



# Deep Integration

A Guide For Affiliate Network

SaaS Providers

Version 1.0

## Integration Goals

1. Allow streamlined importing and management of affiliates.
2. Facilitate easy creation of pay-per-call campaigns.
3. Allow affiliates to easily manage phone numbers without intervention.
4. Integration of data from CallPixels into the local transaction log.

## Account Setup

CallPixels provides customers with an API key that is tied to their account. This API key must be stored locally and passed in as a GET variable with every request. For more information see our API documentation at <http://callpixels.turnapi.com/>. The API key should never be exposed to affiliates as it allows unrestricted access to the CallPixels account.

## Setting up the company

CallPixels allows default click and conversion tracking URLs to be set up. These URLs should be configured for the customer with sensible defaults that will allow us to achieve the goals set out in this document.

See <http://callpixels.turnapi.com/docs/1.0/references-objects-company>

*The default click and conversion ("sale") URLs can be posted to <http://callpixels.com/company.json> without a company ID, so long as the correct API key is being used.*

## Affiliate Import

Affiliates can be managed using the affiliate ID set by your system. Affiliates can be initially imported by using many requests to the CallPixels API; at this time there is no method for posting multiple affiliates at once.

See <http://callpixels.turnapi.com/docs/1.0/references-objects-affiliate>

## Campaign Creation

You can test to see if a campaign exists on CallPixels by sending a GET request as outlined in our API documentation. Use your internal CID to reference the campaign on CallPixels.

See <http://callpixels.turnapi.com/docs/1.0/references-objects-campaign>

A new campaign can be added to CallPixels by directing the user to the CallPixels site. We make no attempt to break out of iframes. Prepopulate the CID and campaign name by sending them to our campaign creation URL like so:

<http://callpixels.com/campaigns/new?cid=34944&name=Test%20Campaign>



## Example code

Ruby

```
Digest::MD5.hexdigest("#{campaign_id}#{affiliate_id}supersecret")
```

PHP

```
md5($campaign_id . $affiliate_id . 'supersecret');
```

## Affiliate Suppression List Access

If your customer is running SMS campaigns they may be making use of our unsubscribe functionality. You can check to see if there are any unsubscribed phone numbers through our suppression list API.

```
> HTTP GET http://callpixels.com/suppression_list/check.json?api_key=
dd2063447e3fdf40190dabbc38e0de33
< 200 OK
< Content-Type: application/json; charset=utf-8
< {"suppression_list":false}
```

This returns true or false depending on if the customer has any suppressed numbers. To provide access to the file, link to

```
http://callpixels.com/embedded/for/cid/209198/afid/113209/suppression_list.txt?key=d3745f5310e37154fb7885387254cbf5&cs=dd2063447e3fdf40190dbbbc38e0de89
```

The parameters on the URL above are the same as in the affiliate number management section.

## Call data integration

As part of the click and conversion callbacks, CallPixels provides many replacement tokens (**Table 1**) that can be used to pass additional information about the call into your system. All callbacks occur after the call has completed, so all tokens are available for all callbacks.

By passing back information about the call in the click (0 second timer) callback, you will ensure that all clicks have complete call data.

**CallPixels Replacements Tokens (Table 1)**

<b>Token</b>	<b>Replaced with</b>
[campaign_id]	
[affiliate_id]	
[affiliate_first_name]	
[affiliate_last_name]	
[sub_id]	Sub-affiliate ID
[called_number]	Number that the caller dialed
[caller_id]	Caller's phone number as seen on CallerID
[caller_state]	
[caller_city]	
[caller_country]	ISO 3166-1 alpha-2 country code
[call_duration]	Total duration of the call
[call_forwarded_duration]	Length of time the call was connected to the target
[call_start_time]	Unix timestamp for when the call began
[call_forwarded_time]	Unix timestamp for when the call was forwarded
[call_finish_time]	Unix timestamp for when the call ended
[call_recording_url]	Full URL to the call recording mp3
[call_uuid]	CallPixels call UUID
[call_forwarded_to]	Number or sip address of the target the call was routed to
[charge_per_minute_in]	
[charge_per_minute_out]	
[charge_total]	