

## Chapter 2

# Data Collection Entries Web Service Method - Upload - POST

from Data Collection Entries API Guide for Workforce  
Now Tempus PI Guide

Published on  
Dec 21, 2020 4:17AM

Last modified  
Jul 27, 2022 2:54AM



## ADP Copyright Information

ADP, the ADP logo, and Always Designing for People are trademarks of ADP, Inc.

Windows is a registered trademark of the Microsoft Corporation.

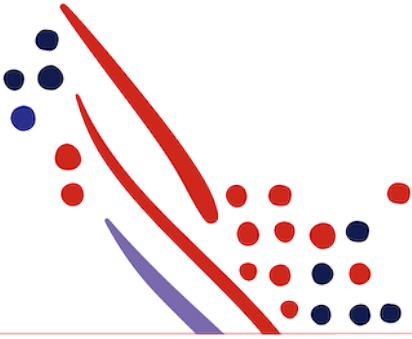
All other trademarks are the property of their respective owners.

Copyright © 2022 ADP, Inc. ADP Proprietary and Confidential - All Rights Reserved. These materials may not be reproduced in any format without the express written permission of ADP, Inc.

These materials may not be reproduced in any format without the express written permission of ADP, Inc. ADP provides this publication "as is" without warranty of any kind, either expressed or implied, including, but not limited to, the implied warranties of merchantability or fitness for a particular purpose. ADP is not responsible for any technical inaccuracies or typographical errors which may be contained in this publication. Changes are periodically made to the information herein, and such changes will be incorporated in new editions of this publication. ADP may make improvements and/or changes in the product and/or the programmes described in this publication.

Published on  
Dec 21, 2020 4:17AM

Last modified  
Jul 27, 2022 2:54AM



# Chapter Contents

## Chapter 2

### **Data Collection Entries Web Service Method - Upload - POST**

Action Code/ Entry Codes Overview

    Action and Entry Code JSON Object Locations

    Action Code/Entry Code Description Matrix

    Action Code/Entry Code Not Supported for Tempus PI

General Comments and Notes

Request Header Parameters

Multiple Punches Sample

Scan/Punch

Lunch Out

Clock In

Clock Out

Sample Supervisor Edit

Alternate Worker ID Samples

    Using PositionID

    Using ASSOCIATEOID

    Using Work Assignment ID (PFID)

Data Collection Entries API Element Glossary

    Events[]

    dataCollectionEntries[]

    workerDataCollectionEntries[]

    Alternate Worker ID's (alternateWokerIDs and alternateActorWokerIDs)

## Chapter 2

# Data Collection Entries Web Service Method - Upload - POST

## Data Collection Entries Web Service Method - Upload - POST

The data-collection-entries.process HTTP POST method is used to upload a payload of punches to the ADPTLM servers for processing.

## Action Code/ Entry Codes Overview

This section describes the validate Action code and Entry Code to be used in the API.

## Action and Entry Code JSON Object Locations

Element Name	JSON Object Location
Action Code	/events/data/transform/dataCollectionEntries/workerDataCollectionEntries/actionCode/codeValue
Entry Code	/events/data/transform/dataCollectionEntries/workerDataCollectionEntries/entryCode/codeValue

## Action Code/Entry Code Description Matrix

Action Code	Entry Code	Description
punch	NA	Scan operation (first scan represents an "IN"; second scan represents an "OUT").
lunchout	NA	Clock out for lunch operation.
clockin	NA	Clock-in operation
clockout	NA	Clock-out operation

## Action Code/Entry Code Not Supported for Tempus PI

Action Code	Entry Code	Description
Transfer	NA	<b>Due to other dependencies transfer functionality is not supported to marketplace clients for TEMPUS PI</b>
earnings	{required}	<b>Earning are not supported for TEMPUS PI</b>

suppEarnings	{required}	Earning are not supported for TEMPUS PI
--------------	------------	---

## General Comments and Notes

- The target clocking employees are always specified in the body by using the Badge value
- For all HTTP Status responses of 202, The "Location" and "Retry-After" response headers indicate that a subsequent request can be made to the data-collection-entries/{eventid} HTTP GET method to obtain an import and/or timecard posting status of the uploaded punches
- The data-collection-entries.process web service is designed to be used asynchronously:
  - Honoring the "Retry-After" response header returned by the data-collection-entries.process and data-collection-entries/{eventid} web service pertains only to the subsequent requests to obtain a status for a particular Event ID (using data-collection-entries/{eventid}).
  - Subsequent requests can be made to the data-collection-entries.process web service to upload additional punches while asynchronously polling/waiting for previous bulk upload events to complete
  - To limit inadvertent DOS (Deinial of Service) incidences, ADP asks consuming applications to honor a wait interval of at least one minute between each subsequent bulk upload request made to data-collection-entries.process.
- Refer to the latest "ADP APIs - Data Dictionary.pdf" document for the following:
  - Creating your Personal Information Exchange (PFX) Certificate.
  - Installing Your PFX Certificate.
  - Installing and Preparing Google Postman
  - Requesting a Bearer Token

## Request Header Parameters

Header Name	Sample value	Comments
Authorization	Bearer {accessToken}	The Token Type (accessToken—i.e. "Bearer") plus the Access Token (access_token—i.e. "edc3c2-6b5d-4251-b258-d1208cbf6339") returned from ADP Marketplace's Token web service.

## Multiple Punches Sample

This is an example of a request that contains multiple time punch entries in the uploading payload

For API related information please refer [Data Collection Entries](#) in API explorer

<b>API:</b>	/events/time/v1/data-collection-entries.process
<b>Method:</b>	POST
<b>Condition:</b>	The Payload has been accepted and stored for further processing by ADP Marketplace. Subsequent requests to the datacollection entries.process, Status service are required by providing the <b>eventID</b> value returned in the <b>confirmMessage</b> object within the response to confirm the processing status of the submitted bulk upload.

The Response Code applies specifically to the POST Method request.

## Scan/Punch

Performs a scan punch operation where the first scan represents an "IN" punch and the next scan represents an "OUT" punch.

For API related information please refer [Data Collection Entries](#) in API explorer

<b>API:</b>	/events/time/v1/data-collection-entries.process
<b>Method:</b>	POST
<b>Condition:</b>	<p>The Payload has been accepted and stored for further processing by ADP Marketplace. Subsequent requests to the datacollection entries.process, Status service are required by providing the <b>eventID</b> value returned in the <b>confirmMessage</b> object within the response to confirm the processing status of the submitted bulk upload.</p> <p>The Response Code applies specifically to the POST Method request.</p>

## Lunch Out

Performs a lunch out operation.

For API related information please refer [Data Collection Entries](#) in API explorer

<b>API:</b>	/events/time/v1/data-collection-entries.process
<b>Method:</b>	POST
<b>Condition:</b>	<p>The Payload has been accepted and stored for further processing by ADP Marketplace. Subsequent requests to the datacollection entries.process, Status service are required by providing the <b>eventID</b> value returned in the <b>confirmMessage</b> object within the response to confirm the processing status of the submitted bulk upload.</p> <p>The Response Code applies specifically to the POST Method request.</p>

## Clock In

Performs a clock-in operation.

For API related information please refer [Data Collection Entries](#) in API explorer

<b>API:</b>	/events/time/v1/data-collection-entries.process
<b>Method:</b>	POST
<b>Condition:</b>	<p>The Payload has been accepted and stored for further processing by ADP Marketplace. Subsequent requests to the datacollection entries.process, Status service are required by providing the <b>eventID</b> value returned in the <b>confirmMessage</b> object within the response to confirm the processing status of the submitted bulk upload.</p> <p>The Response Code applies specifically to the POST Method request.</p>

## Clock Out

For API related information please refer [Data Collection Entries](#) in API explorer

<b>API:</b>	/events/time/v1/data-collection-entries.process
<b>Method:</b>	POST
<b>Condition:</b>	<p>The Payload has been accepted and stored for further processing by ADP Marketplace. Subsequent requests to the datacollection entries.process, Status service are required by providing the <b>eventID</b> value returned in the <b>confirmMessage</b> object within the response to confirm the processing status of the submitted bulk upload.</p> <p>The Response Code applies specifically to the POST Method request.</p>

## Sample Supervisor Edit

For API related information please refer [Data Collection Entries](#) in API explorer

<b>API:</b>	/events/time/v1/data-collection-entries.process
<b>Method:</b>	POST
<b>Condition:</b>	<p>The Payload has been accepted and stored for further processing by ADP Marketplace. Subsequent requests to the datacollection entries.process, Status service are required by providing the <b>eventID</b> value returned in the <b>confirmMessage</b> object within the response to confirm the processing status of the submitted bulk upload.</p> <p>The Response Code applies specifically to the POST Method request.</p>

## Alternate Worker ID Samples

Sometimes a consumer may want to identify the Employee Position for the time punch being uploaded using another commonly known attribute such as an ASSOCIATEOID rather than using badge.

## Using PositionID

For API related information please refer [Data Collection Entries](#) in API explorer

<b>API:</b>	/events/time/v1/data-collection-entries.process
<b>Method:</b>	POST
<b>Condition:</b>	<p>The Payload has been accepted and stored for further processing by ADP Marketplace. Subsequent requests to the datacollection entries.process, Status service are required by providing the <b>eventID</b> value returned in the <b>confirmMessage</b> object within the response to confirm the processing status of the submitted bulk upload.</p> <p>The Response Code applies specifically to the POST Method request.</p>

# Using ASSOCIATEOID

Below is a sample of the ASSOCIATEOID being used to identify the Employee Position that is associated with the Time Punch.

For API related information please refer [Data Collection Entries](#) in API explorer

<b>API:</b>	/events/time/v1/data-collection-entries.process
<b>Method:</b>	POST
<b>Condition:</b>	<p>The Payload has been accepted and stored for further processing by ADP Marketplace. Subsequent requests to the datacollection entries.process, Status service are required by providing the <b>eventID</b> value returned in the <b>confirmMessage</b> object within the response to confirm the processing status of the submitted bulk upload.</p> <p>The Response Code applies specifically to the POST Method request.</p>

# Using Work Assignment ID (PFID)

Below is a sample of the WORKASSIGNMENTID (PFID) being used to identify the Employee Position that is associated with the Time Punch.

For API related information please refer [Data Collection Entries](#) in API explorer

<b>API:</b>	/events/time/v1/data-collection-entries.process
<b>Method:</b>	POST
<b>Condition:</b>	<p>The Payload has been accepted and stored for further processing by ADP Marketplace. Subsequent requests to the datacollection entries.process, Status service are required by providing the <b>eventID</b> value returned in the <b>confirmMessage</b> object within the response to confirm the processing status of the submitted bulk upload.</p> <p>The Response Code applies specifically to the POST Method request.</p>

## Data Collection Entries API Element Glossary

Selected important elements of the payload API are defined below for user reference and information. Any elements not documented below are assumed to be non-applicable and can be ignored for purposes of the data-collection-entries HTTP POST web service method. Elements that have a "Required?" value of "N/A" are data elements that are typically found in the response of an upload request (and would not be needed in the upload request itself but are documented to indicate their importance in subsequent requests that might be made in order to obtain an upload status).

## Events[]

Element Name	Description	Required?
events[]	An array of events. Typically, only one event (events[0]) need ever be supplied. The events[] array element is the parent object of the dataCollectionEntries[] array element.	Yes

events[].eventID	The eventID is the value that can be used to reference an upload event in a subsequent request to obtain upload status. It typically is found in the response API of the initial data-collection-entries.process upload request made.	N/A
events[].serviceCategoryCode.codeValue	Should always have a value of "core" (constant value).	Yes

## dataCollectionEntries[]

The table below describes the elements of the events[], data.transform.dataCollectionEntries[] array objects.

Element Name	Description	Required?
itemID	The itemID is used to identify the dataCollectionEntries array element. It is typically an integer value that represents the indexed position of the array element being described. For example, dataCollectionEntries[0] would have a value of "0", events[1].eventID would have a value of 1, etc.	Yes
terminalName	Represents the ID of the terminal associated with the punches described by the workerDataCollectionEntries[] array (clock name).	NO

## workerDataCollectionEntries[]

The table below describes selected elements of the events[], data.transform.dataCollectionEntries[], workerDataCollectionEntries array objects. All other elements of the workerDataCollectionEntries array objects not defined here can be ignored and are not required.

Element Name	Description	Required?
entryID	The entryID is used to identify the workerDataCollectionEntries array element. It is typically an integer value that represents the indexed position of the array element being described. For example, dataCollectionEntries[0] would have a value of "0", events[1].eventID would have a value of 1, etc.	Yes
entryCode.codeValue	Used to assign the Earning or Supplemental earning code associated with an "earnings" or "suppEarnings" action code. This is a required value whenever an "earnings" or "suppEarnings" action code is specified. Not required for all other cases. See "Standard ADP Available Entry Code Values" for more information about the codes that can be used for this assignment.	Required when used with the "earnings" and "suppEarnings" action codes
badgeID	The badge value associated with the time punch being recorded.	Yes
actorBadgeID	The badge value associated with a "punch edit" made at the clock. Typically, it represents the badge value of the supervisor adding a missed punch or recording an allocation on behalf of an employee.	No
deviceDateTime	The timestamp of the data collection device for the punch being recorded. It has the following format:	

	<p>YYYY-MM-DDTHH:MI:SS-HH:MI</p> <p>For example: 2016-11-11T11:30:15-05:00</p> <p>Where "-05:00" is the hour/minutes offset from UTC.</p>	
entryDateTime	The timestamp of the Punch being recorded for the employee. When the punch is not an edit (either a Supervisor or Employee Edit operation at the clock), the entryDateTime will typically match the deviceDateTime. Uses the same date format as deviceDateTime.	Yes
actionCode	The action code associated with the punch being recorded. See "Action Code/Entry Code Description Matrix" for the set of action codes that can be used for this assignment.	Yes
alternateWokerIDs	Used to identify the Employee Position belonging to the punch by alterantive elements, such as ASSOCIATEOID. To date, only ASSOCIATEOID is presently supported. See "Alternate Worker ID's (alternateWokerIDs and alternateActorWokerIDs)" for property information.	No
alternateActorWokerIDs	Used to identify the Employee Position belonging to the Punch Edit being made (Supervisor Edit) that might belong to a Punch using an alternative element, such as an ASSOCIATEOID. See "Alternate Worker ID's (alternateWokerIDs and alternateActorWokerIDs)" for property information.	No

## Alternate Worker ID's (alternateWokerIDs and alternateActorWokerIDs)

Element Name	Description	Required?
idValue	Value for the Alternate Worker ID being supplied.	Yes
schemeCode.codeValue	Identifies the identifying element/item being supplied. At this time, only "ASSOCIATEOID" is supported.	Yes