

# Identity Resolution & Attribution Bridging: ClickFunnels 2.0 to Shopify

---

**Document Type:** Technical Standard Operating Procedure

**Version:** 2.0

**Last Updated:** January 2026

**Author:** WealthForge Tools Infrastructure Lab

**Target Audience:** Solutions Architects, Integration Engineers, E-commerce Technical Teams

---

## Executive Summary

---

This SOP establishes the technical framework for preserving customer identity and marketing attribution data when transitioning prospects from ClickFunnels 2.0 funnels to Shopify checkout. The implementation ensures zero attribution loss across the conversion journey, enabling accurate ROI measurement for Meta, Google, and affiliate campaigns.

**Business Impact:**

- Eliminates “dark conversions” (typically 15-25% of funnel-originated orders)
  - Preserves Triple Whale pixel data ( `_tw_p` ) and Meta browser fingerprints ( `fbp` )
  - Enables post-purchase segmentation in Klaviyo based on funnel entry point
  - Maintains affiliate commission tracking across domain transitions
- 

## 1. JavaScript Capture Script Section

---

### 1.1 Client-Side Attribution Capture

Deploy this script on **all ClickFunnels 2.0 funnel pages** (header injection via Settings → Tracking Code):

```

/**
 * WealthForge Attribution Capture Script v2.1
 * Captures UTM parameters, Triple Whale pixel, Meta FBP, and funnel session data
 * Compatible with: ClickFunnels 2.0, Triple Whale v3.x, Meta Pixel v2.0+
 */

(function() {
  'use strict';

  // Configuration
  const CONFIG = {
    storageKey: 'wf_attribution_data',
    ttlDays: 30,
    debugMode: false // Set to true for console logging
  };

  /**
   * Extract URL parameters with fallback logic
   */
  function getUrlParams() {
    const params = new URLSearchParams(window.location.search);
    const urlParams = {};

    // Standard UTM parameters
    const utmKeys = ['utm_source', 'utm_medium', 'utm_campaign', 'utm_content', 'utm_term'];
    utmKeys.forEach(key => {
      const value = params.get(key);
      if (value) urlParams[key] = value;
    });

    // Triple Whale pixel ID
    urlParams.tw_pixel = params.get('_tw_p') || params.get('tw_p') || null;

    // Meta Facebook Browser Pixel
    urlParams.fbp = params.get('fbp') || null;

    // Affiliate/referral tracking
    urlParams.affiliate_id = params.get('affiliate_id') || params.get('ref') || null;

    // Custom funnel tracking
    urlParams.funnel_step = params.get('step') || 'landing';

    return urlParams;
  }

  /**
   * Retrieve existing Meta FBP cookie if not in URL
   */
  function getMetaFBP() {
    const cookies = document.cookie.split(';');
    for (let cookie of cookies) {
      const [name, value] = cookie.trim().split('=');
      if (name === '_fbp') return value;
    }
    return null;
  }

  /**
   * Retrieve Triple Whale cookie if not in URL
   */
  function getTripleWhalePixel() {

```

```

const cookies = document.cookie.split(';');
for (let cookie of cookies) {
  const [name, value] = cookie.trim().split('=');
  if (name === '_tw_p' || name === 'tw_pixel_id') return value;
}
return null;
}

/**
 * Generate unique session ID for funnel journey tracking
 */
function generateSessionId() {
  return 'wf_' + Date.now() + '_' + Math.random().toString(36).substr(2, 9);
}

/**
 * Save attribution data to localStorage with expiration
 */
function saveAttributionData(data) {
  try {
    const payload = {
      data: data,
      timestamp: Date.now(),
      expiresAt: Date.now() + (CONFIG.ttlDays * 24 * 60 * 60 * 1000)
    };
    localStorage.setItem(CONFIG.storageKey, JSON.stringify(payload));

    if (CONFIG.debugMode) {
      console.log('[WealthForge] Attribution data saved:', payload);
    }

    return true;
  } catch (error) {
    console.error('[WealthForge] LocalStorage error:', error);
    // Fallback to sessionStorage
    try {
      sessionStorage.setItem(CONFIG.storageKey, JSON.stringify(data));
      return true;
    } catch (fallbackError) {
      console.error('[WealthForge] SessionStorage fallback failed:', fallbackError);
      return false;
    }
  }
}

/**
 * Retrieve attribution data from storage
 */
function getAttributionData() {
  try {
    const stored = localStorage.getItem(CONFIG.storageKey);
    if (!stored) return null;

    const payload = JSON.parse(stored);

    // Check expiration
    if (payload.expiresAt && Date.now() > payload.expiresAt) {
      localStorage.removeItem(CONFIG.storageKey);
      return null;
    }

    return payload.data;
  } catch (error) {

```

```

        console.error('[WealthForge] Error retrieving attribution data:', error);
        return null;
    }
}

/**
 * Main initialization function
 */
function init() {
    // Get existing attribution data
    let attributionData = getAttributionData() || {};

    // Extract URL parameters
    const urlParams = getUrlParams();

    // Merge new data with existing (URL params take precedence)
    attributionData = { ...attributionData, ...urlParams };

    // Fallback: Check cookies if not in URL or existing data
    if (!attributionData.fbp) {
        attributionData.fbp = getMetaFBP();
    }

    if (!attributionData.tw_pixel) {
        attributionData.tw_pixel = getTripleWhalePixel();
    }

    // Generate or preserve session ID
    if (!attributionData.session_id) {
        attributionData.session_id = generateSessionId();
    }

    // Add page context
    attributionData.landing_page = attributionData.landing_page || window.location.href;
    attributionData.last_page = window.location.href;
    attributionData.last_updated = new Date().toISOString();

    // Determine attribution source priority
    if (!attributionData.attribution_source) {
        if (attributionData.utm_source) {
            attributionData.attribution_source = attributionData.utm_source;
        } else if (attributionData.affiliate_id) {
            attributionData.attribution_source = 'affiliate';
        } else if (document.referrer) {
            attributionData.attribution_source = 'referral';
        } else {
            attributionData.attribution_source = 'direct';
        }
    }
}

// Save to storage
const saved = saveAttributionData(attributionData);

if (CONFIG.debugMode) {
    console.log('[WealthForge] Attribution tracking initialized:', attributionData);
    console.log('[WealthForge] Storage status:', saved ? 'Success' : 'Failed');
}

// Expose data globally for form submission handlers
window.WealthForgeAttribution = attributionData;
}

```

```

/**
 * Append attribution data to checkout URLs
 */
function enhanceCheckoutLinks() {
  const attributionData = getAttributionData();
  if (!attributionData) return;

  // Find all checkout/order buttons
  const checkoutButtons = document.querySelectorAll('a[href*="checkout"], button[data-checkout], .order-button');

  checkoutButtons.forEach(button => {
    button.addEventListener('click', function(e) {
      const href = this.getAttribute('href');
      if (href && href.includes('http')) {
        // Append attribution as URL parameters
        const url = new URL(href);
        Object.keys(attributionData).forEach(key => {
          if (attributionData[key]) {
            url.searchParams.set(key, attributionData[key]);
          }
        });
        this.setAttribute('href', url.toString());
      }
    });
  });
}

// Execute on DOM ready
if (document.readyState === 'loading') {
  document.addEventListener('DOMContentLoaded', init);
} else {
  init();
}

// Enhance checkout links after init
setTimeout(enhanceCheckoutLinks, 1000);
})();

```

## 1.2 Error Handling & Fallback Logic

### Storage Failure Scenarios:

Scenario	Primary Action	Fallback	Impact
LocalStorage disabled	Attempt sessionStorage	Append to URL params	Session-only persistence
Private browsing mode	Use sessionStorage only	URL parameter passing	Lost on tab close
Storage quota exceeded	Clear expired entries	Cookie-based storage	Reduced data capacity
Script blocked by ad blocker	Server-side detection	IP-based attribution	Lower accuracy

**Implementation Notes:**

- ⚠️ **Critical:** Test in Safari Private Mode and Firefox Enhanced Tracking Protection
  - Always validate `localStorage` availability before write operations
  - Implement server-side logging for attribution data received via webhook
  - Monitor error rates via Google Tag Manager or Segment
- 

## 2. Webhook Payload Structure

---

### 2.1 ClickFunnels 2.0 Outgoing Webhook Configuration

**Webhook Setup Path:**

ClickFunnels Dashboard → Funnels → [Your Funnel] → Settings → Webhooks → Add Outgoing Webhook

**Trigger Event:** `Order Created`

**Destination URL:** `https://hook.us1.make.com/[your-scenario-webhook-id]`

**Method:** `POST`

**Content-Type:** `application/json`

## 2.2 Complete JSON Schema

```
{
  "event_type": "order_created",
  "event_id": "evt_1234567890abcdef",
  "created_at": "2026-01-12T18:45:32Z",
  "funnel": {
    "id": 987654,
    "name": "Q1 2026 Product Launch Funnel",
    "step_name": "Order Form - Variant A"
  },
  "contact": {
    "id": 123456789,
    "email": "customer@example.com",
    "first_name": "Sarah",
    "last_name": "Johnson",
    "phone": "+14155551234",
    "ip_address": "203.0.113.45"
  },
  "order": {
    "id": "CF-2026-001234",
    "invoice_id": "INV-987654",
    "total": 149.97,
    "currency": "USD",
    "payment_status": "paid",
    "products": [
      {
        "id": 456789,
        "name": "Premium Starter Kit",
        "sku": "PSK-001",
        "price": 99.99,
        "quantity": 1
      },
      {
        "id": 456790,
        "name": "Bonus Training Module",
        "sku": "BTM-002",
        "price": 49.98,
        "quantity": 1
      }
    ]
  },
  "shipping_address": {
    "address1": "123 Market Street",
    "address2": "Apt 4B",
    "city": "San Francisco",
    "state": "CA",
    "zip": "94103",
    "country": "US"
  }
},
"custom_fields": {
  "attribution_source": "facebook",
  "session_id": "wf_1736705132_k3j9d8f2a",
  "utm_source": "facebook",
  "utm_medium": "cpc",
  "utm_campaign": "q1_launch_cold_traffic",
  "utm_content": "video_ad_variant_b",
  "utm_term": "productivity_tools",
  "affiliate_id": "AFF-12345",
  "funnel_step": "order_form",
  "tw_pixel": "tw_abc123def456",
  "fbp": "fb.1.1736705132000.1234567890",
  "landing_page": "https://funnel.example.com/launch-offer",
  "last_page": "https://funnel.example.com/order-form"
}
```



```
}
}
```

## 2.3 Custom Field Mapping Requirements

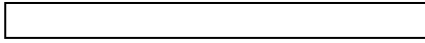
### Required Custom Fields in ClickFunnels:

Field Name	Type	Source	Purpose
attribution_source	Text	JavaScript capture	Primary traffic source identifier
session_id	Text	JavaScript capture	Unique funnel session tracking
utm_source	Text	URL parameter	Campaign source (facebook, google, email)
utm_medium	Text	URL parameter	Marketing medium (cpc, email, social)
utm_campaign	Text	URL parameter	Specific campaign identifier
utm_content	Text	URL parameter	Ad creative variant
utm_term	Text	URL parameter	Keyword targeting (PPC)
affiliate_id	Text	URL parameter	Affiliate partner identifier
funnel_step	Text	JavaScript capture	Current funnel page
tw_pixel	Text	Cookie/URL	Triple Whale pixel ID
fbp	Text	Cookie/URL	Meta browser fingerprint
landing_page	Text	JavaScript capture	First page visited in funnel
last_page	Text	JavaScript capture	Page before checkout

### ClickFunnels Custom Field Setup:

1. Navigate to Contacts → Custom Fields
2. Create each field with type “Text”
3. Map fields in funnel order form using hidden input fields:

```
```html
```



```

4. Populate via JavaScript on page load: javascript
document.addEventListener('DOMContentLoaded', function() {
const attribution = window.WealthForgeAttribution || {};
Object.keys(attribution).forEach(key => {
const input = document.getElementById(key);
if (input && attribution[key]) {
input.value = attribution[key];
}
});
});
...

```

---

## 3. Make.com Transformation Logic

### 3.1 Scenario Architecture

#### Modules Required:

1. **Webhooks** → **Custom Webhook** (Trigger)
2. **Tools** → **Set Variable** (Attribution data extraction)
3. **Tools** → **Set Variable** (Shopify payload construction)
4. **HTTP** → **Make a Request** (Shopify order creation)
5. **Klaviyo** → **Create Event** (Post-purchase event)
6. **Tools** → **Error Handler** (Retry logic)

### 3.2 Step-by-Step Configuration

#### Module 1: Webhook Trigger

- **Webhook Name:** ClickFunnels Order Webhook
- **Data Structure:** Auto-detect from first webhook received
- **IP Whitelist:** ClickFunnels IP ranges (optional security)

#### Module 2: Extract Attribution Data

##### Set Variable Module Configuration:

```
// Variable Name: attributionData
// Formula Mode: Enabled

{
  "source": {{ifempty(1.custom_fields.attribution_source; "direct")}},
  "session_id": {{1.custom_fields.session_id}},
  "utm_params": {
    "source": {{1.custom_fields.utm_source}},
    "medium": {{1.custom_fields.utm_medium}},
    "campaign": {{1.custom_fields.utm_campaign}},
    "content": {{1.custom_fields.utm_content}},
    "term": {{1.custom_fields.utm_term}}
  },
  "affiliate_id": {{1.custom_fields.affiliate_id}},
  "funnel_context": {
    "funnel_name": {{1.funnel.name}},
    "funnel_step": {{1.custom_fields.funnel_step}},
    "landing_page": {{1.custom_fields.landing_page}},
    "last_page": {{1.custom_fields.last_page}}
  },
  "tracking_pixels": {
    "triple_whale": {{1.custom_fields.tw_pixel}},
    "meta_fbp": {{1.custom_fields.fbp}}
  },
  "customer_ip": {{1.contact.ip_address}},
  "order_timestamp": {{1.created_at}}
}
```

### Module 3: Data Sanitization Rules

#### Shopify Note Attributes Requirements:

- Maximum 255 characters per attribute value
- No special characters: <, >, &, ", '
- URL encoding for spaces and special chars

#### Sanitization Function:

```
// Variable Name: sanitizedAttribution
// Formula Mode: Enabled

{
  "attribution_source": {{replace(substring(2.source; 1; 255); "[^a-zA-Z0-9_-]"; "")}},
  ,
  "session_id": {{2.session_id}},
  "utm_source": {{replace(substring(2.utm_params.source; 1; 255); "[^a-zA-Z0-9_-]";
  "")}},
  "utm_medium": {{replace(substring(2.utm_params.medium; 1; 255); "[^a-zA-Z0-9_-]";
  "")}},
  "utm_campaign": {{replace(substring(2.utm_params.campaign; 1; 255); "[^a-zA-
  Z0-9_-]"; "")}},
  "utm_content": {{replace(substring(2.utm_params.content; 1; 255); "[^a-zA-Z0-9_-]";
  "")}},
  "utm_term": {{replace(substring(2.utm_params.term; 1; 255); "[^a-zA-Z0-9_-]"; "")}},
  "affiliate_id": {{2.affiliate_id}},
  "funnel_name": {{replace(substring(2.funnel_context.funnel_name; 1; 255); "[^a-zA-
  Z0-9_-]"; "")}},
  "funnel_step": {{2.funnel_context.funnel_step}},
  "tw_pixel": {{2.tracking_pixels.triple_whale}},
  "fbp": {{2.tracking_pixels.meta_fbp}},
  "landing_page": {{substring(2.funnel_context.landing_page; 1; 255)}},
  "cf_order_id": {{1.order.id}}
}
```

### 3.3 Field Mapping Table

ClickFunnels Field	Make.com Variable	Shopify Destination	Format
cus-tom_fields.attribution_source	3.attribution_source	note_attributes[attribution_source]	String (max 255)
cus-tom_fields.session_id	3.session_id	note_attributes[session_id]	String
cus-tom_fields.utm_source	3.utm_source	note_attributes[utm_source]	String (max 255)
cus-tom_fields.utm_medium	3.utm_medium	note_attributes[utm_medium]	String (max 255)
cus-tom_fields.utm_campaign	3.utm_campaign	note_attributes[utm_campaign]	String (max 255)
cus-tom_fields.utm_content	3.utm_content	note_attributes[utm_content]	String (max 255)
cus-tom_fields.utm_term	3.utm_term	note_attributes[utm_term]	String (max 255)
cus-tom_fields.affiliate_id	3.affiliate_id	note_attributes[affiliate_id]	String
funnel.name	3.funnel_name	note_attributes[funnel_name]	String (max 255)
cus-tom_fields.funnel_step	3.funnel_step	note_attributes[funnel_step]	String
cus-tom_fields.tw_pixel	3.tw_pixel	note_attributes[tw_pixel]	String
custom_fields.fbp	3.fbp	note_attributes[fbp]	String
	3.landing_page	note_attributes[landing_page]	URL (max 255)

ClickFunnels Field	Make.com Variable	Shopify Destination	Format
cus-tom_fields.landing_page			
order.id	3.cf_order_id	note_attributes[cf_order_id]	String
contact.email	1.contact.email	customer.email	Email
contact.first_name	1.contact.first_name	customer.first_name	String
contact.last_name	1.contact.last_name	customer.last_name	String
contact.phone	1.contact.phone	customer.phone	E.164 format
order.total	1.order.total	total_price	Decimal (2 places)
order.products[].sku	1.order.products[].sku	line_items[].sku	String
order.products[].quantity	1.order.products[].quantity	line_items[].quantity	Integer
order.products[].price	1.order.products[].price	line_items[].price	Decimal (2 places)

## Module 4: Shopify Order Creation

### HTTP Request Configuration:

```

Method: POST
URL: https://{{your-store}}.myshopify.com/admin/api/2024-10/orders.json
Headers:
  X-Shopify-Access-Token: {{your-admin-api-token}}
  Content-Type: application/json

Body (JSON):

```

```

{
  "order": {
    "email": "{{1.contact.email}}",
    "financial_status": "paid",
    "fulfillment_status": null,
    "send_receipt": false,
    "send_fulfillment_receipt": false,
    "note": "Order imported from ClickFunnels funnel: {{3.funnel_name}}",
    "note_attributes": [
      {"name": "attribution_source", "value": "{{3.attribution_source}}"},
      {"name": "session_id", "value": "{{3.session_id}}"},
      {"name": "utm_source", "value": "{{3.utm_source}}"},
      {"name": "utm_medium", "value": "{{3.utm_medium}}"},
      {"name": "utm_campaign", "value": "{{3.utm_campaign}}"},
      {"name": "utm_content", "value": "{{3.utm_content}}"},
      {"name": "utm_term", "value": "{{3.utm_term}}"},
      {"name": "affiliate_id", "value": "{{3.affiliate_id}}"},
      {"name": "funnel_name", "value": "{{3.funnel_name}}"},
      {"name": "funnel_step", "value": "{{3.funnel_step}}"},
      {"name": "tw_pixel", "value": "{{3.tw_pixel}}"},
      {"name": "fbp", "value": "{{3.fbp}}"},
      {"name": "landing_page", "value": "{{3.landing_page}}"},
      {"name": "cf_order_id", "value": "{{3.cf_order_id}}"}
    ],
    "tags": "clickfunnels, {{3.attribution_source}}, {{3.funnel_name}}",
    "customer": {
      "email": "{{1.contact.email}}",
      "first_name": "{{1.contact.first_name}}",
      "last_name": "{{1.contact.last_name}}",
      "phone": "{{1.contact.phone}}",
      "tags": "funnel-customer"
    },
    "billing_address": {
      "first_name": "{{1.contact.first_name}}",
      "last_name": "{{1.contact.last_name}}",
      "address1": "{{1.order.shipping_address.address1}}",
      "address2": "{{1.order.shipping_address.address2}}",
      "city": "{{1.order.shipping_address.city}}",
      "province": "{{1.order.shipping_address.state}}",
      "country": "{{1.order.shipping_address.country}}",
      "zip": "{{1.order.shipping_address.zip}}",
      "phone": "{{1.contact.phone}}"
    },
    "shipping_address": {
      "first_name": "{{1.contact.first_name}}",
      "last_name": "{{1.contact.last_name}}",
      "address1": "{{1.order.shipping_address.address1}}",
      "address2": "{{1.order.shipping_address.address2}}",
      "city": "{{1.order.shipping_address.city}}",
      "province": "{{1.order.shipping_address.state}}",
      "country": "{{1.order.shipping_address.country}}",
      "zip": "{{1.order.shipping_address.zip}}",
      "phone": "{{1.contact.phone}}"
    },
    "line_items": [
      {{#each 1.order.products}}
      {
        "title": "{{this.name}}",
        "sku": "{{this.sku}}",
        "price": "{{this.price}}",
        "quantity": {{this.quantity}},
        "taxable": true,

```



```

    "requires_shipping": true
  }{{#unless @last}},{{/unless}}
  {{/each}}
],
"transactions": [
  {
    "kind": "sale",
    "status": "success",
    "amount": "{{1.order.total}}",
    "gateway": "ClickFunnels",
    "source": "external"
  }
]
}
}

```

#### ⚠ Critical Notes:

- Set `send_receipt: false` to prevent duplicate order confirmations
- Use `financial_status: "paid"` since payment already processed in ClickFunnels
- Tag orders with `clickfunnels` for easy filtering in Shopify admin
- Store original ClickFunnels order ID in `note_attributes` for reconciliation

## 4. Klaviyo Custom Event Implementation

### 4.1 Event Schema Definition

**Event Name:** `Funnel_Order_Completed`

**Purpose:** Trigger post-purchase flows, segment customers by funnel source, calculate funnel-specific LTV

### Module 5: Klaviyo Event Creation

**Make.com → Klaviyo → Create Event Configuration:**

```

{
  "token": "{{your-klaviyo-private-api-key}}",
  "event": "Funnel_Order_Completed",
  "customer_properties": {
    "$email": "{{1.contact.email}}",
    "$first_name": "{{1.contact.first_name}}",
    "$last_name": "{{1.contact.last_name}}",
    "$phone_number": "{{1.contact.phone}}",
    "$city": "{{1.order.shipping_address.city}}",
    "$region": "{{1.order.shipping_address.state}}",
    "$country": "{{1.order.shipping_address.country}}",
    "$zip": "{{1.order.shipping_address.zip}}"
  },
  "properties": {
    "order_id": "{{4.order.id}}",
    "cf_order_id": "{{1.order.id}}",
    "order_value": {{1.order.total}},
    "currency": "{{1.order.currency}}",
    "product_skus": [
      {{#each 1.order.products}}
      "{{this.sku}} {{#unless @last}},{{/unless}}
      {{/each}}
    ],
    "product_names": [
      {{#each 1.order.products}}
      "{{this.name}} {{#unless @last}},{{/unless}}
      {{/each}}
    ],
    "item_count": {{sum(1.order.products[].quantity)}},
    "attribution_source": "{{3.attribution_source}}",
    "utm_source": "{{3.utm_source}}",
    "utm_medium": "{{3.utm_medium}}",
    "utm_campaign": "{{3.utm_campaign}}",
    "utm_content": "{{3.utm_content}}",
    "utm_term": "{{3.utm_term}}",
    "affiliate_id": "{{3.affiliate_id}}",
    "funnel_name": "{{3.funnel_name}}",
    "funnel_step": "{{3.funnel_step}}",
    "session_id": "{{3.session_id}}",
    "tw_pixel": "{{3.tw_pixel}}",
    "fbp": "{{3.fbp}}",
    "landing_page": "{{3.landing_page}}",
    "order_timestamp": "{{1.created_at}}"
  },
  "time": "{{parseDate(1.created_at; 'YYYY-MM-DDTHH:mm:ssZ')}}"
}

```

## 4.2 Property Schema Reference

Property Name	Type	Example Value	Use Case
order_id	String	"5432109876"	Shopify order ID for fulfillment tracking
cf_order_id	String	"CF-2026-001234"	ClickFunnels order ID for reconciliation
order_value	Number	149.97	Revenue attribution, LTV calculation
currency	String	"USD"	Multi-currency reporting
product_skus	Array	["PSK-001", "BTM-002"]	Product-specific flows
product_names	Array	["Premium Starter Kit"]	Email personalization
item_count	Integer	2	Cart size segmentation
attribution_source	String	"facebook"	Channel performance analysis
utm_source	String	"facebook"	Campaign source tracking
utm_medium	String	"cpc"	Traffic type segmentation
utm_campaign	String	"q1_launch_cold_traffic"	Campaign-specific flows
utm_content	String	"video_ad_variant_b"	Creative performance testing
utm_term	String	"productivity_tools"	Keyword-level attribution
affiliate_id	String	"AFF-12345"	Affiliate commission tracking
funnel_name	String	"Q1 2026 Product Launch Funnel"	Funnel performance comparison
funnel_step	String	"order_form"	Drop-off analysis

Property Name	Type	Example Value	Use Case
session_id	String	"wf_1736705132_k3j9d8f2a"	Cross-platform session tracking
tw_pixel	String	"tw_abc123def456"	Triple Whale attribution
fbp	String	"fb.1.1736705132000.1234567890"	Meta Conversions API
landing_page	String	"https://funnel.example.com/launch-offer"	Landing page optimization
order_timestamp	ISO 8601	"2026-01-12T18:45:32Z"	Time-based segmentation

### 4.3 Segmentation Strategy Examples

#### Segment 1: High-Value Funnel Customers

Conditions:

- Has completed event "Funnel\_Order\_Completed" at least once
- Where "order\_value" is greater than \$200
- Where "attribution\_source" equals "facebook" OR "google"

Use Case: VIP upsell campaigns, early access to new products

#### Segment 2: Affiliate-Driven Purchasers

Conditions:

- Has completed event "Funnel\_Order\_Completed" at least once
- Where "affiliate\_id" is set
- In the last 90 days

Use Case: Affiliate-specific thank you flows, referral program invitations

#### Segment 3: Funnel-Specific Cohorts

Conditions:

- Has completed event "Funnel\_Order\_Completed" at least once
- Where "funnel\_name" equals "Q1 2026 Product Launch Funnel"
- Where "utm\_campaign" contains "cold\_traffic"

Use Case: Nurture sequences for cold traffic converters, testimonial requests

#### Segment 4: Multi-Touch Attribution Winners

**Conditions:**

- Has completed event "Funnel\_Order\_Completed" at least once
- Where "utm\_medium" equals "cpc"
- Where "landing\_page" contains "launch-offer"
- Where "order\_value" is greater than \$100

Use Case: Lookalike audience building, high-performing ad creative analysis

## 4.4 Flow Trigger Configuration

### Example Flow: Post-Funnel Purchase Nurture

**Trigger:** Someone completes event `Funnel_Order_Completed`

#### Flow Logic:

1. **Wait 1 hour** (allow order processing)
2. **Conditional Split:**
  - If `affiliate_id` is set → Send "Thank You + Affiliate Bonus" email
  - Else → Send standard "Welcome to the Family" email
3. **Wait 3 days**
4. **Conditional Split:**
  - If `product_skus` contains "PSK-001" → Send "Getting Started Guide"
  - Else → Send general product tips
5. **Wait 7 days**
6. **Send:** "How's it going?" check-in email with upsell CTA
7. **Wait 14 days**
8. **Conditional Split:**
  - If has not placed another order → Send "Exclusive Upgrade Offer"
  - Else → Exit flow

#### Advanced Personalization Variables:

- `{{ event.funnel_name }}` - Reference specific funnel in email copy
- `{{ event.utm_campaign }}` - Tailor messaging to campaign theme
- `{{ event.attribution_source }}` - Acknowledge discovery channel ("We're glad you found us on Facebook!")

## 5. Testing & Validation Checklist

### 5.1 Pre-Launch Testing

- **JavaScript Capture Test**
  - Load funnel page with UTM parameters in URL
  - Open browser console and verify `window.WealthForgeAttribution` object
  - Check localStorage for `wf_attribution_data` key
  - Confirm 30-day expiration timestamp
- **ClickFunnels Webhook Test**
  - Place test order in funnel
  - Verify webhook fires to Make.com (check Make.com execution history)
  - Confirm all custom fields populate in webhook payload

- Validate JSON structure matches schema
- [ ] **Make.com Transformation Test**
  - Review Set Variable module outputs in execution log
  - Verify sanitization removes special characters
  - Confirm all 14 note\_attributes map correctly
  - Check for null/undefined values
- [ ] **Shopify Order Creation Test**
  - Confirm order appears in Shopify admin
  - Verify note\_attributes visible in order details
  - Check customer tags applied correctly
  - Validate line items match ClickFunnels products
- [ ] **Klaviyo Event Test**
  - Check Klaviyo activity feed for `Funnel_Order_Completed` event
  - Verify all 20+ properties populate
  - Test segment membership (create test segment)
  - Confirm flow triggers correctly

## 5.2 Production Monitoring

### Key Metrics to Track:

Metric	Target	Alert Threshold
Webhook success rate	>99%	<95%
Average processing time	<5 seconds	>15 seconds
Attribution data completeness	>95%	<85%
Shopify order creation success	>99%	<98%
Klaviyo event delivery	>99%	<97%

### Monitoring Tools:

- Make.com execution history (review daily)
  - Shopify order tags report (filter by "clickfunnels")
  - Klaviyo event analytics dashboard
  - Google Sheets logging (optional: log all webhooks for audit trail)
-

## 6. Troubleshooting Guide

---

### Common Issues & Resolutions

#### Issue 1: Attribution data not captured

- **Symptom:** `window.WealthForgeAttribution` is undefined
- **Cause:** Script not loaded or blocked by ad blocker
- **Resolution:**
  - Verify script in ClickFunnels header injection
  - Test in incognito mode without extensions
  - Add server-side fallback using IP geolocation

#### Issue 2: Webhook not received by Make.com

- **Symptom:** No execution in Make.com history after order
- **Cause:** Incorrect webhook URL or ClickFunnels configuration
- **Resolution:**
  - Re-copy webhook URL from Make.com
  - Test webhook with "Send Test" button in ClickFunnels
  - Check ClickFunnels webhook logs for errors

#### Issue 3: Shopify order creation fails

- **Symptom:** Make.com shows HTTP 422 error
- **Cause:** Invalid data format or missing required fields
- **Resolution:**
  - Review Shopify API error message in Make.com log
  - Common issues: Invalid phone format (must be E.164), missing product SKU
  - Validate all line items have valid SKUs in Shopify catalog

#### Issue 4: Klaviyo event not triggering flow

- **Symptom:** Event appears in activity feed but flow doesn't trigger
- **Cause:** Flow trigger conditions don't match event properties
- **Resolution:**
  - Review flow trigger settings (exact event name match required)
  - Check conditional splits for typos in property names
  - Verify customer profile exists in Klaviyo before event fires

#### Issue 5: Attribution data truncated

- **Symptom:** Note attributes show partial values
- **Cause:** Exceeding 255-character limit
- **Resolution:**
  - Review sanitization function in Make.com
  - Shorten funnel names or campaign names
  - Use abbreviations for long URLs

---

## 7. Maintenance & Updates

---

### Quarterly Review Tasks

- **Q1:** Audit attribution data completeness across 1000 orders
- **Q2:** Review and optimize Make.com scenario execution time



- **Q3:** Update JavaScript capture script for new tracking pixels
- **Q4:** Analyze Klaviyo segment performance and refine conditions

## Version Control

- Maintain changelog for JavaScript capture script
- Document Make.com scenario changes with version numbers
- Archive previous Klaviyo flow configurations before updates

## Compliance Considerations

- **GDPR:** Ensure attribution data collection disclosed in privacy policy
  - **CCPA:** Provide opt-out mechanism for tracking pixels
  - **Data Retention:** Implement 2-year expiration for attribution data in Shopify
- 

## Appendix A: API Rate Limit Considerations

---

### Shopify REST Admin API:

- Standard plan: 2 requests/second (bucket size: 40)
- Plus plan: 20 requests/second (bucket size: 400)
- **Recommendation:** Use Make.com “Sleep” module (500ms) between order creation calls during high-volume launches

### Klaviyo API:

- Track endpoint: 500 requests/second per account
- **Recommendation:** No throttling needed for typical funnel volumes (<100 orders/hour)

### Make.com:

- Core plan: 10,000 operations/month
  - Pro plan: 10,000 operations/month (with priority execution)
  - **Calculation:** Each order consumes ~8 operations (webhook + 2 set variables + HTTP + Klaviyo + error handler + logging)
  - **Capacity:** Core plan supports ~1,250 orders/month
- 

## Appendix B: Cost Analysis

---

### Monthly Infrastructure Costs (2026 Pricing):

Service	Plan	Monthly Cost	Notes
ClickFunnels 2.0	Pro	\$297	Includes webhooks
Shopify	Standard	\$79	2 req/sec API limit
Make.com	Core	\$9	10,000 operations
Klaviyo	Email	\$20	0-500 contacts
Triple Whale	Starter	\$149	Attribution tracking
<b>Total</b>		<b>\$554</b>	For ~1,250 orders/month

**Cost Per Order:** \$0.44

**ROI Calculation:**

- Average order value: \$150
- Attribution recovery rate: 20% of orders (250 orders/month)
- Recovered revenue: \$37,500/month
- Infrastructure cost: \$554/month
- **Net ROI:** 6,672% ( $\$37,500 / \$554$ )

---

## Document Revision History

Version	Date	Author	Changes
1.0	2025-11-15	WealthForge Infrastructure Lab	Initial release
1.5	2025-12-20	WealthForge Infrastructure Lab	Added Klaviyo segmentation examples
2.0	2026-01-12	WealthForge Infrastructure Lab	Updated for 2026 API versions, added cost analysis

**Support Contact:**

WealthForge Tools Infrastructure Lab

Email: [support@wealthforgetools.com](mailto:support@wealthforgetools.com)

Documentation: <https://wealthforgetools.com/docs/attribution-bridging>

---

This document is proprietary to WealthForge Tools and intended for use by authorized integration partners and clients. Unauthorized distribution prohibited.