



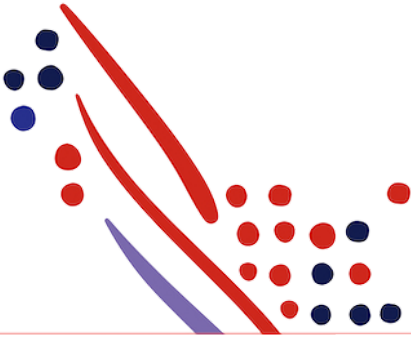
Guide

Benefit Spending Account Enrollment API Guide (HROpen) for ADP Workforce Now

Published on
Mar 23, 2020 11:24AM

Last modified
Mar 25, 2020 3:17PM





ADP Copyright Information

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

Windows is a registered trademark of the Microsoft Corporation.

All other trademarks are the property of their respective owners.

Copyright © 2020 ADP, LLC. ADP Proprietary and Confidential - All Rights Reserved. These materials may not be reproduced in any format without the express written permission of ADP, LLC.

These materials may not be reproduced in any format without the express written permission of ADP, LLC. 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 programs described in this publication.

Published on
Mar 23, 2020 11:24AM

Last modified
Mar 25, 2020 3:17PM

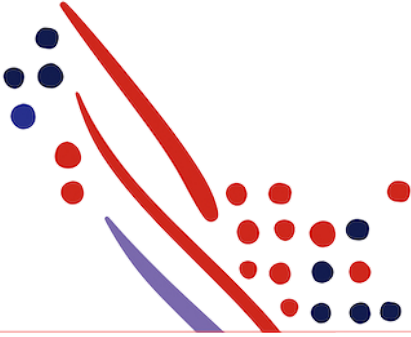


Table of Contents

Chapter 1

About the API

- Summary
- What's New in this Guide?
- Supported Product Version and Customer Base
- Process Overview
- Required Setup Steps
- Postman Collection

Chapter 2

Event Notifications

Chapter 3

Use Case: Getting Spending Account Enrollment Details

- Use Case Description
- API Usage
- Application Scope
- Request Header Parameters
- Supported OData Query Options
- Data Dictionary
- Responses

Chapter 4

Known Issues and Limitations

- Issue 1: Benefit Spending Account Enrollment API dependencies
 - Impacted APIs
 - Issue Description
- Issue 2: API is showing unmasked data without access permissions given by the ADP Workforce Now client practitioner
 - Impacted APIs
 - Issue Description
- Issue 3: Non-spending account enrollments data is not provided
 - Impacted APIs
 - Issue Description
- Issue 4: Get response shows both active and inactive dependent details - Impacted APIs
- Issue 5: No error message is shown when the filter parameter is not used in the URI
 - Impacted APIs

Issue Description

Issue 6: Extra fields in the GET response showing blank values

Impacted APIs

Issue Description

About the API

Summary

The Benefit Spending Account Enrollments Application Programming Interface (API) provides external benefits providers with employee benefits enrollment data. This API provides enrollment data of only those employees who are enrolled in the following plans provided by external benefits providers:

- Parking
- Transit
- MedicalFSA
- Limited Purpose FSA
- Dependent Care FSA

This API also provides worker demographics, employee, dependents, and position information of an employee along with the benefits enrollment data for an employee.

What's New in this Guide?

This is the initial version of the Benefit Spending Account Enrollment API Guide (HROpen) for ADP Workforce Now. In this section, we will announce any new revisions to the Benefit Spending Account Enrollment API and this guide.

Supported Product Version and Customer Base

The Benefit Spending Account Enrollment API is used by external benefit providers conforming to the HROpen industry standards.

Process Overview

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

	Actor	Task Description
1	Client Practitioner	Triggers the API-based data exchange between your application and ADP Workforce Now.
2	Your Application or An external benefit provider	Polls for Event Notifications provided in Event Notifications on page .
3	Client Practitioner	Responds with related Event Notifications along with this Benefit Spending Enrollment API Uniform Resource Locator (URL) in response.
4	Your Application or An external benefit provider	Uses the Uniform Resource Indicator (URI) to Get Benefit Enrollment details along with worker demographics, employee, dependents, and position information of an employee.
5	Client Practitioner	Verifies the information within ADP Workforce Now, under People > Benefits > Enrollments .

Required Setup Steps

1. Verify the external provider exists as a master benefit provider in ADP Workforce Now.
2. Verify the employee is enrolled in the following plans associated with the external benefit provider:
 - o Parking
 - o Transit
 - o Medical FSA
 - o Limited Purpose FSA
 - o Dependent Care FSA
3. Verify your application has subscribed to the canonicals mentioned in Chapter 2: Event Notifications.

Postman Collection

Postman allows you to import a collection of APIs, created by others, so you can try them out. For more information on Postman, see [Making Your First API Call Using Postman](#).

To download API collections for Benefit Spending Account Plan API from the ADP GitHub library and import them to Postman, go to [Benefit Spending Account Enrollments Postman Collections](#).

Chapter 2

Event Notifications

When generated, the following event notifications will show the GET Benefit Spending Account Enrollment API as a part of a notification response body.

For example, consider the sample response payload for an event notification (**legalAddress.change**), which is [Get-Sample-event-notification-response-http-200.json](#).

Scope	Canonical
<code>/hr/workerInformationManagement/workerManagement/associateManagement.worker.deceased.eventNotify.subscribe</code>	<code>/hr/workerInformationManagement/workerManagement/workAssignmentManagement/worker.workAssignment.retire.eventNotify.subscribe</code>
<code>/hr/workerInformationManagement/workerManagement/lifecycleManagement/worker.hire.eventNotify.subscribe</code>	<code>/hr/workerInformationManagement/workerManagement/workAssignmentManagement/worker.transfer.eventNotify.subscribe</code>
<code>/hr/workerInformationManagement/workerManagement/lifecycleManagement/worker.onLeave.eventNotify.subscribe</code>	<code>/hr/workerInformationManagement/dependentInformationManagement/dependentManagement/dependent.change.eventNotify.subscribe</code>
<code>/hr/workerInformationManagement/workerManagement/workerManagement/worker.hire.eventNotify.subscribe</code>	<code>/benefits/benefitsManagement/benefitsElectionManagement/benefitsEnrollment/benefitCoverages.enroll.eventNotify.subscribe</code>

<p>erManagement/lifecycleManagement/worker.terminate.eventNotify.subscribe</p>	
<p>/hr/workerInformationManagement/workerManagement/addressManagement/worker.legalAddress.add.eventNotify.subscribe</p>	<p>/hr/workerInformationManagement/workerManagement/associateManagement/worker.legalName.change</p>
<p>/hr/workerInformationManagement/workerManagement/addressManagement/worker.legalAddress.change.eventNotify.subscribe</p>	<p>/hr/workerInformationManagement/workerManagement/associateManagement/worker.birthdate.change</p>
<p>/hr/workerInformationManagement/workerManagement/communicationManagement/worker.personalCommunication.email.add.eventNotify.subscribe</p>	<p>/hr/workerInformationManagement/workerManagement/associateManagement/worker.gender.change</p>
<p>/hr/workerInformationManagement/workerManagement/communicationManagement/worker.personalCommunication.email.change.eventNotify.subscribe</p>	<p>/hr/workerInformationManagement/workerManagement/communicationManagement/worker.businessCommunication.landline.add</p>
<p>/hr/workerInformationManagement/workerManagement/communicationManagement/worker.personalCommunication.landline.add.eventNotify.subscribe</p>	<p>/hr/workerInformationManagement/workerManagement/communicationManagement/worker.businessCommunication.landline.change</p>
<p>/hr/workerInformationManagement/workerManagement/communicationManagement/worker.personalCommunication.landline.change.eventNotify.subscribe</p>	<p>/hr/workerInformationManagement/workerManagement/communicationManagement/worker.businessCommunication.landline.remove</p>
<p>/hr/workerInformationManagement/workerManagement/workerAssignmentManagement/worker.workAssignment.modify.eventNotify.subscribe</p>	

Use Case: Getting Spending Account Enrollment Details

Use Case Description

For a given employee and effective date the Benefit Spending Account Enrollments API returns the list of benefit spending account enrollments in Flexible Spending Account (FSA) plans associated with the external provider.

This API exposes Benefit enrollments found in the ADP Workforce Now user interface (UI) by selecting **People > Benefits > Enrollments**.



Important

Without the filter parameter, the Benefit Spending Account Enrollments API will respond with a 400 Bad Request.

API Usage

Method	URI	Description	GitHub Sample Response Payload	Notes
GET	<code>/benefits/v1/associates/{aoid}/spending-account-enrollments? \$filter=accountBasedCoverage/effectivePeriod/validFrom%20ge%20'YYYY-MM-DD'%20and%20accountBasedCoverage/effectivePeriod/validTo%20le%20'YYYY-MM-DD'</code>	Get enrollment, demographic, position, dependents(active and inactive) and spending account enrollments data for an employee.	Get-benefit-spending-enrollments-http-200-response.json	The AOID and a single effective date for validFrom and validTo are generated dynamically in the Event Notification responses.

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 to enable this use case:

`/benefits/benefitsManagement/spendingAccountManagement/workerAccountEnrollment Management/enrollments.read`

Request Header Parameters

Parameter Name	Value	Usage
Accept	application/json;masked=false	To display the unmasked values of Tax ID and Date of Birth .

Supported OData Query Options

This use case supports the Query String Options as defined by Open Data Protocol (OData).

For more information about OData, see [Introduction to the ADP API Open Data Protocol \(OData\)](#).

Query String	Description	Example	Notes
\$filter	Returns worker demographic, position, and spending account enrollment data for an employee.	? \$filter=accountBasedCoverage/effectivePeriod/validFrom ge '2019-10-01' and accountBasedCoverage/effectivePeriod/validTo eq '2019-10-01'.	Make sure you give both a validFrom and validTo to the same date. The date format should be YYYY-MM-DD.

Data Dictionary

To access the following resources, use the information under **Navigation in ADP Workforce Now**.

Schema Location	Navigation in ADP Workforce Now	Section or Field Name in ADP Workforce Now	Note
alternatelds	Not Displayed	Not Displayed	
person/name	People > Personal Information > Personal Profile	Name	
person/communication		Not Displayed	
person/communication/address		Addresses	
person/communication/address/line			
person/communication/address/extendedLines			
person/communication/address/countryCode	People > Personal Information > Personal Profile	Addresses	

person/communication/address/countrySubDivisions			
person/communication/address/city			
person/communication/address/postalCode			
person/communication/phone/countryDialingCode		Contact	
person/communication/phone/areaDialingCode			
person/communication/phone/dialNumber			
person/communication/phone/formattedNumber			
person/communication/phone/useCode			
person/communication/email			
person/communication/email/address			
person/communication/email/useCode			
person/communication/email/preference			
person/legalId	People > Personal Information > Personal Profile	Tax ID	
person/legalId/value			
person/legalId/description			
person/birthDate		Demographics	
person/gender			
person/citizenship			
person/residenceCountry			
person/ethnicity			
person/tobaccoUserIndicator			

person/identifyingMarks		Tobacco User	
person/race		Not Displayed	Fields that show blank values in the response, which is expected.
person/nationality			
person/visa			
person/religion			
person/legalDocuments			
person/employmentPermits			
employee/specialClassifications	People > Employment > Employment Profile	Employment	
employee/id	People > Employment > Employment Profile	Employment	
employee/id/value			
employee/citizenship			
employee/residenceCountry			
employee/ethnicity			
employeeWorkLifeCycle		Hire Date	
employeeWorkLifeCycle/hire			
employeeWorkLifeCycle/hire/ originalDate			
deployment		Employment	
deployment/resourceRelationshipCode			
deployment/organizationIds			
deployment/job			
deployment/position			
deployment/employeeReportingCodes		Corporate Groups	
deployment/specialClassifications			

deployment/licenses			
deployment/certifications	People > Employment > Employment Profile	Corporate Groups	
deployment/intervalCode			
deployment/statusCode		Status	
deployment/statusEffectiveDate			
rateBasedCoverage	Not Displayed	Not Displayed	
dependents	People > Benefits > Dependents & Beneficiaries		
dependents/id			
dependents/id/value			
dependents/relationshipCode		Relationship	
dependents/person		Not Displayed	
dependents/person/name		Name	
dependents/person/communication/address		Address	
dependents/person/communication/phone		Phone	
dependents/person/communication/email		Email	
dependents/person/communication/web		Web	
dependents/person/birthDate		Date of Birth	
dependents/person/gender	People > Benefits > Dependents & Beneficiaries	Gender	
dependents/citizenship		Not Displayed	
dependents/residenceCountry			
dependents/ethnicity			
dependents/tobaccoUserIndicator		Tobacco User	
dependents/identifyi		Not Displayed	Some fields that show blank values in

ngMarks			the response, which is expected.
dependents/race			
dependents/nationality			
dependents/visa			
dependents/religion			
dependents/legalDocuments			
dependents/employmentPermits			
accountBasedCoverage	People > Benefits > Enrollments	Consumer Health and Savings Accounts - FSA Health Care	
accountBasedCoverage/ accountBasedProductCode			
accountBasedCoverage/ effectivePeriod			
accountBasedCoverage/ electionType			
accountBasedCoverage/ id			

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](#).

Response Code	Condition	messageText	GitHub Sample Response Payload	Tips to Handle
200 OK	Getting employee enrollment data for a valid effective date.	NA	Get-benefit-spending-enrollments-http-200-response.json	
200 OK	Gets full masking of the Tax ID and Date of Birth when the practitioner is set with full masking settings.	NA	Get-benefit-spending-enrollments-FullMasking-http-200-response.json	Use application/json;masked=false to display the unmasked values of Tax ID and Date of Birth .
200 OK	Gets partial masking of Tax ID and Date of Birth when the practitioner is set with	NA	Get-benefit-spending-enrollments-withMaskedAsFalse-http-	

	partial masking settings.		200-response.json	
400 Bad Request	When the date filter is incorrect or the validFrom and validTo dates are not equal.	{ "type": "ERROR", "code": "BAD_REQUEST", "description": "Invalid Date Filter"}	Get-benefit-spending-enrollments-with-invalid-date-filter-http-400-response.json	You can get the benefits enrollment details as of a single effective date. So, make sure you give both validFrom and validTo the same date. For example, ? \$filter=accountBasedCoverage/effectivePeriod/validFrom ge '2019-10-01' and accountBasedCoverage/effectivePeriod/validTo eq '2019-10-01' .
400 Bad Request	When the employee is not enrolled in any spending account plan effective to the dates mentioned.	{ "type": "ERROR", "code": "BAD_REQUEST", "description": "Enrollment Not Found"}	Get-benefit-spending-enrollments-with-non-effectiveDate-filter-http-400-response.json	Make sure you give a date for which the data is effective.
400 Bad Request	When no filter is used.	NA	NA	For example, /benefits/v1/associates/{aoid}/spending-account-enrollments . IMPORTANT: In this case, the API throws a blank response.

Chapter 4

Known Issues and Limitations

Issue 1: Benefit Spending Account Enrollment API dependencies

Impacted APIs

Method	URI	roleCode Value
GET	/benefits/v1/associates/{aoid}/spending-account-enrollments? \$filter=accountBasedCoverage/effectivePeriod/validFrom%20ge%20'YYYY-MM-DD'%20and%20accountBasedCoverage/effectivePeriod/validTo%20le%20'YYYY-MM-DD'	Practitioner

Issue Description

The Benefit Spending Account Enrollment API **/benefits/v1/workers/{associateOID}/spending-account-enrollment** endpoint is dependent on a few constraints:

1. The Benefit Spending Account Plan API must have been used to create an Benefit Spending Account Plan in ADP Workforce Now for the Benefit Spending Account Enrollment API to recognize the the record.

2. The partner's **consumerApplication** must be set up by the ADP Benefits team to use the API.

Issue 2: API is showing unmasked data without access permissions given by the ADP Workforce Now client practitioner

Impacted APIs

Method	URI	roleCode Value
GET	/benefits/v1/associates/{aoid}/spending-account-enrollments? \$filter=accountBasedCoverage/effectivePeriod/validFrom%20ge%20 'YYYY-MM- DD'%20and%20accountBasedCoverage/effectivePeriod/validTo%20l e%20'YYYY-MM-DD'	Practitioner

Issue Description

Even though there are no access permissions given by the ADP Workforce Now client practitioner to see the unmasked data, the Benefit Spending Account Enrollments API is still showing the unmasked data by passing the **application/json;masked=false** header.

Issue 3: Non-spending account enrollments data is not provided

Impacted APIs

Method	URI	roleCode Value
GET	/benefits/v1/associates/{aoid}/spending-account-enrollments? \$filter=accountBasedCoverage/effectivePeriod/validFrom%20ge%20 'YYYY-MM- DD'%20and%20accountBasedCoverage/effectivePeriod/validTo%20l e%20'YYYY-MM-DD'	Practitioner

Issue Description

The Benefit Spending Account Enrollment API doesn't provide non-spending account enrollment details. Support for non-spending account enrollments should be provided.

Issue 4: Get response shows both active and inactive dependent details - Impacted APIs

Method	URI	roleCode Value
GET	/benefits/v1/associates/{aoid}/spending-account-enrollments? \$filter=accountBasedCoverage/effectivePeriod/validFrom%20ge%20 'YYYY-MM- DD'%20and%20accountBasedCoverage/effectivePeriod/validTo%20l e%20'YYYY-MM-DD'	Practitioner

Issue 5: No error message is shown when the filter parameter is not used in the URI

Impacted APIs

Method	URI	roleCode Value
GET	/benefits/v1/associates/{aoid}/spending-account-enrollments? \$filter=accountBasedCoverage/effectivePeriod/validFrom%20ge%20 'YYYY-MM- DD'%20and%20accountBasedCoverage/effectivePeriod/validTo%20l e%20'YYYY-MM-DD'	Practitioner

Issue Description

When no filter is used in the URI, the Benefit Spending Account API returns a 400 Bad Request with a blank response. For example, **/benefits/v1/associates/{aoid}/spending-account-enrollments**.

Issue 6: Extra fields in the GET response showing blank values

Impacted APIs

Method	URI	roleCode Value
GET	/benefits/v1/associates/{aoid}/spending-account-enrollments? \$filter=accountBasedCoverage/effectivePeriod/validFrom%20ge%20 'YYYY-MM- DD'%20and%20accountBasedCoverage/effectivePeriod/validTo%20l e%20'YYYY-MM-DD'	Practitioner

Issue Description

There are some extra fields in the GET response which are not supported by the API and display blank values.