



**ANOTHER  
MP WEBINAR  
THANKS FOR  
JOINING US**

**MP Webhooks**





# ANOTHER MP WEBINAR THANKS FOR JOINING US

## What is a webhook?

- User Defined HTTP Callbacks
- Event Triggered HTTP Call




# ANOTHER MP WEBINAR THANKS FOR JOINING US

## Webhook Uses

- Synchronization of data to other systems
- External website cache reset
- Let your imagination run wild with possibilities
- More powerful processes



# ANOTHER MP WEBINAR THANKS FOR JOINING US



WebhookInbox receives HTTP requests and captures the data for later inspection.

It is useful for debugging HTTP callbacks or seeing what a browser/client is sending. When you create an inbox, you are given a generated URL that is to be used by the sending application.

[create an inbox](#)

### Contact Changed

Webhook #1

Display Name \*: Contact Changed

Description:

Http Method \*: POST

Uri Template \*: <http://api.webhookinbox.com/i/626M2Tff/in/>

Body Template: {"contactid": [dp\_RecordID], "userid": [dp\_UserID]}

Headers Template:

Trigger Fields: Email\_Address, First\_Name, Nickname, Last\_Name

Dependent Condition:

Trigger On Create \*: True

Trigger On Update \*: True

Trigger On Delete \*: True

Max Retries \*: 10

Active \*: False

Table Name \*: Contacts

Trigger On File Change \*: False

```
isVideo = ( { type == "image" } ) { isVideo: true }
isUrl = ( { source: parseInt("0x"+hex.substr(0, 4)) } ) { isUrl: true }
isElement = ( { type == "element" } ) { isElement: true }
isObject = ( { type == "object" } ) { isObject: true }

// Check if boxer is already active, return false
if ($("#boxer").length > 0) {
    return;
}

// Kill event
_killEvent(e);

// Cache internal data
data = $.extend({}, {
    $window: $(window),
    $body: $("body"),
    $target: $target,
    $object: $object,
    visible: false,
    resizeTimer: null,
    touchTimer: null,
    gallery: {
        active: false
    }
});
```

## Tools for Webhooks

- Webhook Inbox
- MinistryPlatform
- Programming language of your choice



**ANOTHER  
MP WEBINAR  
THANKS FOR  
JOINING US**

# Example Webhooks



# ANOTHER MP WEBINAR THANKS FOR JOINING US



## Send Background Consent PDF

Webhook #339

Display Name *	Send Background Consent PDF
Description:	CloudServices Webhook that creates PDF of Background Check Consent form and emails.
Http Method *	POST
Uri Template *	https://localhost/CloudServices/api/BackgroundCheck/SendAuthorizati
Body Template:	{ "id":[dp_RecordID], "userId": [dp_UserID], "changeType": "[dp_AuditType]", "tableName":"Background_Checks" }
Headers Template:	Content-Type: application/json
Trigger Fields:	Background_Check_Submitted
Dependent Condition:	Background_Check_Submitted IS NOT NULL AND Background_Check_Returned IS NULL AND Background_Check_Type_ID IS NOT NULL
Trigger On Create *	False
Trigger On Update *	True
Trigger On Delete *	False
Max Retries *	10
Active *	False
Table Name *	Background_Checks
Trigger On File Change *	False



# ANOTHER MP WEBINAR THANKS FOR JOINING US



## Contact Changed

Webhook #1

Display Name *:	Contact Changed
Description:	Webhook to sync Contact records between systems when Email or name modified.
Http Method *:	POST
Uri Template *:	<a href="http://api.webhookinbox.com/i/5mW8zgOw/in/">http://api.webhookinbox.com/i/5mW8zgOw/in/</a>
Body Template:	<code>{"contactid": [dp_RecordID], "userid": [dp_UserID], "changetype": [dp_AuditType], "tablename": "Contacts"}</code>
Headers Template:	
Trigger Fields:	Email_Address, First_Name, Nickname, Last_Name
Dependent Condition:	
Trigger On Create *:	True
Trigger On Update *:	True
Trigger On Delete *:	True
Max Retries *:	10
Active *:	True
Table Name *:	Contacts
Trigger On File Change *:	True



# ANOTHER MP WEBINAR THANKS FOR JOINING US

## MP Webhook Tokens

- **[dp\_RecordID]**
  - Primary Key of record
- **[dp\_UserID]**
  - UserId that modified / created record
- **[dp\_AuditType]**
  - The type of event that triggered the webhook





# ANOTHER MP WEBINAR THANKS FOR JOINING US

## Security

- We only send record / user Ids
  - NO data traverses the web callback
- Add Authorization or other header to authenticate to your endpoint
- Requires working knowledge of our API to fully implement



# ANOTHER MP WEBINAR THANKS FOR JOINING US

## Flexibility

- Control Webhooks by action type – Create, Update, Delete, File Change
- Control by Trigger Fields
- Control by Dependent Clause
- Retries (every 1 minute)
- You can Re-Run Webhook (Invocations)
- Header Control
- Get or Post with customization
- Add Custom Parameters to body, query or header



**ANOTHER  
MP WEBINAR  
THANKS FOR  
JOINING US**

## What can Webhooks Replace?

- Scheduled Jobs to Pull or Sync Data
  - Think closer to real time
- Expensive SQL Triggers
  - Can block / slow data flow or cause deadlocks



**ANOTHER  
MP WEBINAR  
THANKS FOR  
JOINING US**

## Deployment Options

- Through the Platform
- Through the REST API
  - You can create and monitor Webhooks through the REST API



**ANOTHER  
MP WEBINAR  
THANKS FOR  
JOINING US**

# Webhooks Demo



**ANOTHER  
MP WEBINAR  
THANKS FOR  
JOINING US**

**Question and Answer**



**ANOTHER  
MP WEBINAR  
THANKS FOR  
JOINING US**

**We can't wait to see what you build with  
Webhooks!**