



# Tungsten Supplier Portal API Guide

Version: 4.9.0

Date: 2024-07-18

**TUNGSTEN**  
**AUTOMATION**

© 2014–2024 Tungsten Automation. All rights reserved.

Tungsten and Tungsten Automation are trademarks of Tungsten Automation Corporation, registered in the U.S. and/or other countries. All other trademarks are the property of their respective owners. No part of this publication may be reproduced, stored, or transmitted in any form without the prior written permission of Tungsten Automation.

# Table of Contents

About this guide.....	4
Basic structure.....	5
Authentication ticket.....	5
ERP system ID.....	5
Content type.....	5
Accept.....	6
Services.....	7
Authentication.....	7
Base URL.....	7
Operations.....	7
Document.....	9
Base URL.....	9
Operations.....	9
Bulk operations.....	18
Synchronization of documents.....	25
Suppliers.....	26
Base URL.....	26
Operations.....	26
SearchHelp.....	34
Base URL.....	34
Operations.....	34
Errors.....	37
500 status code.....	37
403 status code.....	37
Data model.....	39
Process Director Accounts Payable.....	39
Incoming invoice.....	47
Purchase order.....	52
Inner document - PO flip.....	60
Vendor master data.....	61

# About this guide

This guide contains the specifications for the Tungsten Supplier Portal web service API.

# Basic structure

The following HTTP headers are important when using the API.

## Authentication ticket

The authentication ticket identifies a user session context in the Supplier Portal web service. After successful authentication with the web service, an authenticated session ticket is sent back as a response. This ticket allows the web service to identify the authenticated user. It must be included in all subsequent calls to Supplier Portal.

```
pdapp-auth-ticket: <ticket> (required)
```

## ERP system ID

The ERP system ID is a unique identifier that helps Supplier Portal to relate a document to its source ERP system, especially when multiple ERP systems synchronize with a single Supplier Portal tenant. The authenticated web service session is locked in to the particular ERP system.

If only one ERP system is synchronized with Supplier Portal, then supplying the ERP system ID as a header is optional. If supplied, each synchronized document is tagged with a system ID. If not, no tag is added.

If a system ID is supplied, it must be included as a header value during the user authentication call. If the user session is successfully authenticated, that user session is locked in to this system ID and all the synchronization calls affect only the documents that are tagged with the supplied system ID.

```
erpsystemid: <erp system-id> (required)
```

## Content type

The content type defines the format of the data entities that are passed to the web service.

```
Content-Type: <content type> (required)
```

The Supplier Portal web service supports both the JSON and XML formats. It also supports the stream content type that can be used to upload binary content such as images and files.

Example:

- Content-Type: application/xml
- Content-Type: application/json

- Content-Type: application/octet-stream

## Accept

The accept value defines the format of the data entities that are passed as a result of the web service call.

Accept: <content type> (required)

The Supplier Portal web service supports both the JSON and XML formats. It also supports the stream content type that can be used to upload binary content such as images and files.

### Example:

- Accept: application/xml
- Accept: application/json
- Accept: application/octet-stream

# Services

This section defines the services that are included in the Supplier Portal web service API:

- [Authentication](#)
- [Document](#)
- [Suppliers](#)
- [SearchHelp](#)

## Authentication

This service allows the calling client to establish a web service session to send requests to the Supplier Portal web service. Currently, only the cookie-based authentication mechanism is supported, and the authentication credentials must be supplied as a combination of a username and password.

### Base URL

`https://{tenant-endpoint}/supplier-portal/rest/sync/authentication`

Example:

`https://sp-dev.kofaxdev.com/supplier-portal/rest/sync/authentication`

**i** For backward compatibility, the legacy application context name `pdweb-idaho` is still supported, but rerouted to `supplier-portal`.

## Operations

Name	URI	Method
authenticate	/authenticate	POST
logout	/logout	POST

### authenticate

This service is used to authenticate a Supplier Portal user against the Supplier Portal endpoint, using a username and password as authentication credentials.

Example for content-type: application/json:

```
{
  "UserName" : "...",
  "Password" : "..."}
}
```

Example for content-type: application/xml:

```
<AuthenticationCredentials>
  <UserName>.....</UserName>
  <Password>.....</Password>
</AuthenticationCredentials>
```

The authenticating user must have special permissions granted in Supplier Portal to be allowed to make REST API synchronization calls. If the authentication is successful, the service returns the HTTP status code 200, with the following content.

Response example for accept: application/json:

```
{
  "Status" : "true",
  "Ticket" : "this will be a cookie value"
}
```

Response example for accept: application/xml:

```
<Authentication>
  <Status>>true</Status>
  <Ticket>this will be a cookie value</Ticket>
</Authentication>
```

If the authentication fails due to wrong credentials, the service sends the response with the HTTP status code 403, with the response entity.

Response example for accept: application/json:

```
{
  "Code" : 701,
  "Message" : "Username or password is incorrect"
}
```

Response example for accept: application/xml:

```
<Exception>
  <Code>701</Code>
  <Message>Username or password is incorrect</Message>
</Exception>
```

If an error occurs during the authentication process, a similar entity is sent with an appropriate code to identify the error type, as well as a message that provides a description of the error.

Example:

Code	Description
185	Identifies database connectivity issues

## logout

This service must be called to end the authenticated user session. If this service is not called to end the session, the web service automatically ends the session after a certain period of idle time. By default, this time is set to 30 minutes, however, it can be configured in Supplier Portal.



## Document

This service allows communication of documents. In this context, documents are business entities such as invoices or purchase orders that Supplier Portal is configured to handle. The web service allows communication between the following document types and notations:

Document type	Notation
Invoice	IV
Purchase Order	PO
Vendor Master Data	MA
Synchronization Logs	MO

### Base URL

`https://{tenant-endpoint}/supplier-portal/rest/sync/document`

Example:

`https://sp-dev.kofaxdev.com/supplier-portal/rest/sync/document`

### Operations

The following operations can be performed on single documents. For information on operations for multiple documents, see [Bulk operations](#).

Name	URI	Method
archive	<code>/documentID/archive</code>	GET
attachments	<code>/documentID/attachments</code>	GET
changedDocuments	<code>/changedDocuments/{docType}</code>	POST
commitAttachmentSynchronization	<code>/documentID/commitAttachmentSynchronization/{attachmentID}</code>	POST
delete	<code>/documentID/delete</code>	POST
exists	<code>/documentID/exists</code>	POST
exportFields	<code>/exportFields/{docType}</code>	GET
attachment	<code>/documentID/attachment/{attachmentID}</code>	GET
getChanges	<code>/documentID/getChanges</code>	POST
indexReady	<code>/indexReady/{docType}</code>	GET
newAttachments	<code>/documentID/newAttachments</code>	GET
newDocuments	<code>/newDocuments/{docType}</code>	POST
querylist	<code>/querylist</code>	POST
retrieve	<code>/documentID/retrieve</code>	POST

Name	URI	Method
submit	/submit	POST
terminateChangeSynchronization	/{{documentID}}/terminateChangeSynchronization	POST
terminateNewSynchronization	/{{documentID}}/terminateNewSynchronization	POST
update	/update	POST
upload	/{{documentID}}/attachment	POST

## archive

This operation deletes the document and all the related audit trail entries for that document from the database. As a response, this method sends a summary of all the audit trail entries for that document in PDF format, for archiving in the ERP system.

## attachments

This operation gets all the attachments for a specific document.

Example for accept application/json response:

```
{
  "Attachment" : [{
    "AttachmentID" : "{attachment-id}",
    "ActionID" : "{action-id}",
    "AttachmentType" : "{attachment-type}",
    "State" : "{state}",
    "Type" : "{type}"
  }]
}
```

Example for accept: application/xml:

```
<Attachments>
  <Attachment>
    <AttachmentID>{attachment-id}</AttachmentID>
    <ActionID>{action-id}</ActionID>
    <AttachmentType>{action-id}</AttachmentType>
    <State>{state}</State>
    <Type>{type}</Type>
  </Attachment>
</Attachments>
```

## changedDocuments

This operation queries for all the documents that have been changed since the last Synchronizer run. Optionally, a query filter as described in the [querylist](#) operation can be sent to filter the returned results. If filtering is not required, an empty entity should be sent as follows:

`{}` for JSON

or

`<QueryFilter/>` for XML

This operation requires a document type notation as part of the URL parameter. If no query filter is applied, the operation will return all the results that have changed.

Example for accept: application/json response:

```
{
  "Item" : [
    {document-id},
    {document-id},
    {document-id},
    ...
  ]
}
```

Example for accept: application/xml response:

```
<List>
  <Item> {document-id} </Item>
  <Item> {document-id} </Item>
  <Item> {document-id} </Item>
  ...
</List>
```

## commitAttachmentSynchronization

This operation marks the synchronization of a particular attachment of a specified document as complete.

The operation expects a new ID to be set for the synchronized attachment in Supplier Portal. This is useful if an ERP system generates its own IDs and denies externally generated IDs. In such cases, it is important to keep the IDs between the ERP system and Supplier Portal consistent.

If the ERP system accepts external IDs (IDs generated by Supplier Portal), then the same ID should be sent. The ID must be sent as the body of the request.

Upon successful operation, the following is the response of the web service.

Example for accept: application/json response:

```
{
  "Status": true,
  "Message": ""
}
```

Example for accept: application/xml response:

```
<DocumentProcess>
  <Statustrue</Status>
  <Message</Message>
</DocumentProcess>
```

## delete

This operation deletes the document with the specified ID from Supplier Portal. Upon successful operation, the following is the response of the web service.

Example for accept: application/json response:

```
{
  "Status": true,
```

```

    "Message": ""
  }

```

Example for accept: application/xml response:

```

<DocumentProcess>
  <Status>true</Status>
  <Message></Message>
</DocumentProcess>

```

## exists

This operation checks whether or not the document with the specified ID exists in Supplier Portal. Upon successful operation, the web service returns the following response. The status value indicates whether the document exists.

Example for accept: application/json response:

```

{
  "Status": true,
  "Message": ""
}

```

Example for accept: application/xml response:

```

<DocumentProcess>
  <Status>true</Status>
  <Message></Message>
</DocumentProcess>

```

## exportFields

This operation returns detailed information about the data that must be included for a particular document type during the synchronization of the document. Successive calls to Supplier Portal should include the data that is specified in the returned information.

The data to be included is defined in Supplier Portal. This minimizes the amount of data being transmitted back and forth between the client and the web service for each document. Additionally, this serves as added security by limiting the amount of data that is transferred to Supplier Portal from the ERP system.

Upon successful operation, this operation returns the following response entities for the Invoice document type.

Example for accept: application/json response:

```

{
  "Header" : ["DOCNO"],
  "Table" : [{
    "@id" : "ITEM",
    "Field" : ["PO_ITEM", "PO_NUMBER"]
  }, {
    "@id" : "TAX",
    "Field" : ["CURRENCY", "TAX_CODE"]
  }
  ]
}

```

Example for accept: application/xml response:

```

<ExportList>
  <Header>DOCNO</Header>
  <Header>VENDOR_NO</Header>
  <Table @id="ITEM">
    <Field>PO_ITEM</Field>
    <Field>PO_NUMBER</Field>
  </Table>
  <Table @id="TAX">
    <Field>CURRENCY</Field>
    <Field>TAX_CODE</Field>
  </Table>
</ExportList>

```

**i** The returned entities shown above are just examples. The amount of data returned for real entities can be much higher, depending on the fields that are configured in Supplier Portal.

## attachment

This operation returns an attachment with a specified ID, for a document with a specified ID. The response includes these entities as `Content-Disposition` and `Content-Type`, as specified below:

```

Content-Disposition: attachment;mime-type="application/pdf";
size=3087001;attachmentid={id}Content-Type: application/pdf;charset=UTF-8

```

The `mime-type` of the entity depends on the type of attachment.

## getChanges

This operation returns a document with a specified ID, containing only the changed data. The data returned is similar to the [retrieve](#) operation.

## indexReady

This operation returns detailed information about the database readiness to accept a document of a particular type.

Upon successful operation, this operation returns the following response entities for the Invoice document type.

Example for accept: `application/json` response:

When the database is ready for acceptance:

```

{
  "Status": "true"
}

```

When the database is not ready for acceptance:

```

{
  "Status": "false",
  "ChangesDone": "64135",
  "ChangesTotal": "64736"
}

```

The value that is returned represents the number of documents already indexed and the total number of documents to be indexed.

## newAttachments

This operation returns all the attachments that have been newly added to the document since the last `commitAttachmentSynchronization` operation. The response is similar to the [attachment](#) operation.

## newDocuments

This operation returns the IDs of all the new documents. Optionally, a query filter as described in the [querylist](#) operation can be sent to filter the returned results. If filtering is not required, an empty entity should be sent, as follows.

`{}` for JSON

or

`<QueryFilter/>` for XML.

This operation requires a document type notation as part of the URL parameter. If no query filter is applied, the operation returns all the results that have been marked as new.

## querylist

This operation allows you to perform a query in Supplier Portal, based on the specified filter.

With `Headers` set to `false`, the operation returns a list of document IDs that match the query filter. With `Headers` set to `true`, the operation returns a list of document IDs that match the query filter, together with the header fields. The size of the returned list can be controlled by the `MaxHits` value.

Example for `application/json` content-type:

```
{
  "SingleFilter": [
    {
      "Field": "VENDOR_NO",
      "Value": "1998"
    },
    {
      "Field": "GROSS_AMOUNT",
      "Value": "100"
    }
  ],
  "RangeFilter": [
    {
      "Field": "MOD_DATE",
      "Start": "20120301",
      "End": "20120329"
    }
  ],
  "MaxHits": "100",
  "Headers": false,
  "DocumentType": "IV"
}
```

Example for `application/xml` content-type:

```
<QueryFilter>
```

```

<DocumentType>IV</DocumentType>
<Headers>>false</Headers>
<SingleFilter>
  <Field>VENDOR_NO</Field>
  <Value>1998</Value>
</SingleFilter>
<SingleFilter>
  <Field>GROSS_AMOUNT</Field>
  <Value>100</Value>
</SingleFilter>
<RangeFilter>
  <Field>MOD_DATE</Field>
  <Start>20120301</Start>
  <End>20120329</End>
</RangeFilter>
<MaxHits>100</MaxHits>
</QueryFilter>

```

For single filters, if you do not want to filter, specify ALL as the filter value. For example, if you want to list all the invoices (document type IV), the filter would be:

```

<QueryFilter>
  <DocumentType>IV</DocumentType>
  <Headers>>false</Headers>
  <SingleFilter>
    <Field>VENDOR_NO</Field>
    <Value>ALL</Value>
  </SingleFilter>
  <MaxHits>100</MaxHits>
</QueryFilter>

```

Example for accept: application/json response:

```

{
  "Item" : [
    {document-id},
    {document-id},
    {document-id},
    ...
  ]
}

```

Example for: accept: application/xml

```

<List>
  <Item>{document-id}</Item>
  <Item>{document-id}</Item>
  <Item>{document-id}</Item>
  ...
</List>

```

## retrieve

This operation returns a document with a specified ID. The document ID to be retrieved is passed as a part of the operations URL.

## submit

This operation allows you to submit a document to Supplier Portal. The document to be submitted is included as the request body.

Upon successful operation, the following is the response of the web service.

Example for accept: application/json response:

```
{
  "Status": true,
  "Message": "{External-ID}"
}
```

Example for accept: application/xml response:

```
<DocumentProcess>
  <Status>true</Status>
  <Message>{External-ID}</Message>
</DocumentProcess>
```

The value that is returned as `External-ID` represents the document ID that should be used to refer to the submitted document in the future.

## terminateChangeSynchronization

This operation is called to mark a specific changed document as having been completely synchronized. This guarantees that immediate calls to the `getChanges` operation exclude this document until the next change.

The document ID is passed as a part of the operations URL. Upon successful operation, the web service returns the following response:

Example for accept: application/json response:

```
{
  "Status": true,
  "Message": ""
}
```

Example for accept: application/xml response:

```
<DocumentProcess>
  <Status>true</Status>
  <Message></Message>
</DocumentProcess>
```

## terminateNewSynchronization

This operation is called to mark a specific new document as having been completely synchronized. This guarantees that future calls to the `newDocuments` operation exclude this document.

The document ID is passed as part of the operations URL. The operation expects the updated document as a request body. The updated document contains the updates after it has been submitted to the target ERP system.

Upon successful operation, the web service returns the following response:

Example for accept: application/json response:

```
{
  "Status": true,
  "Message": ""
}
```



Example for accept: application/xml response:

```
<DocumentProcess>
  <Status>true</Status>
  <Message></Message>
</DocumentProcess>
```

## update

This operation updates the document. The content of the updated document is passed as the request body.

Upon successful operation, the web service returns the following response:

Example for accept: application/json response:

```
{
  "Status": true,
  "Message": ""
}
```

Example for accept: application/xml response:

```
<DocumentProcess>
  <Status>true</Status>
  <Message></Message>
</DocumentProcess>
```

## upload

This operation allows you to upload an attachment to the document with the specified ID. The document ID is passed as part of the operations URL. In addition, the request must contain the proper entity for Content-Disposition, as shown below:

```
Content-Disposition: attachment;mime-type="application/pdf";
size=3087001;attachmentid={id};description={description};
filename={filename};name={name};actionid={actionid};show={show}
```

The header value for Content-Type should be set as

```
application/octet-stream
```

Upon successful operation, the web service returns the following response:

Example for accept: application/json response:

```
{
  "Status": true,
  "Message": ""
}
```

Example for accept: application/xml response:

```
<DocumentProcess>
  <Status>true</Status>
  <Message></Message>
</DocumentProcess>
```

## Bulk operations

The following operations can be performed on multiple documents.

Name	URI	Method
attachmentsInfos	/attachmentsInfos	POST
commitAttachmentsSynchronization	/commitAttachmentsSynchronization	POST
exists (bulk)	/exists	POST
retrieve (bulk)	/retrieve	POST
submitbulk	/submitbulk	POST
terminateChangeSynchronization (bulk)	/terminateChangeSynchronization	POST
terminateNewSynchronization (bulk)	/terminateNewSynchronization	POST
updatebulk	/updatebulk	POST

### attachmentsInfos

Bulk operation for the [attachments](#) operation.

This operation receives a list of document IDs and returns the attachment metadata (IDs, size, file type, exact document ID, and so on) from Supplier Portal for the specified documents.

Request example for accept: application/json:

```
{
  "id_list": [
    {
      "@id": "0000005557"
    },
    {
      "@id": "0000005558"
    },
    {
      "@id": "0000005560"
    }
  ]
}
```

Response example for accept: application/json:

```
{
  "Attachment": [
    {
      "AttachmentID": "005056B517531EEA939434A49F128F09",
      "AttachmentType": "ZEIC_PDF",
      "State": 0,
      "Type": 0,
      "Size": 0,
      "@id": "005056B517531EEA9394265739FA4EF4"
    },
    {
      "AttachmentID": "005056B517531EEA93944B3B7AEB0F23",
      "AttachmentType": "ZEIC_PDF",

```

```

        "State": 0,
        "Type": 0,
        "Size": 0,
        "@id": "005056B517531EEA9394265739FA4EF4"
    },
    {
        "AttachmentID": "005056B517531EEA939463FDDA77CF41",
        "AttachmentType": "ZEIC_PDFA",
        "State": 0,
        "Type": 0,
        "Size": 0,
        "@id": "005056B54A821EEA93AE77F833BAB89E"
    },
]
}

```

## commitAttachmentsSynchronization

Bulk operation for the [commitAttachmentSynchronization](#) operation.

This operation receives a list with document ID, attachment ID and optionally new attachment ID (GUID from SAP) entries and return a status message and the list of documents, as well as any errors that occurred.

Request example for accept: application/json:

```

{
  "attachment_list": [
    {
      "documentGUID": "62c6d1d1eb9bb326a7f9e6673001a58f",
      "attachmentId": "6E715688804F463BA338CA5E0B2C037B"
    },
    {
      "documentGUID": "62c6d1d1eb9bb326a7f9e6673001a58f",
      "attachmentId": "4AC5186A46F04E2B933305783194F403",
      "newAttachmentId": "D7A3E612D169472DA5C0B234050A4526"
    },
    {
      "documentGUID": "62c6d1d1eb9bb326a7f9e66730019869",
      "attachmentId": "4AC5186A46F04E2B933305783194F403",
      "newAttachmentId": "4AC5186A46F04E2B933305783194F402"
    }
  ]
}

```

Response example for accept: application/json:

If everything is OK:

```

{
  "Status": "true"
}

```

If errors occur:

```

{
  "Status": "false",
  "Messages": [
    {
      "guid": "InvNewBulk21",
      "Status": 703,
      "Message": "Document with GUID InvNewBulk21 already exists in the system."
    }
  ]
}

```

```

    },
    {
      "guid": "InvNewBulk20",
      "Status": 703,
      "Message": "Document with GUID InvNewBulk20 already exists in the system."
    }
  ]
}

```

## exists (bulk)

Bulk operation for the [exists](#) operation.

This operation receives a list of document IDs and returns a list of document IDs that exist in Supplier Portal.

Request example for accept: application/json:

```

{
  "id_list": [
    {
      "@id": "0000005557"
    },
    {
      "@id": "0000005558"
    },
    {
      "@id": "0000005560"
    }
  ]
}

```

Response example for accept: application/json:

```

{
  "Item": [
    "0000005557",
    "0000005560"
  ]
}

```

## retrieve (bulk)

Bulk operation for the [retrieve](#) operation.

This operation receives a list of document IDs and returns a list of document data.

Request example for accept: application/json:

```

{
  "id_list": [
    {
      "@id": "0000005557"
    },
    {
      "@id": "0000005558"
    },
    {
      "@id": "0000005560"
    }
  ]
}

```

}

Response example for accept: application/json:

```
[
  {
    "_id": "66d3cf67160d6246c4c19d7362120b12",
    "_rev": "1-19eeb177a2d3f80f495a114a840f656f",
    "_name": "$MNG_displayname.",
    "header": {...},
    "table": [...],
    "metaData": {
      "entry": [
        {
          "key": "systemID",
          "value": "DEV"
        },
        {
          "key": "tobeCalculated",
          "value": "true"
        }
      ],
      "@state": "0"
    },
    "@id": "005056B517531ED9BEFB005796C54581",
    "@state": "0",
    "@type": "IV",
    "@xmlns": "http://www.tungstenautomation.com/documentData",
    "@rootElementName": "structuredDocument"
  },
  {...}
]
```

## submitbulk

Bulk operation for the [submit](#) operation.

This operation receives a list of document data and returns a status message and the list of documents, as well as any errors that occurred.

Request example for accept: application/json:

```
{
  "doc_list": [
    {
      "@id": "INVGUID01",
      "@type": "IV",
      "@xmlns": "http://www.tungstenautomation.com/documentData",
      "@rootElementName": "structuredDocument",
      "header": {...},
      "table": [...],
      "uploadInfo": [...]
    },
    {
      "@id": "INVGUID02",
      "@type": "IV",
      "@xmlns": "http://www.tungstenautomation.com/documentData",
      "@rootElementName": "structuredDocument",
      "header": {...},
      "table": [...],
      "uploadInfo": [...]
    }
  ]
}
```

```
]
}
```

Response example for accept: application/json:

If everything is OK:

```
{
  "Status": "true"
}
```

If errors occur:

```
{
  "Status": "false",
  "Messages": [
    {
      "guid": "InvNewBulk21",
      "Status": 703,
      "Message": "Document with GUID InvNewBulk21 already exists in the system."
    },
    {
      "guid": "InvNewBulk20",
      "Status": 703,
      "Message": "Document with GUID InvNewBulk20 already exists in the system."
    }
  ]
}
```

## terminateChangeSynchronization (bulk)

Bulk operation for the [terminateChangeSynchronization](#) operation.

This operation receives a list of document IDs and returns a status message and the list of documents, as well as any errors that occurred.

Request example for accept: application/json:

```
{
  "id_list": [
    {
      "@id": "0000005557"
    },
    {
      "@id": "0000005558"
    },
    {
      "@id": "0000005560"
    }
  ]
}
```

Response example for accept: application/json:

If everything is OK:

```
{
  "Status": "true"
}
```

If errors occur:

```
{
```

```

"Status": "false",
"Messages": [
  {
    "guid": "InvNewBulk21",
    "Status": 703,
    "Message": "Document with GUID InvNewBulk21 already exists in the system."
  },
  {
    "guid": "InvNewBulk20",
    "Status": 703,
    "Message": "Document with GUID InvNewBulk20 already exists in the system."
  }
]
}

```

## terminateNewSynchronization (bulk)

Bulk operation for the [terminateNewSynchronization](#) operation.

This operation receives a list of document IDs and returns a status message and the list of documents, as well as any errors that occurred.

Request example for accept: application/json:

```

{
  "doc_list": [
    {
      "@id": "INVGUID01",
      "@type": "IV",
      "@xmlns": "http://www.tungstenautomation.com/documentData",
      "@rootElementName": "structuredDocument",
      "header": {...},
      "table": [...],
      "uploadInfo": [...]
    },
    {
      "@id": "INVGUID02",
      "@type": "IV",
      "@xmlns": "http://www.tungstenautomation.com/documentData",
      "@rootElementName": "structuredDocument",
      "header": {...},
      "table": [...],
      "uploadInfo": [...]
    }
  ]
}

```

Response example for accept: application/json:

If everything is OK:

```

{
  "Status": "true"
}

```

If errors occur:

```

{
  "Status": "false",
  "Messages": [
    {
      "guid": "InvNewBulk21",

```

```

    "Status": 703,
    "Message": "Document with GUID InvNewBulk21 already exists in the system."
  },
  {
    "guid": "InvNewBulk20",
    "Status": 703,
    "Message": "Document with GUID InvNewBulk20 already exists in the system."
  }
]
}

```

## updatebulk

Bulk operation for the [update](#) operation.

This operation receives a list of document data and returns a status message and the list of documents, as well as any errors that occurred.

Request example for accept: application/json:

```

{
  "doc_list": [
    {
      "@id": "INVGUID01",
      "@type": "IV",
      "@xmlns": "http://www.tungstenautomation.com/documentData",
      "@rootElementName": "structuredDocument",
      "header": {...},
      "table": [...],
      "uploadInfo": [...]
    },
    {
      "@id": "INVGUID02",
      "@type": "IV",
      "@xmlns": "http://www.tungstenautomation.com/documentData",
      "@rootElementName": "structuredDocument",
      "header": {...},
      "table": [...],
      "uploadInfo": [...]
    }
  ]
}

```

Response example for accept: application/json:

If everything is OK:

```

{
  "Status": "true"
}

```

If errors occur:

```

{
  "Status": "false",
  "Messages": [
    {
      "guid": "InvNewBulk21",
      "Status": 703,
      "Message": "Document with GUID InvNewBulk21 already exists in the system."
    },
    {

```



```

    "guid": "InvNewBulk20",
    "Status": 703,
    "Message": "Document with GUID InvNewBulk20 already exists in the system."
  }
]
}

```

## Synchronization of documents

The following sequence of operations must be executed to synchronize documents.

### Synchronization of new documents on the Supplier Portal side

Name	Purpose
newDocuments	Get the document IDs of all the documents that have been created since the last Synchronizer run.
retrieve	For each document ID, get the complete document.
attachments	For each document ID, get the list of attachments.
getAttachment	For each attachment in the list, get the attachment itself.
commitAttachmentSynchronization	Finish the attachment synchronization for that attachment (it will not be displayed again in the list of new attachments).
terminateNewSynchronization	For each document, finish the synchronization.

### Synchronization of changed documents on the Supplier Portal side

Name	Purpose
changedDocuments	Get the document IDs of all the documents that have been changed since the last Synchronizer run.
retrieve / getChanges	For each document ID, get the complete document or only the changes.
newAttachments	For each document ID, get the list of attachments that have been added since the last Synchronizer run.
getAttachment	For each attachment in the list, get the attachment itself.
commitAttachmentSynchronization	Finish the attachment synchronization for that attachment (it will not be displayed again in the list of new attachments).
terminateChangeSynchronization	For each document, finish the synchronization.

### Synchronization of new documents on the ERP side

Name	Purpose
submit	Submit all new documents that have been created on the ERP side since the last Synchronizer run.
upload	Upload all the attachments for those new documents.

## Synchronization of changed documents on the ERP side

Name	Purpose
update	Update all the documents that have been changed on the ERP side since the last Synchronizer run. Send either the whole document, or only the changes.
upload	Upload all the attachments that have been added since the last Synchronizer run, for each document.

## Suppliers

This service allows you to perform supplier-related operations in Supplier Portal. It includes dedicated operations for gathering information regarding the suppliers in Supplier Portal.

### Base URL

`https://{tenant-endpoint}/supplier-portal/rest/sync/suppliers`

Example:

`https://sp-dev.kofaxdev.com/supplier-portal/rest/sync/suppliers`

### Operations

Name	URI	Method
registered	/registered	GET
registeredIDs	/registeredIDs	GET
registeredSupplierUsers	/registeredSupplierUsers	GET
registeredSupplierUsersAsSimplejson	/registeredSupplierUsersAsSimplejson	GET
registered	/registered	POST
registeredSupplierUsersByCompCode	/registeredSupplierUsersByCompCode/{compCode}	POST
registeredSupplierUsersByVendor	/registeredSupplierUsersByVendor/{vendor}	POST

### registered

This operation returns the list of registered suppliers, as well as their access and authorization rights.

Example for accept: application/json response:

```
{
  "Supplier": [
    {
      "SupplierID": "100008",
```

```

    "Rights": [
      {
        "DocumentType": "PO",
        "Right": [
          "CHANGE",
          "SUBMIT",
          "CREATE",
          "ATTACH"
        ]
      },
      {
        "DocumentType": "IV",
        "Right": [
          "CHANGE",
          "SUBMIT",
          "ATTACH",
          "CREATE"
        ]
      }
    ]
  }
]
}

```

Example for: accept: application/xml

```

<RegisteredSuppliers>
  <Supplier>
    <SupplierID>100008</SupplierID>
    <Rights>
      <DocumentType>PO</DocumentType>
      <Right>CHANGE</Right>
      <Right>ATTACH</Right>
      <Right>CREATE</Right>
      <Right>SUBMIT</Right>
    </Rights>
    <Rights>
      <DocumentType>IV</DocumentType>
      <Right>CHANGE</Right>
      <Right>CREATE</Right>
      <Right>SUBMIT</Right>
      <Right>ATTACH</Right>
    </Rights>
  </Supplier>
</RegisteredSuppliers>

```

## registeredIDs

This operation returns the list of registered supplier IDs.

Example for accept: application/json response:

```

{
  "SupplierId": [
    "170420",
    "170421",
    "170422",
    "170423",
    "170424",
    "170425"
  ]
}

```

Example for accept: application/xml:

```
<RegisteredSuppliersIds>
  <SupplierId>170420</SupplierId>
  <SupplierId>170421</SupplierId>
  <SupplierId>170422</SupplierId>
  <SupplierId>170423</SupplierId>
  <SupplierId>170424</SupplierId>
  <SupplierId>170425</SupplierId>
</RegisteredSuppliersIds>
```

## registeredSupplierUsers

This operation returns the list of supplier users in the portal who are associated with vendors from the submitted system ID, along with those associated vendors.

### Request

Headers:

erpsystemid: systemID

pdapp-auth-ticket: session\_token

Content-Type: application/json

accept: application/json

Body: Empty

### Response

Array of supplier users in the portal.

Entry: user email, vendors with company codes.

Example for application/json response:

```
{
  "table": [{
    "line": [{
      "field": [{
        "value": "user1@web.de",
        "@id": "email",
        "@state": "0"
      }
    ],
    "table": [{
      "line": [{
        "field": [{
          "value": "1000",
          "@id": "vendorNo",
          "@state": "0"
        }
      ],
      "@id": "0",
      "@state": "0"
    }
  ],
  "@id": "vendors",
  "@state": "0"
}
```

```

    }, {
      "field": [{
        "value": "user2@tungstenautomation.com",
        "@id": "email",
        "@state": "0"
      }
    ],
    "table": [{
      "line": [{
        "field": [{
          "value": "1080",
          "@id": "vendorNo",
          "@state": "0"
        }
      ],
      "@id": "0",
      "@state": "0"
    }
  ],
  "@id": "vendors",
  "@state": "0"
},
"@id": "1",
"@state": "0"
},
"@id": "0",
"@state": "0"
},
"@state": "0",
"@xmlns": "http://www.tungstenautomation.com/documentData",
"@rootElementName": "structuredDocument"
}

```

## registeredSupplierUsersAsSimplejson

This operation returns the list of supplier users in the portal who are associated with vendors from the submitted system ID, along with those associated vendors as a simple JSON file.

### Request

Headers:

erpsystemid: systemID

pdapp-auth-ticket: session\_token

Content-Type: application/json

accept: application/json

Body: Empty

### Response

Array of supplier users in the portal in simple JSON format.

Entry: user email, vendors with company codes.

Example for application/json response:

```
{
  "Users": [
    [
      "",
      "1000",
      "user1@tungstenautomation.com"
    ],
    [
      "",
      "1",
      " user2@tungstenautomation.com "
    ],
    [
      "",
      "1000",
      " user3@tungstenautomation.com "
    ]
  ]
}
```

## registered

This operation is similar to the [registered](#) service, except that it takes the list of supplier IDs as returned by the [registeredIDs](#) service. When there is a huge number of suppliers, getting bulk information about the registered suppliers via the service is time consuming and takes up a lot of memory resources. Thus, this operation returns the registered suppliers and their access and authorization rights only for the supplier IDs returned by the `registeredIDs` service.

Example for application/json content-type:

```
{
  "SupplierId": [
    "170420",
    "170421",
    "170422",
    "170423",
    "170424",
    "170425"
  ]
}
```

Example for application/xml content-type:

```
<RegisteredSuppliersIds>
  <SupplierId>170420</SupplierId>
  <SupplierId>170421</SupplierId>
  <SupplierId>170422</SupplierId>
  <SupplierId>170423</SupplierId>
  <SupplierId>170424</SupplierId>
  <SupplierId>170425</SupplierId>
</RegisteredSuppliersIds>
```

## registeredSupplierUsersByCompCode/{compCode}

This operation returns the list of supplier users in the portal who are associated with vendors from the submitted system ID, and of the submitted company code, along with those associated vendors.

### Request

Headers:

```
erpsystemid: systemID
pdapp-auth-ticket: session_token
Content-Type: application/json
accept: application/json
```

Body: Empty

### Response

Array of supplier users assigned to the company code parameter.

Entry: user email, vendors with company codes.

Example for application/json response (/registeredSupplierUsersByCompCode/1000):

```
{
  "table": [{
    "line": [{
      "field": [{
        "value": "user3@tungstenautomation.com",
        "@id": "email",
        "@state": "0"
      }
    ],
    "table": [{
      "line": [{
        "field": [{
          "value": "1000",
          "@id": "vendorNo",
          "@state": "0"
        }
      ],
      "table": [{
        "line": [{
          "field": [{
            "value": "1000",
            "@id": "compCode",
            "@state": "0"
          }
        ],
        "@id": "0",
        "@state": "0"
      }
    ],
    "@id": "comp_codes",
    "@state": "0"
  }
}
```

```

    ],
    "@id": "0",
    "@state": "0"
  }
],
"@id": "vendors",
"@state": "0"
}
],
"@id": "0",
"@state": "0"
}
],
"@id": "0",
"@state": "0"
}
],
"@state": "0",
"@xmlns": "http://www.tungstenautomation.com/documentData",
"@rootElementName": "structuredDocument"
}

```

## registeredSupplierUsersByVendor/{vendor}

This operation returns the list of supplier users associated with the submitted vendor.

### Request

Headers:

erpsystemid: systemID

pdapp-auth-ticket: session\_token

Content-Type: application/json

accept: application/json

Body: Empty

### Response

Array of supplier users assigned to the vendor parameter.

Entry: user email, vendors with company codes.

Example for application/json response (/registeredSupplierUsersByVendor/1000):

```

{
  "table": [{
    "line": [{
      "field": [{
        "value": "user4@web.de",
        "@id": "email",
        "@state": "0"
      }
    ],
    "table": [{
      "line": [{
        "field": [{
          "value": "1000",

```



```

        "@id": "vendorNo",
        "@state": "0"
      }
    ],
    "@id": "0",
    "@state": "0"
  }
],
"@id": "vendors",
"@state": "0"
}, {
"@id": "0",
"@state": "0"
}, {
  "field": [{
    "value": "user5@tungstenautomation.com",
    "@id": "email",
    "@state": "0"
  }
],
"table": [{
  "line": [{
    "field": [{
      "value": "66",
      "@id": "vendorNo",
      "@state": "0"
    }
  ],
  "@id": "0",
  "@state": "0"
}, {
  "field": [{
    "value": "1000",
    "@id": "vendorNo",
    "@state": "0"
  }
],
  "@id": "1",
  "@state": "0"
}
],
"@id": "vendors",
"@state": "0"
}, {
"@id": "1",
"@state": "0"
}, {
  "field": [{
    "value": "user6@tungstenautomation.com",
    "@id": "email",
    "@state": "0"
  }
],
"table": [{
  "line": [{
    "field": [{
      "value": "1075",
      "@id": "vendorNo",
      "@state": "0"
    }
  ],
  "@id": "0",

```

```

        }, {
          "field": [{
            "value": "1000",
            "@id": "vendorNo",
            "@state": "0"
          }
        ],
        "@id": "1",
        "@state": "0"
      }
    ],
    "@id": "vendors",
    "@state": "0"
  }
],
"@id": "2",
"@state": "0"
}
],
"@state": "0",
"@xmlns": "http://www.tungstenautomation.com/documentData",
"@rootElementName": "structuredDocument"
}

```

## SearchHelp

This service is used to handle search help submission, update and deletion. In Supplier Portal, depending on the document type, you can configure a search help for the header fields.

### Base URL

`https://{tenant-endpoint}/supplier-portal/rest/sync/searchhelp`

Example:

`https://sp-dev.kofaxdev.com/supplier-portal/rest/sync/searchhelp`

## Operations

Name	URI	Method
delete	<code>/docType/{field}/delete</code>	POST
submit	<code>/docType/{field}/submit</code>	POST
update	<code>/docType/{field}/update</code>	POST

### delete

This operation deletes the search help data for the given document type and field. The request body does not expect any data, despite being a POST method.

Upon successful operation, the web service returns the following response:

Example for accept: application/json response:

```
{
  "Status": true,
  "Message": ""
}
```

Example for accept: application/xml response:

```
<DocumentProcess>
  <Status>true</Status>
  <Message></Message>
</DocumentProcess>
```

## submit

This operation allows you to submit search help data for the given document type and field. The `documentType` and `field` should be sent as a part of the URL, and the search help data as the request body.

Example for application/json content-type:

```
{
  "name" : "VENDOR_NO",
  "line" : [
    {
      "@id" : "1",
      "field" : [
        {
          "@id" : "VENDOR_NO",
          "value" : "1000"
        },
        {
          "@id" : "VENDOR_NAME",
          "value" : "Tungsten Automation Germany"
        }
      ]
    }
  ]
}
```

Example for application/xml content-type:

```
<searchhelpdata>
  <name>VENDOR_NO</name>
  <line id="1">
    <field id="VENDOR_NO">
      <value>1000</value>
    </field>
    <field id="VENDOR_NAME">
      <value>Tungsten Automation Germany</value>
    </field>
  </line>
</searchhelpdata>
```

Upon successful operation, the web service returns the following response:

Example for accept: application/json response:

```
{
  "Status": true,
  "Message": "{search-help-id}"
}
```

Example for accept: application/xml response:

```
<DocumentProcess>
  <Status>true</Status>
  <Message>{search-help-id}</Message>
</DocumentProcess>
```

## update

Similar to the [submit](#) operation, this operation updates the search help data for the given document type and field. The `documentType` and `field` should be sent as a part of the URL, and the search help data as the request body. However, the search help data contains only the changes for the given document type and field.

Upon successful operation, the web service returns the following response:

Example for accept: application/json response:

```
{
  "Status": true,
  "Message": ""
}
```

Example for accept: application/xml response:

```
<DocumentProcess>
  <Status>true</Status>
  <Message></Message>
</DocumentProcess>
```

# Errors

This section briefly explains how the web service behaves in case there is an error. Errors in the web service happen for many known and unknown reasons. In both cases, the web service responds by sending a proper HTTP status code and a response entity that contains the code identifying the nature of the error, as well as a message providing a short description of the error.

In general, the web service returns errors with two different HTTP status codes.

HTTP status code	Returned
500	When the web service encounters unexpected system errors.
403	When the web service knows the cause of the error.

## 500 status code

When the 500 status code is returned, the web service sends the following generic response entity.

Example for accept: application/json response:

```
{
  "Code" : 500,
  "Message" : "The Application has encountered a problem. It cannot fulfill the operation
you have requested"
}
```

Example for accept: application/xml response:

```
<Exception>
  <Code>500</Code>
<Message>The Application has encountered a problem. It cannot fulfill the operation you
have requested
</Message>
</Exception>
```

## 403 status code

When the 403 status code is returned, the web service returns a response entity with a proper error code identifying the nature of the error, and a message that provides a short description of the error.

The following table lists the error codes that are sent.

Error code	Purpose
101	Expected headers are missing from the request.

Error code	Purpose
102	Request headers have empty values.
105	The session does not exist for the session ID token value that was sent for the <code>pdapp-auth-ticket</code> request header.
106	The session ID token is invalid due to a session timeout.
107	The session cannot be determined for some other reason; unlikely to occur.
185	The web service has problems connecting with the backend, rendering the service as unavailable. The included message provides a hint regarding the connectivity problem.
700	The document on which an operation is performed cannot be found.
701	The operation performed for a document is not authorized.
702	The document that is currently being worked on fails validation.
703	The document that is currently being worked on is a duplicate document.
704	Wrong parameters are sent in the request body entity.
705	The document that is currently being worked on fails a check.
706	A request body entity exceeds the allowed size. Currently, the web service denies request entities that exceed (approximately) 20 MB.
720	An operation is interrupted due to access problems with the database, such as a timeout. This is not a permanent error, though. Successively retrying the operation might be successful.
725	An update conflict is encountered; that is, a newer copy of the document that is in operation exists in Supplier Portal. In this case, the client should attempt to retrieve the latest document from the web service before sending an operation that will result in an update in Supplier Portal.
730	An internal error occurred while fulfilling the requested operation. The included message provides a hint regarding the problem that occurred.

# Data model

The fields listed in this section are transferred as part of the standard.

## Process Director Accounts Payable

The data for Process Director Accounts Payable documents is retrieved by the `/COCKPIT/API_DATA_GET` function module.

<b>E_STR_HEADER &gt; HEADER</b>	
INVOICE_GUID	Unique document identification
ORIGIN	Origin of Process Director document
DOCNO	Process Director document number
FI_MM_FLG	FI or MM document
STATUS	Document status
INVOICE_IND	Flag to identify document as invoice
CP_DOC_TYPE	Process Director document type
DOC_TYPE	Document type
DOC_DATE	Document date in document
VENDOR_NO	Account number of vendor or creditor
DIFF_INV	Different invoicing party
PO_NUMBER	Purchasing document number
PSTNG_DATE	Posting date in document
FIS_PERIOD	Fiscal period
REF_DOC_NO	Reference document number
COMP_CODE	Company code
GL_ACCOUNT	G/L account number
CURRENCY	Currency key
GROSS_AMOUNT	Process Director gross amount
NET_AMOUNT	Process Director net amount
CALC_TAX_IND	Automatic tax calculation
BLINE_DATE	Baseline date for due date calculation
PMNTRMS	Terms of payment key
DSCT_DAYS1	Cash discount days 1

<b>E_STR_HEADER &gt; HEADER</b>	
DSCT_DAYS2	Cash discount days 2
NETTERMS	Net payment terms period
DSCT_PCT1	Cash discount percentage 1
DSCT_PCT2	Cash discount percentage 2
PYMT_METH	Payment method
HEADER_TXT	Document header text
DEL_COSTS	Unplanned delivery costs
DEL_COSTS_TAXC	Tax code for delivery costs
EXT_DOC_NO	External document number
SAP_DOC_NO	Document number of invoice document
FISCAL_YEAR	Fiscal year
BVORG	Number of cross-company code posting transactions
REVERSAL_DOC	Reverse document number
REV_DOC_YEAR	Reverse document fiscal year
VENDOR_NAME_EXT	External vendor name
POSTAGE_AMOUNT	Postal charge
INSURANCE_AMOUNT	Insurance amount
FREIGHT_AMOUNT	Freight costs
PACKAGE_AMOUNT	Packing costs
SGTXT	Item text
BVTYP	Partner bank type
HBKID	Short key for house bank
SCBANK_IND	State central bank indicator
SUPCOUNTRY	Supplying country
SUPCOUNTRY_ISO	Supplier country ISO code
BLLSRV_IND	Service indicator (foreign payment)
KURSF	Exchange rate
KURSR	Hedged exchange rate
GBETR	Hedged amount in foreign currency
TBTKZ	Indicator: Subsequent debit/credit
SHEET_NO	Entry sheet number
ALLOC_NMBR	Assignment number
LFSNR	Number of external delivery notes
DSCT_AMOUNT	Cash discount amount in document currency



<b>E_STR_HEADER &gt; HEADER</b>	
BUS_AREA	Business area
WWERT	Translation date
PMNT_BLOCK	Payment block key
FOLLOW_UP	Follow-up flag ID (not available since Process Director Accounts Payable 7.2)
NETDT	Due date for net payment
NETDT1	Due date days 1
NETDT2	Due date days 2
AUGDT	Clearing date
AUGCP	Clearing entry date
AUGBL	Document number of clearing document
AMOUNT_SPLIT	Document posted using invoice split
SKFBT1	Discount amount for discount days/rate 1
SKFBT2	Discount amount for discount days/rate 2
SKGAINED	Gained discount amount
SKLOST	Lost discount amount
REP_EXTRACT	Reporter data export carried out
VATDATE	Tax reporting date
PAYMT_REF	Payment reference
PMTMTHSUPL	Payment method supplement
INV_REF_NO	Number of the invoice the transaction belongs to
INV_YEAR	Fiscal year of relevant invoice (for credit memo)
EC_ARC_ID	Content repository identification
EC_DBC_ID	SAP ArchiveLink: Document ID
EC_ARC_OBJ	Document type
TTIME	Processing time by Process Director Accounts Payable
TEWCTIME	Processing time in workflow
CHECK_TYPE	Validation type for additional checks in Process Director
PAYR_HBKID	Short key for house bank
PAYR_HKTID	ID for account details
PAYR_CHECT	Check number
BUPLA	Business place
J_1BNFTYPE	Nota fiscal type
J_1BDOCNUM	Document number

<b>E_STR_HEADER &gt; HEADER</b>	
J_1BREGIO	Region of NF-e issuer
J_1BYEAR	Year of NF-e posting date
J_1BMONTH	Month of NF-e posting date
J_1BSTCD1	CNPJ number of NF-e issuer
J_1BMODEL	Model of nota fiscal
J_1BDOCNUM9	NF-e: Random number
J_1BCHECKDIGIT	Check digit for NF-e access key
J_1BNFEAUTHCODE	NF-e: Protocol number
J_1BPREFNO	Service NF-e number provided by prefecture
J_1BCHECOD	Checking code of Service NF-e
J_1BNFEXMLVER	NF-e: XML version
ESRNR	POR subscriber number
ESRPZ	POR check digit
ESRRE	POR reference number
VENDOR_NAME1	Business partner name
VENDOR_NAME2	Business partner name
VENDOR_CITY	City
VENDOR_STREET	House number and street
CR_DATE	Date on which record was created
CR_TIME	Time at which record was created
HD_OFFICE	Head office account number
HD_OFFICE_DESCR	Head office description
COMP_CODE_DESCR	Company code description
BUS_AREA_DESCR	Business area description
DIFF_INV_DESCR	Description of alternative invoicing party
GROSS_AMT_DISP	Gross amount in Process Director Accounts Payable (negative values)
NET_AMT_DISP	Net amount in Process Director Accounts Payable (negative values)
CR_USER	Processor of Process Director document
CR_TIMESTAMP	UTC time stamp in long form (YYYYMMDDhhmmss,mmmuuun)
CH_USER	Last person to change a record in Process Director
CH_TIMESTAMP	UTC time stamp in long form (YYYYMMDDhhmmss,mmmuuun)
CI_COCKPIT_HDR	All fields in include CI_COCKPIT_HDR

<b>E_TAB_ITEM &gt; ITEM</b>	
INVOICE_GUID	Unique document identification
INVOICE_ITEM	Document item in invoice document
PO_NUMBER	Purchasing document number
PO_ITEM	Item number of purchasing document
REF_DOC	Document number of reference document
REF_DOC_YEAR	Fiscal year of current period
REF_DOC_IT	Item of reference document
ITEM_AMOUNT	Process Director net amount
GROSS_AMOUNT	Process Director gross amount
QUANTITY	Quantity
PO_UNIT	Order unit
MATERIAL	Material number
ITEM_TEXT	Item text
TBTKZ	Indicator: Subsequent debit/credit
SHEET_NO	Entry sheet number
LFSNR	Number of external delivery note
DISTRIB	Distribution code for multiple account assignment
COND_TYPE	Condition type
COND_ST_NO	Step number
COND_COUNT	Condition counter
FINAL_INV	Final invoice indicator
RETPO	Item return
PART_INV	Partial invoice indicator
PACKNO	Package number of service
INTROW	Line number of service
DA_QUANTITY	Quantity
DA_AMOUNT	Process Director net amount
TAX_CODE	Tax code
TAX_RATE	Tax rate
TAX_AMOUNT	Tax amount in document currency
TAXJURCODE	Jurisdiction for tax calculation - tax jurisdiction code
APPROVED	Line item/account assignment approved
APPROVER	Approver ID
APPROVER_TYPE	Approver type

<b>E_TAB_ITEM &gt; ITEM</b>	
APPROVAL_STATUS	Line item/account assignment approved - display only
CURRENT_APPROVER	Current item processor
CURRENT_APPROVER_TYPE	Current item processor type
NEXT_PROC	Next item processor
NEXT_PROC_TYPE	Next item processor type
Table name	DDIC table name of data table
MATERIAL_TEXT	Material description
FOLLOW_UP_ICON	Follow-up flag
CI_COCKPIT_ITEM	All fields in include CI_COCKPIT_ITEM

<b>E_TAB_TAX &gt; TAX</b>	
INVOICE_GUID	Unique document identification
TAX_CODE	Tax code
TAX_RATE	Tax rate
TAX_AMOUNT	Tax amount in document currency
TAXJURCODE	Jurisdiction for tax calculation - tax jurisdiction code
SHKZG	Debit/credit indicator
XEXTN_TXP	Indicator: External tax calculation
CI_COCKPIT_TAX	All fields in include CI_COCKPIT_TAX

<b>E_TAB_ACCOUNT &gt; ACCOUNT</b>	
INVOICE_GUID	Unique document identification
GL_ACCOUNT	G/L account number
NET_AMOUNT	Process Director net amount
GROSS_AMOUNT	Process Director gross amount
TEXT	Item text
ALLOC_NMBR	Assignment number
SHKZG	Debit/credit indicator
BSCHL	Posting key
COSTCENTER	Cost center
ORDERID	Order number
WBS_ELEMENT	Work Breakdown Structure Element (WBS Element)
SALES_ORD	Sales order number
S_ORD_ITEM	Item number of SD document
COST_OBJ	Cost object

<b>E_TAB_ACCOUNT &gt; ACCOUNT</b>	
CO_BUSPROC	Receiver business process
NETWORK	Network number for account assignment
NETWORK_OP	Network activity
CO_MATERIAL	Receiving material
PLANT	Plant for receiving material
PROFIT_CTR	Profit center
BUS_AREA	Business area
COMP_CODE	Company code
TAX_CODE	Tax code
TAX_RATE	Tax rate
TAX_AMOUNT	Tax amount in document currency
TAXJURCODE	Jurisdiction for tax calculation - tax jurisdiction code
REC_STATUS	Check box
IDOC_NUMBER	IDoc number
IDOC_GUID	Unique document identification
ORIGIN	Origin of Process Director document
DOCNO	Process Director document number
FI_MM_FLG	FI or MM document
INVOICE_IND	Flag to identify document as invoice
CP_DOC_TYPE	Process Director document type
DOC_DATE	Document date in document
VENDOR_NO	Account number of vendor or creditor
PO_NUMBER	Purchasing document number
REF_DOC_NO	Reference document number
CURRENCY	Currency key
HEADER_TXT	Document header text
PAOBJNR	Profitability segment number (CO-PA)
PASUBNR	Profitability segment changes (CO-PA)
COPA_ICON	Process Director document profile
TR_ACTIVITY	Trip activities
TR_NETAMOUNT	Process Director net amount
TR_CURRENCY	Currency key
TR_AMOUNT_TR	Travel expenses
TR_DATE	Document date in document

<b>E_TAB_ACCOUNT &gt; ACCOUNT</b>	
TR_TEXT_OLD	Item text
APPROVED	Line item/account assignment approved
APPROVER	Approver ID
APPROVER_TYPE	Approver type
APPROVAL_STATUS	Line item/account assignment approved - display only
CURRENT_APPROVER	Current item processor
CURRENT_APPROVER_TYPE	Current item processor type
NEXT_PROC	Next item processor
NEXT_PROC_TYPE	Next item processor type
GL_ACCOUNT_TEXT	G/L account short text
CI_COCKPIT_ACCT	All fields in include CI_COCKPIT_ACCT

<b>E_TAB_ACCT_ASSIGN &gt; not implemented</b>	
CLIENT	Client
INVOICE_GUID	Unique document identification
VERSION	Version of Process Director Accounts Payable data record
POS_NO	Item number
PO_NUMBER	Purchasing document number
PO_ITEM	Item number of purchasing document
SUBPOS	Subitem
SERIAL_NO	Sequential number of account assignment
QUANTITY	Quantity
ITEM_AMOUNT	Process Director net amount
GL_ACCOUNT	G/L account number
COSTCENTER	Cost center
ORDERID	Order number
WBS_ELEMENT	Work Breakdown Structure Element (WBS Element)
SALES_ORD	Sales order number
S_ORD_ITEM	Item number of SD document
COST_OBJ	Cost object
NETWORK	Network number for account assignment
NETWORK_OP	Network activity
PROFIT_CTR	Profit center
ASSET_NO	Main asset number

<b>E_TAB_ACCT_ASSIGN &gt; not implemented</b>	
SUB_NUMBER	Asset sub-number
BUS_AREA	Business area
CO_AREA	Controlling area
CI_COCKPIT_ACCAS	All fields in include CI_COCKPIT_ACCAS

<b>E_TAB_FLAG &gt; Follow-up</b>	
CLIENT	Client
INVOICE_GUID	Unique document identification
FLAG	Follow-up flag ID
CR_TIMESTAMP	UTC time stamp in long form (YYYYMMDDhhmmss,mmmuuun)

The data for Process Director Accounts Payable documents is retrieved by the `/COCKPIT/WC_DB_WEBCYCLE_GET` function module.

<b>ES_TWC &gt; HEADER</b>	
CLIENT	Client
INVOICE_GUID	Unique document identification
WC_NO	Workflow number
WC_VERSION	Version of workflow data
VERSION	Version of Process Director Accounts Payable data record
DOC_STATUS	Document status
WC_ID	Workflow ID
WC_DUEDATE	Workflow due date
WC_STATUS	Workflow status
WC_POS	Workflow step position
WC_CH_TIMESTAMP	UTC time stamp in long form (YYYYMMDDhhmmss,mmmuuun)
ARCHIVED	Check box
WC_INITIATOR	Last person to change a record in Process Director

## Incoming invoice

The data for incoming invoices is transferred to Process Director Accounts Payable by the `/COCKPIT/RFC_DATA_SUBMIT` function module.

<b>HEADER &gt; I_STR_HEADER</b>	
INVOICE_IND	Flag to identify document as invoice
CP_DOC_TYPE	Process Director document type

<b>HEADER &gt; I_STR_HEADER</b>	
ORIGIN	Origin of Process Director document
DOC_TYPE	Document type
DOC_DATE	Document date in document
VENDOR_NO	Account number of vendor or creditor
DIFF_INV	Different invoicing party
PO_NUMBER	Purchasing document number
PSTNG_DATE	Posting date in document
REF_DOC_NO	Reference document number
COMP_CODE	Company code
GL_ACCOUNT	General ledger account
CURRENCY_ISO	ISO currency code
CURRENCY_SAP	Currency key
GROSS_AMOUNT	Amount in document currency
NET_AMOUNT	Amount in document currency
PMNTTRMS	Terms of payment key
BLINE_DATE	Baseline date for due date calculation
DSCT_DAYS1	Cash discount days 1
DSCT_DAYS2	Cash discount days 2
NETTERMS	Net payment terms period
DSCT_PCT1	Cash discount percentage 1
DSCT_PCT2	Cash discount percentage 2
PYMT_METH	Payment method
HEADER_TXT	Document header text
DEL_COSTS	Unplanned delivery costs
DEL_COSTS_TAXC	Tax code for delivery costs
EXT_DOC_NO	External document number
SAP_DOC_NO	Document number of invoice document
USERNAME	User name
VENDOR_NAME_EXT	External vendor name
POSTAGE_AMOUNT	Unplanned delivery costs
INSURANCE_AMOUNT	Unplanned delivery costs
FREIGHT_AMOUNT	Unplanned delivery costs
PACKAGE_AMOUNT	Unplanned delivery costs
TOLL_AMOUNT	Unplanned delivery costs



<b>HEADER &gt; I_STR_HEADER</b>	
SGTXT	Item text
BVTYP	Partner bank type
HBKID	Short key for house bank
SCBANK_IND	State central bank indicator
SUPCOUNTRY	Supplier country
SUPCOUNTRY_ISO	Supplier country ISO code
BLLSRV_IND	Service indicator (foreign payment)
KURSF	Exchange rate
KURSR	Hedged exchange rate
GBETR	Amount in document currency
TBTkZ	Indicator: Subsequent debit/credit
SHEET_NO	Entry sheet number
ALLOC_NMBR	Assignment number
LFSNR	Number of external delivery note
DSCT_AMOUNT	Cash discount amount in local currency
BUS_AREA	Business area
PMNT_BLOCK	Payment block key
VATDATE	Tax reporting date
PAYMT_REF	Payment reference
PMTMTHSUPL	Payment method supplement
FISCAL_YEAR	Fiscal year
WWERT	Translation date
EC_ARC_ID	Content repository identification
EC_DBC_ID	SAP ArchiveLink: Document ID
EC_ARC_OBJ	Document type
BUPLA	Business place
J_1BNFTYPE	Nota fiscal type
J_1BDOCNUM	Document number
J_1BREGIO	Region of NF-e issuer
J_1BYEAR	Year of NF-e posting date
J_1BMONTH	Month of NF-e posting date
J_1BSTCD1	CNPJ number of NF-e issuer
J_1BMODEL	Model of nota fiscal
J_1BDOCNUM9	NF-e: Random number

<b>HEADER &gt; I_STR_HEADER</b>	
J_1BCHECKDIGIT	Check digit for NF-e access key
J_1BNFEAUTHCODE	NF-e: Protocol number
J_1BPREFNO	Service NF-e number provided by prefecture
J_1BCHECOD	Checking code of Service NF-e
J_1BNFEXMLVER	NF-e: XML version
CI_COCKPIT_HDR	All fields in include CI_COCKPIT_HDR

<b>ITEM &gt; I_TAB_ITEM</b>	
INVOICE_ITEM	Document item in invoice document
PO_NUMBER	Purchase order number
PO_ITEM	Item number of purchasing document
REF_DOC	Document number of reference document
REF_DOC_YEAR	Fiscal year of current period
REF_DOC_IT	Item of reference document
TBTkZ	Indicator: Subsequent debit/credit
TAX_CODE	Tax code
TAX_RATE	Tax rate
TAX_AMOUNT	Tax amount in document currency
TAXJURCODE	Jurisdiction for tax calculation - tax jurisdiction code
ITEM_AMOUNT	Amount in document currency
GROSS_AMOUNT	Amount in document currency
QUANTITY	Quantity
PO_UNIT_ISO	ISO code for PO unit of measure
PO_UNIT_SAP	Order unit
MATERIAL	Material number
ITEM_TEXT	Item text
SHEET_NO	Entry sheet number
LFSNR	Number of external delivery note
COND_TYPE	Condition type
CI_COCKPIT_ITEM	All fields in include CI_COCKPIT_ITEM

<b>TAX &gt; I_TAB_TAX</b>	
TAX_CODE	Tax on sales/purchases code
TAX_RATE	Tax rate
TAX_AMOUNT	Tax amount in document currency

<b>TAX &gt; I_TAB_TAX</b>	
TAXJURCODE	Jurisdiction for tax calculation - tax jurisdiction code
CI_COCKPIT_TAX	All fields in include CI_COCKPIT_TAX

<b>ACCOUNT &gt; I_TAB_ACCOUNT</b>	
GL_ACCOUNT	G/L account number
NET_AMOUNT	Amount in document currency
GROSS_AMOUNT	Amount in document currency
TEXT	Item text
HEADER_DATA	Check box
ALLOC_NMBR	Assignment number
SHKZG	Debit/credit indicator
BSCHL	Posting key
COSTCENTER	Cost center
ORDERID	Order number
WBS_ELEMENT	Work Breakdown Structure Element (WBS Element)
SALES_ORD	Sales order number
S_ORD_ITEM	Item number of SD document
COST_OBJ	Cost object
CO_BUSPROC	Receiver business process
NETWORK	Network number for account assignment
NETWORK_OP	Network activity
CO_MATERIAL	Receiving material
PLANT	Plant for receiving material
PROFIT_CTR	Profit center
BUS_AREA	Business area
COMP_CODE	Company code
TAX_CODE	Tax on sales/purchases code
TAX_RATE	Tax rate
TAX_AMOUNT	Tax amount in document currency
TAXJURCODE	Jurisdiction for tax calculation - tax jurisdiction code
REC_STATUS	Check box
IDOC_NUMBER	IDoc number
IDOC_GUID	Unique document identification
ORIGIN	Origin of Process Director document

<b>ACCOUNT &gt; I_TAB_ACCOUNT</b>	
DOCNO	Process Director document number
FI_MM_FLG	FI or MM document
INVOICE_IND	Flag to identify document as invoice
CP_DOC_TYPE	Process Director document type
DOC_DATE	Document date in document
VENDOR_NO	Account number of vendor or creditor
PO_NUMBER	Purchasing document number
REF_DOC_NO	Reference document number
CURRENCY	Currency key
HEADER_TXT	Document header text
CI_COCKPIT_ACCT	All fields in include CI_COCKPIT_ACCT

## Purchase order

The data for Process Director Accounts Payable documents is retrieved by the `BAPI_PO_GETDETAIL` function module.

<b>PO_HEADER &gt; HEADER</b>	
PO_NUMBER	Purchasing document number
CO_CODE	Company code
DOC_CAT	Purchasing document category
DOC_TYPE	Purchasing document type
CNTRL_IND	Control indicator for purchasing document type
DELETE_IND	Deletion indicator in purchasing document
STATUS	Status of purchasing document
CREATED_ON	Date on which record was created
CREATED_BY	Person who created the object
ITEM_INTVL	Item number interval
LAST_ITEM	Last item number
VENDOR	Vendor's account number
LANGUAGE	Language key
PMNTRMS	Terms of payment key
DSCNT1_TO	Cash discount days 1
DSCNT2_TO	Cash discount days 2
DSCNT3_TO	Cash discount days 3

<b>PO_HEADER &gt; HEADER</b>	
CASH_DISC1	Cash discount percentage 1
CASH_DISC2	Cash discount percentage 2
PURCH_ORG	Purchasing organization
PUR_GROUP	Purchasing group
CURRENCY	Currency key
EXCH_RATE	Direct quoted exchange rate
EX_RATE_FX	Indicator: Fixing of exchange rate
DOC_DATE	Purchasing document date
VPER_START	Start of validity period
VPER_END	End of validity period
APPLIC_BY	Closing date for applications
QUOT_DEAD	Deadline for submission of bid/quotation
BINDG_PER	Binding period for quotation
WARRANTY	Warranty date
BIDINV_NO	Bid invitation number
QUOTATION	Quotation number
QUOT_DATE	Quotation submission date
REF_1	Customer's or vendor's internal reference
SALES_PERS	Responsible salesperson in vendor's office
TELEPHONE	Vendor's telephone number
SUPPL_VEND	Supplying vendor
CUSTOMER	Customer number
AGREEMENT	Number of principal purchasing agreement
REJ_REASON	Reason for rejection of quotations and sales orders
COMPL_DLV	Complete delivery stipulated for each purchase order
GR_MESSAGE	Indicator: Goods receipt message
SUPPL_PLNT	Supplying (issuing) plant in case of stock transport order
RCVG_VEND	Field not used
INCOTERMS1	Incoterms (part 1)
INCOTERMS2	Incoterms (part 2)
TARGET_VAL	Cumulative planned value
COLL_NO	Collective number
DOC_COND	Number of the document condition
PROCEDURE	Procedure (pricing, output control, acct. det., costing, and so on)

<b>PO_HEADER &gt; HEADER</b>	
UPDATE_GRP	Update group for statistics update
DIFF_INV	Different invoicing party
EXPORT_NO	Number of foreign trade data in MM and SD documents
OUR_REF	Our reference
LOGSYSTEM	Logical system
SUBITEMINT	Item interval for sub-items
MAST_COND	Document with time-dependent conditions
REL_GROUP	Release group
REL_STRAT	Release strategy
REL_IND	Release indicator: Purchasing document
REL_STATUS	Release status
SUBJ_TO_R	Release not yet completely in effect
TAXR_CNTRY	Country for tax return
SCHED_IND	Indicator for scheduling agreement release documentation
VEND_NAME	Name 1
CURRENCY_ISO	ISO currency code
EXCH_RATE_CM	Indirect quoted exchange rate
HOLD	Purchase order not yet complete

<b>PO_HEADER_TEXTS &gt; HEADER_TEXTS</b>	
TEXT_LINE	Text line

<b>PO_ITEMS &gt; ITEM</b>	
PO_NUMBER	Purchasing document number
PO_ITEM	Item number of purchasing document
DELETE_IND	Deletion indicator in purchasing document
STATUS	RFQ status
CHANGED_ON	Purchasing document item change date
SHORT_TEXT	Short text
MATERIAL	Material number
PUR_MAT	Material number
CO_CODE	Company code
PLANT	Plant
STORE_LOC	Storage location
TRACKINGNO	Requirement tracking number

<b>PO_ITEMS &gt; ITEM</b>	
MAT_GRP	Material group
INFO_REC	Number of purchasing information record
VEND_MAT	Material number used by vendor
TARGET_QTY	Target quantity
QUANTITY	Purchase order quantity
UNIT	Order unit
ORDERPR_UN	Order price unit (purchasing)
CONV_NUM1	Numerator for conversion of order price unit to order unit
CONV_DEN1	Denominator for conversion of order price unit to order unit
CONV_NUM2	Numerator for conversion of order unit to base unit
CONV_DEN2	Denominator for conversion of order unit to base unit
NET_PRICE	Net price in purchasing document (in document currency)
PRICE_UNIT	Price unit
NET_VALUE	Net order value in PO currency
GROS_VALUE	Gross order value in PO currency
QUOT_DEAD	Deadline for submission of bid/quotation
GR_PR_TIME	Goods receipt processing time in days
TAX_CODE	Tax on sales/purchases code
SETT_GRP1	Settlement group 1 (Purchasing)
QUAL_INSP	Stock type
INFO_UPD	Indicator: Update information record
PRNT_PRICE	Print price
EST_PRICE	Estimated price indicator
NUM_REMIND	Number of reminders or urging messages (expeditors)
REMINDER1	Number of days until first reminder/urging letter (expeditor)
REMINDER2	Number of days until second reminder/urging letter (expeditor)
REMINDER3	Number of days until third reminder/urging letter (expeditor)
OVERDELTO	Overdelivery tolerance limit
UNLIMITED	Indicator: Unlimited overdelivery allowed
UNDER_TOL	Underdelivery tolerance limit
VAL_TYPE	Valuation type
VAL_CAT	Valuation category
REJ_IND	Rejection indicator
COMMENT	Internal comment on quotation

<b>PO_ITEMS &gt; ITEM</b>	
DEL_COMPL	"Delivery completed" indicator
FINAL_INV	Final invoice indicator
ITEM_CAT	Item category in purchasing document
ACCTASSCAT	Account assignment category
CONSUMPT	Consumption posting
DISTRIB	Distribution indicator for multiple account assignment
PART_INV	Partial invoice indicator
GR_IND	Goods receipt indicator
GR_NON_VAL	Goods receipt, non-valuated
IR_IND	Invoice receipt indicator
GR_BASEDIV	Indicator: GR-based invoice verification
ACKN_REQD	Order acknowledgment requirement
ACKNOWL_NO	Order acknowledgment number
AGREEMENT	Number of principal purchasing agreement
AGMT_ITEM	Item number of principal purchasing agreement
RECON_DATE	Reconciliation date for agreed cumulative quantity
AGRCUMQTY	Agreed cumulative quantity
FIRM_ZONE	Firm zone (go-ahead for production)
TRADE_OFF	Trade-off zone (go-ahead for material procurement)
BOM_EXPL	Indicator: BOM explosion
EXCLUSION	Exclusion in outline agreement item with material class
BASE_UNIT	Base unit of measure
SHIPPING	Shipping instructions
OUTL_TARGV	Target value for outline agreement in document currency
NOND_ITAX	Non-deductible input tax
RELORD_QTY	Standard release order quantity
PRICE_DATE	Date of price determination
DOC_CAT	Purchasing document category
EFF_VALUE	Effective value of item
COMMITMENT	Item affects commitments
CUSTOMER	Customer
ADDRESS	Address
COND_GROUP	Condition group for vendor
NO_C_DISC	No cash discount granted on this item



<b>PO_ITEMS &gt; ITEM</b>	
UPDATE_GRP	Update group for statistics update
PLAN_DEL	Planned delivery time in days
NET_WEIGHT	Net weight
WEIGHTUNIT	Unit of weight
TAX_JUR_CD	Jurisdiction for tax calculation - tax jurisdiction code
PRINT_REL	Indicator: Print-relevant schedule lines exist
SPEC_STOCK	Special stock indicator
SETRESERNO	Settlement reservation number
SETTLITMNO	Item number of settlement reservation
NOT_CHGBL	Quality inspection indicator cannot be changed
CTR_KEY_QM	Control key for quality management in procurement
CERT_TYPE	Certificate type
EAN_UPC	International article number (EAN/UPC)
CONF_CTRL	Confirmation control key
REV_LEV	Revision level
FUND	Fund
FUNDS_CTR	Funds center
CMMT_ITEM	Commitment item
BA_PARTNER	Business area reported to partner
PTR_ASS_BA	Assumed business area of business partner
PROFIT_CTR	Profit center
PARTNER_PC	Partner profit center
PRICE_CTR	Pricing date category (controls date of price determination)
GROSS_WGHT	Gross weight
VOLUME	Volume
VOLUMEUNIT	Volume unit
INCOTERMS1	Incoterms (part 1)
INCOTERMS2	Incoterms (part 2)
ADVANCE	Advance procurement: Project stock
PRIOR_VEND	Prior vendor
SUB_RANGE	Vendor sub-range
PCKG_NO	Package number
STATISTIC	Item is statistical
HL_ITEM	Higher-level item in purchasing documents

<b>PO_ITEMS &gt; ITEM</b>	
GR_TO_DATE	Latest possible goods receipt
SUPPL_VEND	Vendor to be supplied/who is to receive delivery
SC_VENDOR	Subcontracting vendor
CONF_MATL	Cross-plant configurable material
MAT_CAT	Material category
KANBAN_IND	Kanban indicator
ADDRESS2	Number of delivery address
INT_OBJ_NO	Configuration (internal object number)
ERS	Evaluated Receipt Settlement (ERS)
GRSETTFROM	Start date for GR-based settlement
LAST_TRANS	Last transmission
TRANS_TIME	Time
SER_NO	Sequential number
PROMOTION	Promotion
ALLOC_TBL	Allocation table number
AT_ITEM	Item number of allocation table
POINTS	Number of points
POINTS_UN	Points unit
SEASON_TY	Season category
SEASON_YR	Season year
SETT_GRP_2	Settlement group 2 (rebate settlement, purchasing)
SETT_GRP_3	Settlement group 3 (rebate settlement, purchasing)
SETT_ITEM	Item relevant to subsequent (period-end rebate) settlement
ML_AKT	Material ledger activated at material level
REMSHLIFE	Minimum remaining shelf life
RFQ	RFQ number
RFQ_ITEM	Item number of RFQ
CONFIG_ORG	Origin of configuration
QUOTAUSAGE	Quota arrangement usage
SPSTCK_PHY	Special stock indicator for physical stock transfer
PREQ_NO	Purchase requisition number
PREQ_ITEM	Item number of purchase requisition
MAT_TYPE	Material type
SI_CAT	Sub-item category, purchasing document

<b>PO_ITEMS &gt; ITEM</b>	
SUB_ITEMS	Sub-items exist
SUBTOTAL_1	Condition subtotal 1 from calculation schema
SUBTOTAL_2	Condition subtotal 2 from calculation schema
SUBTOTAL_3	Condition subtotal 3 from calculation schema
SUBTOTAL_4	Condition subtotal 4 from calculation schema
SUBTOTAL_5	Condition subtotal 5 from calculation schema
SUBTOTAL_6	Condition subtotal 6 from calculation schema
SUBITM_KEY	Processing key for sub-items
MAX_CMG	Maximum cumulative materials go-ahead quantity
MAX_CPGO	Maximum cumulative production go-ahead quantity
RET_ITEM	Return item
AT_RELEV	Relevant to allocation table
ORD_REAS	Reason for ordering
DEL_TYP_RT	Delivery type for returns to vendors
PRDTE_CTRL	Material freight group
MANUF_PROF	Manufacturer part profile
MANU_MAT	Manufacturer part number
MFR_NO	Manufacturer number
MFR_NO_EXT	External manufacturer code name or number
ITEM_CAT_EXT	Item category in purchasing document
PO_UNIT_ISO	Order unit in ISO code
ORDERPR_UN_ISO	ISO code for purchase order price unit
BASE_UOM_ISO	Base unit of measure in ISO code
WEIGHTUNIT_ISO	ISO code for unit of weight
VOLUMEUNIT_ISO	Volume unit in ISO code
POINTS_UN_ISO	ISO code for points unit
CONF_MATL_EXTERNAL	Long material number (future development) for CONF_ field
CONF_MATL_GUID	External GUID (future development) for CONF_MATL field
CONF_MATL_VERSION	Version number (future development) for CONF_MATL field
MATERIAL_EXTERNAL	Long material number (future development) for MATERIAL field
MATERIAL_GUID	External GUID (future development) for MATERIAL field
MATERIAL_VERSION	Version number (future development) for MATERIAL field
PUR_MAT_EXTERNAL	Long material number (future development) for PUR_M field
PUR_MAT_GUID	External GUID (future development) for PUR_MAT field

<b>PO_ITEMS &gt; ITEM</b>	
PUR_MAT_VERSION	Version number (future development) for PUR_MAT field

## Inner document - PO flip

The data for Process Director Accounts Payable documents is retrieved by the `BAPI_PO_GETDETAIL` function module.

<b>PO_ITEMS &gt; inner document</b>	
PO_NUMBER	Purchasing document number
PO_ITEM	Item number of purchasing document
DELETE_IND	Deletion indicator in purchasing document
SHORT_TEXT	Short text
MATERIAL	Material number
QUANTITY	Purchase order quantity
UNIT	Order unit
ORDERPR_UN	Order price unit (purchasing)
CONV_NUM1	Numerator for conversion of order price unit to order unit
CONV_DEN1	Denominator for conversion of order price unit to order unit
CONV_NUM2	Numerator for conversion of order unit to base unit
CONV_DEN2	Denominator for conversion of order unit to base unit
NET_PRICE	Net price in purchasing document (in document currency)
PRICE_UNIT	Price unit
TAX_CODE	Tax on sales/purchases code
OVERDELTO	Overdelivery tolerance limit
UNLIMITED	Indicator: Unlimited overdelivery allowed
UNDER_TOL	Underdelivery tolerance limit
FINAL_INV	Final invoice indicator
ITEM_CAT	Item category in purchasing document

<b>PO_ITEM_HISTORY_TOTALS &gt; inner document</b>	
DELIV_QTY	Quantity of goods received
IV_QTY	Quantity invoiced

## Vendor master data

The data for Process Director Accounts Payable documents is retrieved by the `BAPI_VENDOR_GETDETAIL` function module.

<b>GENERALDETAIL &gt; HEADER</b>	
VENDOR	Account number of vendor or creditor
NAME	Name 1
NAME_2	Name 2
NAME_3	Name 3
NAME_4	Name 4
CITY	City
DISTRICT	District
PO_BOX	P.O. box
POBX_PCD	P.O. box postal code
POSTL_CODE	Postal code
REGION	Region (state, province, county)
STREET	House number and street
COUNTRY	Country key
COUNTRYISO	Country ISO code
POBX_CTY	PO box of city
LANGU	Language key
LANGU_ISO	Language according to ISO 639
BAPIVENDOR_04	All fields from append BAPIVENDOR_04

<b>BANKDETAIL &gt; BANKDETAIL</b>	
VENDOR	Account number of vendor or creditor
BANK_CTRY	Bank country key
BANK_KEY	Bank key
BANK_ACCT	Bank account number
CTRL_KEY	Bank control key
PARTNER_BK	Partner bank type
COLL_AUTH	Indicator: Is there collection authorization?
BANK_REF	Reference specifications for bank details
BAPIVENDOR_06_APPEND	All fields from append BAPIVENDOR_06_APPEND

Data retrieved from the LFA1 table:

Table LFA1 > HEADER	
LOEVM	Central deletion flag for master record
SPERR	Central posting block

Data retrieved from the TIBAN table:

Table LFA1 > HEADER	
IBAN	IBAN (International Bank Account Number)

The data for Process Director Accounts Payable documents is retrieved by the BAPI\_VENDOR\_GETDETAIL function module.

Table COMPANYDETAIL > COMPANYDETAIL	
VENDOR	Account number of vendor or creditor
COMP_CODE	Company code
CLERK	Accounting clerk
HD_OFFICE	Head office account number
ALT_PAYEE	Account number of alternative payee
CUVD_CLEAR	Indicator: Clearing between customer and vendor
PMNTTRMS	Terms of payment key
ACT_AT_VEN	Our account number with vendor
VEND_USER	Clerk at vendor
INTERNET	Internet address of partner company clerk
FAX	Accounting clerk's fax number at the customer/vendor
BAPIVENDOR_05	All fields from append BAPIVENDOR_05

Data retrieved from the LFB table:

Table LFB1 > COMPANYDETAIL	
LOEVM	Central deletion flag for master record
SPERR	Central posting block

Data retrieved from the LFAS table:

Table LFAS > VAT_NUMBER	
LAND1	Country key
STCEG	VAT registration number

Data retrieved from the WYT3 table:

Table WYT3 > BUS_PARTNER	
LIFN2	Reference to other vendor