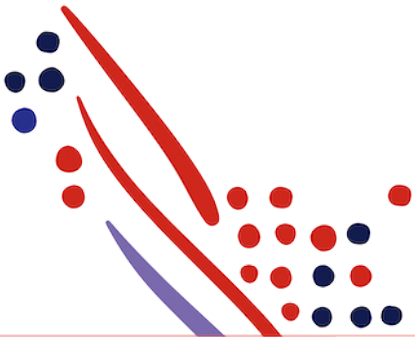
Guide

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

Published on
May 05, 2020 3:30PM

Last modified
Jun 08, 2023 2:30PM

Always Designing for People™



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 © 2023 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 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 05, 2020 3:30PM

Last modified
Jun 08, 2023 2:30PM

Table of Contents

Chapter 1

About this API

- Summary
- What's New in this Guide?
  - April 2022
  - August 2020
- Supported Product Version and Customer Base
- Process Overview
- Required Setup Steps
- Terminology
- Postman Collection

Chapter 2

Use Case: Creating and Updating a Benefit Spending Account Plan

- Use Case Description
- API Usage
- Application Scope
- Supported Actors
- Sequence of Interactions
- Data Dictionary
- Responses

Chapter 3

Unsupported Fields

Chapter 4

Known Issues and Limitations

- Issue 1: TA4742036 - META Information is not supported the Benefit Spending Account Plan API
  - Impacted APIs
  - Suggested Workaround
- Issue 2: On updating a Benefit Plan, the Response status code and the message thrown by the API are inappropriate
  - Impacted APIs
  - Issue Description
  - Suggested Workaround
- Issue 3: While creating a plan with the planAnniversary field as blank, the Benefit Spending Account API returns a 500 Internal Server Error instead of a 400 Bad Request.

Impacted APIs

Issue Description

Suggested Workaround

Issue 4: TA4757122 - API doesn't support validation for the alternateID field

Impacted APIs

Issue Description

Issue 5: Updating the planStartDate and alternateID is not supported through the Benefit Spending Account API

Impacted APIs

Suggested Workaround

Issue 6: Event Notifications are not supported by this Benefit Spending Account API

Impacted APIs

Suggested Workaround

Chapter 1

About this API

Summary

The Benefit Spending Account Plan Application Programming Interface (API) allows external benefit providers to create and update spending account plans in ADP Workforce Now. It currently supports the creation of the following plans:

- Health Savings Account (HSA)
- Parking
- Transit
- Medical Flexible Spending Account (MedicalFSA)
- Limited Purpose FSA
- Dependent Care FSA

Plans created using the Benefit Spending Account Plan API are in the in-progress state. The Practitioner, or Plan Administrator, reviews the incomplete plans and provides the necessary information to complete the plan setup.

What's New in this Guide?

April 2022

Updated API endpoints and samples to link to API explorer.

November 2020

Updated Required Setup Steps section in this chapter with an important Note.

August 2020

- Updated the **Postman Collection** in this chapter.
- Updated the **Sample Request Payloads** under the **API Usage** and **Responses** sections in Chapter 2 - Use Case: Creating and Updating a Benefit Spending Account Plan.

Supported Product Version and Customer Base

The Benefit Spending Account Plan API is used by external benefit providers conforming to the HR Open industry standards, and is available for workers in the United States.

Process Overview

The following table shows an illustration of 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	Posts a spending account plan to be created in ADP Workforce Now using the Benefit Spending Account Plan API.
3	Your Application or an external benefit provider	Creates an in-progress spending account plan in ADP Workforce Now.
4	Your Application or an external benefit provider	Updates an in-progress spending account plan in ADP Workforce Now.
5	Client Practitioner	Gets a Message notification to complete the Plan setup.
6	Client Practitioner	Acts as needed to complete the Plan setup under Setup > Benefits > Plan Setup.
7	Your Application or an external benefit provider	Updates a practitioner reviewed spending account plan in ADP Workforce Now if needed.

Required Setup Steps

important

Please reach out to Matt Rollins and Carrier Connections team for information to integrate with WFN using this API

You need to verify the external provider exists as a master benefit provider within the ADP Workforce Now user interface (UI) by selecting **Setup > Benefits > Benefit Providers**.

Terminology

Term	Description
Account Based Product Code	Each spending account has an Account Based Product Code, which is decided by the external benefit providers.
Master Benefit Provider	ADP Workforce Now has few benefit providers who can add benefit plans for the company. These benefit providers are considered Master Benefit Providers.
Spending Account Plans	These are FSA and HSA plans.

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 the Benefit Spending Account Plan API from the ADP GitHub library and import them to Postman, go to Postman Benefits Spending Account Collection.

Chapter 2

Use Case: Creating and Updating a Benefit Spending Account Plan

Use Case Description

- The Benefit Account Plan API allows the external benefit provider to create and update an in-progress spending account plan in ADP Workforce Now.
- After the API is successful, the in-progress plan can be viewed and accessed by Practitioners in ADP Workforce Now by selecting **Setup > Benefits > Plan Setup**.

The Benefit Account Plan API allows the external benefit provider to update the plan even after reviewed and approved by the ADP Workforce Now Practitioner.

Important:

While updating an already existing plan make sure that you don't change the following details of the payload. They should match with the details which are provided while creating the Benefit Spending Account Plan.

- **id/value**
- **providerId/value**
- **productCode**

In case any one of these is changed when compared to the one's that are used while creating the plan, a new plan gets created in ADP Workforce Now.

API Editor guide link : Spending Accounts Plan For ADP WFN

API Usage

Method	Uniform Resource Indicator (URI)	Description
POST	/benefits/v1/spending-account-plans	Creates an in-progress spending account plan in ADP Workforce Now.

Application Scope

The canonical URI corresponding to the Benefit Spending Account Plan 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/organizationAccountPlanManagement/accountPlan.add

Supported Actors

Request Parameter roleCode Value	Usage
Practitioner	System user that creates a Benefit Spending Account Plan.

Sequence of Interactions

1. Your consumer application makes a **/benefits/v1/spending-account-plans** request to the ADP API endpoint.
2. The ADP API endpoint responds to the consumer application with the details of **/benefits/v1/spending-account-plans**.

Data Dictionary

Plans created using the Benefit Spending Account Plan API will be available in the ADP Workforce Now user interface (UI) by selecting **Setup > Benefits > Plan Setup.** Resources listed in the following table can be accessed by selecting **Setup > Benefits > Plan Setup.**

When selecting the created and incomplete plan the fields in the following table are shown under the **Plan settings**, **Eligibility Group Details**, and **Define contributions** sections.

Important: All the fields mentioned in the following table cannot be edited by the ADP Workforce Now Client Practitioner. So, make sure you send all the required details in the payload to complete the plan setup.

Schema Location	Field Name in ADP Workforce Now UI	Is Required (Y/N)	Section in ADP Workforce Now UI	Note
id/value				
alternateId/value	**Policy Number**	Y	**Plan settings**	
legalPlanName	**Plan Name**	Y		
providerId		Y		Predefined ID provided by the external provider. **Important:** The external provider name is shown under the **Provider** field in the ADP Workforce Now UI.
providerId/value				
productCode	**Plan Type**	Y		Supported plan types are:

				• MedicalFSA • Limited Purpose FSA • Dependent Care FSA • Parking • Transit The following are the **productCode** types: • "Parking" • "Transit" • "MedicalFSA" • "LimitedPurpose FSA" • "DependentCare FSA"
effectivePeriod				This is the Eligibility Group effective date.
effectivePeriod/validFrom	**Start Date**	N	**Plan settings**	
originalEffectiveDate	Not displayed			
planAnniversary	**Renewal Date [MM/DD]**	Y	Plan settings	This is the plan anniversary day and month.
WaitingRuleType			**Eligibility Group Details**	
WaitingRuleType/rollForward Indicator	**First of month...**			An indicator must determine whether the 'First of month' waiting rule type can be rolled forward to the first of the following month when the date falls on the first of the month.
WaitingRuleType/ruleValue				Value indicating the number of months or days for the waiting rule type code.
WaitingRuleType/waitingRuleEventCode		Y		The waiting rule can be one of the following: • **WorkEvent** • **NewHire**
WaitingRuleType/waitingRuleTypeCode		Y		One of the following values can be used for the **waitingRuleTypeCode**: • **HireDate** • **FirstOfMonth AfterXMonths** • **AfterXMonths** • **AfterXDays** • **FirstOfMonth AfterXDays**

				• **OtherProbation** • **FirstOfMonthAfterHire** • **FirstPayPeriodAfterXDays** • **NoWaitingRule**
/accountBasedPlanSetupType/ employerContributionType	**Employer Contribution**	Y	**Define contributions**	Contribution Type can either be **Flat** or **Percent**.
/accountBasedPlanSetupType/ accountBasedCoverageLevels/employerContributionType/ contributionAmount				
/accountBasedPlanSetupType/ accountBasedCoverageLevels/employerContributionType/ contributionAmount/ Currency				Currency code is USD.
/accountBasedPlanSetupType/ accountBasedCoverageLevels/employerContributionType/ contributionAmount/ value				
/accountBasedPlanSetupType/ accountBasedCoverageLevels/employerContributionType/ contributionType				
/accountBasedPlanSetupType/ accountBasedCoverageLevels/minimumContribution	**Minimum Employee Contribution**	Y	**Define contributions**	
/accountBasedPlanSetupType/accountBasedCoverageLevels/ minimumContribution/currency				Currency code is USD.
/accountBasedPlanSetupType/accountBasedCoverageLevels/ minimumContribution/value				
/accountBasedPlanSetupType/accountBasedCoverageLevels/ maxi	**Minimum Employee Contribution**	Y	**Define contributions**	

mumContribution				
/accountBasedPlanSetupType/ accountBasedCoverageLevels/maximumContribution/currency				Currency code is USD.
/accountBasedPlanSetupType/accountBasedCoverageLevels/ maximumContribution/value				
/accountBasedPlanSetupType/accountBasedCoverageLevels/ coverageLevelType	**Coverage level**	Y	**Define contributions**	Visible when Health Savings Account (HSA) plan is created Supported Coverage level types are **Employee** and **Employee Family**.

The following table consists of the following:

- Combinations that can be used for waiting rule type.
- Corresponding field names are enabled in the ADP Workforce Now based on the combinations passed.

The following resources can be accessed in the ADP Workforce Now UI by selecting **Setup > Benefits > Plan Setup.** When selecting the created and incomplete plan the following fields are shown under **Eligibility Group Details**.

Section in ADP Workforce Now UI	Combinations	Field Name in ADP Workforce Now UI
New Hire Waiting Period	"waitingRuleEventCode": "NewHire","waitingRuleTypeCode": "HireDate" Or "waitingRuleEventCode": "NewHire","waitingRuleTypeCode": "NoWaitingRule"	**Date of hire (no waiting period)**
New Hire Waiting Period	"waitingRuleEventCode": "NewHire","waitingRuleTypeCode": "FirstOfMonthAfterXMonths" Or "waitingRuleEventCode": "NewHire","waitingRuleTypeCode": "FirstOfMonthAfterXDays"	**First of month...**
New Hire Waiting Period	"waitingRuleEventCode": "NewHire","waitingRuleTypeCode": "AfterXMonths" Or "waitingRuleEventCode": "NewHire","waitingRuleTypeCode": "AfterXDays"	**following date of hire**
Work Event Waiting Period	"waitingRuleEventCode": "WorkEvent","waitingRuleTypeCode": "HireDate"OR "waitingRuleEventCode": "WorkEvent","waitingRuleTypeCode": "NoWaitingRule"	**Date of work event (no waiting period)**
Work Event Waiting Period	"waitingRuleEventCode": "WorkEvent","waitingRuleTypeCode": "FirstOfMonthAfterXMonths" OR "waitingRuleEventCode": "WorkEvent", "waitingRuleTypeCode": "FirstOfMonthAfterXDays"	**First of month...**
Work Event Waiting	"waitingRuleEventCode": "NewHire","waitingRuleTypeCode":	**following date of work event**

Period	"AfterXMonths" OR "waitingRuleEventCode": "NewHire","waitingRuleTypeCode": "AfterXDays"	

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.

API Editor guide link : Spending Accounts Plan For ADP WFN

Response Code	Condition	messageText	Tips to Handle
Response Code	Condition	messageText	Tips to Handle
201 Created	Adding or creating a Benefit Plan Flexible Spending Account (FSA) in ADP Workforce Now.	{ "type":"SUCCESS", "code": "CREATED", "description": "SUCCESS"}	
201 Created	Adding or creating a Healthcare Spending Account (HSA) Plan in ADP Workforce Now.	{ "type":"SUCCESS", "code": "CREATED", "description": "SUCCESS"}	
201 Created	Adding or creating a Benefit Plan without passing **waitingRuleType**.	{ "type":"SUCCESS", "code": "CREATED", "description": "SUCCESS"}	
201 Created	Updating a Practitioner approved benefit plan.	{ "type":"SUCCESS", "code": "CREATED", "description": "SUCCESS"}	To update a benefit plan, make sure that you provide the following details same as what you provided while adding a plan • id/value • providerId/value • productCode

201 Created	Updating a created plan, which is in a pending state (not reviewed and approved by the ADP Workforce Now Practitioner).	{
		"type": "SUCCESS",
		"code": "OK",
		"description": "SUCCESS"
		}	

400 Bad Request	Adding or updating a Benefit plan with a maximum contribution more than the plan limits.	{
		"type": "ERROR",
		"code": "BAD_REQUEST",
		"description": " Maximum contribution exceeds plan limits. "
		}
			Make sure the maximum contribution is within the IRS plan limits of the respective benefit spending account plans.

400 Bad Request	Adding or updating a Benefit spending account plan with an invalid coverage level type.	"userMessage": { "codeValue": "JSON Error ", "messageTxt": "JSON parse error : Cannot deserialize value of type `com.adp.wfn.benefits.externalfsa.dto.AccountBasedCoverageLevel$CoverageLevelType` from String \"Emplo\": not one of the values accepted for Enum class:	For now, supported **coverageLevel Types** are the following: • Employee for FSA plans, transit, parking • Employee and EmployeeFamily for HSA plans.

		[EmployeeDependent, EmployeeFamily, EmployeeSpouse, ChildOnly, EmployeeChildren, Employee, SpouseOnly];	
400 Bad Request	Adding a plan with an invalid **productCode**.	userMessage": { "codeValue": "JSON Error ", "messageTxt": "JSON parse error: Cannot deserialize value of type `com.adp.wfn. benefits.plansetup. pojo.ProductCode List` from String \"Hsp\": value not one of declared Enum instance names: [LimitedPurposeFSA, MedicalFSA, DependentCareFSA, HSA, Parking, Transit];}	Supported plan types are: • MedicalFSA, • Limited Purpose FSA • Dependent Care FSA • Parking, Transit The **productCode** types are: • "Parking" • "Transit" • "MedicalFSA" • "LimitedPurposeFSA" • "DependentCareFSA"
400 Bad Request	Adding a plan with invalid **ProviderID**.	{ "code": "BAD_REQUEST", "description": "Invalid PCProvider Id."}	Give a pre-determined **providerID** provided by the external benefit provider.
400 Bad Request	Adding a plan with blank **ProviderID** value.	{ "code": "BAD_REQUEST", "description": "Invalid PCProvider Id."}	
400 Bad Request	Adding a plan with a blank **productCode** value.	"userMessage": { "codeValue": "JSON Error ", "messageTxt": "JSON parse error: Cannot deserialize value of type`com.adp.wfn. benefits.plansetup. pojo.ProductCode List` from String \"Hsp\": value not one of declared Enum instance names: [LimitedPurposeFSA, MedicalFSA, DependentCareFSA, HSA, Parking, Transit];}	
400 Bad Request	Adding a plan with blank **alternateID** value.	{ "code": "BAD_REQUEST", "description": "Alternate Id is not present. "}	
400 Bad Request	Adding a plan with a blank or Invalid **waitingRule EventCode** value.	"userMessage": { "codeValue": "JSON Error ", "messageTxt": "JSON parse error: Cannot deserialize value of type`com.adp.wfn. benefits.plansetup. pojo.WaitingRule Type$WaitingRuleEventCode` from String \"\": value not one of declared Enum instance names: [WorkEvent, NewHire, LifeEvent]}	The waiting rule event can be one of the following: • **WorkEvent** • **NewHire**
400 Bad Request	Adding a plan with blank or Invalid **waitingRule TypeCode** value.	"userMessage": { "codeValue": "JSON Error ", "messageTxt": "JSON parse error: Cannot deserialize value of type `com.adp.wfn. benefits.plansetup. pojo.WaitingRule Type$WaitingRule TypeCode . from String \"\": value not one of declared Enum instance names: [OtherProbation, HireDate, AfterXDays, FirstOfMonthAfter Hire, FirstOfMonthAfterXMonths, FirstPayPeriodAfterXDays, AfterXMonths, NoWaitingRule, FirstOfMonthAfterXDays]}	One of the following values can be used for the **waitingRuleTypeCode**: • "HireDate" • "FirstOfMonth AfterXMonths" • "AfterXMonths" • "AfterXDays" • "FirstOfMonthAfterXDays " • "Other Probation" • "FirstOfMonth AfterHire" • "NoWaiting Rule"

500 Internal Server Error	Adding a plan with blank plan **Anniversary**.	{ "type": "ERROR", "code": "INTERNAL_SERVER_ERROR"}	
500 Internal Server Error	When the request payload is incorrect.	{ "type": "ERROR", "code": "INTERNAL_SERVER_ERROR"}	

Chapter 3

Unsupported Fields

The following ADP Workforce Now fields are not supported through the Benefit Spending Account API.

API Editor guide link : Spending Accounts Plan For ADP WFN

	Section	Field	Note
1	**Plan Settings**	**Plan Group**	**Editable** by an ADP Workforce Now Client Practitioner.
	Plan Settings	**Additional Fields**	
	Plan Settings	**Allow Contribution from Prior Employer**	This is a check box field, editable by an ADP Workforce Now Client Practitioner.
	Plan Settings	**COBRA Eligible Plan**	This is a check box field and cannot be edited by an ADP Workforce Now Client Practitioner.
2	**Eligibility Rules**		**Editable** by an ADP Workforce Now Client Practitioner.
3	**Eligibility Group Details**	**End of Coverage for Terminated Employees**	**Cannot be edited** by an ADP Workforce Now Client Practitioner.
	Eligibility Group Details	**Enrollment Settings**	**Editable** by an ADP Workforce Now Client Practitioner.
4	**Define Contributions**	**Default Cost Period for Employee Contribution**	**Editable** by an ADP Workforce Now Client Practitioner.
	Define Contributions	**Distribution of Employer Contribution**	
5	**Deductions**		**Editable** by an ADP Workforce Now Client Practitioner.

Chapter 4

Known Issues and Limitations

Issue 1: TA4742036 - META Information is not supported the Benefit Spending Account Plan API

Impacted APIs

Method	URI	roleCode Value
POST	/benefits/v1/spending-account-plans/	Practitioner

Suggested Workaround

The workaround is to look through the request payload available in Chapter 2 - Use Case: Creating and Updating a Benefit Spending Account Plan, within the **API Usage** and **Data Dictionary** sections.

Issue 2: On updating a Benefit Plan, the Response status code and the message thrown by the API are inappropriate

Impacted APIs

Method	URI	roleCode Value
POST	/benefits/v1/spending-account-plans/	Practitioner

Issue Description

When a pending or approved plan is updated, the Benefit Spending Account API is throwing 201 Created status code with the following response. But, in this case the API should return a 200 OK status with the response message stating that the plan is updated, instead of displaying 201 Created, as this might confuse the user.


```
{
  "type": "SUCCESS",
  "code": "OK",
  "description": "SUCCESS"
}
```


Suggested Workaround

The workaround is to check with the ADP Workforce Now Practitioner if the plan is updated or a new plan is generated in case of any confusion.

Issue 3: While creating a plan with the planAnniversary field as blank, the Benefit Spending Account API returns a 500 Internal Server Error instead of a 400 Bad Request.

Impacted APIs

Method	URI	roleCode Value
POST	/benefits/v1/spending-account-plans/	Practitioner

Issue Description

While creating a plan with the **planAnniversary** field set at blank, the Benefit Spending Account API returns a 500 Internal Server Error instead of a 400 Bad Request. This might create confusion to the user and this, in turn, adds support tickets.

Suggested Workaround

The workaround is to check with the ADP Workforce Now Practitioner if the plan is updated or a new plan is generated in case of any confusion.

Issue 4: TA4757122 - API doesn't support validation for the alternateID field

Impacted APIs

Method	URI	roleCode Value
POST	/benefits/v1/spending-account-plans/	Practitioner

Issue Description

When a plan is added with an invalid **alternateID**, the Benefit Spending Account Plan API doesn't return any errors.

Issue 5: Updating the planStartDate and alternateID is not supported through the Benefit Spending Account API

Impacted APIs

Method	URI	roleCode Value
POST	/benefits/v1/spending-account-plans/	Practitioner

Suggested Workaround

There are no workarounds available.

Issue 6: Event Notifications are not supported by this Benefit Spending Account API

Impacted APIs

Method	URI	roleCode Value
POST	/benefits/v1/spending-account-plans/	Practitioner

Suggested Workaround

There are no workarounds available.