Guide

# Payroll Data Input API Guide for ADP Workforce Now

Published on
May 11, 2020 4:54PM

Last modified
Sep 22, 2022 8:52AM

Always Designing
for People™

# ADP Copyright Information

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 inaccurancies 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 programes described in this publication.

Published on
May 11, 2020 4:54PM

Last modified
Sep 22, 2022 8:52AM

# Table of Contents

Chapter 1

**About this API**

- Summary
- What's New in this Guide?
  - September 2022
  - August 2022
  - April 2021
  - March 2022
  - December 2020
  - October 2020
  - August 2020
  - May 2020
- Recent Webinar Overview
- Process Overview
- Required Setup Steps
  - **Setup Pages in ADP Workforce Now**
- Before You Begin Using the Payroll Data Input API
  - Client Requirements
  - API User Requirements
  - Payroll Cycle Requirements
  - Employee Requirements
- Data Dictionary
  - Batch and Employee Information Data Dictionary
  - Regular and Overtime Earning Inputs Data Dictionary
  - Processing Rules
  - Coded Hours and Earnings Data Dictionary
    - Processing Rules
  - Deductions Data Dictionary
    - Processing Rules
  - Memo Data Dictionary
  - Temporary Cost Number, Temporary Department Number, and Allocation Data Dictionary
- Working with Temporary Cost Number, Temporary Department Number, and Allocations in ADP Workforce Now
  - Setting the Temporary Cost Number in ADP Workforce Now
  - Setting the Temporary Department Number in ADP Workforce Now
  - Setting the Allocations in ADP Workforce Now
  - Processing Rules

Chapter 2

Use Case: Adding Entries in a Pay Data Batch

Use Case Description

API Usage

Application Scope

Supported Actors

Sequence of Interactions

Responses

Other Scenarios

Chapter 3

Use Case: Appending Entries to an Existing Pay Data Batch

Use Case Description

API Usage

Application Scope

Supported Actors

Data Dictionary

Responses

Chapter 4

Frequently Asked Questions (FAQs)

Question 1: What's the recommended batch size to have the best performance?

Question 2: If the client created templates for pay data, would the template be used for the Payroll Data Input API?

Question 3: If a future hire is included in the Request Payload, would it get processed?

Question 4: Does the Payroll Data Input API import Paid Time Off (PTO) hours?

Question 5: Does the Payroll Data Input API import double overtime for California workers?

Question 6: Does the Payroll Data Input API support earnings, such as cash tips?

Question 7: How is piece Rate handled in pay data?

Question 8: Does vacation/sick hours entered in Paydata auto deduct time off balances in ADP Workforce Now?

Chapter 5

Known Issues and Limitations

Issue: Meta Call is not Giving any Codelist Details for ratecodes

Impacted APIs

Description

Suggested Workaround

US1719297: Update Pay Data Input API  to reflect true batch name

Impacted APIs

Description

Suggested Workaround

Issue: Event notification is not supported

Impacted APIs

Description

Suggested Workaround

Chapter 6

Appendixes

Appendix A: Pay Data Types Handled by the Payroll Data Input API

Appendix B: Pay Data Fields Not Handled by the Payroll Data Input API

Chapter 1

# About this API

**About this API**

## Summary

**ADP Workforce Now enables users to create a Pay Data batch for a payroll cycle to include the following:**

- Earnings
- Deductions
- Reimbursements
- Reportable Earnings and Benefits
- Temporary Cost Number
- Temporary Department Number
- Allocations

**The Payroll Data Input Application Programming Interface (API) enables you to build a data connector application for clients to transmit data that is sourced in your application to a specific Pay Data batch in ADP Workforce Now.**

## What's New in this Guide?

# September 2022

- Added support for 3 new pay data fields
  - Cancel Automatic Payment
  - Shift
  - Tax Frequency

# August 2022

- Added clarity regarding passing pay rates and highlighting the depreciation of 'codedHoursEarningInputs'
  - Info: All hours and earnings are supported by the 'earningInputs' object and should be the only object used to submit hours or earnings. In order to not break existing integrations that use 'codedHoursEarningInputs' the fields will be accepted but may not have the same behavior as 'earningInputs'.
  - info: If sending the rate for one earning in the array, a rate should be sent for all codes in that array. If only 1 rate is sent that rate would be used for all records on that row.

# April 2021

- Added a note point for the section Regular and Overtime Earning Inputs Data Dictionary under Chapter 1- About this API

# March 2022

- Updated Menu Access requirements to include new Payroll Dashboard.

# December 2020

- Added "ADP Workforce Now API Release Plan" in Chapter 5 - Known Issues and Limitations
- Added below information in Chapter 6-Appendix B: Pay Data Fields Not Handled by the Payroll Data Input API
  ADP client practitioner could use file based import as a work around.

# October 2020

- Added link in Chapter 1 for Video recording of an overview of this API

# August 2020

- Removed **Grid Template Setup in ADP Workforce Now** information in this chapter.
- Added the **API[/events/payroll/v1/pay-data-input.modify/meta?companyCode=94N&fileNumber=078765]** with response under the **API Usage** section in Chapter 2 - Use Case: Adding Entries in a Batch.

# May 2020

- Revised the following in this chapter:
  - Added a schema location for **Rate Code** in **Regular and Overtime Earning Inputs Data Dictionary**.
  - Added codeList APIs for **Temporary Cost Number** and **Temporary Department Number** in **Temporary Cost Number, Temporary Department Number, and Allocation Data Dictionary** section.
  - Updated the **APPEND** functionality in the Pay Data Input API Postman Collection.
- Revised the following in Chapter 2 - Use Case: Adding Entries in a Pay Data Patch:
  - Added negative test cases on rate code along with GitHub links in the **Responses** section.
  - Updated the responses for meta API in the API Usage section.
- Added Chapter 3 - Use Case: Appending Entries to an Existing Pay Data Batch
- Revised the following in Chapter 5 - Known Issues and Limitations**:**
  - Added **Issue: Meta Call is not giving any codeList details for ratecodes**.
  - Added **Issue: Event notification is not supported**.
  - Added US1719297: Update Pay Data Input API to reflect the true batch name
- Moved the following to the Appendixes from **Chapter 1 - About this API**:
  - **Appendix A: Pay Data Types Handled by the Payroll Data Input API**
  - **Appendix B: Pay Data Fields Not Handled by the Payroll Data Input API**

## Recent Webinar Overview

- You can view an overview of this API provided on October 2nd, 2020 using this link Passcode: PhD6zumk
- Payload collection used in the video overview is available at here

## Process Overview

The following table illustrates how clients use your data connector application, which is built using the Payroll Data Input API.

|  | Actor | Task Description |
|---|---|---|
| 1 | Payroll Practitioner | Selects the company code, employees, and file numbers to be imported. |
| 2 | Payroll Practitioner | Enters a batch number and initiating the Pay Data import. |
| 3 | Your application | Automatically creates a new batch and imports Pay Data in ADP Workforce Now. |
| 4 | Payroll Practitioner | Reviews the import result on the ADP Workforce Now **Paydata** page. |
| 5 | Payroll Practitioner | Takes actions to balance and adjust, as needed.<br>**Note:** To keep your data in sync with what's on the **Paydata** page in ADP Workforce Now, data must be adjusted in your data connector application. |

## Required Setup Steps

# Setup Pages in ADP Workforce Now

The Payroll Data Input API supports all earning and deduction codes, including the earning and deduction codes added by clients. Clients need to make sure the codes and numbers used by your data connector application are added and configured in ADP Workforce Now by selecting **Setup > Tools > Validation Tables > Payroll > Paydata**.

The following table lists the records in your data connector application and how they correspond to the fields on the **Paydata** page in ADP Workforce Now.

| Records in Your Data Connector Application | Fields on the Paydata Page in ADP Workforce Now |
|---|---|
| Hours and Earnings Codes | **Hours and Earning Codes** |
| Deduction Code | **Deductions** |
| Temporary Cost Number | **Cost Number** |
| Temporary Department Number | **Department** |

## Before You Begin Using the Payroll Data Input API

The following sections document the requirements for using the Payroll Data Input API.

# Client Requirements

The client must meet the following requirements:

- Must be the United States, Canada (AutoPay), or cross border clients.
- Due to a limitation, the client must have a **Memo Code** of 5 and **Hours & Earnings Code** of T created for each of their active companies. Within the ADP Workforce Now UI, select **Setup > Tools > Validation Tables > Paydata > Hours & Earnings Codes** and **Setup > Tools > Validation Tables > Paydata > Memo**. If the parameter **autoGenerateMemoCode=false** is used, **Memo Code** 5 and **Earning Code** T are not required.

# API User Requirements

The API user making the request must be assigned to a profile having the following permissions:

- ADP Workforce Now menu access to either: **Process > Payroll > Pay Data OR Process > Payroll > Payroll Dashboard**
- Payroll and People Access: View and edit access to target company codes

# Payroll Cycle Requirements

To process the API request, the payroll cycle status must be either **Entering Payroll Information** or **Correcting Input**, otherwise the request will cause an error.

# Employee Requirements

Before including any employee in the API request, make sure the employee is in an ADP Workforce Now paid position.

## Data Dictionary

# Batch and Employee Information Data Dictionary

Using the resources listed in the following table, data sent through the Payroll Data Input API eliminates the need for the user to enter batch and employee information data manually in ADP Workforce Now.

| Schema Location | Field Name | Business Rules | Notes |
|---|---|---|---|
| /eventContext/payrollGroupCode/codeValue | **Company Code** | Always Required | A client could have multiple company codes. To create a Pay Data batch, the client needs to select the company code first. |
| /eventContext/payrollProcessingJobID | **Batch ID** | | The client needs to assign the batch an ID. The ID doesn't need to be unique, as the system would append system generated characters if the ID is used by an existing batch. |

|  |  |  |  |
|---|---|---|---|
| /payDataInput/payeePayInputs/associateOID | Not Displayed |  | To retrieve the value for an employee, use GET workers. |
| /payDataInput/payeePayInputs/payrollProfilePayInputs /payrollFileNumber | **File #** |  |  |
| /payDataInput/payeePayInputs/payNumber | **Pay Number** |  | The client needs to assign a **payNumber** to be used for all entries included in the batch. |

# Regular and Overtime Earning Inputs Data Dictionary

Using the resources listed in the following table, regular and overtime earning input data sent through the Payroll Data Input API eliminates the need to enter this data manually in ADP Workforce Now.

Code **R** is reserved for **Regular Earnings** and **O** is reserved for **Overtime Earnings**.

| **Schema Location** | **Paydata Page Location** | **Notes** |
|---|---|---|
| /payDataInput/payeePayInputs/payrollProfilePayInputs /payInputs/earningInputs/earningCode/codeValue | NA | The Earning Code.<br>Use the **earning code** values which are listed under **"/payDataInput/payeePayInputs/payrollProfilePayInputs/payInputs/earningInputs/earningCode"** in the Meta call response. |
| /payDataInput/payeePayInputs/payrollProfilePayInputs/payInputs/earningInputs/rateCode | **Rate Code** |  |
| /payDataInput/payeePayInputs/payrollProfilePayInputs /payInputs/earningInputs/rate/rateValue | **Temp Rate** | The **Temp Rate** associated with the Earning Code. |
| /payDataInput/payeePayInputs/payrollProfilePayInputs /payInputs/earningInputs/rate/currencyCode | NA | The Currency Code. |
| /payDataInput/payeePayInputs/payrollProfilePayInputs /payInputs/earningInputs/numberOfHours | **Reg Hours** for code **R O/T Hours** for code **O** | The Hours associated with the Earning Code. |
| /payDataInput/payeePayInputs/payrollProfilePayInputs | **Reg Earnings** | The Amount associated with the Earning Code. |

|  |  |  |
|---|---|---|
| /payInputs/earningInputs/earningsAmount/amountValue |  |  |
| /payDataInput/payeePayInputs/payrollProfilePayInputs/payInputs/earningInputs/earningsAmount/currencyCode | NA | The Currency Code. |
| /payDataInput/payeePayInputs/payrollProfilePayInputs/payInputs/earningInputs/earnedPayPeriodWeekNumber | **FLSA WorkWeek** | Indicates whether the hours entered for employees are for week 1 or week 2. |

# Processing Rules

Based on the values included in the payload, ADP Workforce Now will process data according to business rules. ADP Workforce Now has the following out of box earning codes:

- **R = Regular Earning**
- **O = Overtime Earning**

The following table outlines various input combinations and process results.

|  | **Common Use** | **/rate/rateValue** | **/numberOfHours** | **/earningsAmount/amountValue** | **Displayed on Paydata Page When Request is Processed Successfully:** |  |  |  |  | **Notes** |
|---|---|---|---|---|---|---|---|---|---|---|
|  |  |  |  |  | **RegHours** | **O/THours** | **TempRate** | **RegEarnings** | **O/TEarnings** |  |
| 1 | Send in **Hours** for **Regular Earning**. | Null | Not Null | Null | Not Null | Null | Null | Null | Null |  |
| 2 | Send in **Hours** for **Overtime Earning**. | Null | Not Null | Null | Null | Not Null | Null | Null | Null |  |
| 3 | Send in **Temp Rate** and **Hours** for **Reg** | Not Null | Not Null | Null | Not Null | Null | Not Null | Null | Null | Used for **Hourly Employee** only |

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | **ular Earning**. | | | | | | | | | (**Pay Profile > Regular Pay Rate = Hourly**). The **Temp Rate** will be used to replace the **Regular Rate** on the employee record. |
| 4 | Send in **Temp Rate** and **Hours** for **Overtime Earning**. | Not Null | Not Null | Null | Null | Not Null | Not Null | Null | Null | Used for **Hourly Employee** only (**Pay Profile > Regular Pay Rate = Hourly**). The **Temp Rate** will be used to replace the **Overtime Rate**. You cannot submit a **Temp Rate** for **Regular Hours** and another **Temp Rate** for the **Overtime** hours for the same file number in the same batch. |
| 5 | Send in **Amount** for **Regular Earning**. | Null | Null | Not Null | Null | Null | Null | Not Null | Null | Used for **Salaried/Daily Employee** only |

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | | **(Pay Profile > Regular Pay Rate = Salary or Daily**). |
| 6 | Send in **Amount** for **Overtime Earning**. | Null | Null | Not Null | Null | Null | Null | Null | Not Null | Used for **Salaried/Daily Employee** only (**Pay Profile > Regular Pay Rate = Salary or Daily**). |
| 7 | Send in both **Hours** and **Amounts** for **Regular Earning**. | Null | Not Null | Not Null | Not Null | Null | Null | Null | Null | The **AmountValue** will be disregarded. |
| 8 | Send in both **Hours** and **Amounts** for **Overtime Earning**. | Null | Not Null | Not Null | Null | Not Null | Null | Null | Null | The **AmountValue** will be disregarded. |

# Coded Hours and Earnings Data Dictionary

Deprecating codedHoursEarningInputs

All hours and earnings are supported by the 'earningInputs' object and should be the only object used to submit hours or earnings. In order to not break existing integrations that use 'codedHoursEarningInputs' the fields will be accepted but may not have the same behavior as 'earningInputs'.

Using the resources listed in the following table, coded hours and earnings data sent through the Payroll Data Input API eliminates the need to enter this data manually in ADP Workforce Now.

| Schema Location | Field Name | Notes |
|---|---|---|
| /payDataInput/payeePayInputs/payrollProfilePayInpu | Various, depends on the Earning | The Earning Code Value. |

| | | |
|---|---|---|
| ts/payInputs/earningInputs/earningCode/codeValue | | |
| /payDataInput/payeePayInputs/payrollProfilePayInputs/payInputs/earningInputs/earningsAmount/amountValue | **Amount** | The Earning Amount Value. |
| /payDataInput/payeePayInputs/payrollProfilePayInputs/payInputs/earningInputs/earningsAmount/currencyCode | Not Displayed | The Currency Code. |
| /payDataInput/payeePayInputs/payrollProfilePayInputs/payInputs/earningInputs/rate/rateValue | **Rate** | The Rate associated with the earning.<br>Rate Amount<br>If sending the rate for one earning in the array, a rate should be sent for all codes in that array. If only 1 rate is sent that rate would be used for all records on that row. |
| /payDataInput/payeePayInputs/payrollProfilePayInputs/payInputs/earningInputs/rate/currencyCode | Not Displayed | The Currency Code. |
| /payDataInput/payeePayInputs/payrollProfilePayInputs/payInputs/earningInputs/numberOfHours | **Hours** | The Hours associated with the earning. |

## Processing Rules

Based on values included in the payload, ADP Workforce Now will process data according to business rules. The following table outlines various input combinations and processing results.

| | **Common Use** | **/rate/rateValue** | **/numberOfHours** | **/earningsAmount/amountValue** | **Displayed on Paydata PageWhen Request isProcessed Successfully** | **Notes** |
|---|---|---|---|---|---|---|
| 1 | Send in **Hours** for client configured **Hours and Earnings** Field type: **Other Hours** | Null | Not Null | Null | Data sent from **/numberOfHours** is displayed under **Other Hours - (code)**. | If data is sent from **rateValue** and **amountValue**, they are disregarded. |
| 2 | Send in **Amount** for client | Null | Null | Not Null | Data sent from **/amountValue** is displayed | If data is sent from **rateValue** and **numberOfHou** |

| | | | | | | |
|---|---|---|---|---|---|---|
| | configured **Hours and Earnings**<br>Field type: **Other Hours** | | | | under **Other Earnings - (code)**. | **rs**, they are disregarded. |
| 3 | Send in **Hours**, **Rate**, and **Amount** for client configured **Hours and Earnings**<br>Field type: **Hours/Earnings**<br>Field value: **3** | Null | Not Null | Null | Data sent from **/numberOfHours** is displayed under **Other Hours - (code)**. | If data is sent from **rateValue** and **amountValue**, they are disregarded. |
| 4 | Send in **Hours**, **Rate**, and **Amount** for client configured **Hours and Earnings**<br>Field type: **Hours/Earnings**<br>Field value: **4** | Null | Null | Not Null | Data sent from **/amountValue** is displayed under **Other Earning - (code)**. | If data is sent from **rateValue** and **numberOfHours**, they are disregarded. |

# Deductions Data Dictionary

Using the resources listed in the following table, deductions data sent through the Payroll Data Input API eliminates the need to enter this data manually in ADP Workforce Now.

| Schema Location | Field Name | Notes |
|---|---|---|
| /payDataInput/payeePayInputs/payrollProfilePayInputs/payInputs/deductionInputs/deductionCode/codeValue | Various, depends on the Deduction | The Deduction Code. |
| /payDataInput/payeePayInputs/payrollProfilePayInputs/payInputs/deductionInputs/deductionRate/rateValue | **Rate** | The Rate associated with the deduction code. |
| /payDataInput/payeePayInputs/payrollProfilePayInputs/payInputs/deductionInputs/deductionRate/currencyCode | Not Displayed | The Deduction Currency Code. |
| /payDataInput/payeePayInputs/payrollProfilePayInputs/payInputs/deductionInputs/deductionRate/baseUnitCode/codeValue | Use the value from meta | |

| | | |
|---|---|---|
| /payDataInput/payeePayInputs/payrollProfilePayInputs/payInputs/deductionInputs/configurationTags/tagCode | Use the value from meta | |
| /payDataInput/payeePayInputs/payrollProfilePayInputs/payInputs/deductionInputs/configurationTags/tagType/dataTypeCode | Use the value from meta | |
| /payDataInput/payeePayInputs/payrollProfilePayInputs/payInputs/deductionInputs/configurationTags/tagValue | Use the value from meta | |

## Processing Rules

The value in **Rate** must be part of the payload to post a deduction adjustment or deduction entry.

# Memo Data Dictionary

Using the resources listed in the following table, memo data sent through the Payroll Data Input API eliminates the need to enter this data manually in ADP Workforce Now.

| Schema Location | Field Name | Notes |
|---|---|---|
| /payDataInput/payeePayInputs/payrollProfilePayInputs/payInputs/memoInputs | Various, depends on the Memo. | The memo-related entries. |
| /payDataInput/payeePayInputs/payrollProfilePayInputs/payInputs/memoCode | **Memo** | |
| /payDataInput/payeePayInputs/payrollProfilePayInputs/payInputs/memoCode/codeValue | **Code** | Use the value from the meta. |
| /payDataInput/payeePayInputs/payrollProfilePayInputs/payInputs/memoInputs/memoAmount | NA | |

| | | |
|---|---|---|
| /payDataInput/payeePayInputs/payrollProfilePayInputs/payInputs/memoInputs/memoAmount/amountValue | **Amount** | |
| /payDataInput/payeePayInputs/payrollProfilePayInputs/payInputs/memoInputs/memoAmount/currencyCode | NA | |

# Temporary Cost Number, Temporary Department Number, and Allocation Data Dictionary

Using the resources listed in the following table, your data connector application can post Temporary Cost Number, Temporary Department Number, and Allocation position details through the Payroll Data Input API. This eliminates the need to enter this data manually in ADP Workforce Now using the **Paydata** page.

| POST | Schema Location | Field Name | Notes |
|---|---|---|---|
| Temporary Cost Number | /payDataInput/payeePayInputs/payrollProfilePayInputs/payInputs/payAllocation | Not Displayed | |
| Temporary Cost Number | /payDataInput/payeePayInputs/payrollProfilePayInputs/payInputs/payAllocation/allocationID | Not Displayed | **allocationID:2** – Indicates the Cost Number |
| Temporary Cost Number | /payDataInput/payeePayInputs/payrollProfilePayInputs/payInputs/payAllocation/allocationTypeCode | Not Displayed | |
| Temporary Cost Number | /payDataInput/payeePayInputs/payrollProfilePayInputs/payInputs/payAllocation/allocationTypeCode/codeValue | **Temporary Cost Number** | Use codeList API to get values<br>**API: /codelists/hr/v3/worker-management/cost-numbers/WFN/1** |
| Temporary Cost Number | /payDataInput/payeePayInputs/payrollProfilePayInputs/payInputs/payAllocation/allocationTypeCode/shortName | Not Displayed | |
| Temporary Department Number | /payDataInput/payeePayInputs/payrollProfilePayInputs/payInputs/payAllocation | Not Displayed | |
| Temporary Department Number | /payDataInput/payeePayInputs/payrollProfilePayInputs/payInputs/payAllocation/allocationID | Not Displayed | **allocationID:3** – Indicates the Department Number. |
| Temporary Department Number | /payDataInput/payeePayInputs/payrollProfilePayInputs/payInputs/payAllocation/allocationTypeCode | Not Displayed | |
| Temporary Department Number | /payDataInput/payeePayInputs/payrollProfilePayInputs/payInputs/payAllocation/allocationTypeCode/codeValue | **Temporary Department** | Use codelist API to get values<br>**API: /codelists/hr/v3/worker-management/departments/WFN/1** |

| | | | |
|---|---|---|---|
| Temporary Department Number | /payDataInput/payeePayInputs/payrollProfilePayInputs/payInputs/payAllocation/allocationTypeCode/shortName | Not Displayed | |
| Allocation (ALA) | /payDataInput/payeePayInputs/payrollProfilePayInputs/payInputs/payAllocation | Not Displayed | |
| Allocation (ALA) | /payDataInput/payeePayInputs/payrollProfilePayInputs/payInputs/payAllocation/allocationID | Not Displayed | **allocationID:1** – Indicates the Allocations |
| Allocation (ALA) | /payDataInput/payeePayInputs/payrollProfilePayInputs/payInputs/payAllocation/allocationTypeCode | Not Displayed | |
| Allocation (ALA) | /payDataInput/payeePayInputs/payrollProfilePayInputs/payInputs/payAllocation/allocationTypeCode/codeValue | **ALA Position** | |
| Allocation (ALA) | /payDataInput/payeePayInputs/payrollProfilePayInputs/payInputs/payAllocation/allocationTypeCode/shortName | Not Displayed | |

## Working with Temporary Cost Number, Temporary Department Number, and Allocations in ADP Workforce Now

Using the components listed in the table above, sending Temporary Cost Number, Temporary Department Number, and Allocations using the Payroll Data Input API achieves the same result as if the user follows the steps in the following sections within ADP Workforce Now.

# Setting the Temporary Cost Number in ADP Workforce Now

1. On the **Paydata** page, select **Insert > Column**.
2. On the **Insert Column** window, select the **Paydata** list. Select the **Temporary Cost Number** position.
3. Click **Done**.
4. Add a column. Under the new column, enter a value for an employee.

The **allocationID:2** value in the payload indicates the **Temporary Cost Number**. To create the batch with **Temporary Cost Number**, ADP Workforce Now should have the **Labor Distribution** and **Allocations** set to **Cost Number** using the following steps:

1. Select **Setup > Payroll > Company Options**. Select the company code and click **Labor Distribution**.
2. Set the **Labor Distribution** to **Cost Number**.
3. Select **Setup > Payroll > Company Options.** Select the company code and click **Allocations**.
4. Set **Allocations** to **Cost Number**.

# Setting the Temporary Department Number in ADP Workforce Now

1. On the **Paydata page**, select **Insert > Column.**
2. On the **Insert Column** window, select the **Paydata** list. Select the **Temporary Department Number** position.

3. Click **Done**.
4. Add a column. Under the new column, enter a value for an employee.

The **allocationID:3** value in the payload indicates the **Temporary Department Number**. To create the batch with **Temporary Department Number**, ADP Workforce Now should have the **Labor Distribution** and **Allocations** set to **Department Number** using the following steps:

1. Select **Setup > Payroll > Company Options**. Select the company code and click **Labor Distribution**.
2. Select **Setup > Payroll > Company Options**. Select the company code.
3. Click **Allocations**.

## Setting the Allocations in ADP Workforce Now

1. On the **Paydata** page, select **Insert > Column**.
2. On the **Insert Column** window, select the **Paydata** list. Select **Allocations Position Number**.
3. Click **Done**.
4. Add a column. Under the new column, enter a value for an employee.

The **allocationID:1** value in the payload indicates **Allocation** positions for **Cost Number** and **Department Number**.

1. Select **Setup > Payroll > Company Options**.
2. Select either **Cost Number** or **Department** as the Allocation method your company uses to allocate employee hours and earnings.
3. Select the **Automatic Allocation by Percentage** to allow your company to distribute an employee's hours and earnings to multiple departments or cost numbers on a regular basis, by percentage. This should total 100 percent.
4. Select **Manual Posting of Allocation in Paydata** to allow your company to distribute an employee's hours and earnings to multiple departments or cost numbers through postings in paydata.

The following conditions for allocations in the payload exist based on your selections in steps 3 and 4:

- When both **Automatic Allocation by Percentage** and **Manual Posting of Allocation** are checked, it takes values from **1-99** and **""**.
- When only **Automatic Allocation by Percentage** is checked, it takes the values as **""** and **1**.

When only **Manual Posting of Allocation** is checked, it takes values from **2-99** and **""**.

When using the **Allocation** positions, the employee should have allocations set up under **People > Employment > Employment Profile > Corporate Groups > Allocations** and either use **Temporary Cost Number** or **Temporary Department Number**.

## Processing Rules

Based on values included in the payload, ADP Workforce Now processes data according to business rules. The following table outlines the various input combinations and processing results.

| | **Common Use** | **/payAllocation/allocationID** | **/payAllocation/allocationTypeCode/codeValue** | **Displayed on the Paydata Page When Request is Processed Successfully:** | **Notes** |
|---|---|---|---|---|---|

| | Common Use | /payAllocation/allocationID | /payAllocation/allocationTypeCode/codeValue | Displayed on the Paydata Page When Request is Processed Successfully: | Notes |
|---|---|---|---|---|---|
| 1 | Send in **Temporary Cost Number**. | **2** | **001000** (Any value which is set up in the Validation table for **Cost Number**) | Data sent from the following is displayed under **Temporary Cost Number**: /allocationTypeCode / codeValue. | The values sent in **allocationTypeCode** should be set up in the Validation table for **Cost Number** before being used. Otherwise, the result is an error. |
| 2 | Send in **Temporary Department Number**. | **3** | **002000** (Any value which is set up in the Validation table for **Department**) | Data sent from the following is displayed under **Temporary Department Number**: /allocationTypeCode / codeValue. | The values sent in **allocationTypeCode** should be set up in the Validation table for **Department** before used. Otherwise, the result is an error. |
| 3 | Send in **Allocation** for **Cost Number** with both **Automatic** and **Manual** set up in **Company Options**. | **1** | **"" and 1-99** | Data sent from /allocationTypeCode /codeValue is displayed under **ALA Position**. | When **Automatic** and Manual is set up, it takes values as **"" and 1-99**. |
| 4 | Send in **Allocation** for **Department Number** with both **Automatic** and **Manual** set up in **Company Options**. | | | | |
| 5 | Send in **Allocation** for **Cost Number** with only **Automatic** set up in **Company Options**. | | **"" and 1** | | When **Automatic** is set up, it takes values as **"" and 1**. |
| 6 | Send in **Allocation** for **Department Number** with only **Automatic** set up in **Company Options**. | | | | |
| 7 | Send in **Allocation** for **Cost Number** with only **Manual** set up in **Company Options**. | **1** | **"" and 2-99** | Data sent from /allocationTypeCode /codeValue is displayed under **ALA Position**. | When **Manual** is set up, it takes values as **"" and 2-99**. |
| 8 | Send in **Allocation** for **Department Number** with only **Manual** set up in **Company Options**. | | | | |

Chapter 2

# Use Case: Adding Entries in a Pay Data Batch

## Use Case Description

This use case sends entries to a Pay Data batch. It is commonly used to send the following data from a different system or application to the ADP Workforce Now **Paydata** page:

- Earnings
- Deductions
- Reimbursement
- Reportable Earnings and Benefits
- Temporary Cost Number
- Temporary Department Number
- Allocations

## API Usage

For API related information please refer Pay Data Input under API Explorer.

| Method | Uniform Resource Identifier (URI) | Description |
|---|---|---|
| GET | /events/payroll/v1/pay-data-input.modify/meta | Returns an event metadata.<br>**Note:** Sample payload can be built from the meta call. |
| GET | /events/payroll/v1/pay-data-input.modify/meta?companyCode=94N&fileNumber=078765 | Returns a response based on the applied filter<br>Filter<br>The following filter can be used for meta API call to retrieve the codes like deductions, earnings etc based on company code |
| POST | /events/payroll/v1/pay-data-input.modify | Adds a new pay data batch. |

In the given payload (Single Associate Request, Multiple Associate Request), the following is the purpose of the EventContext and Transform sections:

- **eventContext**: A set of keys, identifying the subject. In the payload, the associateID field is present under eventContext. The associateID identifies the subject.
- **transform**: Provides the values added or changed with respect to the subject keys defined in the eventContext section.

## Application Scope

The canonical URI corresponding to the Payroll Data Input API needs to be added in the Consumer Application Registry (CAR) for the subscription following which a user can access this API and make successful API calls.

The following canonical needs to be added to your application scope to enable this use case:

**/payroll/payrollManagement/payrollProcessing/payDataInputManagement/pay-data-input.modify**

## Supported Actors

| Request Parameter roleCode Value | Usage |
|---|---|
| practitioner | System user that sends pay data for a pay data batch. |

## Sequence of Interactions

1. Your consumer application makes a request for /events/payroll/v1/pay-data-input.modify/meta to the ADP API endpoint.
2. The ADP API endpoint responds to your consumer application with the meta payload. See the table in the following section.
3. Your consumer application processes the meta payload to validate data included in the payload in step 4 and prompts the user to fix any validation issues. Otherwise, compose the payload for the /events/payroll/v1/pay-data-input.modify request.
4. Your consumer application makes a /events/payroll/v1/pay-data-input.modify request to the ADP API endpoint.
5. The ADP API endpoint responds to the consumer application concerning the details of /events/payroll/v1/pay-data-input.modify.

## Responses

You may encounter exceptions outside your common success scenarios. You must account for these exceptions during your initial development.

For more information, see API Common Exceptions and Tips for Handling.

For API related information please refer Pay Data Input under API Explorer.

| Response Code | Condition | messageText | Tips to Handle |
|---|---|---|---|
| 200 OK | Request processes successfully. | | |
| 400 Bad Request | If the payroll cycle status is not one of the following:<br>• **Entering Payroll Information**<br>• **Correcting Input** | "**messageText**": "eventContext/ payrollGroupCode is invalid. eventContext/ payrollGroupCode | Check the **Payroll Cycle** value for the company code, which is currently used in the request. |
| 400 Bad Request | An invalid company code is passed in the request. | "**messageText**": "eventContext/payrollGroupCode is invalid. eventContext/ payrollGroupCode" | Check if the input data is the value. |
| 400 Bad Request | An invalid Associate Organization ID (AOID) exists. This is the AOID associated with the file number passed in the request. | "**messageText**": "Label not found, missing following key: access.invalid associateOid: G3E82PE9GT1Q8VENN" | Check if the employee is still active and eligible for payroll processing. |
| 400 Bad Request | An Invalid **payrollFileNumber** is passed in the request. | "**messageText**": "payrollFileNumber:8118799 is invalid. payrollFileNumber:8118799" | Check if the employee is still active and eligible for payroll processing. |
| 400 Bad Request | When an invalid **payNumber** is passed in the request. | "**messageText**": "payNumber:100 is invalid. payNumber:100" | Check and correct pay number value. |

| | | | |
|---|---|---|---|
| 400 Bad Request | The **hoursEarningCode** is not present in the Validation table. | "**messageText**": "codedHoursEarning: VV is invalid. codedHoursEarning: VV" | Your data connector application should validate that the **hoursEarningCode** is set up for the **Codedhours** in the Validation table. |
| 400 Bad Request | The deduction code is not present in the Validation table. | "**messageText**": "payrollProfilePay Inputs/payInputs /deductionInputs /deductionCode /codeValue is invalid. PayrollProfilePay Inputs/payInputs /deductionInputs /deductionCode /codeValue DD" | Your data connector application should validate that the deduction code is set up for the specific deduction in the Validation table. |
| 400 Bad Request | Request for single **Associate** with **Amount** and **Inactive Memo** code value. | "**messageText**": "Label not found, missing following key: Memo codeValue does not exist in Validation Table, or MemoCode is not an active memo code in the Company G" | |
| 400 Bad Request | Request for single **Associate** with **Amount** and **Memo** code value as empty. | "**messageText**": "Label not found, missing following key: Memo codeValue does not exist in Validation Table, or MemoCode is not an active memo code in the Company " | |
| 400 Bad Request | Request for single **Associate** with **Amount** and **Invalid Memo** code value. | "**messageText**": "Label not found, missing following key: Memo codeValue does not exist in Validation Table, or MemoCode is not an active memo code in the Company aaa" | |
| 400 Bad Request | Request for single **Associate Memo** code and invalid amount. | "**messageText**": "Label not found, missing following key: Memo amountValue - The value must be between -99,999,999.99 and 99,999,999.99. 6000000000000" | |
| 400 Bad Request | Request for a **Single Associate** with **Single Pay Number** and **Temporary Cost Number** when **costNumber** is not present in the Validation table. | | Your data connector application should validate that the **Cost Number** is set up for the specific **Cost Number** in the Validation table. |
| 400 Bad Request | Request for a **Single Associate** with **costNumber** when the **Company** set up options includes a department number. | "messageText": "Provide correct message for the key - Error creating paydataGrid column Temporary Cost Number" | Your data connector application should validate that the **Cost Number** is set up for both **Labor Distribution** and **Allocations** in ADP Workforce Now by selecting **Setup > Payroll > Company Options**. |
| 400 Bad Request | Request for a **Single Associate** with **Temporary Cost Number** and **codeValue=""**. | "**messageText**": "Label not found, missing following key: payAllocation/allocationTypeCode/code Value cannot be empty or null" | Check and pass the valid **Value** for **Temporary Cost Number**. |
| 400 Bad Request | Request for a **Single Associate** with **Temporary Cost Number** when **codeValue** for **costNumber** is more than 50. | "**messageText**": "Label not found, missing following key: cost number cannot be more than 50 " | **Temporary Cost Number** must be 50 characters. |
| 400 Bad Request | Request for a **Single Associate** with **Temporary Department** and **codeValue=""**. | "**messageText**": "Label not found, missing following key: payAllocation/allocationTypeCode/code Value cannot be empty or null " | Check and pass the valid value. |
| 400 Bad Request | Request for a **Single Associate** with **Temporary Department** | "**messageText**": "Label not found, missing following key: Allocation code is | Check and pass the valid value for **Temporary Department Number**. |

| | | | |
|---|---|---|---|
| | **Number** when **Department Number** is not present in the Validation table. | invalid " | |
| 400 Bad Request | Request for a **Single Associate** with **Department Number** when **Company** set up options includes a **Cost Number**. | "**messageText**": "Label not found, missing following key: Error creating paydataGrid column Temporary Department 3" | Your data connector application should validate that **Department Number** is set up for both **Labor Distribution** and **Allocations** in ADP Workforce Now by selecting **Setup > Payroll > Company Options**. |
| 400 Bad Request | Request for a **Single Associate** and **Single Pay Number** with **Allocation Number** when **Company** set up options are set to **enable[Automatic and Manual]** for **costNumber**, and:<br>• Allocations are in **employment profile > allocations**<br>• **codeValue=""** | "**messageText**": "Label not found, missing following key: payAllocation/allocationTypeCode/code Value cannot be empty or null " | |
| 400 Bad Request | Request for a **Single Associate** and **Single Pay Number** with **Allocation Number** when **Company** set up options are set to **enable[Manual]** for **costNumber**, and:<br>• Allocations are in **employment profile > allocations**<br>• **codeValue=1** | "**messageText**": "Label not found, missing following key: Allocation code is cannot be 1 if Automatic Allocation by Percentage not checked code: 1" | |
| 400 Bad Request | Request for a **Single Associate** and **Single Pay Number** with **Allocation Number** when **Company** set up options are set to **enable[Manual]** for **costNumber**, and:<br>]]></ac:plain-text-body></ac:structured-macro><br>• Allocations are in **employment profile > allocations**<br>• **codeValue=""** | "**messageText**": "Label not found, missing following key: payAllocation/allocationTypeCode/code Value cannot be empty or null" | |
| 400 Bad Request | Request for a **Single Associate** and **Multiple Pay Number** with **Allocation Number** when **Company** set up options are set to **enable[Manual]** for **costNumber**, and:<br>• Allocations are in **employment profile > allocations- add > allocation position**<br>• **codeValue=1 and 1** | "**messageText**": "Label not found, missing following key: Allocation code is cannot be 1 if Automatic Allocation by Percentage not checked code: 1" | |

|  |  |  |  |
|---|---|---|---|
| 400 Bad Request | Request for a **Single Associate** and **Multiple Pay Number** with **Allocation Number** when **Company** set up options are set up to **enable[Automatic]** for **costNumber**, and:<br>• Allocations are in **employment profile > allocations- add > allocation position**<br>• **codeValue="" and ""** | "**messageText**": "Label not found, missing following key: payAllocation/allocationTypeCode/code Value cannot be empty or null" |  |
| 400 Bad Request | Request for a **Single Associate** and **Multiple Pay Number** with **Allocation Number** when **Company** set up options are set. | "**messageText**": "Label not found, missing following key: payAllocation/ allocationType Code/codeValue cannot be empty or null" |  |
| 400 Bad Request | Request for a **Single Associate** and **Single Pay Number** with **Allocation Number When:** Company set up options are set to **enable[Automatic]** for **costNumber**, and:<br>• Allocations are in **employment profile > allocations- add > allocation position**<br>• **codeValue=""** | "**messageText**": "Label not found, missing following key: payAllocation /allocationType Code/codeValue cannot be empty or null" |  |
| 400 Bad Request | Request for a **Single Associate** and **Multiple Pay Number** with **Allocation Number** when **Company** set up options are set to **enable[Automatic]** for **Department Number**.<br>Allocations are in **employment profile > allocations- add > allocation position** is **3 and 10[allocation position not there in employment profile]**. | "**messageText**": "Label not found, missing following key: Allocation code is cannot be 2-99 if Manual Posting of Allocation in Paydata not checked code: 10" |  |
| 400 Bad Request | Request for a **Single Associate** and **Multiple Pay Number** with **Allocation Number** when **Company** set up options are set to **enable[Automatic]** for **Department Number**, and:<br>• Allocations in **employment profile > allocations- add > allocation position**<br>• **codeValue="" and ""** | "**messageText**": "Label not found, missing following key: payAllocation/allocationTypeCode/code Value cannot be empty or null" |  |
| 400 Bad Request | Request for a **Single Associate** and **Multiple Pay** | "**messageText**": "Label not found, missing following key: payAllocation/ |  |

|  |  |  |  |
|---|---|---|---|
|  | **Number** with **Allocation Number** when **Company** set up options are set to **enable[Automatic]** for **Department Number**, and **codeValue="" and "1"**. | allocationType Code/codeValue cannot be empty or null" |  |
| 400 Bad Request | Request for a **Single Associate** and **Multiple Pay Number** with **Allocation Number** when **Company** set up options are set to **enable[Automatic]** for **Department Number**, and:<br>- Allocations are in **employment profile > allocations- add > allocation position**<br>- **codeValue="4" and "3"** | **"messageText"**: "Label not found, missing following key: Allocation code is cannot be 2-99 if Manual Posting of Allocation in Paydata not checked code: 4" |  |
| 400 Bad Request | Request for a **Single Associate** with **Allocation Number** when **Company** set up options are set to **enable[Automatic and Manual]** for **DepartmentNumber**, and: ]]></ac:plain-text-body></ac:structured-macro><br>- Allocations are set to **employment profile > allocations**<br>- **codeValue=""** | **"messageText"**: "Label not found, missing following key: payAllocation/allocationTypeCode/codeValue cannot be empty or null" |  |
| 400 Bad Request | Request for a **Single Associate** with **Allocation Number** when **Company** set up options are set to **enable[Manual]** for **DepartmentNumber**, and:<br>- Allocations are in **employment profile > allocations- add > allocation position**<br>- **codeValue=1** | **"messageText"**: "Label not found, missing following key: Allocation code is cannot be 1 if Automatic Allocation by Percentage not checked code: 1" |  |
| 400 Bad Request | Request for a **Single Associate** with **Allocation Number** when **Company** set up options are set to **enable[ Manual]** for **DepartmentNumber**, and:<br>- Allocations are in **employment profile > allocations- add > allocation position**<br>- **codeValue=""** | **"messageText"**: "Label not found, missing following key: payAllocation/ allocationType Code/codeValue cannot be empty or null" |  |
| 400 Bad Request | Request for a **Single Associate** with **Allocation Number** when **Company** set up options are set to **enable[ Manual]** for **DepartmentNumber**. Allocations are in **employment profile >** | **"messageText"**: "Label not found, missing following key: Allocation code is invalid code: 5" |  |

| | | | |
|---|---|---|---|
| | **allocations- add > allocation position** is **"5"[which is not there in allocation]** | | |
| 400 Bad Request | Request for a **Single Position-Multiple PayNumber** with **Allocation Number** when **Company** set up options has **enable[ Manual]** for **DepartmentNumber**, and: ]]></ac:plain-text-body></ac:structured-macro><br>• Allocations are in **employment profile-> allocations**<br>• **codeValue="1"** and **1** | **"messageText"**: "Label not found, missing following key: Allocation code is cannot be 1 if Automatic Allocation by Percentage not checked code: 1" | |
| 400 Bad Request | When adding a batch with multiple employees and if any error exists in the payload | "resourceMessageInfoList": [ { "resourceMessageID": null, "processMessageInfoList": [ { "processMessageID": null, "messageTypeCode": "error", "messageText": "payrollFileNumber:81187 is invalid. payrollFileNumber:81187" } ] }, { "resourceMessageID": null, "processMessageInfoList": [ { "processMessageID": null, "messageTypeCode": "error", "messageText": "Label not found, missing following key: access.invalid associateOid: G3E82PE9GT1Q8VE" } ] } ], | In this case, if there are any errors in the payload, the API fails and batch will not create. |
| 400 Bad Request | Add a pay data with incorrect Rate code | **"messageText"**: "earningInputs/rateCode/codeValue is invalid. earningInputs/rateCode/codeValue" | |
| 400 Bad Request | Add a pay data with both rate code and rate value | "**messageText**": "Label not found, missing following key: A paydata transaction may contain either temporary rate or temporary rate code. " | |

## Other Scenarios

The following is a list of common scenarios that you may want to consider for your application.

**Note:** The sample responses listed all have the response code of 200.

For API related information please refer Pay Data Input under API Explorer.

| **Scenario Description** | **Notes** |
|---|---|
| **Scenario Description** | **Notes** |
| A batch already exists. | When a batch with the same Batch ID is requested, a new batch will be created with a system generated Batch ID (characters append to the same Batch ID). |

| | |
|---|---|
| Send in **Hours** for client configured **Hours and Earnings** Earning Code Type: **Other Hours** | |
| Send in **Hours**, **Rate**, and **Amount** for client configured **Hours and Earnings** Earning Code Type: **Hours/Earnings** Field Value: **3** | |
| Send in **Hours**, **Rate**, and **Amount** for client configured **Hours and Earnings** Earning Code Type: **Hours/Earnings** Field Value: **4** | |
| Send in **Hours** for **Regular Earning** | |
| Send in **Hours** for **Overtime Earning** | |
| Send in **Temp Rate** and **Hours** for **Regular Earning** | |
| Send in **Temp Rate** and **Hours** for **Overtime Earning** | |
| Send in **Amount** for **Regular Earning** | |
| Send in **Amount** for **Overtime Earning** | |
| Send in both **Hours** and **Amounts** for **Regular Earning** | |
| Send in both **Hours** and **Amounts** for **Overtime Earning**. | |
| Request with the same code for multiple associates for codedhoursEarnings Inputs.<br>**Note: DT** is an Hours/Earnings type. | |
| Request with different codes for | |

| | |
|---|---|
| multiple associates for codedhoursEarningsInputs. | |
| Request with multiple associates with different earning code, deduction, and codedhoursEarnings. | |
| Request with multiple associates with the same deduction codes, earning code, and different codedhoursEarnings. | |
| Request for multiple associates with different deduction codes, earning code, and different codedhoursEarnings. | |
| Request for a single associate with earning code, deduction, and codedhoursEarnings. | |
| Request for a single associate with hours for client configured Hours and Earnings Earning Code Type/Field type: **OH** is an Hours type | |
| Request for a single associate with Amount for client configured Hours and Earnings Earning Code Type/Field Type: **Other Earnings** (**N** is an Other Earnings type) | |
| Request for a single associate with hours for client configured Hours and Earning Earning Code Type/Field Type: **Hours/Earnings** | |
| Request for a single associate with **Amount** for client configured **Hours and Earnings** Earning Code Type/Field Type: **Hours/Earnings** | |
| Request with the same code for multiple | |

| | |
|---|---|
| associates for codedHoursEarningInputs for the Hours/Earnings type with the same deduction codes.<br>**Note: DT** is an Hours/Earnings type. | |
| Request with the same code for multiple associates for codedHoursEarningInputs for the Other Earnings type with the same deduction codes.<br>**Note: N** is an Earnings type. | |
| Request with the same code for multiple associates for codedHoursEarningInputs for the Other Hours type with the same deduction codes.<br>**Note: OH** is an Other Hours type. | |
| Request with the same code for multiple associates for codedHoursEarningInputs for the Hours/Earnings type with different deduction codes.<br>**Note: DT** is an Hours/Earnings type. | |
| Request with the same code for multiple associates for codedHoursEarningInputs for the Other Earnings type with different deduction codes.<br>**Note: N** is an Earnings type. | |
| Request with the same code for multiple associates for codedHoursEarningInputs for the Other Hours type with different deduction codes.<br>**Note: OH** is an Other Hours type. | |
| Request with different codes for multiple associates for codedHoursEarningInputs. | |

| | |
|---|---|
| • **DT** – Hours /Earnings type<br>• **OH** – Other Hours type<br>• **N** – Earnings type | |
| Request for a single associate with same Amount and different Memo code value. | |
| Request for a single Associate with different Amount and same Memo code value. | |
| Request for a single associate with Temporary Cost Number and code value. | |
| Request for a single associate with Temporary Cost Number and code value, which is in the **Inactive** status. | |
| Request for a single associate with the following:<br>• Multiple Pay Number<br>• Temporary Cost Number<br>• codeValue is **001000**<br>• **AllocationID:2** | |
| Request for multiple associates with the following:<br>• Multiple Pay Number<br>• Temporary Cost Number<br>• codeValue<br>• allocationID | |

| | |
|---|---|
| Request for single associate with the following:<br>- Temporary Department<br>- codeValue | |
| Request for a single associate with the following:<br>- Temporary Department Number<br>- codeValue, which is in the **Inactive** status | |
| Request for a single associate with the following:<br>- Multiple Pay Number<br>- Department Number<br>- codeValue is **001000**<br>- **allocation ID:3** | |
| Request for multiple associates with the following:<br>- Multiple Pay Number<br>- Temporary Department Number<br>- codeValue<br>- allocationID is **3** | |
| Request for a single associate and **Multiple Pay Number** with **Allocation Number** when **Company** set up options are set to **enable[Automatic and Manual]** for **costNumber**, and:<br>- Allocations are | |

| | |
|---|---|
| in **employment profile > allocations** with department and percentage<br>• **codeValue=3 and 4** | |
| Request for a single associate with **Single Pay Number** and **Allocation Number** when **Company** set up options are **enable[Automatic and Manual]** for **costNumber**, and:<br>• Allocations are in **employment profile-> allocations**<br>• **codeValue=3** | |
| Request for a single associate and **Single Pay Number** with **Allocation Number** when **Company** set up options are set to **enable[Automatic and Manual]** for **costNumber**, and:<br>• Allocations are in **employment profile > allocations**<br>• **codeValue=1** | |
| Request for a **Single Associate**, **Single Pay Number**, and **Allocation Number** when **Company** set up options are set | |

| | |
|---|---|
| to **enable[Manual]** for **costNumber**, and:<br>- Allocations are in **employment profile > allocations**<br>- **codeValue=4** | |
| Request for a **Single Associate**, **Single Pay Number**, and **Allocation Number** when **Company** set up options are set to **enable[Manual]** for **costNumber**, and:<br>- Allocations are in **employment profile > allocations**<br>- **codeValue=10** | |
| Request for a **Single Associate** and **Multiple Pay Number** with **Allocation Number** when **Company** set up options are set to **enable[Manual]** for **costNumber**, and:<br>- Allocations are in **employment profile > allocations** with department and percentage<br>- **codeValue=5 and 10** | |
| Request for a **Single Associate** and **Multiple Pay Number** with **Allocati** | |

| | |
|---|---|
| on Number when **Company** set up options are set to **enable[Automatic]** for **costNumber**, and:<br>• Allocations are in **employment profile > allocations**<br>• **codeValue=1 and 1** | |
| Request for a **Single Associate** and **Single Pay Number** with **Allocation Number** when **Company** set up options are set to **enable[ Automatic ]** for costNumber and **codeValue="1"**. | |
| Request for a single associate and **Multiple Pay Number** with **Allocation Number** when **Company** set up options are set to **enable[Automatic and Manual]** for **costNumber**, and:<br>• Allocations are in **employment profile > allocations** with department and percentage<br>• **codeValue=3 and 4** | |
| Request for a single associate with **Single Pay Number** and **Allocation Number** when **Company** set up options are **enable[Automatic and Manual]** for **costNumber**, and: | |

| | |
|---|---|
| • Allocations are in **employment profile-> allocations**<br>• **codeValue=3** | |
| Request for a **Single Associate** and **Multiple Pay Number** with **Allocation Number** when **Company** set up options are set to **enable[Automatic]** for **DepartmentNumber** and **codeValue="1" and "1"**. | |
| Request for a **Single Associate** with **Allocation Number** when Company set up options are set to **enable[Automatic and Manual]** for **DepartmentNumber** and **codeValue="1"**. | |
| Request for a **Single Associate** with **Allocation Number** when **Company** set up options are set to **enable[Manual]** for **DepartmentNumber**, and:<br>• Allocations are in **employment profile > allocations**<br>• **codeValue="10"** | |
| Request for a **Single Associate** with **Allocation Number** when **Company** set up options are set up to **enable[Automatic and Manual]** for **DepartmentNumber**, and:<br>• Allocations are | |

| | |
|---|---|
| in **employment profile > allocations**<br>- **codeValue="3"** | |
| Add paydata with Rate code | |

Chapter 3

# Use Case: Appending Entries to an Existing Pay Data Batch

## Use Case Description

This use case **appends** entries to an existing Pay Data batch. It is commonly used to append the following data from a different system or application to the ADP Workforce Now **Paydata** page:

- Earnings
- Deductions
- Reimbursement
- Reportable Earnings and Benefits
- Temporary Cost Number
- Temporary Department Number
- Allocations
- Memo's

## API Usage

For API related information please refer Pay Data Input under API Explorer.

| **Method** | **Uniform Resource Identifier (URI)** | **Description** |
|---|---|---|
| GET | /events/payroll/v1/pay-data-input.modify/meta | Returns an event metadata.<br>**Note:** Sample payload can be built from the meta call. |
| POST | /events/payroll/v1/pay-data-input.modify | Appends entries to an existing pay data batch. |

**Important:**

**Functionality of APPEND:**

When performing an APPEND to an existing pay data batch, we need to include **"_modificationTypeCode": "Append"** in the request payload. Refer to the Github Sample Request Payload: payrollProcessingJobID[BatchID].

**Functionality of the APPEND:**

- Note, that the batch is created with the batch name*[provided in request]* appending random numeric values ***[Example: Batch_654321] in the UI.***
  IF payrollProcessingJobID = 'Batch_654321' Then it will **append** to Batch_654321

- When there are multiple payrollProcessingJobID(i.e Batch ID) with the same name.The Append functionality is performed to the latest batch that is added.
- **Example**:
  - Batch_123456
  - Batch_654321
  - Batch_987654

If payrollProcessingJobID = 'Batch' Then it will append to Batch_987654

## Application Scope

The following canonical needs to be added to your application scope to enable this use case:

**/payroll/payrollManagement/payrollProcessing/payDataInputManagement/pay-data-input.modify**

## Supported Actors

| **Request Parameter roleCode Value** | **Usage** |
|---|---|
| practitioner | Appends the entries to an existing pay data batch as a practitioner. |

## Data Dictionary

Refer to Chapter 1 - About this API, under the **Temporary Cost Number, Temporary Department Number, and Allocation Data Dictionary** section. Need to include the following request payload for the **APPEND** functionality.

| **Schema Location** | **Field Name in WFN UI** | **Notes** |
|---|---|---|
| /payDataInput/payeePayInputs/payrollProfilePayInputs/payInputs/_modificationTypeCode | Not Displayed | **"_modificationTypeCode": "Append"** indicates the **APPEND** functionality of the API, which appends entries to the existing batch. |

## Responses

You may encounter exceptions outside your common success scenarios. You must account for these exceptions during your initial development.

For more information, see API Common Exceptions and Tips for Handling.

For API related information please refer Pay Data Input under API Explorer.

| **Response Code** | **Condition** | **messageText** | **Tips to Handle** |
|---|---|---|---|

| | | | |
|---|---|---|---|
| 200 OK | Append the paydata entries for an existing paydata batch. | NA | |
| 200 OK | Append the paydata batch with coded hrs and earning. | NA | |
| 200 OK | Append the paydata batch with rate codes for multiple associates. | NA | |
| 200 OK | Append the paydata batch with cost number. | NA | |
| 200 OK | Append the paydata batch with Department number. | NA | |
| 400 Bad Request | Append the paydata batch with invalid **payrollProcessingJobID**. | **messageText**": "eventContext/payrollProcessingJobID is invalid. eventContext/payrollProcessingJobID" | |

Chapter 4

# Frequently Asked Questions (FAQs)

## Question 1: What's the recommended batch size to have the best performance?

**Answer:** 100 rows.

## Question 2: If the client created templates for pay data, would the template be used for the Payroll Data Input API?

**Answer:** No.

## Question 3: If a future hire is included in the Request Payload, would it get processed?

**Answer:** If the position Start Date is in the current payroll cycle then it will be processed, otherwise no.

## Question 4: Does the Payroll Data Input API import Paid Time Off (PTO) hours?

**Answer:** Yes, you can configure Hours and Earning codes for each type in ADP Workforce Now and import the data.

## Question 5: Does the Payroll Data Input API import double overtime for California workers?

**Answer:** If your application uses California double overtime earning rules and calculates earnings, then the client practitioner could setup Other Earnings codes for each overtime container and Import Earnings without hours for each Earning code. Otherwise, the client practitioner could setup Other Hours Codes for each overtime container and send a request for ADP service to set up special calculations for each code. Then, you can import hours for each code.

## Question 6: Does the Payroll Data Input API support earnings, such as cash tips?

**Answer:** Currently, the API supports reportable earnings with known defects under the following conditions:

- Client does not use **Earning Code** of T and **Memo Code** of 5.
- Client uses API for only one type of earning that can be reported.

Your application must do the following:

- Send the amountValue for **Memo Code** of 5 through reportableEarningAndBenefitInputs. It populates data under the **Memo Code** of 5 and data under **Earning Code** of T.
- If the employee already received the payment (such as cash tips), you also need to send a negative Deduction to offset the **Earning Code** of T. For example, if the employee needs to report tips of $100 cash, the API should send the memo through /reportableEarningAndBenefitInputs/amount: 100 for code T (it's a known defect as the code should be 5), and deduction with a matching code in the amount of 100.

## Question 7: How is piece Rate handled in pay data?

**Answer:** In the case where an employee gets paid by piece rate, the rate and pieces could be entered in the Pay Data.
For example, if a fruit picking company pays its worker for each piece of picked fruit, the data could be entered in Pay Data.
Let's say a worker picked 100 oranges at $0.01 each and 200 apples at $0.02 each. The person should receive 100 x $0.01 = $1 and 200 x $0.02 = $4 for a total of $5.

Follow these steps to add the this in the Pay Data:

1. The client creates two Other Earning Codes. For instance, **O** for Orange and **A** for Apple.
2. The client creates two Matching Memo Codes. For instance, **O** and **A**.
3. The client needs to call ADP to setup a Rate of Pay Payroll Calculation of $0.01 for **O** and $0.02 for **A** in the ADP Payroll engine.

The above entries will result in the earning **O** = $1 and **A** = $4 in the worker's pay statement.

## Question 8: Does vacation/sick hours entered in Paydata auto deduct time off balances in ADP Workforce Now?

**Answer:** It depends on the following:

- Answer is Yes if the client doesn't use ADP Workforce Now accruals (Autopay Benefit Accruals). The hours entered in the batch will reduce the balance (allowed and taken).
- If the client uses ADP Workforce Now Time off Based Accruals, the codes must be mapped to the Time Off Policy and a Time Off Request must be submitted. Entries for the code entered within the batch do not impact the balance. The Time Off Accepted requests are processed through the HBA001 batch, which drives the balance.

Chapter 5

# Known Issues and Limitations

Please refer to the ADP Workforce Now API Release Plan to see if any limitation(s) are scheduled for upcoming ADP Workforce releases.

### Issue: Meta Call is not Giving any Codelist Details for ratecodes

## Impacted APIs

For API related information please refer Pay Data Input under API Explorer.

| Method | URI | roleCode Value |
|---|---|---|
| POST | /events/payroll/v1/pay-data-input.modify | practitioner |
| GET | /events/payroll/v1/pay-data-input.modify/meta | practitioner |

## Description

The rate code details in earningInputs are not returned in the meta API response to use them in the request payload.

## Suggested Workaround

There are no workarounds available.

### US1719297: Update Pay Data Input API  to reflect true batch name

## Impacted APIs

For API related information please refer Pay Data Input under API Explorer.

| Method | URI | roleCode Value |
|---|---|---|
| POST | /events/payroll/v1/pay-data-input.modify | practitioner |
| GET | /events/payroll/v1/pay-data-input.modify/meta | practitioner |

## Description

A random number is appending to **payrollProcessingJobID[e.g.Batch_654321]** in the ADP Workforce Now UI after a successful API call.

## Suggested Workaround

There are no workarounds available.

### Issue: Event notification is not supported

## Impacted APIs

For API related information please refer Pay Data Input under API Explorer.

| Method | URI | roleCode Value |
|---|---|---|
| POST | /events/payroll/v1/pay-data-input.modify | practitioner |
| GET | /events/payroll/v1/pay-data-input.modify/meta | practitioner |

## Description

Currently, event notification is not supported when a pay data batch is added.

## Suggested Workaround

There are no workarounds available.

Chapter 6

# Appendixes

### Appendix A: Pay Data Types Handled by the Payroll Data Input API

The following table lists the pay data transmitted through the Payroll Data Input API to ADP Workforce Now.

For API related information please refer Pay Data Input under API Explorer.

| Type | Common Uses | Schema Location |
|---|---|---|
| Regular and Overtime Earnings | Sends regular and overtime earnings to ADP Workforce Now. | /payDataInput/payeePayInputs/payrollProfilePayInputs<br>/payInputs/earningInputs |
| Coded Hours and Earnings | Sends client defined and other earnings to ADP Workforce Now. | /payDataInput/payeePayInputs/payrollProfilePayInputs<br>/payInputs/earningInputs |

| | | |
|---|---|---|
| Deductions | Sends deductions to ADP Workforce Now, including out of box deductions and client-defined deductions. | /payDataInput/payeePayInputs/payrollProfilePayInputs /payInputs/ deductionInputs |
| Deduction Adjustment 11 | Recommend using Deductions.<br>**Important:** ADP Workforce Now will stop supporting these resources after 2019. | /payDataInput/payeePayInputs/payrollProfilePayInputs /payInputs/reimbursementInputs |
| Allocation Position | Sends the allocation position details to ADP Workforce Now. Specifies the number of the allocations associated with an entry. This is for Automatic Labor Allocation (ALA). | /payDataInput/payeePayInputs/payrollProfilePayInputs /payInputs/ payAllocation |
| Cancel Automatic Payment | This true/false indicator will cancel automatic payments for an employee for companies set up as Automatic Pay.<br>Values: true, false | /payDataInputs/payeePayInputs/payrollProfilePayInputs/payInputs/ cancelAutomaticPayIndicator |
| Tax Frequency | The company must be set up for Proportionate Labor Distribution.<br>Entry is not allowed if the employee has an entry for Tax Frequency for the same Pay # in this file or in another batch.<br>Values: 0-9, S, M, D, T, B, F, H,C | /payDataInputs/payeePayInputs/payrollProfilePayInputs/payInputs/ taxInputs/taxCycleCode |
| Shift | Send shift for special calculations valid<br>Values: 2-9 | /payDataInputs/payeePayInputs/payrollProfilePayInputs/payInputs/earningInputs/configurationTags/tagCode<br>/payDataInputs/payeePayInputs/payrollProfilePayInputs/payInputs/earningInputs/configurationTags/tagValues |
| Temp Cost Number | Sends the Cost Number Details to ADP Workforce Now. Specifies the temporary cost number. | /payDataInput/payeePayInputs/payrollProfilePayInputs /payInputs/ payAllocation |
| Temporary Department Number | Sends the Department Details to ADP Workforce Now. Specifies the temporary Department Number. | /payDataInput/payeePayInputs/payrollProfilePayInputs /payInputs/ payAllocation |
| Memos | Sends the Memo details to ADP Workforce Now. | /payDataInput/payeePayInputs/payrollProfilePayInputs/payInputs /memoInputs |

## Appendix B: Pay Data Fields Not Handled by the Payroll Data Input API

Currently, the Payroll Data Input API doesn't handle the following columns on the **Paydata** page. **ADP client practitioner** could use file based import as a work around.

For API related information please refer Pay Data Input under API Explorer.

| **ADP Workforce Now Field Name** | **Description** |
|---|---|
| **ADP Workforce Now Field Name** | **Description** |
| **Tax adjustment fields** | Adjusts the tax that ADP calculates by adding or subtracting this amount from the calculated amount for the following fields:<br>• Federal tax<br>• State tax<br>• Local tax<br>• Lived-in state and local tax<br>• Worked-in state and local tax<br>• Other tax fields |

| | |
|---|---|
| **Medicare, SUI/SDI, Soc Sec** | Adjusts the Medicare and State Unemployment Insurance (SUI) amount ADP calculates by adding or subtracting this amount from the calculated amount. |
| **Advance Pay Date** | Specifies the advance pay date to use with the Wage Garnishment Processing Service (WGPS). |
| **Clock** | Specifies the employee clock value associated with the entry.<br>**Note:** The default clock value is used by the Payroll Data Input API. If an employee has multiple clock values, the ADP Workforce Now **Paydata** page enables clients to select a value other than the default. |
| **Deduction by Week Nb** | Specifies the payroll week numbers for which scheduled deductions are in effect for the selected pay check. |
| **Ded Forecast Wk 1 thru Wk5** | Specifies the deduction forecasting week 1 through 5. |
| **FLSA Special Processing Code** | Indicates whether the transaction is processed as Exclude or Compare Rates. |
| **Other Period Beginning Date** | Specifies the other period beginning date. |
| **Other Period Ending Date** | Specifies the other period ending date. |
| **Override Total Hours Worked - Pay Level** | Specifies the override total hours worked - pay level. |
| **Paycheck Tax Frequency** | Specifies the number of weeks in the pay cycle. |
| **Replacement of Ded Code, Amount, Federal, State Lived-in, Medicare, Soc Sec, SUI/SDI, Worked** | Replaces ADP calculated amounts. |
| **Special Action** | Specifies the special action. |
| **Temp Lived Local, State, School,** and **Worked Local Codes** | Specifies the following:<br>• Lived-in local<br>• State code<br>• Reciprocity allocation code<br>• School district tax code<br>• Worked-in State code<br>• Worked-in Local code |