

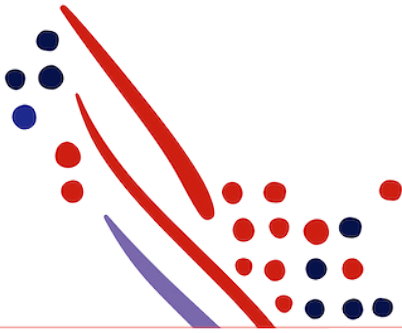


Guide

Pay Data Input APIs Guide for ADP Lyric

Published on
May 13, 2025, 10:19 PM

Last modified
May 29, 2025, 02:07 PM



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 © 2025 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
May 13, 2025, 10:19 PM

Published on
May 29, 2025, 02:07 PM

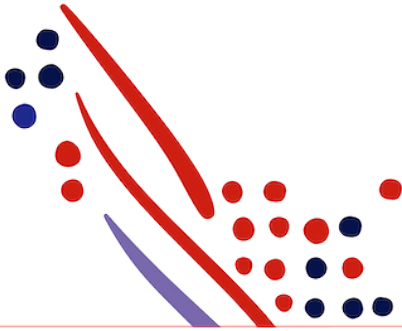


Table of Contents

Chapter 1

CHAPTER 1: About this API

- Summary
- What's New in this Guide?
 - May 2025
- Supported Product Version and Customer Base
- Process Overview
- Required Setup
- Supported Pay Data Types for ADP Lyric

Chapter 2

CHAPTER 2: Use Case: Adding Entries in a Pay Data Batch

- Use Case Description
- API Usage
- Application Scope
- Responses
 - Responses for Retrieving a Pay Data Input Batch for a Specific Batch Request
- Data Dictionary
 - Earning Inputs
 - Memo

Chapter 3

Chapter 3: Known Issues and Limitations

- Limitation 1: Support for payAllocation, taxInputs, timeInputs
 - Impacted API(s):
 - Description
 - Suggested Workaround
- Limitation 2: Support for numberOfHours for Earnings
 - Impacted API(s):
 - Description
 - Suggested Workaround
- Limitation 3: Support for Pay Data Input.Add META API
 - Impacted API(s):
 - Description
 - Suggested Workaround
- Limitation 4: payrollWeekNumber is not supported by API



Impacted API(s):
Description
Suggested Workaround

CHAPTER 1: About this API

Summary

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

- Deductions
- Earnings
- Memos
- Reimbursement

Method	Uniform Resource Identifier (URI)	Description	Note
POST	/events/payroll/v1/pay-data-input.add	Adds a new pay data batch.	<ul style="list-style-type: none"> • The Pay Data Input - Add API enables you to build a data connector application for clients to transmit data sourced in your application to a specific Pay Data batch in ADP Lyric. • This API only supports one offs from the above use cases. • This API does not support recurring pay data. E.g. recurring deductions. • The Payroll Data Input API call is an Asynchronous call. More details on the implementation in Chapter 3.

What's New in this Guide?

In this section, revisions to the Payroll Data Input API and this guide will be announced.

May 2025

- Updated API Guide for Paydata Input.Add for ADP Lyric
- Updated API endpoints and samples to link to API explorer.

Supported Product Version and Customer Base

This API is available with ADP Lyric version 25.18 or higher.

Process Overview

The following table shows an illustration of how your data connector application would be used by a client.

Sl. No.	Actor	Task Description
1	Payroll practitioner	Selects the payroll group code, employees, and payroll file numbers to be imported.
2	Payroll practitioner	Enters the Pay Data inputs and initiating the Pay Data import.

Sl. No.	Actor	Task Description
3	Your application	Automatically creates a new batch and imports Pay Data in ADP Lyric.
4	Payroll practitioner	Reviews the import result on the Pay Data Entry page in the ADP Lyric user interface (UI)
5	Payroll practitioner	Takes actions to balance and adjust, as needed.

Required Setup

Client Factor Setup

- Client Factors shape the creation of an Employee Actual when new transactional data comes into the system.
- When we refer to the "Client Factor," we are referring to the configuration (something like a template) for the incoming data, not the actual data itself.
- A Client Factor is required to be configured by the ADP Lyric's Implementation Team for every unique earning, deduction, Memo, Reimbursement, etc.
- Clients need to make sure the Client Factor codes used by your application are added and configured in the ADP Lyric.
- The following table lists the records in your data connector application and how they correspond to the fields on the **Setup Information** page under **Payroll** in the ADP Lyric

Records	Fields in ADP Lyric HCM UI
Earning Codes	Earnings
Deduction Codes	Deductions
Memo Codes	Memo
Reimbursement Code	Reimbursement

Supported Pay Data Types for ADP Lyric

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

Type	Common Uses	Schema Location
Earnings	Sends one-off client-defined and other earnings to ADP Lyric	/payDataInput/payeePayInputs/payrollProfilePayInputs/payInputs/earningInputs
Deductions	Sends one-off deductions to ADP Lyric	/payDataInput/payeePayInputs/payrollProfilePayInputs /payInputs/ deductionInputs

Type	Common Uses	Schema Location
Memos	Sends the memo details to ADP Lyric	/payDataInput/payeePayInputs/payrollProfilePayInputs/payInputs/memoInputs
Reimbursement	Sends reimbursement details to ADP Lyric	/payDataInput/payeePayInputs/payrollProfilePayInputs/payInputs/reimbursementInputs

NOTE: ADP Lyric has ongoing enhancements to support more functional types.

Chapter 2

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's commonly used to send the following data from a different system or application to the **Pay Data Entry** page in ADP Lyric:

- Earnings
- Deductions
- Memo
- Reimbursement

Note: The Client Practitioner can check whether the Pay Data Entry status is successful or not in ADP Lyric by navigating to **Manage Associate's Pay Data** and select the corresponding Associate

API Usage

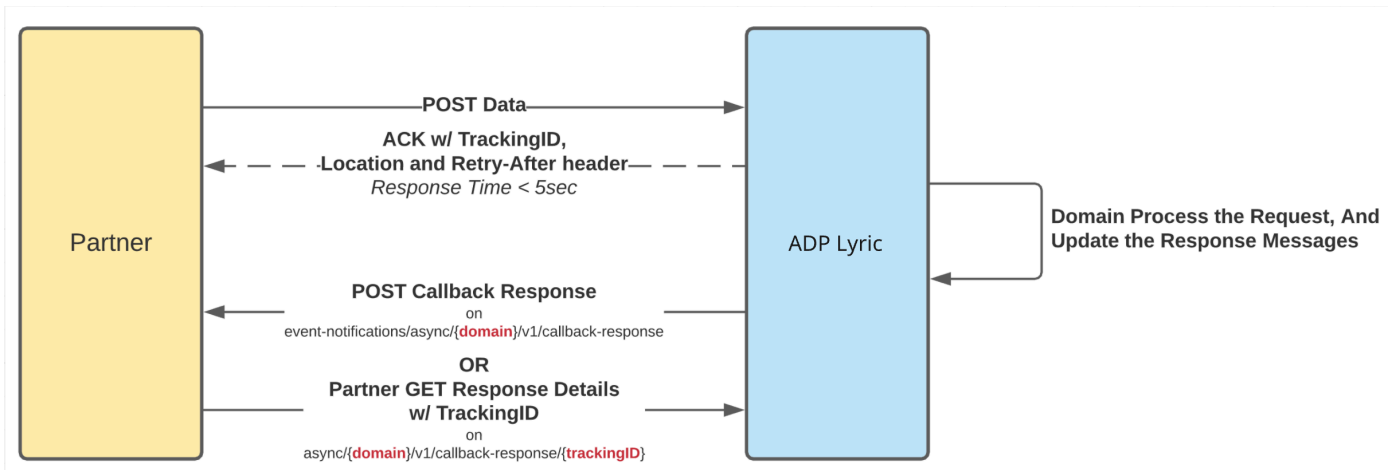
Method	Uniform Resource Identifier (URI)	Description
POST	/events/payroll/v1/pay-data-input.add	Posts a one-off Pay Data entry for a corresponding payroll cycle

Async Job Processing Summary Design

The Payroll Data Input API call is an Asynchronous call, which means when you try to add a Pay Data batch through the API, it throws a **200 Accepted** as a status code along with Message ID and Request ID.

ADP Lyric supports additional uniform resource locator (URL) which takes Message ID from the above API response. On hitting this URL, it will throw a **200 OK** as a status code along with a response payload, which displays a generic message stating if the Import Pay Data process is passed or failed.

Additionally, to keep API response time is within RANGE (API Proxy Default Timeout is 10 seconds, for Marketplace consumer experience, API response time is expected to be within 5 seconds). ADP Lyric HCM has put together the following design to help partners adhere to the 5 seconds response time limitation. The design outlines creating and responding Partners with a request tracking ID within split seconds. And, subsequently post a callback response to the partners and maintain the data store for the callback responses.



Type	API	Description
GET	/payroll/v1/worker-payroll-management/processing-jobs/{job-id}/summary	API to retrieve the processing status of a worker payroll management batch job
POST	/event-notifications/payroll/v1/worker-payroll-management/processing-job.summary.change	Event Notification notify the processing status of a worker payroll management batch job

In the above response, Request Status & Processing Status convey the status of the API call and the record correspondingly.

requestStatus: [success, partial-success, partial-failure, failure, partial-success]

processingStatus: [received, validated, started, in-progress, completed, waiting-practitioner-action]

Application Scope

The canonical URI corresponding to the 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 in order to enable this use case:

/payroll/payrollManagement/payrollProcessing/payDataInputManagement/pay-data-input.add

/payroll/payrollManagement/payrollInstructionManagement/processingJobManagement/processingJob.summary.read

/payroll/payrollManagement/payrollInstructionManagement/processingJobManagement/processingJob.summary.change.eventNotify

Responses

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

Responses for Retrieving a Pay Data Input Batch for a Specific Batch Request

API	Response Code	Condition	messageTxt	Notes
/events/payroll/v1/pay-data-input.add	200	When the pay-data-input.add call to ADP Lyric is successful.	<pre> { "confirmMessage": { "confirmMessageID": { "idValue": " {{messageID}}" }, "createDateTime": "2025-05-15T19:37:11.181Z", "requestID": { "idValue": " {{requestID}}" }, "processingStatusCode": { "codeValue": "received" }, "processMessages": [{ "processMessageID": { "idValue": " {{processMessageID}}" }, "messageTypeCode": { "codeValue": "received" }, "userMessage": { "message": " {{processMessageID}} " } }] } } </pre>	On successfully receiving the API Payload, the API will return a 200 response with the schema.
/payroll/v1/worker-payroll-management/processing-jobs/{job-id}/summary	200 OK (Async call)	When the pay data batch import process is successful and the corresponding entry gets added into ADP Lyric.	<pre> { "submittedDateTime": "2025-05-16T18:52:47.000Z", "processingCounts": { "errorCount": 0, "successCount": 0 }, "processingStatus": "completed", "requestStatus": "success", "applicationID": "LIFION", </pre>	{job-id} in the param should map to {{confirmMessageID}} from the response /events/payroll/v1/pay-data-input.add API

API	Response Code	Condition	messageTxt	Notes
			<pre> "messageID": "194c3d53cb844420 a96a85d9263fd0be", "requestID": "aa436f81d86e454b 8ad96d5121592c5e" ' "messages": [{ }], "meta": { "completeIndicator": true } } </pre>	
/payroll/v1/worker-payroll-management/processing-jobs/{job-id}/summary	200 OK (Async call)	When the pay data batch import process failed/ partially successful and all/ few entries did not get added into ADP Lyric.	<pre> { "submittedDateTime": "2025-05-15T19:46:52.000Z", "processingCounts": { "errorCount": 1, "successCount": 0 }, "processingStatus": "completed", "requestStatus": "success", "applicationID": "LIFION", "messageID": "6e33c6064076499f 933f78dbda816576", "messageDateTime": "2025-05-15T19:46:58.000Z", "requestID": "cbe3b9d490ce4881 9fe1fd06599719bd", "messages": [{ "messageCode": "400", "messageTypeCode": "error", "messageText": "PayTypeInput is invalid", "resourceID": "0196d57c-466c- 7221-87ca- 14b2f53b1183" }], "meta": { </pre>	<p>This can be due to any invalid data sent through the payload.</p> <p>Inside messages array each item represents an entry in the payload request and userMessage, which gets populated with the reason of failure in case if the entry failed to get added into the ADP Lyric system.</p> <p>Ensure valid Code(s) (for earnings, memos, and so on) along with relevant dates and identifiers are send to the Pay Data input API</p> <p>NOTE: Codes setup in Lyric should correspond to the cores setup in the partner system. Please work with the Client Practitioner and/ or ADP Lyric Service team to setup the necessary codes.</p>

API	Response Code	Condition	messageTxt	Notes
			<pre>"completeIndicator": true } }</pre>	
/payroll/v1/worker-payroll-management/processing-jobs/{job-id}/summary	200 OK (Async call)		<pre>{ "submittedDateTime": "2025-05-16T18:58:04.000Z", "processingCounts": { }, "processingStatus": "waiting-practitioner-action", "applicationID": "LIFION", "messageID": "0b7267b82a71466f9b64e11430b74665", "requestID": "300f648d56074650a3a38d64dc615525", "messages": [{ }], "meta": { } }</pre>	"processingStatus": "waiting-practitioner-action" refers to intermediate state where the pay data entry is received and is in 'Draft' state. Client Practitioner can review and process the records.

Data Dictionary

Resources in the following table can be accessed in the ADP Lyric UI in Client Factor Setup section mentioned in the previous chapter.

Using the resources listed in the following table Deduction, Earning, Memo, Reimbursement input data sent through the Payroll Data Input API eliminates the need to enter this data manually in ADP Lyric.

NOTE: Pay Data Input.Add API only supports One-offs i.e. the entry will be processed for the corresponding Pay Period.

Deduction Input:

Schema Location	Field Name	Business Rules	Notes
events/data/eventContext/payrollGroupCode/codeValue	Pay Group 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.
events/data/transform/payDataInput/payeePayInputs/associateOID	Associate OID	Always Required	To retrieve the AOID value for an employee, use Worker API (GET)
events/data/transform/payDataInput/payeePayInputs/payPeriodStartDate	Pay Period Start Date	Always Required	Start Date of the corresponding Pay Data Input entry.
events/data/transform/payDataInput/payeePayInputs/payPeriodEndDate	Pay Period End Date	Optional	End Date of the corresponding Pay Data Input entry. NOTE: Pay Data Input.Add API only supports One-offs i.e. the entry will be processed for the corresponding Pay Period. For Recurring Deductions Please see Payroll Instructions API
events/data/transform/payDataInput/payeePayInputs/payrollProfilePayInputs/payrollFileNumber	Payroll File Number	Optional	Associate's Payroll ID in Lyric
events/data/transform/payDataInput/payeePayInputs/payrollProfilePayInputs/payInputs/deductionInputs/deductionCode/codeValue	Deduction Code	Always Required	Deduction Code for the Pay Data Input entry. NOTE: Codes setup in Lyric should correspond to the codes setup in the partner system. Please work with the Client Practitioner and/ or ADP Lyric Service team to setup the necessary codes.
events/data/transform/payDataInput/payeePayInputs/payrollProfilePayInputs/payInputs/deductionInputs/deductionAmount/amountValue	Deduction Amount	Always Required	Deduction Amount

Earning Inputs

Schema Location	Field Name	Business Rules	Notes
events/data/eventContext/payrollGroupCode/codeValue	Pay Group 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.
events/data/transform/payDataInput/payeePayInputs/associateOID	Associate OID	Always Required	To retrieve the AOID value for an employee, use Worker API (GET)
events/data/transform/payDataInput/payeePayInputs/payPeriodStartDate	Pay Period Start Date	Always Required	Start Date of the corresponding Pay Data Input entry

Schema Location	Field Name	Business Rules	Notes
events/data/transform/payDataInput/payeePayInputs/payPeriodEndDate	Pay Period End Date	Optional	End Date of the corresponding Pay Data Input entry NOTE: Pay Data Input.Add API only supports One-offs i.e. the entry will be processed for the corresponding Pay Period.
events/data/transform/payDataInput/payeePayInputs/payrollProfilePayInputs/payrollFileNumber	Payroll File Number	Optional	Associate's Payroll ID in Lyric
events/data/transform/payDataInput/payeePayInputs/payrollProfilePayInputs/payInputs/earningInputs/earningCode/codeValue	Earning Code	Always Required	Earning Code for the Pay Data Input entry NOTE: Codes setup in Lyric should correspond to the codes setup in the partner system. Please work with the Client Practitioner and/ or ADP Lyric Service team to setup the necessary codes.
events/data/output/payDataInput/payeePayInputs/payrollProfilePayInputs/payInputs/earningInputs/rate/rateValue	Rate Value	Always Required	Earning Amount
events/data/output/payDataInput/payeePayInputs/payrollProfilePayInputs/payInputs/earningInputs/rate/currencyCode	Currency Code	Optional	If no Currency Code is provided, Lyric will default it to USD

Memo

Schema Location	Field Name	Business Rules	Notes
events/data/eventContext/payrollGroupCode/codeValue	Pay Group 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.
events/data/transform/payDataInput/payeePayInputs/associateOID	Associate OID	Always Required	To retrieve the AOID value for an employee, use Worker API (GET)
events/data/transform/payDataInput/payeePayInputs/payPeriodStartDate	Pay Period Start Date	Always Required	Start Date of the corresponding Pay Data Input entry.
events/data/transform/payDataInput/payeePayInputs/payPeriodEndDate	Pay Period End Date	Optional	End Date of the corresponding Pay Data Input entry. NOTE: Pay Data Input.Add API only supports One-offs i.e. the entry will be processed for the corresponding Pay Period.
events/data/transform/payDataInput/payeePayInputs/payrollProfilePayInputs/payrollFileNumber	Payroll File Number	Optional	Associate's Payroll ID in Lyric

Schema Location	Field Name	Business Rules	Notes
events/data/output/payDataInput/payeePayInputs/payrollProfilePayInputs/payInputs/earningInputs/memoInputs/memoCode/codeValue	Memo Code	Always Required	Memo Code for the Pay Data Input entry. NOTE: Codes setup in Lyric should correspond to the cores setup in the partner system. Please work with the Client Practitioner and/ or ADP Lyric Service team to setup the necessary codes. Attaching/ Linking a Memo to another Payroll Code needs to be setup in Lyric while configuring the code.
events/data/output/payDataInput/payeePayInputs/payrollProfilePayInputs/payInputs/earningInputs/memoInputs/memoAmount/amountValue	Amount Value	Always Required	Amount Value

Reimbursement

Schema Location	Field Name	Business Rules	Notes
Schema Location	Field Name	Business Rules	Notes
events/data/eventContext/payrollGroupCode/codeValue	Pay Group 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.
events/data/transform/payDataInput/payeePayInputs/associateOID	Associate OID	Always Required	To retrieve the AOID value for an employee, use Worker API (GET)
events/data/transform/payDataInput/payeePayInputs/payPeriodStartDate	Pay Period Start Date	Always Required	Start Date of the corresponding Pay Data Input entry.
events/data/transform/payDataInput/payeePayInputs/payPeriodEndDate	Pay Period End Date	Optional	End Date of the corresponding Pay Data Input entry. NOTE: Pay Data Input.Add API only supports One-offs i.e. the entry will be processed for the corresponding Pay Period.
events/data/transform/payDataInput/payeePayInputs/payrollProfilePayInputs/payrollFileNumber	Payroll File Number	Optional	Associate's Payroll ID in Lyric
events/data/transform/payDataInput/payeePayInputs/payrollProfilePayInputs/payInputs/reimbursementInputs/reimbursementCode/codeValue	Reimbursement Code	Always Required	Reimbursement Code for the Pay Data Input entry. NOTE: Codes setup in Lyric should correspond to the cores setup in the partner system. Please work with the Client Practitioner and/ or ADP Lyric Service team to setup the necessary codes.

Schema Location	Field Name	Business Rules	Notes
events/data/transform/payDataInput/payeePayInputs/payrollProfilePayInputs/payInputs/reimbursementInputs/reimbursementAmount/amountValue	ReimbursementAmount	Always Required	Deduction Amount

Chapter 3

Chapter 3: Known Issues and Limitations

Limitation 1: Support for payAllocation, taxInputs, timeInputs

Impacted API(s):

Method	URI	roleCode
POST	/events/payroll/v1/pay-data-input.add	Practitioner

Description

Currently **/events/payroll/v1/pay-data-input.add** does not support payAllocation, taxInputs, timeInputs.

Suggested Workaround

There are no workarounds available.

Limitation 2: Support for numberOfHours for Earnings

Impacted API(s):

Method	URI	roleCode
POST	/events/payroll/v1/pay-data-input.add	Practitioner

Description

Currently `/events/payroll/v1/pay-data-input.add` does not support importing no. of hours worked. It only supports the earning amount.

Suggested Workaround

There are no workarounds available.

Limitation 3: Support for Pay Data Input.Add META API

Impacted API(s):

Method	URI	roleCode
GET	<code>/events/payroll/v1/pay-data-input.add/meta</code>	Practitioner

Description

Currently `/events/payroll/v1/pay-data-input.add/meta` is not supported

Suggested Workaround

Refer API Explorer section for sample responses. This includes examples for the 4 use cases currently supported.

Limitation 4: payrollWeekNumber is not supported by API

Impacted API(s):

Method	URI	roleCode
POST	<code>/events/payroll/v1/pay-data-input.add</code>	Practitioner

Description

When this field is sent in through the Payroll Data Input API while creating a pay data batch the import gets failed.

Suggested Workaround

There are no suggested workarounds.