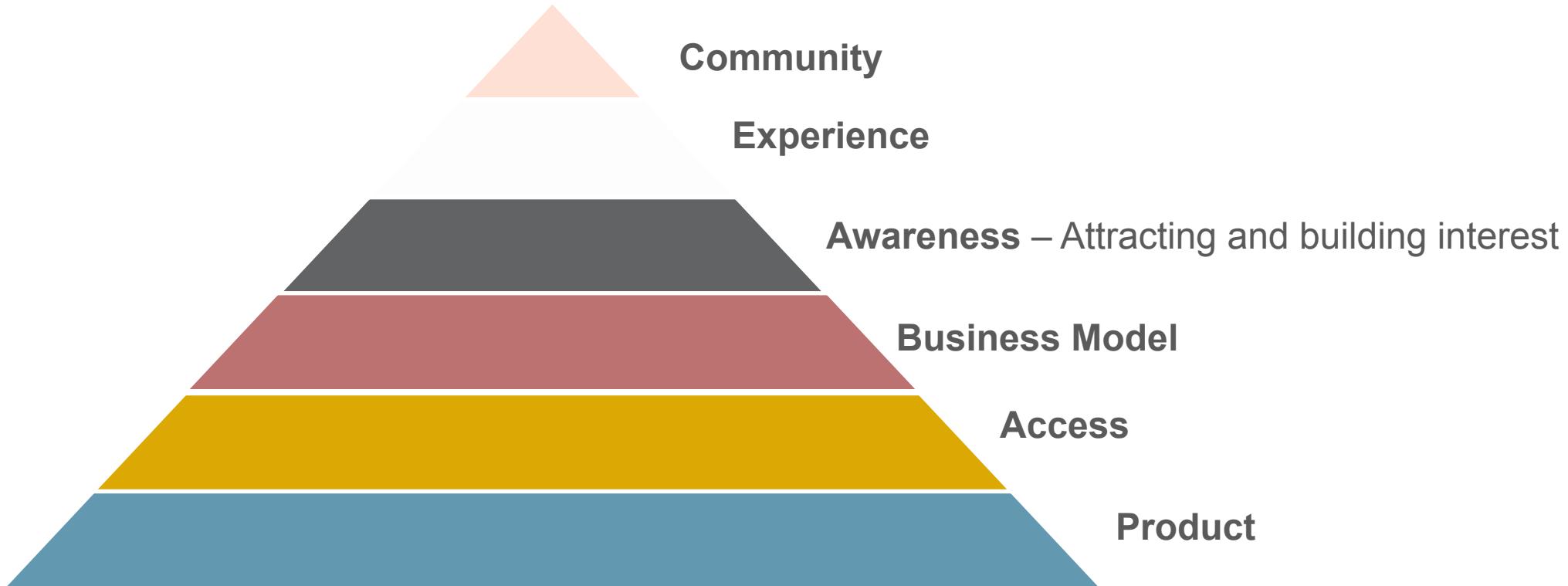


Business Model: best practices

- Your business model needs to be a win-win for you and your developers
- Give to get: Twilio gives you \$30 worth of text messages
- Business models evolve: Netflix changed to a partner model which drove client availability.
- Learn and adapt the business model to the opportunity



API Developer Program Success



Awareness

Awareness

Target the right type of developers and reach them through as many channels as possible

How to reach them?

API Type

Who uses the API?	App Types	Actions
Internal Developers (Internal API) 	B2E, S2S B2B, B2C,	Establish and promote a contest
Partner Developers (Partner API) 	B2C	Evangelize within your partners' development teams
Developers Anywhere (Open API) 	B2C	Publish your API and advertise on dev sites

Building Awareness

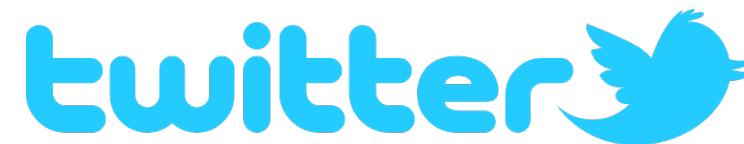
- Public Relations
- Social media
- Forums
- API/App contests
- Developer Days



HACKATHON



readwrite



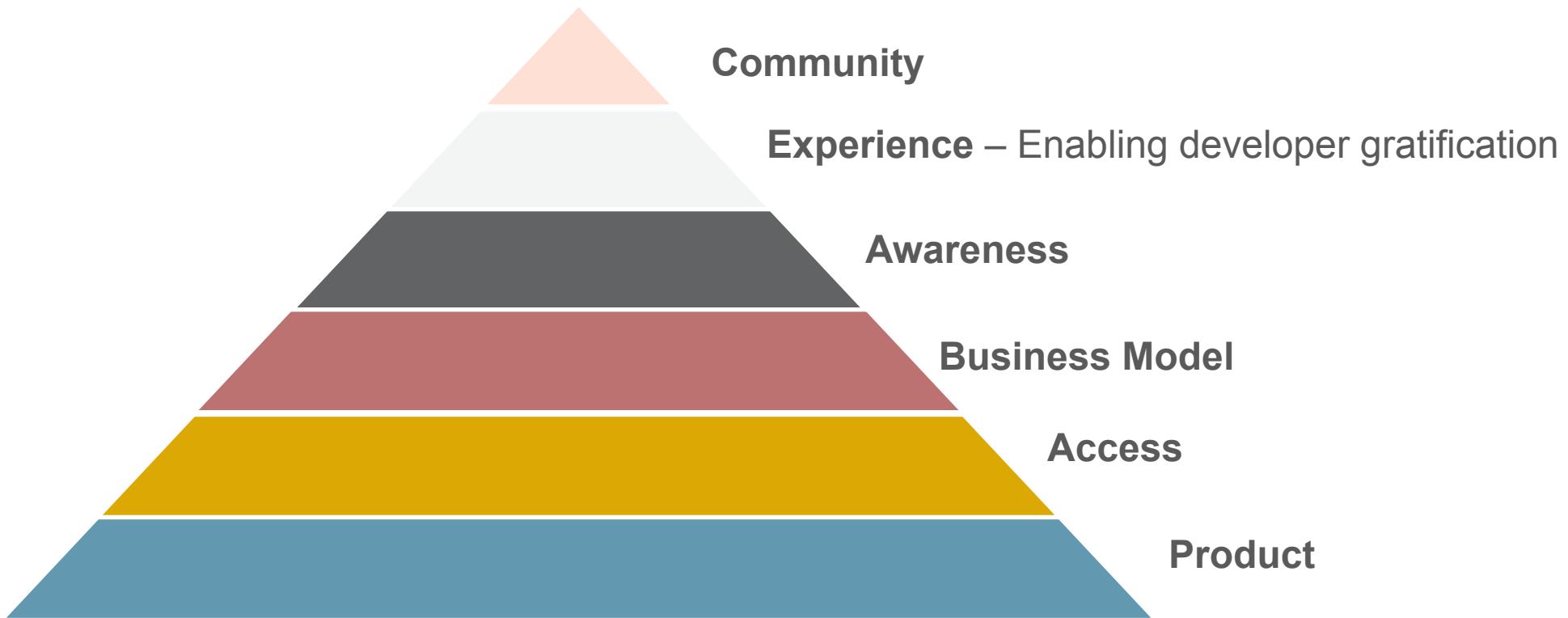
AUG
18

Money 20/20 Hackathon on Oct 24

The world's largest Fintech hackathon will be held in Las Vegas on Oct 24 2015 as part of the Money 20/20 conference. The Money 20/20 conference is organized once a year and is one of the most awaited event in the payments and financial tech space. For the conference this year, there will be 10,000+ attendees from 75 countries, 1000 CEO's, great speakers and a line-up of engaging events. More information is available in the website - <http://www.money2020.com/>

READ MORE

API Developer Program Success

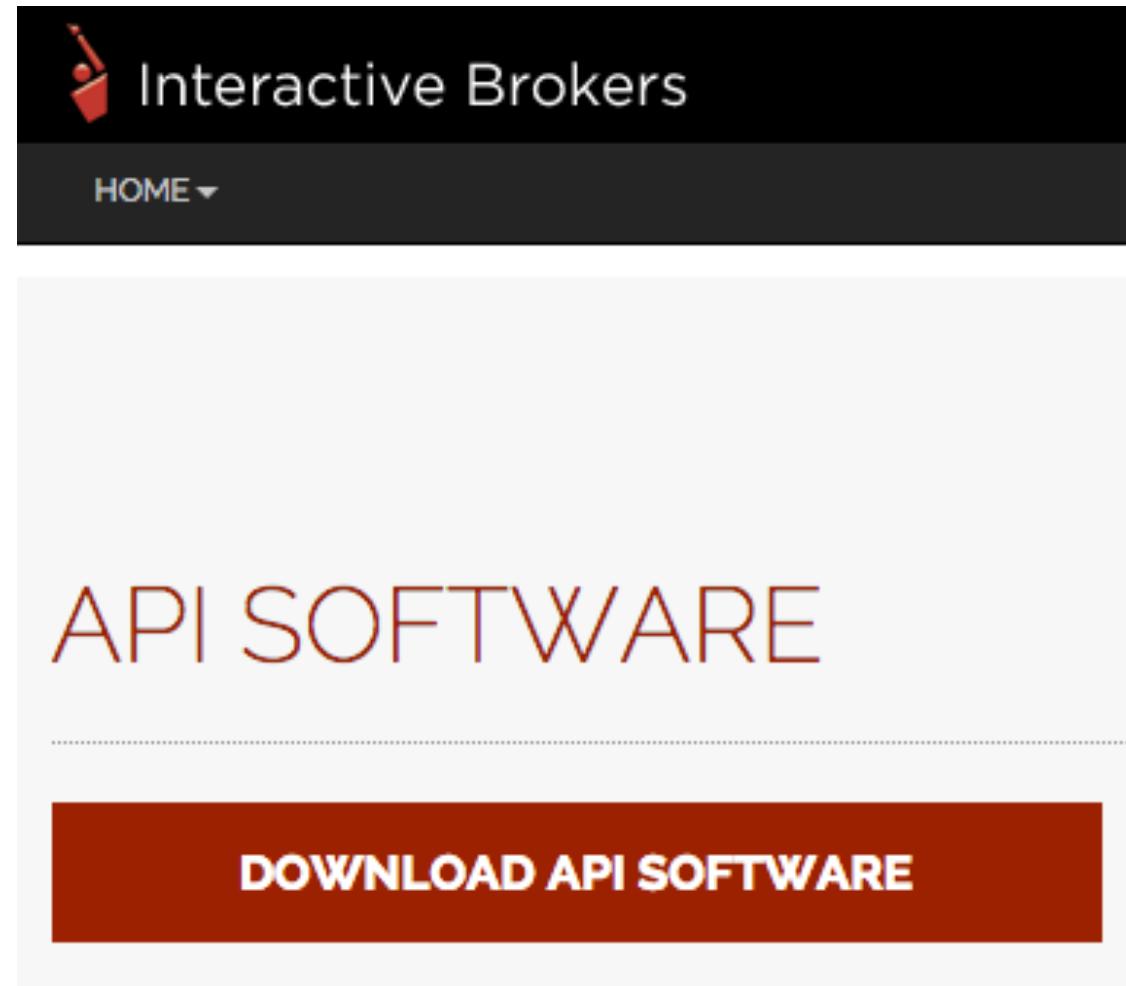


Experience

- The synergy of Access, Product, Community, and Awareness
 - Immersive, rich, and streamlined
 - Developer portal is the heart of developer experience
 - Should create confidence, inspiration, and solve their problems
-
- Easy onboarding/instant gratification
 - Wiki, forums, blogs, FAQs, support
 - SDKs can ease adoption



You do not download an API.



You do not install API software.

The screenshot shows a software application window with a title bar containing icons for Home, Overview, About the APIs, Installing the API Software (which is selected), Run the API through TWS, Run the API through the IB, Read-Only Mode, Recommendations, API Orders and TWS Precautions, API Order IDs, Trader Workstation API Settings, Uninstalling and Re-installing, DDE for Excel, ActiveX, C++, Java, Running the Sample Program, Running the Java Test Client, Java EClientSocket Methods, and Java EClientSocket Methods. The main content area has a heading 'Installing the API Software' and a sub-section 'To install the API software' with a numbered list of 7 steps.

Installing the API Software

To install the API software

1. Download the latest API software from the IB website:
2. From the IB website menu, click **Trading Technology > API Solutions**.
3. Click **IB API**, then click the **API Software** button.
4. In the popup window, read the license agreement then click **I Agree**.
5. Click the button corresponding to the Windows- or Mac/UNIX-based production or beta API version you want to install.
6. Save the installation file to your computer.
7. Run the downloaded installation program to install the API software on your computer.

Nobody wants to read a 600-page PDF.

The days of RTFM are over.

API Reference Guide

600

updateMktDepthL2()

This method is called when the Level II market depth changes.

```
void updateMktDepthL2(int tickerId, int position, String marketMaker, int operation, int side, double price, int size)
```

Parameter	Type	Description
tickerId	int	The ticker Id that was specified previously in the call to reqMktDepth()
position	int	Specifies the row id of this market depth entry.
marketMaker	String	Specifies the exchange hosting this order.
operation	int	Identifies the how this order should be applied to the market depth. Valid values are: <ul style="list-style-type: none"> • 0 = insert (insert this new order into the row identified by 'position') • 1 = update (update the existing order in the row identified by 'position') • 2 = delete (delete the existing order at the row identified by 'position')
side	int	Identifies the side of the book that this order belongs to. Valid values are: <ul style="list-style-type: none"> • 0 = ask • 1 = bid
price	double	The order price.
size	int	The order size.

updateNewsBulletin()

This method is triggered for each new bulletin if the client has subscribed (i.e. by calling the reqNewsBulletins() method).

```
void updateNewsBulletin(int msgId, int msgType, String message, String origExchange)
```

Parameter	Type	Description
msgId	int	The bulletin ID, incrementing for each new bulletin.
msgType	int	Specifies the type of bulletin. Valid values include: <ul style="list-style-type: none"> • 1 = Regular news bulletin • 2 = Exchange no longer available for trading • 3 = Exchange is available for trading
message	String	The bulletin's message text.
origExchange	String	The exchange from which this message originated.

Create a new order

POST /v1/accounts/:account_id/orders

Input Data Parameters (inside body)

instrument:* *Required* Instrument to open the order on.

units: *Required* The number of units to open order for.

side: *Required* Direction of the order, either 'buy' or 'sell'.

type: *Required* The type of the order 'limit', 'stop', 'marketIfTouched' or 'market'.

expiry: *Required* If order type is 'limit', 'stop', or 'marketIfTouched'. The order expiration time in UTC. The value specified must be in a valid datetime format.

price: *Required* If order type is 'limit', 'stop', or 'marketIfTouched'. The price where the order is set to trigger at.

lowerBound: *Optional* The minimum execution price.

upperBound: *Optional* The maximum execution price.

stopLoss: *Optional* The stop loss price.

takeProfit: *Optional* The take profit price.

trailingStop: *Optional* The trailing stop distance in pips, up to one decimal place.

Example ('market' order)

```
$curl -X POST -d "instrument=EUR_USD&unit=s=2&side=sell&type=market" "https://api-fxtrade.oanda.com/v1/accounts/12345/orders"
```

Response

Header

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: 204
```

Body

```
{
  "instrument": "EUR_USD",
  "time": "2013-12-06T20:36:06Z", // Time that order was executed
  "price": 1.37041, // Trigger price of the order
  "tradeOpened": {
    "id": 175517237, // Order id
    "units": 2, // Number of units
    "side": "sell", // Direction of the order
    "takeProfit": 0, // The take-profit associated with the order, if any
    "stopLoss": 0, // The stop-loss associated with the order, if any
    "trailingStop": 0 // The trailing stop associated with the order, if any
  },
  "tradesClosed": [],
  "tradeReduced": []
}
```

API

ACTIVITY SERVICES

Activity Detail

- List Activities
- GPS Data
- List Activities By Experience Type
- Add Activities
- Aggregate Sport Data

ERROR MESSAGES +

REFERENCE +

TEST CONSOLE >

USER CONNECTIONS +

WEBHOOKS +

SDK

IOS SDK -

Quickstart Guide

ACTIVITY DETAIL

GET

The Activity Detail (**GET v1/me/sport/activities/{activityId}**) endpoint provides details of the user's Activities. Each Activity has a unique Activity ID. Activity details include the Activity's type, start time, and duration, the NikeFuel earned and calories burned during the Activity, and the type of device that recorded the Activity.

The Metric Type values returned by the Activity Detail endpoint will vary depending on the Activity. Devices and apps can use these Metric Type values (e.g., **fuel**) to display information to enhance the user's experience.

Note: Use the ALL_DAY Activity Type to receive NikeFuel from multiple or overlapping Activities. See NikeFuel Across Activities for more information.

Note: This endpoint does not return GPS data. Use the GPS Data endpoint to retrieve GPS data one Activity at a time.

TEST CONSOLE

Select an Endpoint

/v1/me/sport/activities/{activityId}

activityId

LOG IN



Apple Pay Android Pay

GET CODING NOW



Smart payments in your own language.

Build your business with Payeezy, the simplest way to accept transactions online.

GET CODING NOW

MERCHANTS

Docs & Sandbox

Availability of certain payment methods depend on merchant's country of domicile. [Click here](#) for details.

Make Payments

METHOD

DESCRIPTION

POST

Credit Card Payments[/transactions](#)

Use this method to submit payments credit and debit cards. Supported transaction types are 'authorize' & 'purchase'.

POST

PayPal Transactions[/transactions](#)

Use this method for PayPal based transactions. Supported transaction types are 'authorize', 'purchase' & 'order'.

POST

Apple Pay™ Transactions[/transactions](#)

Use this method to do transactions using cryptogram which is supported by

POST

Credit Card Payments

Use this method to submit payments credit and debit cards. Supported transaction types are 'authorize' & 'purchase'.

Resource URL

Header Parameters

Try it out !

Authorize

Purchase

Split-Tender

Authorize

Sample Authorization payload for a credit card transaction is shown below.

```

1 { "merchant_ref": "Astonishing-Sale",
2   "transaction_type": "authorize",
3   "method": "credit_card",
4   "amount": "1299",
5   "currency_code": "USD",
6   "credit_card": {
7     "type": "visa",
8     "cardholder_name": "John Smith",
9     "card_number": "4788250000028291",
10    "exp_date": "1020",
11    "cvv": "123"
12 }
```

Resource Summary

Category Make Payments

Note: The below request payload shows "billing_address", "soft_descriptors", "level2" & "level3" objects which are optional.

["Click here"](#) to know more about how to make use of these optional objects in your request.

Request

Response

Errors

Test Cards

```
{
  "merchant_ref": "{string}",
  "transaction_type": "{string}",
  "method": "{string}",
  "amount": "{string}",
  "partial_redemption": "{string}",
  "split_tender_id": "{string}",
  "currency_code": "{string}",
  "credit_card": {
    "type": "{string}",
    "cardholder_name": "{string}",
    "card_number": "{string}",
    "exp_date": "{string}",
    "soft_descriptors": {
      "level2": "{string}",
      "level3": "{string}"
    }
  }
}
```

Property name	Type	Description
---------------	------	-------------

merchant_ref (optional)	string	Merchant reference code – used by Payeezy system will be reflected in your settlement records and webhook notifications. It is an "optional" field
----------------------------	--------	--

POST

Credit Card Payments

Use this method to submit payments credit and debit cards. Supported transaction types are 'authorize' & 'purchase'.

Resource URI

Environment	URL
Sandbox	https://api-cert.payeezy.com/v1/transactions ✓
Live	https://api.payeezy.com/v1/transactions

Header Parameters

Try it out !

Authorize **Purchase** **Split-Tender**

Authorize
Sample Authorization payload for a credit card transaction is shown below.

```

1 {
2   "merchant_ref": "Astonishing-Sale",
3   "transaction_type": "authorize",
4   "method": "credit_card",
5   "amount": "1299",
6   "currency_code": "USD",
7   "credit_card": {
8     "type": "visa",
9     "cardholder_name": "John Smith",
10    "card_number": "4788250000028291",
11    "exp_date": "1020",
12    "cvv": "123"
13  }
14}

```

Resource Summary

Category Make Payments

Note: The below request payload shows "billing_address", "soft_descriptors", "level2" & "level3" objects which are optional.

"Click here" to know more about how to make use of these optional objects in your request.

Request Response Errors Test Cards

```
{
  "merchant_ref": "{string}",
  "transaction_type": "{string}",
  "method": "{string}",
  "amount": "{string}",
  "partial_redemption": "{string}",
  "split_tender_id": "{string}",
  "currency_code": "{string}",
  "credit_card": {
    "type": "{string}",
    "cardholder_name": "{string}",
    "card_number": "{string}",
    "exp_date": "{string}"
  }
}
```

Property name	Type	Description
merchant_ref (optional)	string	Merchant reference code – used by Payeezy system will be reflected in your settlement records and webhook notifications. It is an "optional" field
method (required)	string	Inputted transaction method. Ex:"credit_card"
transaction_type (required)	string	Type of transaction that merchant would want to process. values = authorize/purchase
amount (required)	string	Processed Amount in cents. 1299 = \$12.99
currency_code (required)	string	ISO 4217 currency span Ex: USD
credit_card (required)	object	JSON object that holds the credit/debit card info used to process the transaction.

Try it out !



Authorize

Purchase

Split-Tender

Authorize

Sample Authorization payload for a credit card transaction is shown below.

```
1 {  
2   "merchant_ref": "Astonishing-Sale",  
3   "transaction_type": "authorize",  
4   "method": "credit_card",  
5   "amount": "1299",  
6   "currency_code": "USD",  
7   "credit_card": {  
8     "type": "visa",  
9     "cardholder_name": "John Smith",  
10    "card_number": "4788250000028291",  
11    "exp_date": "1020",  
12    "cvv": "123"  
13  }  
14 }
```

SEND

RESET

SEND

RESET

Request

Response

HTTP/1.1 201 Created

Access-Control-Allow-Headers: Content-Type
Access-Control-Allow-Methods: GET
Access-Control-Allow-Origin: http://localhost:8080
Access-Control-Max-Age: 3628800
Access-Control-Request-Headers: origin
Connection: keep-alive
Content-Language: en-US
Content-Length: 536
Content-Type: application/json; charset=UTF-8
Date: Thu, 17 Sep 2015 15:51:15 GMT
OPTR_CXT: 01000100000b86a6cb-fb5c-45bc-9a6f-9cd28da340a20000000-0000-0000-000000000000-1
HTTP ;
Server: Apigee Router
X-Archived-Client-IP: 10.180.205.250
X-Backside-Transport: OK OK
X-Client-IP: 10.180.205.250
X-Powered-By: Servlet/3.0

```
{  
    "correlation_id": "228.1442505074821",  
    "transaction_status": "approved",  
    "validation_status": "success",  
    "transaction_type": "authorize",  
    "transaction_id": "ET103988",  
    "transaction_tag": "62860604",  
    "method": "credit_card",  
    "amount": "1299",  
    "currency": "USD",  
    "cvv2": "M",  
    "token": {  
        "token_type": "FDToken",  
        "token_data": {  
            "value": "9831481087258291"  
        }  
    },  
    "card": {  
        "type": "visa",  
        "cardholder_name": "John Smith",  
        "card_number": "8291",  
        "exp_date": "1020"  
    },  
}
```

SEND

RESET

Request

Response

POST /v1/transactions HTTP/1.1

```
Accept: */*
Accept-Encoding: gzip
Accept-Language: en-US
apikey: y6pWAJNyJyjGv66lsVuWnkIkKUPFbb0a
Authorization:
NmE3NjNIMWE5MjRkYjA4MDJmOWUwYmVkJNTIiNjk0NTY2NWJh
YjlzMTExM2M0OGJIZjhjOTJkMDY0N2lwYTImNQ==
Content-Length: 309
Content-Type: application/json
Host: api-cert.payeezy.com
nonce: 6838818385265767000
timestamp: 1442505818049
token: fdoa-
a480ce8951daa73262734cf102641994c1e55e7cdf4c02b6
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_9_5)
AppleWebKit/537.36 (KHTML, like Gecko)
X-Forwarded-For: 66.83.254.34
X-Forwarded-Port: 443
X-Forwarded-Proto: https
```

```
{
  "merchant_ref": "Astonishing-Sale",
  "transaction_type": "authorize",
  "method": "credit_card",
  "amount": "1299",
  "currency_code": "USD",
  "credit_card": {
    "type": "visa",
    "cardholder_name": "John Smith",
    "card_number": "4788250000028291",
    "exp_date": "1020",
    "cvv": "123"
  }
}
```

Header Parameters

Name	Value	Description
apikey (required)	y6pWAJNyJyjGv66lsVt	This is used to identify the app.
token (required)	fdoa-a480ce8951daa7	token required for authorization
Content-type (required)	application/json	content type the payload requires
Authorization (required)	ZDM5OWRjZmViYWUx	HMAC of payload signed by consumer secret
	GENERATE HMAC <small>i</small>	
nonce (required)	423809991357848060	Secure random number
timestamp (required)	1442505023533	Epoch time stamp

Header Parameters

Name	Value
apikey (required)	y6pWAJN
token (required)	fdoa-a48
Content-type (required)	application
Authorization (required)	ZDM5OW
nonce (required)	4238099

GENERATE HMAC

Construct the data param by appending the parameters below in the same order as shown. a. apikey - API key of the developer. b. nonce - secure random number. c. timestamp - epoch timestamp in milliseconds. d. token - Merchant Token. e. payload - Actual body content passed as post request. *Compute HMAC SHA256 hash on the above data param using the key below* f. apiSecret - Consumer Secret token for the given api key *Calculate the base64 of the hash which would be our required Authorization header value.*

— Payeezy Developer Guide

Objective-C

PHP

Python

```
// import required header files
#import <CommonCrypto/CommonDigest.h>;
#import <CommonCrypto/CommonHMAC.h>

NSString* apiKey = @<your api key>;
NSString* apiSecret = @<your consumer secret>;
NSString* token = @<Merchant Token>;
NSString* payload = @<For POST - Request body / For GET - empty string>;
NSString* nonce = @<Cryptographically strong random number>;
NSString* timestamp = @<Epoch timestamp in milli seconds>;
NSString *hmacData =
[NSString stringWithFormat:@"%@%@", apiKey, nonce, timestamp, token, payload];
```

CLOSE

Resource Summary

Category Make Payments

Note: The below request payload shows "billing_address", "soft_descriptors", "level2" & "level3" objects which are optional.

"Click here" to know more about how to make use of these optional objects in your request.

Request	Response	Errors	Test Cards
<pre>{ "merchant_ref": "{string}", "transaction_type": "{string}", "method": "{string}", "amount": "{string}", "partial_redemption": "{string}", "split_tender_id": "{string}", "currency_code": "{string}", "credit_card": { "type": "{string}", "cardholder_name": "{string}", "card_number": "{string}", "exp_date": "{string}" } }</pre>			
Property name	Type	Description	
merchant_ref (optional)	string	Merchant reference code – used by Payeezy system will be reflected in your settlement records and webhook notifications. It is an optional field	
method (required)	string	Inputted transaction method. Ex:"credit_card"	
transaction_type (required)	string	Type of transaction that merchant would want to process. values = authorize/purchase	
amount (required)	string	Processed Amount in cents. 1299 = \$12.99	
currency_code (required)	string	ISO 4217 currency span Ex: USD	
credit_card (required)	object	JSON object that holds the credit/debit card info used to process the transaction.	
credit_card.type	string	values - "American Express", "Visa",	

OPTIONAL AVS, SOFT DESCRIPTORS, LEVEL2 & LEVEL3 PROPERTIES

X

Address object for AVS Check

POST

Credit Card

Use this method to submit payment requests. Transaction types are 'authorization' or 'capture'.

Resource URL

Header Parameters

Name	Value
------	-------

apikey (required)	y6pWAJN
----------------------	---------

token (required)	61...420...22511...7
---------------------	----------------------

```
"billing_address": {  
    "street": "{string}",  
    "city": "{string}",  
    "state_province": "{string}",  
    "country": "{string}",  
    "zip_postal_code": "{string}",  
    "email": "{string}",  
    "phone": {  
        "type": "{string}",  
        "number": "{string}"  
    }  
}
```

CLOSE

token required for
authorization

"amount": "{string}",

and shows "billing_address",
& "level3" objects which are

to make use of these optional

Errors

Test Cards

en object" fields only if the

I Token based services with

Response would contain "avs & token object" fields only if the merchant has enabled AVS check and Token based services with Payeezy.

```
{
  "method": "{string}",
  "amount": "{string}",
  "currency": "{string}",
  "avs": "{string}",
  "cvv2": "{string}",
  "card": {
    "type": "{string}",
    "cardholder_name": "{string}",
    "card_number": "{string}",
    "exp_date": "{string}"
  },
  "token": {
    "token_type": "{string}",
    "token_data": {
      "value": "{string}"
    }
  },
  "transaction_status": "{string}",
  "validation_status": "{string}",
  "transaction_type": "{string}",
  "transaction_id": "{string}",
  "transaction_tag": "{string}",
  "bank_resp_code": "{string}",
  "bank_message": "{string}",
  "gateway_resp_code": "{string}",
  "gateway_message": "{string}",
  "correlation_id": "{string}"
}
```

Property name	Type	Description
method	string	Inputted transaction method. Ex:"3DS" / "Credit_Card" / "EMV"
amount	string	Processed Amount in cents. 1299 = \$12.99
currency	string	ISO 4217 currency span Ex: USD
avs	string	Address Verification System response returned from the gateway. Click here to see the full list of response codes.
cvv2	string	Card Verification Value (CVV2)

Sample Error Messages & Error Codes.

```
{  
    "method": "{\'credit_card\' or \'3DS\' or \'EM  
V\'}",  
    "amount": "{string ex:12.99 = 1299}",  
    "currency": "{string}",  
    "card": {  
        "type": "visa",  
        "cvv": "{string}",  
        "cardholder_name": "{string}",  
        "card_number": "{string}",  
        "exp_date": "{string}"  
    },  
    "transaction_status": "Not Processed",  
    "validation_status": "failed",  
    "transaction_type": "authorize",  
    "Error": {
```

Property	Type	Description
messages	array	Contains an array of JSON objects with error codes and description as shown above

HTTP Status Codes	API Error Codes
-------------------	-----------------

Status Code	Description
200,201,202	Created / Accepted Transaction request - All OK.
400 Bad Request	Input Request is invalid or incorrect.
401 Unauthorized	Invalid API Key and Token.
404 Not Found	The requested resource does not exist.
500, 502, 503, 504 Server errors	Server Error at Payeezy end.

Request	Response	Errors	Test Cards																		
<p>Expiry Date: Any future date.</p> <p>Cvv:Any 3 digit number for Visa, Mastercard & Discover and 4 digit number for American Express</p>																					
<table border="1"><thead><tr><th>Type</th><th>Card Number</th></tr></thead><tbody><tr><td>Visa</td><td>4012 0000 3333 0026</td></tr><tr><td></td><td>4005 5192 0000 0004</td></tr><tr><td>MasterCard</td><td>5424 1802 7979 1732</td></tr><tr><td></td><td>5526 3990 0064 8568</td></tr><tr><td></td><td>5405 0101 0000 0016</td></tr><tr><td>American Express</td><td>3739 5319 2351 004</td></tr><tr><td></td><td>3411 1159 7241 002</td></tr><tr><td>Discover</td><td>6510 0000 0000 1248</td></tr></tbody></table>				Type	Card Number	Visa	4012 0000 3333 0026		4005 5192 0000 0004	MasterCard	5424 1802 7979 1732		5526 3990 0064 8568		5405 0101 0000 0016	American Express	3739 5319 2351 004		3411 1159 7241 002	Discover	6510 0000 0000 1248
Type	Card Number																				
Visa	4012 0000 3333 0026																				
	4005 5192 0000 0004																				
MasterCard	5424 1802 7979 1732																				
	5526 3990 0064 8568																				
	5405 0101 0000 0016																				
American Express	3739 5319 2351 004																				
	3411 1159 7241 002																				
Discover	6510 0000 0000 1248																				

View Guides



- **Apple Pay™**
- **iOS**
- **Payeezy JS**
- **Direct API**
- **Android**

 payeezy / payeezy_direct_API Watch ▾ 16  Star 10  Fork 17

Payeezy bindings for PHP, Ruby, nodeJS, Python and Curl

33 commits

1 branch

0 releases

5 contributors



Branch: master ➔ payeezy_direct_API / +

Code

Issues 0

Pull requests 0

Wiki

Pulse

Graphs

HTTPS clone URL

<https://github.com/> You can clone with [HTTPS](#), [SSH](#), or [Subversion](#).  Clone in Desktop Download ZIP

Update api_methods.py

 nohup-atulparmar authored 2 days agolatest commit bbf398ce5c 

 guide	update in code and documents	3 months ago
 payeezy_curl	Update README.md	3 months ago
 payeezy_java	July 1 release code and documentation	3 months ago
 payeezy_node	Renamed getToken function appropriately.	2 months ago
 payeezy_php	Create payeezyCSS.css	13 days ago
 payeezy_python	Update api_methods.py	2 days ago
 payeezy_ruby	July 1 release code and documentation	3 months ago
 README.md	update in code and documents	3 months ago

 README.md

Payeezy Direct API

User Action: Buyer does a checkout on the merchant hosted site by clicking on buy/submit button.

This needs to be a secure page.

1. Upon Submit the form gets posted to the Merchant server
2. Merchant's backend server invokes the Payeezy API over a SSL connection and submits the order information securely. The call requires providing an authorization header with a HMAC hash. Refer to API docs section for more information on this.
3. Payeezy process the transactions (such as authorize, purchase) with the provided transaction details and sends back the response
4. Success/failure message needs to be handled accordingly by the merchants server.

A welcome message with f

POST Credit Ca

Use this method to submit pay
transaction types are 'authoriz

Resource URL

Header Parameters

Try it out !

Authorize

Authorize

Sample Authorization payload

```
1 {  
2   "merchant_ref": "  
3   "transaction_type":  
4   "method": "credit"  
5   "amount": "1299",  
6   "currency_code":  
7   "credit_card": {  
8     "type": "visa",  
9     "cardholder_name":  
10    "card_number":  
11    "exp_date": "10/  
12    "cvv": "123"  
13  }  
14 }
```



SIGN UP TO TEST DRIVE NOW

First Name

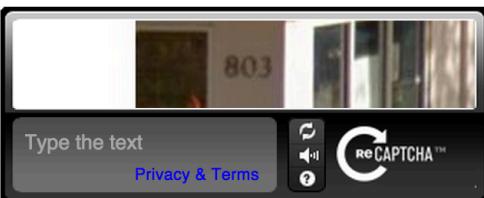
Last Name

E-mail address

A valid e-mail address. Once registered cannot be changed in future. All e-mails from the system will be sent to this address. The e-mail address is not made public and will only be used if you wish to receive notifications by e-mail.

CAPTCHA

This question is for testing whether or not you are a human visitor and to prevent automated spam submissions.



CREATE YOUR ACCOUNT

CREATE ACCOUNT

LOGIN

X

d shows "billing_address",
" & "level3" objects which are

to make use of these optional

Errors Test Cards

,
ng}",

ring}",
g}",
",

ing}",
",

reference code – used by
system will be reflected in your
records and webhook
ons.
optional" field

transaction method.
_card"

[Docs & Sandbox](#)[Integration](#)[Support](#)[Forums](#)[Blog](#)[MY MERCHANTS](#)[MY APIs](#)[MY ACCOUNT](#)

1. LITE

2. GET CERTIFIED

3. ADD MERCHANTS

4. LIVE!

My APIs

Add more, edit or delete them as you like.

[**+ ADD A NEW API**](#)**test****SANDBOX**

test

SANDBOX

KEYS

PRODUCTS

DELETE "TEST"

EDIT "TEST"

ANALYTICS

API Product: Sandbox

Status

Approved

My Merchants

SANDBOX

LIVE

What's this ?

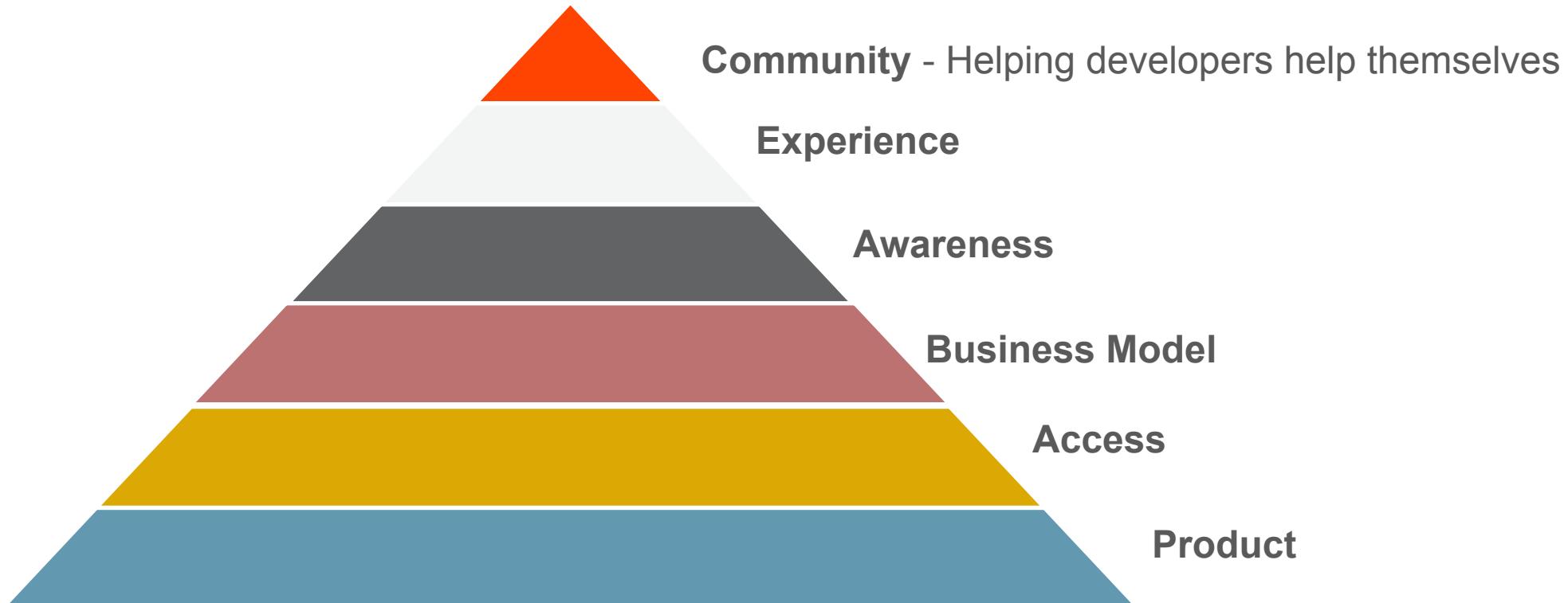
Merchant ID	Merchant Name	CSR	App Label	Token	JS Security Key	View Details	Status
317675295 5	Acme Sock	Download	PLE002.Sa ndBoxApp	fdoa- a480ce895 1daa73262 734cf10264 1994c1e55 e7cdf4c02 b6	js- df37ef4331 6d026d0b aecfd54f b465783bf 438c0479e 6a8	View Details	Approved

We've provisioned you with the above merchant account in Sandbox. Provide this merchant token as the "token" header parameter in your API call.

Note: For integrating with Apple Pay, here is your CSR. For further instructions on how to use the CSR, please refer to our documentation

[Next > Try it out!](#)

API Developer Program Success



Community: Nurturing within the portal



Docs & Sandbox Integration Support **Forums** Blog

[CREATE ACCOUNT](#)

[LOGIN](#)

Forums

[VIEW FORUMS](#)

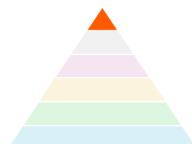
[ACTIVE TOPICS](#)

[UNANSWERED TOPICS](#)

New Posts

No New Posts

Forum	Topics	Posts	Last post
General Discussion Questions & conversations about general aspects of Payeezy development.	63	218	Capture payment... by rohitrajagopal3538 22 hours 10 min ago
Payments Help Payments, refunds, voids and a host of other payment specific topics.	6	19	How to check the... by rohitrajagopal3538 09/08/2015 - 16:56
Integration	50	184	Python Odoo... by krupeshlaiya5212



apigee

©2015 Apigee Corp. All Rights Reserved.

[SIGN IN](#)[REGISTER](#) [API Programs](#)[App Gallery](#)[Developer Events](#)[Press Room](#)[RSS Feed](#)

Welcome to our Developer Portal

GET

Started
with documentation

Building
with access to a key



App Gallery

Sep 08

[Introducing Winkflash: The newest Photo Prints API](#)[Integration](#)

Walgreens is happy to announce our newest Photo Prints API integration with Winkflash, whose free app can now be found on the Apple App Store!

Announcements

Sep 01

[Are you going to Mobile Photo Connect? We are!](#)

Walgreens is very excited to kickoff our financial year with our Gold Sponsorship at the Mobile Photo Connect 2015 conference.

Aug 25

Apr 27

[WebMD: New Platform Balance Rewards & Prescription](#)

Walgreens API Tweets

**Walgreens API**
[@WalgreensAPI](#)[Follow @WalgreensAPI](#)

Welcome to the Photo Prints API Family, [@Winkflash!](#)
t.co/ThNHf9tozr

1 week 2 days ago.

[Reply](#) [Retweet](#) [Favorite](#)**Walgreens API**
[@WalgreensAPI](#)

Home > Developer Events

Developer Events

Are you going to Mobile Photo Connect? We are!

Tue, 09/01/2015 - 14:10 - Drew Schweinfurth

[Sign In to Comment](#)

Walgreens is very excited to kickoff our financial year with our Gold Sponsorship at the [Mobile Photo Connect 2015](#) conference. This year the event is being held at the amazing Golden Gate Club in the beautiful area of Presidio, San Francisco.

Making an in-person from our team to represent Walgreens, along with the [many other leaders](#) across the mobile photo industry, will be Joe Rago (Director, Mobile Innovation Program and Drew Schweinfurth (Developer Evangelist).

Excitement levels rise as we see more and more people attending the conference every year! This year a large portion of the [Show & Tell Presentations](#) are represented by members of our great and growing Photo Prints API eco-system!

- MEA Mobile <http://printicular.com/>
- Canvsly <http://canvsly.com/>
- Rainhut <http://www.rainhut.com/>
- PicPocket <http://www.picpocket.com/>
- SnapsBoard <https://www.snapsboard.com/welcome>
- iGreet <http://igreet.co/>

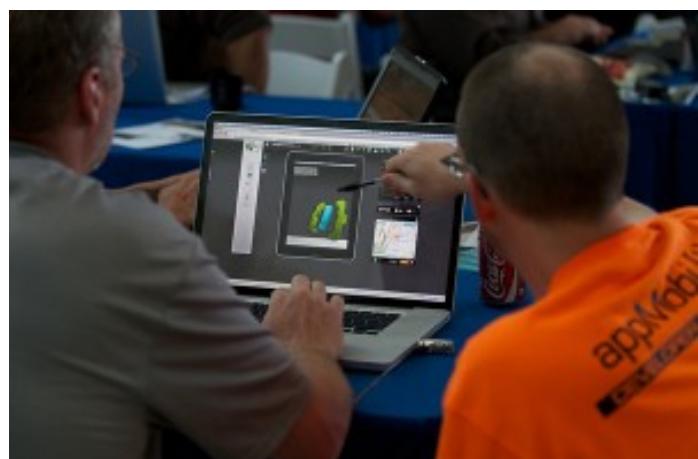
Categories

- Announcements (31)
- Developer Events (28)
- App Gallery (111)
- New Features (4)
- Prescription API (15)
- Photo Prints (88)
- Balance Rewards (18)
- Healthcare Clinic Scheduler (1)

Archives

- September 2015 (2)
- August 2015 (1)
- July 2015 (2)
- June 2015 (1)
- April 2015 (3)
- March 2015 (2)

Community: events build community



Community: external events

FUTURE OF WEB APPS

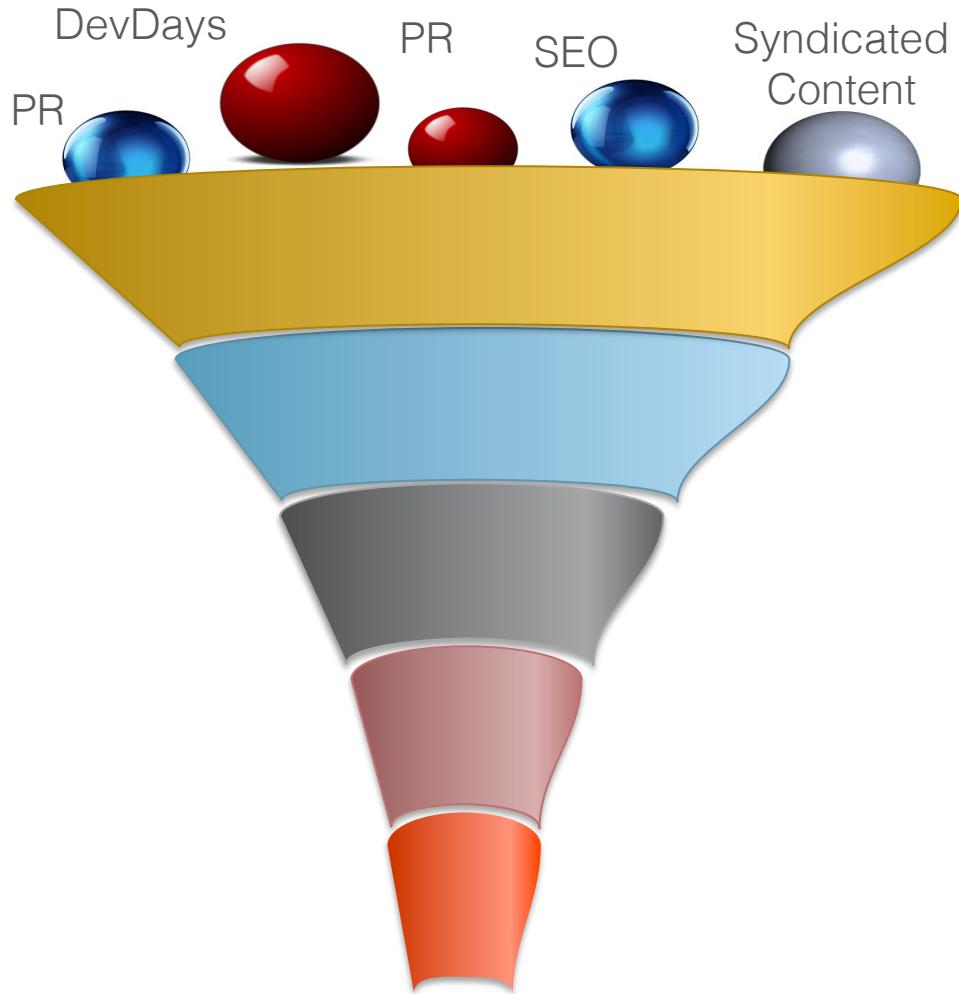
DIGIDAY:APPS



MOBILE APP
HACK-A-THON



For every buying process there is a ‘funnel’ and stages to measure



Stage	KPI	%	Typical %
Visits	1000	100%	
Signups	50	5%	2-5%
Trials	15	30%	20-40%
Apps	2	13%	15-30%

Improving conversions

Conversion	Consider
New visits	<ul style="list-style-type: none">• How do all demand gen options rank?• PR, TechCrunch or Programmable Web mentions• Improve SEO capability
Visits to signup	<ul style="list-style-type: none">• How are portal forums performing? Greater personal investment in responding• Improve TTR
Signup to trials	<ul style="list-style-type: none">• Easy to find the API, to take the next step?• Can you test the API in the portal?• Do you show how easy it is to use API?
Trials to completed Apps	<ul style="list-style-type: none">• Sample code to accelerate development?• A showcase for completed apps?
Completed to Show Apps	<ul style="list-style-type: none">• Seed more apps in the app store

Analytics are Essential



The Top 10 Best Practices

1

Be responsive and adjust to have the right developer **product**

3

Multichannel approach to generate **awareness**

5

Be perceived as cool, innovative

7

Make the API **experience** easy for developers

9

Make the first impression count

2

Measure and adapt for success

4

Support your **community** of developers and evangelists

6

Streamline the processes for easy **access**

8

Show them the money with the right **business model**

10

Know who you are targeting and how to reach them

Technical Tips

The RESTifarian

“

A RESTifarian is a zealous proponent of the REST software architectural style as defined by Roy T. Fielding in Chapter 5 of his PhD. dissertation at UC Irvine. You can find RESTifarians in the wild on the REST-discuss mailing list. But be careful, RESTifarians can be extremely meticulous when discussing the finer points of REST ...”

-Mike Schinkel, software architect and entrepreneur

“

If you dig a hole and it's in the wrong place, digging it deeper isn't going to help.

-Seymour Chwast, graphic designer

Design for Adoption



Resources

- <https://vimeo.com/11690980> – Darwinian Finches
- <https://www.youtube.com/user/apigee>
- <https://groups.google.com/forum/#!forum/api-craft>
- <http://apistudio.io>
- <https://apigee.com/about/resources>
- <http://iloveapis.com> - lgrey50off





Thank you
