



PUBLIC

Document Version: 2608 – 2026-08

Guided Buying Functional Documents API

SAP Ariba Buying

SAP Ariba Buying and Invoicing



Content

Guided Buying Functional Documents API.	3
Overview of the Guided Buying Functional Documents API.	4
Workflow for Guided Buying Functional Documents API.	5
Creating and Publishing an Outbound API.	6
Example Outbound API.	7
Receiving Data in Client Applications and Sending It Back to Guided Buying.	13
Important Disclaimers and Legal Information.	16
API-Specific Disclaimers and Legal Information.	18

Guided Buying Functional Documents API

This guide is for buying organizations with SAP Ariba Procurement solutions that want to use externally managed forms.

This API makes it possible for client applications to update the status of externally managed forms, which are forms that are filled out in guided buying and processed by external systems. They allow you to use guided buying as a repository of general-purpose forms and to connect guided buying with client applications.

If you use the "Support for externally managed forms" feature, you must use this API.

This guide applies to:

- SAP Ariba APIs
- SAP Ariba Developer Portal
- SAP Ariba Buying
- SAP Ariba Buying and Invoicing

Related Guides:

[Guided Buying Administration](#)

Overview of the Guided Buying Functional Documents API

This API makes it possible to use client applications to update externally managed forms in guided buying.

This API includes the following endpoints:

PATCH <code>/externalDocuments/{documentId}</code>	Updates the status of externally managed form in guided buying
GET <code>/externalDocuments/{documentId}</code>	(Optional) gets the details of an externally managed form from guided buying

This API supports the following document type:

- Externally managed form

Prerequisites

- You must create an outbound API to send form data from guided buying to your client application. For more information, see [Creating and Publishing an Outbound API \[page 6\]](#).
- To create an application and request access to use this API, you must have access to the SAP Ariba Developer Portal. For details refer to [Steps to Start Using the SAP Ariba APIs](#) in [Help for the SAP Ariba Developer Portal](#).
- All queries made to this API must be authenticated using OAuth authentication. For details refer to [SAP Ariba Developer Portal Authentication](#) in [Help for the SAP Ariba Developer Portal](#).
- To get complete form field values, your client application must call the Custom Forms API after receiving an outbound API call. Custom field values are available only through the Custom Forms API. For more information, see [Custom Forms API](#).
- Guided buying must be configured to support externally managed forms. For more information, see [Guided Buying Administration](#).

Restrictions

Your client application must call the outbound API before calling the Guided Buying Functional Documents API; the outbound API provides data required by the inbound API.

Workflow for Guided Buying Functional Documents API

The **Guided Buying Functional Documents API** is an inbound API that allows client application to update data in forms within guided buying.

Your client application receives data through an outbound API that you provide and it transmits data back through the inbound **Guided Buying Functional Documents API**. It has the information necessary to use the inbound API only if it first receives data from the outbound API.

Note

Your organization must create this outbound API and publish it on the SAP Ariba Developer Portal. For more information about the outbound API, see [Creating and Publishing an Outbound API \[page 6\]](#).

The following steps are an overview of the workflow for this API:

1. An application developer creates an outbound API that enables guided buying to send data to your client application.
For more information, see [Creating and Publishing an Outbound API \[page 6\]](#).
2. An application developer configures your client application to receive data through the outbound API and send data through the inbound **Guided Buying Functional Documents API**.
For more information, see the documentation for your external system.
3. An API administrator registers your client application on the SAP Ariba Developer Portal, which assigns it an OAuth client ID, client secret, and API key. Use the API key in every API call.
For more information, see [Help for the SAP Ariba Developer Portal](#).
4. An API administrator obtains your outbound API and publishes it on the SAP Ariba Developer Portal.
For more information, see [Creating and Publishing an Outbound API \[page 6\]](#).
5. Guided buying users submit an externally managed form.
6. Guided buying uses the outbound API to send identifying information for the form to your client application.
7. Your client application uses the Custom Forms API GET operation to retrieve the list of custom form instances and their data. Then, it searches this list by ID and a date from the outbound API call to find custom fields and their values.
8. Your client application processes the form data and sends status information back to guided buying through the **Guided Buying Functional Documents API**.
For more information, see [Receiving Data in Client Applications and Sending It Back to Guided Buying \[page 13\]](#).

Creating and Publishing an Outbound API

Your organization must provide an outbound API that sends data from guided buying to your client application. Use this procedure to create or purchase this API for your particular client application.

Context

The outbound API might be available from the software vendor that provided your client application. You can create this API yourself or use one from a third-party, but it must pass the fields supplied by guided buying.

The outbound API must use the REST protocol and it must be in swagger format.

For an example, see [Example Outbound API \[page 7\]](#).

After your organization obtains the outbound API, an API administrator publishes it to the SAP Ariba Developer Portal.

Procedure

1. Sign in to the Portal.
2. Click **Manage** in the left pane and click the **My APIs** tab.
3. Publish the API, either by clicking **Get Started** for your first API or by clicking the plus sign (near the **Search** bar on the left) for subsequent APIs.

Use the following parameters in the portal:

Parameter	Description	Recommendation
API name	The name used when the portal displays lists of APIs; for example, ACME Externally Managed Form Outbound API .	You'll use the same name for the value of the guided buying parameter EXTERNALLY_MANAGED_OSG_RULE_NAME , so that guided buying knows which outbound API and client to use.
Target URL	The URL that guided buying uses for the POST endpoint.	This URL is your external system's gateway for API access. It must begin with https:// .
Description	The text that displays in the portal for your API.	You can enter any description.
Allowed tenants	The list of SAP Ariba realms that can use your API.	Select the guided buying realm.
Tags	Optional tags that help you organize APIs.	You can use these tags to filter APIs in lists.

Parameter	Description	Recommendation
Header parameters	Static parameters and values to use in the HTTP header for each call.	Use any parameters required by your client application.
URL query parameters	Static parameters and values to use in the URL query parameters table.	Use any parameters required by your client application.
Authentication type	Method that the server uses to authenticate each call.	Use OAuth 2.0 .

For more information about these parameters, see [SAP Ariba Developer Portal Guide for Administrators](#).

4. In the **Detailed Documentation** section, click **Browse file...**, select your swagger API file, and click **Open**.
5. Click **Publish**.
6. Configure guided buying to use the new outbound API:
 - a. Sign in to guided buying and click **Admin** on the home page.
 - b. In the **Parameters** section, click **Manage parameters**.
 - c. Enter the same value you entered for **API name** for the guided buying parameter named **EXTERNALLY_MANAGED_OSG_RULE_NAME**.

Example Outbound API

Create an outbound API that sends data from guided buying to your client application. It must pass fields supplied by guided buying.

The outbound API must pass the following fields:

Field name	Type	Description	Example
realm	string	The name of the SAP Ariba realm that hosts the guided buying instance where the form was submitted.	ACMECorporate
documentId	string	The guided buying unique identifier for form instances of type "Externally Managed."	EM546
documentProcessName	string	The name of your organization's process under which the form was created.	SupplierApprovalProcess
title	string	The name that the user assigned to the form instance.	Approval request for BigSupplier Inc.
erpVendorId	string	The unique identifier for the supplier, which is assigned by your ERP system.	sid2076
supplierName	string	The name of the supplier that was selected by the user.	BigSupplier Inc.

Field name	Type	Description	Example
formtemplateId	string	The guided buying identifier for the form template, which is assigned during template design.	8a76a11d-97cd-40e2-4125-2641cb7eb74a
formDocumentId	integer	The unique guided buying identifier for the form instance, which is assigned when the user submits a form.	30298
formCreatedDate	date/time	The date and time (ISO 8601) when the user created the form instance.	2020-07-04T12:08:56.235+05:30
formSubmittedDate	date/time	The date and time (ISO 8601) when the form was submitted.	2020-07-04T12:18:56.235+05:30
formApprovalDate	date/time	The date and time (ISO 8601) when the form was approved.	2020-07-04T12:28:56.235+05:30
formLastUpdatedDate	date/time	The date and time (ISO 8601) when the form was last changed.	2020-07-04T12:38:56.235+05:30

The following example swagger listing shows how you might create the outbound API.

Note

You can create a swagger view of this example by copying it and pasting it into the Swagger editor at editor.swagger.io.

```
{
  "swagger": "2.0",
  "info": {
    "description": "API for sending externally managed form data to an external service",
    "version": "1.0",
    "title": "Externally Managed Form Outbound API"
  },
  "tags": [
    {
      "name": "Externally Managed Form Outbound API",
      "description": "Externally Managed Form Outbound API"
    }
  ],
  "schemes": [
    "https"
  ],
  "paths": {
    "/externalDocuments": {
      "post": {
        "tags": [
          "Externally Managed Form Outbound API"
        ],
        "summary": "Submit externally managed form data",
        "operationId": "externalDocumentsPost",
        "consumes": [
          "application/json"
        ]
      }
    }
  }
}
```

```

    "produces": [
      "application/json"
    ],
    "parameters": [
      {
        "in": "body",
        "name": "body",
        "required": true,
        "schema": {
          "$ref": "#/definitions/ExternallyManagedDocumentOutboundRequest"
        },
        "description": "Externally managed form input details"
      }
    ],
    "responses": {
      "200": {
        "description": "Successfully sent externally managed form",
        "schema": {
          "$ref": "#/definitions/ExternallyManagedDocumentOutboundResponse"
        }
      },
      "400": {
        "description": "Invalid externally managed form request",
        "schema": {
          "$ref": "#/definitions/ErrorResponse"
        }
      },
      "401": {
        "description": "Unauthorized",
        "schema": {
          "$ref": "#/definitions/ErrorResponse"
        }
      },
      "403": {
        "description": "Forbidden",
        "schema": {
          "$ref": "#/definitions/ErrorResponse"
        }
      },
      "500": {
        "description": "Internal server error",
        "schema": {
          "$ref": "#/definitions/ErrorResponse"
        }
      }
    }
  },
  "definitions": {
    "ExternallyManagedDocumentOutboundRequest": {
      "description": "Input for sending externally managed form data to external service",
      "type": "object",
      "required": [
        "realm",
        "documentId",
        "documentProcessName",
        "title",
        "erpVendorId",
        "supplierName",
        "formTemplateId",
        "formDocumentId",
        "formCreatedDate"
      ],
      "properties": {
        "realm": {
          "type": "string",

```

```

    "minLength": 2,
    "maxLength": 192,
    "description": "Realm where the externally managed form was created"
  },
  "documentId": {
    "type": "string",
    "description": "Externally managed form ID"
  },
  "documentProcessName": {
    "type": "string",
    "minLength": 1,
    "maxLength": 255,
    "description": "Process name defined for the externally managed form"
  },
  "title": {
    "type": "string",
    "minLength": 1,
    "maxLength": 50,
    "description": "Externally managed form title"
  },
  "erpVendorId": {
    "type": "string",
    "minLength": 1,
    "maxLength": 50,
    "description": "erpVendorId (uniqueName)"
  },
  "supplierName": {
    "type": "string",
    "minLength": 1,
    "maxLength": 255,
    "description": "Supplier Name"
  },
  "smVendorId": {
    "type": "string",
    "minLength": 1,
    "maxLength": 255,
    "description": "smVendorId"
  },
  "anId": {
    "type": "string",
    "minLength": 1,
    "maxLength": 255,
    "description": "Ariba Network ID"
  },
  "otherSupplierIds": {
    "type": "array",
    "description": "A list of domain/value pairs for other IDs for the supplier",
    "items": {
      "$ref": "#/definitions/OrganizationID"
    }
  },
  "isPreferredSupplier": {
    "type": "boolean",
    "example": false,
    "description": "Is preferred supplier"
  },
  "justification": {
    "type": "string",
    "minLength": 2,
    "maxLength": 1024,
    "description": "Justification text entered by the user when submitting the form"
  },
  "formTemplateId": {
    "type": "string",
    "minLength": 2,
    "maxLength": 256,

```

```

        "description": "Form template ID"
    },
    "formDocumentId": {
        "type": "integer",
        "description": "Internal form document ID"
    },
    "formCreatedDate": {
        "type": "string",
        "description": "The time the form was created (in ISO8601, GMT)",
        "format": "date-time",
        "example": "2019-07-04T12:08:56.235+05:30"
    },
    "formApprovedDate": {
        "type": "string",
        "description": "If the form was approved, the time the form was approved (in ISO8601, GMT)",
        "format": "date-time"
    },
    "formSubmittedDate": {
        "type": "string",
        "description": "The time the form was submitted (in ISO8601, GMT)",
        "format": "date-time"
    },
    "formLastUpdatedDate": {
        "type": "string",
        "description": "The time the form was last updated (in ISO8601, GMT)",
        "format": "date-time"
    }
}
},
"OrganizationID": {
    "type": "object",
    "required": [
        "domain"
    ],
    "properties": {
        "domain": {
            "type": "string",
            "description": "domain name"
        },
        "value": {
            "type": "string",
            "description": "domain value"
        }
    },
    "description": "The domain/pair for a supplier, possible domain names: ANORG, internalsupplierid, S4, transactionnetworkid, SSP, sourcingnetworkid etc."
},
"ExternallyManagedDocumentOutboundResponse": {
    "type": "object",
    "properties": {
        "status": {
            "type": "string",
            "description": "Status returned by the external system for its internal purpose"
        },
        "messageId": {
            "type": "string",
            "description": "Message ID returned by the external system for its internal purpose"
        }
    }
},
"ErrorResponse": {
    "type": "object",
    "required": [
        "status",

```

```
    "error":  
    ],  
    "properties": {  
      "status": {  
        "type": "integer",  
        "format": "int32",  
        "description": "HTTP status code"  
      },  
      "error": {  
        "type": "string",  
        "description": "Error message"  
      },  
      "description": {  
        "type": "string",  
        "description": "Error description"  
      }  
    },  
    "description": "The error response"  
  }  
}
```

Receiving Data in Client Applications and Sending It Back to Guided Buying

This procedure describes how to receive data in client applications and send it back to guided buying. When users submit an externally managed form, guided buying sends it to your client application. Your client application receives it, processes it, and sends status information back to guided buying, which adds it to the form.

Context

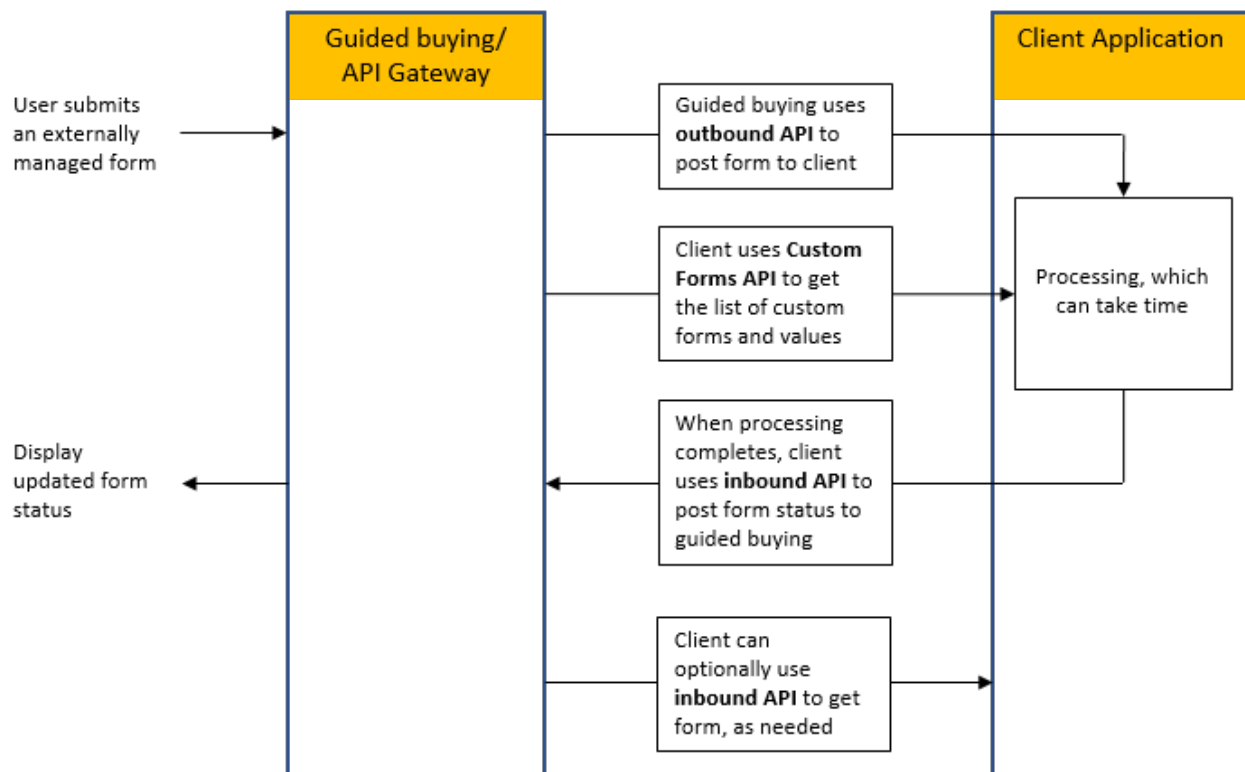
Guided buying calls the outbound API and later, your client application calls the inbound API.

To set up the outbound API, see [Creating and Publishing an Outbound API \[page 6\]](#).

Note

In the sample URLs in this topic, replace `{{runtime_url}}` with the desired runtime URL from the **Environment Details** table on the SAP Ariba Developer Portal discovery page for this API.

The following figure illustrates how data flows from an externally managed form in guided buying to the client application, and back to guided buying.



Procedure

1. A user fills out and submits an externally managed form and guided buying saves it in the **Other requests** tab in the **Your requests** page.
2. Guided buying uses the value of the **EXTERNALLY_MANAGED_OSG_RULE_NAME** parameter to identify the client and outbound API to use.
3. Guided buying uses the outbound API POST endpoint to send the data from the form to your client application. For example:

```
POST {{runtime_url}}/externalDocuments
```

4. Your client application uses the Ariba Custom Forms API to get the list of custom form instances and their field data:

```
GET {{runtime_url}}/forms
```

5. Your client application searches the retrieved data by `formDocumentId` and any date value (`formCreateDate`, `formApprovalDate`, `formSubmittedDate`, or `formLastUpdatedDate`) from the outbound API message to find custom field values.
6. Your client application begins processing the data.

The following points describe error handling:

- If the outbound request fails (for example, if your client application isn't available), guided buying retries every hour up to 24 times. If there's no response after 24 attempts, guided buying discontinues and sets the status of the form to **Error**, which is visible to the user.
 - If you receive multiple outbound requests for the same form (for example, if guided buying is interrupted after sending the form), your client application should process the last-received form and use it to replace any existing ones.
7. (Optional) if your client application needs to refresh its internal data with data from the form, it uses the inbound API GET endpoint:

```
GET {{runtime_url}}/externalDocuments/{documentId}{realm}
```

where `documentId` is the ID of the externally managed form assigned by guided buying and `realm` is the name of your guided buying realm.

For example:

```
https://openapi.ariba.com/api/externalDocuments/EM794?realm=OurRealm
```

Guided buying responds with the data from the form.

8. After your client application finishes processing the data, it creates a short status update document. For example:

```
{
  "externalReferenceId": "EM794",
  "externalStatus": "Approved",
  "externalTransactionId": "DD37328",
  "externalTransactionType": "CONTRACTWORKSPACE",
  "requestedAction": "Processed"
}
```

The `externalReferenceId`, `externalStatus`, and `externalTransactionId` attributes can have any values.

The `externalTransactionType` attribute must have the value `CONTRACTCOMPLIANCE`, `CONTRACTWORKSPACE`, or `REQUISITION`.

The `requestedAction` attribute must have the value `Processed`.

9. Your client application sends form status back to guided buying by using the inbound API PATCH endpoint:

```
PATCH {{runtime_url}}/externalDocuments{documentId}
```

📌 Note

Your client application can take as long as it needs to send results back; guided buying uses the `documentId` parameter to find the original form. The client can update the form multiple times with multiple PATCH operations, if the form status changes

10. Guided buying adds the received data to the form. Users can view the form in the **Other requests** tab in the **Your requests** page.

Important Disclaimers and Legal Information

Coding Samples

Any software coding and/or code lines / strings ("Code") included in this documentation are only examples and are not intended to be used in a productive system environment. The Code is only intended to better explain and visualize the syntax and phrasing rules of certain coding. Neither SAP nor CDC warrants the correctness and completeness of the Code given herein, and SAP and CDC shall not be liable for errors or damage caused by the usage of the Code.

Accessibility

The information contained in the Documentation represents both CDC's and SAP's current view of accessibility criteria as of the date of publication; it is in no way intended to be a binding guideline on how to ensure accessibility of software products. SAP and CDC disclaim any liability in relation to this Guide .. Furthermore, this Guide does not result in any direct or indirect contractual obligations of SAP and /or CDC

Gender-Neutral Language

As far as possible, SAP documentation is gender neutral. Depending on the context, the reader is addressed directly with "you", or a gender-neutral noun (such as "sales person" or "working days") is used. If when referring to members of both sexes, however, the third-person singular cannot be avoided or a gender-neutral noun does not exist, SAP reserves the right to use the masculine form of the noun and pronoun. This is to ensure that the documentation remains comprehensible.

Internet Hyperlinks

The Documentation may contain hyperlinks to the Internet. These hyperlinks are intended to serve as a hint about where to find related information. Neither CDC nor SAP warrants the availability and correctness of this related information or the ability of this information to serve a particular purpose. SAP and CDC shall not be liable for any damage caused by the use of related information. Links are categorized for transparency (see: <http://help.sap.com/disclaimer>).

Copyright © 2016-2024 SAP SE and its Affiliates. All rights reserved.

This Documentation, as well as the solutions, software and/or services described in it, contain proprietary information. They are provided under a license or other agreement containing restrictions on use and disclosure and are also protected by copyright, patent and/or other intellectual property laws. Except as permitted by such agreement, no part of the document may be reproduced or transmitted in any form by any means, electronic, mechanical or otherwise, without the prior written permission of SAP. Neither SAP nor CDC assumes any responsibility or liability for any errors or inaccuracies that may appear in this Guide. The information contained in this Guide is subject to change without notice.

'SAP' and 'Ariba', and the solutions, as well as their respective logos, are trademarks or registered trademarks of Ariba, Inc. in the United States and other countries. Please see <http://www.ariba.com/legal/trademarks> for additional trademark information and notices.

SAP Ariba Sourcing Solutions are protected by one or more of the following patents, including without limitation:

U.S. Patent Nos. 6,199,050; 6,216,114; 6,223,167; 6,230,146;

6,230,147; 6,285,989; 6,408,283; 6,499,018; 6,564,192; 6,871,191;

6,952,682; 7,010,511; 7,072,061; 7,130,815; 7,146,331;

7,152,043; 7,225,152; 7,277,878; 7,249,085; 7,283,979; 7,283,980;
7,296,001; 7,346,574; 7,383,206; 7,395,238; 7,401,035; 7,407,035;
7,444,299; 7,483,852; 7,499,876; 7,536,362; 7,558,746; 7,558,752;
7,571,137; 7,599,878; 7,634,439; 7,657,461; 7,693,747; 8,364,577;
and 8,392,317. Patents pending.

Other SAP Ariba Solutions are protected by one or more of the following patents:

U.S. Patent Nos. 6,199,050, 6,216,114, 6,223,167, 6,230,146,
6,230,147, 6,285,989, 6,408,283, 6,499,018, 6,564,192, 6,584,451,
6,606,603, 6,714,939, 6,871,191, 6,952,682, 7,010,511, 7,047,318,
7,072,061, 7,084,998; 7,117,165; 7,225,145; 7,324,936; 7,536,362;
8,364,577; and 8,392,317. Patents pending.

Certain SAP Ariba Solutions may include third party software or other intellectual property licensed from a third party. For information regarding software or other intellectual property licensed from a third party, go to <http://www.ariba.com/copyrights.cfm>.

API-Specific Disclaimers and Legal Information

The SAP Ariba Developer Portal included in the SAP Ariba APIs product and the APIs made available on this site are provided solely at the discretion of SAP without warranty of any kind, and SAP may change, suspend, or cancel any or all features or functions of the SAP Ariba APIs product or revise the website at any time. Any production use of or commercialization of applications containing any APIs provided on this website is prohibited without a written agreement between your company and SAP governing such activities.

Access to this API is available to you as a subscriber to this solution as part of the SAP Cloud Service Level Agreement. However, it is not considered part of the solution. Use of this API is purely optional and is subject to restrictions stated in the documentation, including the Terms of Use and the documentation found at the SAP Ariba Developer Portal (see <https://developer.ariba.com/api>). If you wish to connect a third-party service using this API, first confirm that the company is participating in the SAP partner program and is authorized to provide a connection to this solution using this API. You will be required to submit written consent to SAP to authorize the exchange of data with the third-party service.

Important Disclaimers and Legal Information

Hyperlinks

Some links are classified by an icon and/or a mouseover text. These links provide additional information.

About the icons:

- Links with the icon  : You are entering a Web site that is not hosted by SAP. By using such links, you agree (unless expressly stated otherwise in your agreements with SAP) to this:
 - The content of the linked-to site is not SAP documentation. You may not infer any product claims against SAP based on this information.
 - SAP does not agree or disagree with the content on the linked-to site, nor does SAP warrant the availability and correctness. SAP shall not be liable for any damages caused by the use of such content unless damages have been caused by SAP's gross negligence or willful misconduct.
- Links with the icon  : You are leaving the documentation for that particular SAP product or service and are entering an SAP-hosted Web site. By using such links, you agree that (unless expressly stated otherwise in your agreements with SAP) you may not infer any product claims against SAP based on this information.

Videos Hosted on External Platforms

Some videos may point to third-party video hosting platforms. SAP cannot guarantee the future availability of videos stored on these platforms. Furthermore, any advertisements or other content hosted on these platforms (for example, suggested videos or by navigating to other videos hosted on the same site), are not within the control or responsibility of SAP.

Beta and Other Experimental Features

Experimental features are not part of the officially delivered scope that SAP guarantees for future releases. This means that experimental features may be changed by SAP at any time for any reason without notice. Experimental features are not for productive use. You may not demonstrate, test, examine, evaluate or otherwise use the experimental features in a live operating environment or with data that has not been sufficiently backed up.

The purpose of experimental features is to get feedback early on, allowing customers and partners to influence the future product accordingly. By providing your feedback (e.g. in the SAP Community), you accept that intellectual property rights of the contributions or derivative works shall remain the exclusive property of SAP.

Example Code

Any software coding and/or code snippets are examples. They are not for productive use. The example code is only intended to better explain and visualize the syntax and phrasing rules. SAP does not warrant the correctness and completeness of the example code. SAP shall not be liable for errors or damages caused by the use of example code unless damages have been caused by SAP's gross negligence or willful misconduct.

Bias-Free Language

SAP supports a culture of diversity and inclusion. Whenever possible, we use unbiased language in our documentation to refer to people of all cultures, ethnicities, genders, and abilities.

Copyright © 2026 Ariba, Inc. All rights reserved.

This documentation, as well as the Ariba solutions, software and/or services described in it, contain proprietary information. They are provided under a license or other agreement containing restrictions on use and disclosure and are also protected by copyright, patent and/or other intellectual property laws. Except as permitted by such agreement, no part of the document may be reproduced or transmitted in any form by any means, electronic, mechanical or otherwise, without the prior written permission of Ariba, Inc.

Ariba, Inc. assumes no responsibility or liability for any errors or inaccuracies that may appear in the documentation. The information contained in the documentation is subject to change without notice.

Ariba and Ariba products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of Ariba, Inc. in the United States and other countries. Please see <http://www.ariba.com/legal/trademarks> for additional trademark information and notices.

Ariba Sourcing solutions (On Demand and software) are protected by one or more of the following patents, including without limitation: U.S. Patent Nos. 6,199,050; 6,216,114; 6,223,167; 6,230,146; 6,230,147; 6,285,989; 6,408,283; 6,499,018; 6,564,192; 6,871,191; 6,952,682; 7,010,511; 7,072,061; 7,130,815; 7,146,331; 7,152,043; 7,225,152; 7,277,878; 7,249,085; 7,283,979; 7,283,980; 7,296,001; 7,346,574; 7,383,206; 7,395,238; 7,401,035; 7,407,035; 7,444,299; 7,483,852; 7,499,876; 7,536,362; 7,558,746; 7,558,752; 7,571,137; 7,599,878; 7,634,439; 7,657,461; 7,693,747; 8,364,577; and 8,392,317. Patents pending.

Other Ariba product solutions are protected by one or more of the following patents:

U.S. Patent Nos. 6,199,050, 6,216,114, 6,223,167, 6,230,146, 6,230,147, 6,285,989, 6,408,283, 6,499,018, 6,564,192, 6,584,451, 6,606,603, 6,714,939, 6,871,191, 6,952,682, 7,010,511, 7,047,318, 7,072,061, 7,084,998; 7,117,165; 7,225,145; 7,324,936; 7,536,362; 8,364,577; and 8,392,317. Patents pending.

Certain Ariba products may include third party software or other intellectual property licensed from a third party. For information regarding software or other intellectual property licensed from a third party, go to <http://www.ariba.com/copyrights.cfm>.

