Print-Mailing Dispatch Preparation API

v 3.18.0-1987-da209f0

Division: Post & Parcel Germany

This API is used for dispatching **Dialogue Marketing** print mailings.

- Calculate and optimize prices
- Create shipping documents
- Franking
- Place and change your order
- Use an easy way for dispatch preparation in your software

Region: Germany
Used for: Shipping, Advertising

Overview

Automate all steps for dispatching your print mailing. With the **Print-Mailing Dispatch Preparation API**, you calculate the prices, including postage and production optimization, frank your items using IT franking and generate all accompanying documents required for shipping, such as the posting list.

With the Print-Mailing API for Dispatch Preparation, you can use your software to simply and automatically prepare your print mailing for shipping:

- Calculate prices by entering address data and production parameters

- Optimize postage by entering production parameters such as the shipment format. Current requirements of Deutsche Post are automatically taken into account
- The dispatch preparation supports franking types such as the franking wave or franking impression. IT franking is an integral element.
- You can create, change or cancel orders using the order management system.
- Create accompanying documentation such as the posting list and bundle, tray and pallet labels. Digital integration into customer processes.
- Post items yourself or organize their pickup automatically.

The benefits for you

- Prepare the dispatch of your print mailing with one software for all Deutsche Post dialogue marketing products
- Full integration into your system landscape and process flows
- No license fees
- New functions, data and prices without time-consuming software updates
- Compliant with GDPR and Postal Act
- Data processing in Germany and the EU (private cloud of Deutsche Post DHL) For more information, see https://www.deutschepost.de/de/p/print-mailing/partner.html.

The web services covers the following postal products of DPAG:

- DIALOGPOST
- POSTWURFSPEZIAL
- POSTAKTUELL.

The Dispatch Preparation API perfectly harmonises with the Targeting API from Deutsche Post, which will support you to select regions or postal codes resulting in the required input for the POSTAKTUELL or POSTWURFSPEZIAL pricecalcualtion.

Find more information about the **Targeting API** here.

Additional services

With **IT franking**, you organize your mail and merchandise shipments in a simple, cost-effective and efficient way via your own IT systems. It is not possible to transfer existing IT contracts. As is the case for other applications, a new IT contract must be additionally created and/or agreed for the Print-Mailing API for Dispatch Preparation. Here you will find all the necessary information and access requirements.

The intelligent process for your digital address maintenance: Click here to learn all about our **Premiumadress** product.

When you post Postwurfspezial items for the first time, you must conclude a permanent contract on the use of **Postwurfspezial** partial addresses with Deutsche Post before the partial addresses are given to you. Please contact your customer advisor about this.

The API transmits the result of your optimization directly to the **Deutsche Post electronic order management system (OMS)**. Not only can you create new orders, you can also change or cancel existing orders. When using the API, you do not require separate access data for data exchange with the OMS.

Audience information

The primary audience of this documentation are readers with a technical background, who are interested in

- estimating the complexity of integrating the service
- understanding the business and technical requirements and
- integrating the respective APIs to the service.

Get Access

The API functions for price calculation of the COST CALCULATION API resources are freely available to you, without having credentials:

- Simple/Standard cost calculation for DIALOGPOST
- Standard cost calculation for POSTWURFSPEZIAL
- Standard cost calculation for POSTAKTUELL

In order to **process and book orders and receive shipment documents** with the PROCESSING API resources, you need to have an own user environment and credentials.

1. Please fill the Einrichtungsauftrag_Dispatchprepation*.docx and
2. send it to it-csp@deutschepost.de.
3. Use "[#dis] your company name" as e-mail subject header.
4. Our staff will contact you directly.

Initially, only the access data for the test environment will be created. Following successful acceptance, the access data for the live system can be made available. Further information is available at Approval Terms.

To find the **External Customer ID** of the Group Developer Portal, you need to register.

After log-in, the id can be found in the User Profile.

Environments

We offer two environments which may differ in their version and documentation

- Production environment (PROD)
  - use this Documentation
- Test Environment (UAT)
  - use this Documentation for UAT
  - serves as a sandbox
  - *Requests made to the test environment never affect productive systems and are not billed*.

See the servers drop-down menu in the Reference Docs for the environments URLs.

There are two different technical specifications for both the production environment (PROD) and test environment (UAT). Updates will be available in UAT first. These updates then also appear with a time lag on PROD. You can see from the release notes of both documentations whether the technical specification for the UAT is more up-to-date than for the PROD.

Using the API

This API is built conform to the REST architectural style to provide interoperability.

Exposed API resources can be manipulated using HTTP methods. Requests to a resource URL generate a response with JSON payload containing detailed information about the requested resource. HTTP response codes are used to indicate API errors.

We use a token-based approach to authorize requests coming from your system.

Http requests

Apply the standardized HTTP method semantics to communicate with the Print-Mailing Automation API.

The API offers clearly identifiable **resources** ("things of the Print-Mailing Automation universe"). You can use a resource's **representation** to create, update, delete and execute resources. There are different types of resources:

- **Single resource**: E.g. a systemuser, a campaign and its corresponding properties. A PUT updates properties of an existing single resource identified by an id.
- **List resource**: A (pageable) list of single resources e.g. a list of systemusers. Usually, a list resource doesn't include all properties of the single resources. A POST adds a new single resource to a list resource.
- **Processing resource**: Triggers a usually long running asynchronous process. This might be a price calculation or a campaign order. A POST is used to trigger processing resources.
- **Lookup resource**: Offer permitted (enum) field values for simpler communication with the API. API clients can see which values they can expect and have to include in requests. E.g. values for campaign states (active, paused, ...). Usually, lookup resources include an internationalized label.

POST requests are used to create single resources on a list resource. The semantic for list endpoints is best described as 'please add the enclosed representation to the list resource identified by the URL'. On a successful POST request, the server will create one or multiple new resources and provide their URLs in the response. Successful POST requests will usually generate 201.

PUT requests are used to update an **entire** resources, which already exists and thus can be clearly identified by an id. The semantic is best described as "please put the enclosed representation at the resource mentioned by the URL, replacing any existing resource". On successful PUT requests, the server will replace the entire resource addressed by the URL with the representation passed in the payload (subsequent reads will deliver the same payload). Successful PUT requests will usually generate 200.

GET, **DELETE** and **OPTIONS** requests follow the standardized HTTP method semantics.

Headers

You can (and sometimes have to) set HTTP headers. This could be the 'Content-Type', 'Accept', 'Accept-Encoding' or 'Authorization' headers, among others. Please only set headers that are necessary.

Caution

Be aware that HTTP headers with empty values are not allowed for security reasons and such requests will be rejected.

Payload and media types

The standard media type for requests and responses is **application/json** with default encoding **UTF-8**. However, other media types like **application/xml** are supported, too. See the concrete resource definition for all accepted media types.

Tip

Set the header **'Accept: application/json'** in your JSON requests in order to get a detailed JSON response in case of an error.

Use the same semantics for **null** and **absent** properties. OpenApi 3.x allows to mark properties as required and as nullable to specify whether properties may be absent ({}) or null ({"example":null}). If a property is defined to be not required and nullable (see 2nd row in Table below), this rule demands that both cases must be handled in the exact same manner by specification.

The following table shows all combinations and whether the examples are valid:

required	nullable	{}	{"example":null}
true	true	No	Yes
false	true	Yes	Yes
true	false	No	No
false	false	Yes	No

boolean properties must not be null. A boolean is essentially a closed enumeration of two values, true and false.

HTTP response compression

The Print-Mailing Automation API will compress all text-based responses using your favorite compression algorithm.

Tip

Please set a correct **Accept-Encoding** http header for all your request, if applicable!

This increases the API user experience and saves bandwidth.

Date and time formats

Use the date and time formats defined by RFC 3339:

- For "date" use strings matching yyyy-MM-dd, for example: 2026-05-28
- For "date-time" use strings matching yyyy-MM-dd'T'HH:mm:ss.SSS'Z', for example 2026-05-28T14:07:17.000Z

Note that the OpenApi format "date-time" corresponds to "date-time" in the RFC and "date" corresponds to "full-date" in RFC. Both are specific profiles, a subset of the international standard ISO 8601.

A zone offset must not be used. Date time values are always interpreted in **UTC** without offsets. Localization of date time should be done by the api client, if necessary.

Status codes and common error structures

Apply the standardized HTTP status code semantics to communicate with the Print-Mailing Automation API.

The OpenApi specification of every resource provides information about specific success and error responses. It is not a complete list to avoid an overload with common sense information. See below the list of most commonly used status codes in the Print-Mailing Automation API:

Status code	Meaning
200 OK	Request completed successfully.
201 Created	Request completed successfully. Resource created.
400 Bad Request	Invalid/not parseable JSON payload. Non existing enum value recognized. Concurrent update of a resource.
401 Unauthorized	Missing access token in the request headers.
403 Forbidden	Access denied because of missing authority. E.g. customer id is not allowed.

404 Not Found	The resource is not found.
422 Unprocessable Entity	The request could be parsed but the contents are invalid. See the business validation error code for further information.
500 Internal Server Error	A generic error indication for an unexpected server execution problem.
503 Service Unavailable	Service is (temporarily) not available.

In case of an error or input data validation failure, the services will respond with the following error representation in the response body (including internationalized messages).

For general errors (e.g. service unavailable):

```
{
  "timestamp": "2026-02-18T16:11:13.456Z",
  "status": 503,
  "correlationId": "3ea93639-3d7f-4a5c-88ee-f6b6d01dffac",
  "error": "Service Unavailable",
  "errors": [{
    "errorCode": "DM_CO_0012",
    "errorMessage": "The requested functionality is currently not available.Try again later.",
    "arguments": [],
    "fieldName": null
  }]
}
```

For specific validation of a sent field:

```
{
  "timestamp": "2026-02-18T16:12:05.352Z",
  "status": 422,
  "correlationId": "08a884be-3aec-4fa5-b647-7f82149144d5",
  "error": "Unprocessable Entity",
  "errors": [{
    "errorCode": "NotBlank",
    "errorMessage": "must not be empty",
    "arguments": [],
    "fieldName": "password"
    }]
}
```

Tip

Notice the unique **errorCode**: It describes a specific error case which is documented in the OpenApi. Let your application handle these specific error cases and don't only look for http status codes!

If you want to get in contact with us because of an error response, please also supply the **correlationId**. This is a unique id which is generated for every request and can help us to trace requests through our backend.

Common representation structures

Most of the resource representations include common fields:

```
{
   "id":87075,
   "createdOn":"2020-04-02T11:48:51.000Z",
   "changedOn":"2020-04-29T13:02:56.000Z",
   "version":44,
   ".....": "....."
}
```

Use this **id** to address the resource. The **version** is a simple counter informing about resource changes.

A **paged list resource** has the following structure:

```
{
   "elements":[
      {
         "id":549,
         "createdOn":"2020-03-04T09:05:11.000Z",
         "changedOn":null,
         "version":1,
         ".....": "....."
      },
      { ... }
   ],
   "page":{
      "size":50,
      "totalElements":13,
      "totalPages":1,
      "number":0
   },
   "links":[
      {
         "rel":"self",

"href":"https://print-mailing-api.deutschepost.de/user/customers
?page=0&size=50&sort=company,asc"
      }
   ]
}
```

Pagination and filtering list resources

Several list resources of the Print-Mailing Automation API support pagination and filtering to reduce the amount of returned data and improve the API client experience. If no resource fits to the given filtering criteria an empty list is returned. An HTTP 404 is never returned form a list resource.

Request	Meaning

`?page=0&size=50`	Get the first page, maximum 50 items per page.
`?page=0&size=50&company=Dynp`	Get the first page, maximum 50 items per page. Search in field "company" with value "Dynp".
`?page=0&size=50&sort=company,asc`	Get the first page, maximum 50 elements per page. Sort ascending by fieldName "company".
`?sort=company,asc&sort=type,desc`	Get the default number of elements per page. Sort ascending by fieldName "company" and descending by "type".

Caution

The maximum number of returned elements per list resource is always **2000**. You MUST use paging, if you expect bigger result sets.

List resources also include **link relation fields** for easier pagination:

- **self**: The hyperlink pointing to the same page.
- **prev**: The hyperlink pointing to the previous page.
- **next**: The hyperlink pointing to the next page.

Lookup resources

Lookup resources offer permitted (enum) field values for simpler communication with the API. There are cases where a API client has to send a specific value from a set of permitted values. If you have a frontend application, these kind of values usually can be picked from a drop down menu.

A lookup resource publishes all of this permitted values (which are mostly ids) together with an internationalized label.

```
{
   "Font": [{
      "id": 10,
      "label": "Arial"
   },
   {
      "id": 11,
      "label": "Arial Bold"
   }, "...."]
}
```

Error code lookup resources

Error code lookup resources offer a complete list of error codes, specific resources may return in an error response. This resource can be helpful during API client development. An error code lookup resource publishes all of this codes with an internationalized label.

```
{
   "CommonBusinessError":[
      {
         "value":"DM_CO_0001",
         "label":"Auswahlwert existiert nicht."
      }, "..."
   ],
   "RecipientBusinessError":[
      {
         "value":"DM_RC_0001",
         "label":"Die Adressfelder des Print-Mailings sind nicht
definiert."
      },
      {
         "value":"DM_RC_0002",
         "label":"Das Print-Mailing existiert nicht."
      }, "..."
   ]
}
```

Error code lookup resources may include **common business errors**. This type of errors is common to all resources.

Versioning

The API has explicit API versioning within the URL (`/v1/`). However, we apply the following rules to evolve the Print-Mailing Automation API in a backward-compatible way to stay on v1 as long as possible:

- Add only optional, never mandatory fields.
- Never change the semantic of fields (e.g. changing the semantic from customer-ekp to customer-id, as both are different unique customer keys)
- Input fields may have (complex) constraints being validated via server-side business logic. Never change the validation logic to be more restrictive and make sure that all constraints are clearly defined in description.
- Support redirection in case an URL has to change 301 (Moved Permanently).

API clients should apply the following principles:

- Be conservative with API requests and data passed as input, e.g. avoid to exploit definition deficits like passing megabytes of strings with unspecified maximum length.
- Be tolerant in processing and reading data of API responses.
- Be tolerant with unknown fields in the payload, i.e. ignore new fields **but do not eliminate them from payload if needed for subsequent PUT requests**. Remember, the semantics of PUT.
- Be prepared that enum return parameter may deliver new values; either be agnostic or provide default behavior for unknown values.
- Be prepared to handle HTTP status codes not explicitly specified in endpoint definitions. Note also, that status codes are extensible. Default handling is how you would treat the corresponding 2xx code (see RFC 7231 Section 6).
- Follow the redirect when the server returns HTTP status code 301 (Moved Permanently).

Authentication

Most of the Print-Mailing Automation API resources are secured using OAuth 2.0.

The first step accessing the Print-Mailing Automation API is to create an access token JWT (json web token) using the `user/authentication` resource.

Add this access token, which usually starts with 'ey...', to the `Authorization http header` of every request:

```
Authorization: Bearer eyJJKK78Bhu53KB.......iJKV1QiL
GET https://print-mailing-api.deutschepost.de/resource/4711
```

You can decode (base64) the Print-Mailing Automation JWT to get further information for communicating with the API. A JWT contains the following payload:

```
{
    "sub": "paul",
    "iss": "com.dpdhl.dialogmarketing",
    "fullName": "Paul Watson",
    "customerIds": [ "4" ],
    "exp": 1588272706,
    "locale": "de",
    "iat": 1588269106,
    "userId": 2,
    "authorities": [ "ROLE_PMP_ACCESS", "ROLE_DIS_ACCESS",
"ROLE_DIS_MANAGE_DIALOGPOST" ]
}
```

The most important fields are:

- **customerIds**: Usually the Ids of the customer(s) you have access to. In the most cases, this will be exactly one. You need this id for subsequent requests. Depending on the API you access, this might be you employer. In this case, you have access to several other customers and you are offered a resource which returns the list of accessible customer Ids.
- **exp**: Expiry datetime of the token in UTC (jwt standard field).
- **locale**: Your chosen local for internationalization.
- **userId**: This is your systemuser id.

- **authorities**: An array containing all your access rights.

Every access token has a specific **valid time** (about 30 minutes). An expired access token can never be used again and lets the Print-Mailing Automation refuse the whole request. You have to call the `user/authentication/reauth` resource with a valid token to get a fresh access token with the maximum valid time. We recommend to get a new access token a couple of minutes before the expiration time.

Warning

Handle your access token like a password!

Depending on your systemuser account, you have a specific number of **authorities** ("access rights"). The OpenApi specification of every resource provides information about the authorities you have to own to access a resource.

Creating access tokens

As an API systemuser you received the following login data from us:

- **username**: Your unique username.
- **password**: A secret for authentication.

Use the `/user/authentication/businesslogin` resource to create an access token, issued with user account details and granted rights. Your token will include the authority 'ROLE_DIS_ACCESS' for accessing the dispatch preparation api. Others, like the authority 'ROLE_DIS_MANAGE_DIALOGPOST', are granting rights to access specific api resources.

Access rights are assigned to your user account with your initial request for account creation. If you need more access rights, please ask the IT-CSP support to update your user account accordingly.

Example request and response for a partnersystem login:

```
POST
https://print-mailing-api.deutschepost.de/user/authentication/businesslogin
Accept: application/json
```

```
Accept-Encoding: gzip
Content-Type: application/json

{
  "username": "dis_ddh3k9dns",
  "password": "SECRET",
  "locale": "de"
}

HTTP/1.1 200
{
  "jwtToken": "eyJ0eXiOi........rJGiHMTcrn_3T9Og"
}
```

COST CALCULATION API

This section walks you through the currently available resources for cost calculation. You don't need any authentication to create a cost calculation.

DIALOGPOST: Create a cost calculation

You can calculate costs with (standard cost calculation) or without (simple cost calculation) mailing items. You have to specify mailing items (postcal codes) to use the standard cost calculation and you can determine whether the response includes sorted mailing items or not. The simple cost calculation is primary designed for Onlines Shops in order to get a roughly price estimation. The standard cost calculation returns the exact price based on standard values for the production.

Note

By now there is no postal code validation implemented in cost calculation resources.

The following internal **production parameters** are used for standard cost calculations:

- Franking Type: FRANKIERVERMERK(1)
- Franking Type remaining items: POSTWERTZEICHEN(11)
- entitledToDeductPreTax: Optimizer pays taxes

- notEnabledForAutomation: Mailing items are enabled for automation
- dialogpostDisabled: This is a promotional mailing
- trayType: SIZE_1(1)
- dialogpostKoop: Dialogpost-Koop is not used
- remainingItemsDisabled: Remaining items are enabled
- maxNumberExtraItems: 0 (only relevant if unwrapped is true)
- unwrapped: false
- gogreen: false
- trackmatch: false
- Only one variant is possible
- variantProductionType: EINSTROM(1)
- Bundles are not used
- alwaysProducePallets: Do not always produce pallets
- emptyWeightInGram of pallet: 22000
- maxGrossWeightInGram of pallet: 700000
- traysStackable: false
- minNumberItemsLr Leitregion: 0 (means produce as much LR pallets as possible)
- minNumberItemsLz Leitzone: 0 (means produce as much LZ pallets as possible)
- minSavingToProduceExtraPallet: 0 (means any amount of gained rebate will be enough to produce an extra pallet)
- Not partial posting compliant (Teileinlieferung)

Example request and response for creating a **simple** cost calculation:

```
POST
https://print-mailing-api.deutschepost.de/dispatchpreparation/dialogpost/simplecostcalculation
Content-Type: application/json
Accept: application/json
Accept-Encoding: gzip

{
  "quantity": 6000,
  "lengthInDeciMm": 2200,
  "widthInDeciMm": 1100,
  "heightInDeciMm": 20,
  "weightInGram": 20,
  "inductionDate": "2021-02-03T12:26:24.782Z",
  "entitledToDeductPreTax": true,
  "frankingType": 1,
  "frankingTypeRemainingItems": 3,
  "lrDpPossible": false,
  "notEnabledForAutomation": false,
  "mailingItemTypePostcard": false,
  "dialogPostDisabled": false
}
```

```
HTTP/1.1 200
{
  "gtcVersion": "2021-1",
  "costs": 214200,
  "dialogpost": {
    "baseFormatId": 1,
    "dialogpostLr": false,
    "dialogpostEasy": false,
    "postages": {
      "numberExtraItems": 0,
      "numberItemsTotal": 6000,
      "postageAllItems": 180000,
      "postageAdditionsTotal": 0,
      "rebateExtraItems": 0,
      "rebateFrankingTotal": 0,
      "rebateTotal": 0,
      "postageTotal": 180000,
      "postageVat": 34200,
      "postageSingleItem": 30
    }
  },
  "remainingItems": {
    "variants": [
      {
        "baseFormatId": 0,
        "postages": {
          "numberRealItems": 0,
          "postageAllItems": 0,
          "rebateFrankingTotal": 0,
          "postageTotal": 0,
          "postageSingleItem": 0
        }
      }
    ]
  }
}
```

Example request and response for creating a **standard** cost calculation with mailing items:

```
POST
https://print-mailing-api.deutschepost.de/dispatchpreparation/dialogpost/standardcostcalculation
Content-Type: application/json
Accept: application/json
Accept-Encoding: gzip

{
  "lengthInDeciMm":2200,
  "widthInDeciMm":1100,
  "heightInDeciMm":20,
  "weightInGram":20,
  "inductionDate":"2021-08-03T11:21:45.000Z",
  "entitledToDeductPreTax":true,
  "frankingType":3,
  "frankingTypeRemainingItems":11,
  "notEnabledForAutomation":false,
  "mailingItemTypePostcard":false,
  "dialogpostDisabled":false,
  "createSortedMailingItems":true,
  "mailingItems":[

{"id":1,"zip":"64293"},{"id":2,"zip":"64293"},{"id":3,"zip":"64293"}, ...
  ]
}


HTTP/1.1 200
{
  "gtcVersion": "2021-1",
  "creationDateTime": "2021-08-03T12:06:45.389Z",
  "statusSeverityId": 1,
  "statusMessages": [
    {
      "statusSeverityId": 1,
      "code": 1001,
      "message": "All postages are without obligation. In any case the current rates, tariffs and taxes are valid."
    },
    {
      "statusSeverityId": 1,
      "code": 1502,
      "message": "PLZ bundles/trays will not be produced because the item format is not 'Groß'."
    }
  ],
  "dialogpost": {
    "baseFormatId": 1,
    "dialogpostLr": true,
    "dialogpostEasy": false,
    "postages": {
      "variantId": 0,
      "numberExtraItems": 0,
      "numberItemsTotal": 500,
      "postageAllItems": 15000,
      "postageAdditionsTotal": 0,
      "rebateExtraItems": 0,
      "rebateFrankingTotal": 150,
      "rebateTotal": 150,
      "numberRealItems": 500,
      "numberItemsInContainersRebatedLr": 0,
      "numberItemsInContainersRebatedPlz": 0,
      "numberItemsInPalletsRebatedLz": 0,
      "numberItemsInPalletsRebatedLr": 0,
      "rebateContainersLr": 0,
      "rebateContainersPlz": 0,
      "rebatePalletsLr": 0,
      "rebatePalletsLz": 0,
      "rebateProductionTotal": 0,
      "weightAllItemsInGram": 10000,
      "postageExtraServicesTotal": 0,
      "postageFrankierserviceTotal": 0,
      "postageTotal": 14850,
      "postageVat": 2822,
      "postageSingleItem": 30,
      "rebatePercentageContainersLr": 0,
      "rebatePercentageContainersPlz": 0,
      "rebatePercentagePalletsLz": 0,
      "rebatePercentagePalletsLr": 0,
      "rebatePercentageExtraItems": 0,
      "rebatePrepaid": 0,
      "postageVatTotal": 2822,
      "postageTotalGross": 17672,
      "postageToPay": 17672,
      "numberItemsInContainersNotRebatedLr": 0,
      "numberItemsInPalletsNotRebatedLz": 0,
      "numberItemsInPalletsNotRebatedLr": 0,
      "notEnabledForAutomation": null
    },
    "items": [
      {
        "id": 1,
        "originalId": 1
      },
      {
        "id": 2,
        "originalId": 2
      },
      {
        "id": 3,
        "originalId": 3
      },
      .....
    ]
  },
  "remainingItems": null,
  "ignoredItems": null,
  "usedValues": {
    "itemDefinition": {
      "variants": [
        {
          "maxNumberItemsPerContainer": 200,
          "numberItemsForTrayHalfFull": 100
        }
      ]
    },
    "productionDefinition": {
      "trays": {
        "trayType": 1
      },
      "pallets": null
    }
  }
}
```

The most important fields are:

- **createSortedMailingItems**: Is set to true in order to include a sorted list of mailing items, as well as the remaining mailing items and the ignored mailing items in the response.
- **dialogpost.postages**: Detailed information about postage.
- **items**: The sorted list of mailing items.

See the specific OpenAPI definitions in Reference Docs for a detailed description of the representation structure.

POSTWURFSPEZIAL: Create a cost calculation

You need the quantity result of a "POSTWURFSPEZIAL Target Group Selection" to create the request. Example request and response for creating a simple cost calculation:

```
POST
https://print-mailing-api.deutschepost.de/dispatchpreparation/postwurfspezial/simplecostcalculation
Content-Type: application/json
Accept: application/json
Accept-Encoding: gzip

{
  "quantity": 6000,
  "lengthInDeciMm": 2200,
  "widthInDeciMm": 1100,
  "heightInDeciMm": 20,
  "weightInGram": 98,
  "inductionDate": "2021-02-03T12:26:24.782Z",
  "mailingItemTypePostcard": false,
  "notEnabledForAutomation": false
}
```

```
HTTP/1.1 200
{
  "gtcVersion": "2021-1",
  "costs": 221340,
  "creationDateTime": "2021-07-30T12:42:17.556Z",
  "postwurfspezial": {
    "postageTotal": 186000,
    "pwspEasy": true,
    "postageSingleItem": 26,
    "postageAllItems": 156000,
    "postageAdditionsTotal": 30000,
    "postageExtraServicesTotal": 0,
    "postageVat": 35340,
    "numberExtraItems": 0
  }
}
```

See the specific OpenAPI definitions in Reference Docs for a detailled description of the representation structure.

POSTAKTUELL: Create a cost calculation

You need the result of a "POSTAKTUELL Selection" to create the request. To do so you call dispatch-information from the Targeting API, take the result and finally call postaktuell/simplecostcalculation.

For more details on the dispatch information call from the Targeting API please refer to Targeting API documentation.

Example request and response for creating a simple cost calculation:

```
POST
https://print-mailing-api.deutschepost.de/dispatchpreparation/postaktuell/simplecostcalculation
Content-Type: application/json
Accept: application/json
Accept-Encoding: gzip

{
  "numberItemsTariffZoneA": 34367,
  "numberItemsTariffZoneB": 1129,
  "itemWeightInGram": 98,
  "paProductVariant": 1,
  "deliveryDistrictSelection": true,
  "inductionDate": "2021-03-23T14:06:33.261Z"
}

HTTP/1.1 200
{
  "gtcVersion": "2021-1",
  "creationDateTime": "2021-03-23T14:12:02.402Z",
  "costs": 853194,
  "postaktuellValues": {
    "postagesTariffA": {
      "numberItems": 34367,
      "costsPer1000": 19100,
      "costsTotal": 656410
    },
    "postagesTariffB": {
      "numberItems": 1129,
      "costsPer1000": 22200,
      "costsTotal": 25064
    },
    "totalNet": 716970,
    "totalVat": 136224,
    "totalGross": 853194,
    "additionalChargeForMinimumOrder": 0,
    "additionalChargeDeliveryDistrictSelection": 35496
  }
}
```

See the specific OpenAPI definitions in Reference Docs for a detailed description of the representation structure.

PROCESSING API

"Processing" your mailing means that the Dispatchpreparation API takes all the necessary input data (shipment type, franking type, quantity, measures, ...) and prepares the necessary documents for posting and preparation.

Note

For mailing processing you need a tenant context within the Dispatchpreparation system which is identified by a **customer Id**. This tenant context comprises all of your specific processing and master data. Please contact us at it-csp@deutschepost.de with a mail subject "[#dis] API access" to get your personal authentication data.

Prior to processing, you may add the following master data:

- **DV Contracts**: If you use DV-Freimachung as your frankingtype for Dialogpost you have to define all relevant contract information you negotiated with your DV franking consultant. For more information about DV-Freimachung please refer to here.
- **Addresses**: You have to define all involved parties mentioned e.g. in contracts, Deutsche Post Ordermanagement (AM) Registration or processings.
- **Production definitions**: There is a set of standard production definitions. However, you may add additional production definitions if you have specific requirements regarding the production.

PROCESSING API - Common

Master data

Master data is used within your processings. You may add, update or delete you master data as needed. The next chapters explain each master data resource.

Addresses

You need a **valid access token** including the access right ROLE_DIS_ACCESS to continue (see this section).

For several API functions it is necessary to define addresses of involved parties. You have to define all involved parties mentioned e.g. in AM Registration or processings.

A party is identified by an EKP number (Unique Customer ID of Deutsche Post), has a postal address and a business contact.

- **Originator EKP**: The company who is the sender of the mailing.
- **Submitter EKP**: The company responsible for handing over the mailing at the induction date to Deutsche Post.

Note

All EKPs used in a processing must be **defined as address** master data prior to processing execution.

Example request and response to show all your addresses:

```
GET https://print-mailing-api.deutschepost.de/dispatchpreparation/addresses?customerId=74578
Accept: application/json
Accept-Encoding: gzip
Authorization: Bearer ACCESS-TOKEN
```

```
HTTP/1.1 200
{
  "elements": [
    {
      "id": 2543,
      "ekp": "5045287962",
      "name": "Dispatch Corp.",
      "street": "221B Baker Street",
      "zip": "W1U 4HG",
      "city": "London",
      "country": "EN",
      "customerContact": "",
      "tel": "",
      "fax": "",
      "email": "",
      "epost": ""
    },
    {
      "id": 2544,
      "ekp": "0000000001",
      "name": "Print First GmbH",
      "street": "Frankenstr. 33",
      "zip": "90443",
      "city": "Nürnberg",
      "country": "DE",
      "customerContact": "Robert Helm",
      "tel": "",
      "fax": "",
      "email": "",
      "epost": ""
    }
  ],
  "page": {
    "size": 2,
    "totalElements": 2,
    "totalPages": 1,
    "number": 0
  },
  "links": []
}
```

Example request and response to create a new address:

```
POST
https://print-mailing-api.deutschepost.de/dispatchpreparation/addresses
Accept: application/json
Accept-Encoding: gzip
Content-Type: application/json
Authorization: Bearer ACCESS-TOKEN
```

```
{
  "customerId": 74578,
  "ekp": "1234567890",
  "name": "My GmbH",
  "street": "Musterstraße 12",
  "zip": "12345",
  "city": "Bonn",
  "country": "DE",
  "customerContact": "Max Mustermann",
  "tel": "00491234123456",
  "fax": "0049987654373",
  "email": "max.mustermann@deutschepost.de",
  "epost": "string"
}

HTTP/1.1 201 Created
{
    "id": 3721,
    "ekp": "1234567890",
    "name": "My GmbH",
    "street": "Musterstraße 12",
    "zip": "12345",
    "city": "Bonn",
    "country": "DE",
    "customerContact": "Max Mustermann",
    "tel": "00491234123456",
    "fax": "0049987654373",
    "email": "max.mustermann@deutschepost.de",
    "epost": "string"
}
```

The most important fields are:

- **ekp**: A unique (within your tenant context) number identifying one of your addresses.

Default payment information (POSTAKTUELL and POSTWURFSPEZIAL)

For POSTAKTUELL and POSTWURFSPEZIAL processings you don't need contracts to define payment information like for Dialogpost. You can pay with Deutsche Post Postcard (POSTAKTUELL and POSTWURFSPEZIAL) or a bank account (POSTAKTUELL). You can either define default payment information (using the `defaultpayment` resource) or define them in every processing. **You can define one default payment per customerId you have access to.**

The `defaultpayment` resource allows for storing a single payment information which is applied as a default to your processings. You may choose from different payment methods. Find the supported types in lookup resource `PaymentMethodType`. Make sure to create an address (see this section), corresponding to the 'payerEkp' beforehand.

Example request and response to add default payment information:

```
POST
https://print-mailing-api.deutschepost.de/dispatchpreparation/defaultpayment
Accept: application/json
Accept-Encoding: gzip
Content-Type: application/json
Authorization: Bearer ACCESS-TOKEN

{
    "customerId": 74578,
    "payerEkp": "0000000001",
    "entitledToDeductPreTax": true,
    "paymentMethod": 3,
    "cardNo": "2502001",
    "iban": null,
    "bic": null,
    "bankName": null
}

HTTP/1.1 201 Created
{
    "id": 1564,
    "payerEkp": "0000000001",
    "entitledToDeductPreTax": true,
    "paymentMethod": 3,
    "cardNo": "2502001",
    "iban": null,
    "bic": null,
    "bankName": null
}
```

Default induction information (POSTAKTUELL and POSTWURFSPEZIAL)

For POSTAKTUELL and POSTWURFSPEZIAL processings you don't need contracts to define induction information like for Dialogpost. You can either define default induction information (using the `defaultinduction` resource) or define them in every processing. **You can define one default induction per customerId you have access to.**

The `defaultinduction` resource allows for storing a single induction information which is applied as a default to your processings. Note that the **induction type** determines whether your mailing items are carried to a Deutsche Post depot or are picked up from an address you define.

Example request and response to add induction information:

```
POST
https://print-mailing-api.deutschepost.de/dispatchpreparation/de
faultinduction
Accept: application/json
Accept-Encoding: gzip
Content-Type: application/json
Authorization: Bearer ACCESS-TOKEN

{
  "customerId": 74578,
  "inductionType":1,
  "depotName": null,
  "pickUpLocationStreet": "Musterstraße",
  "pickUpLocationHouseNumber":"12",
  "pickUpLocationZip":"12345",
  "pickUpLocationCity":"Musterstadt",
  "pickUpLocationCountry":"DE"
}

HTTP/1.1 201 Created
{
  "id": 657,
  "depotName": null,
  "pickUpLocationStreet": "Musterstraße",
  "pickUpLocationHouseNumber":"12",
  "pickUpLocationZip":"12345",
  "pickUpLocationCity":"Musterstadt",
  "pickUpLocationCountry":"DE",
  "inductionType": 1,
  "createdOn": "2021-11-18T14:06:30.000Z",
  "changedOn": null
}
```

Production definitions

You need a **valid access token** including the access right ROLE_DIS_ACCESS to continue (see this section).

You have to describe your mailing specification parameters before starting a processing. They are similar to the ones you already know from the cost calculation resources. In addition, you also have to define mailing production parameters. Mailing production parameters are devided in three groups:

- bundles
- pallets
- trays

You have to specify the neccessary parameters for bundles (exclusive) or trays. In general you will use trays since you are only allowed to use bundles for format GROSS. If the overall gross weight of your mailings is more than 500kg you have to specify the neccessary parameters for pallets.

Note

There exist three **default production definitions** for each product. They are called **'LetterC5'**, **'LetterDL'**, **'PostcardDL'**, **'LetterC5Postwurfspezial'**, **'LetterDLPostwurfspezial'** and **'PostcardDLPostwurfspezial'**. You may use them or create a copy of them with a new production definition name.

Example request and response to create a production definition using bundles:

```
POST
https://print-mailing-api.deutschepost.de/dispatchpreparation/pr
oductiondefinitions
Accept: application/json
Accept-Encoding: gzip
Content-Type: application/json
Authorization: Bearer ACCESS-TOKEN

{
  "name":"Letter C4 Endkundendokumentation",
  "customerId": 74578,
  "mailingItemType": 1,
  "lengthInDeciMm": 3240,
  "widthInDeciMm": 2290,
  "heightInDeciMm": 30,
  "weightInGram": 250,
  "notEnabledForAutomation": false,
  "productionDefinitionFabricationType": 2,
  "trays": null,
  "bundles": {
    "maxNumberItemsPerContainer": 40,
    "productionDefinitionOptimizationType": 1,
    "productionDefinitionNumberItemsDependentOptimizationType":
1,
    "minNumberItemsForContainerLr": 20,
    "minNumberItemsForContainerPlz": 20
  },
  "pallets": {
    "usePallets": true,
    "alwaysProducePallets": false,
    "traysStackable": false,
    "emptyWeightInGram": 20000,
    "maxNumberTraysPerPallet": 32,
    "maxNumberTraysPerLayer": 4,
    "maxNumberMailingItemsPerPallet": 5000,
    "productionDefinitionOptimizationType": 1,
    "minSavingToProduceExtraPallet": 0
  }
}

HTTP/1.1 201 Created
```

```
{
  "name": "Letter C4 Endkundendokumentation",
  "mailingItemType": 1,
  "lengthInDeciMm": 3240,
  "widthInDeciMm": 2290,
  "heightInDeciMm": 30,
  "dialogpostKoop": false,
  "notEnabledForAutomation": false,
  "unwrapped": false,
  "weightInGram": 250,
  "productionDefinitionFabricationType": 2,
  "trays": null,
  },
  "bundles": {
    "maxNumberItemsPerContainer": 40,
    "productionDefinitionOptimizationType": 1,
    "productionDefinitionNumberItemsDependentOptimizationType": 1,
    "minNumberItemsForContainerLr": 20,
    "minNumberItemsForContainerPlz": 20
  },
  "pallets": {
    "usePallets": true,
    "alwaysProducePallets": false,
    "traysStackable": false,
    "emptyWeightInGram": 20000,
    "maxNumberTraysPerPallet": 32,
    "maxNumberTraysPerLayer": 4,
    "maxNumberMailingItemsPerPallet": 5000,
    "productionDefinitionOptimizationType": 1,
    "minSavingToProduceExtraPallet": 0
  },
  "id": 2440
}
```

The most important fields are:

- **name**: This has to be unique (within your tenant context).

- **productionDefinitionFabricationType**: This is tray **OR** bundle production. Depending on your choice, add the tray OR bundle parameters to your request.

Note: Though you can only specify the height of a mailing in mm the maxNumberItemsPerContainer can consider the height in 1/10mm.

Order Records of Deutsche Post Auftragsmanagement (AM)

In the order records resouce, you can find detailed records of the creation of your order in the Deutsche Post Auftragsmanagement (AM). You may change details of the order in AM or cancel the order as needed.

Note

Please be advised that it lies within your responsibility and best interest to verify your Order Creation attempts via this resource. Warnings and error messages are displayed within the amStatus field.

For every processing you can choose if you want to create orders in Deutsche Post Auftragsmanagement (AM). Notice this depends on the **simulation** parameter in the `processings` resource.

Every order record is linked to your processing by the prcessing identifier. A single order record is identified by the alphanumeric message ID from AM (example '210910114130A10001001'). You can find the order identifier from the AM in 'amOrderId' (e.g: 4836374), once the creation was successfully completed.

Example request and response for order records. Note that the current day is set as default:

```
GET
https://print-mailing-api.deutschepost.de/dispatchpreparation/orderrecords?customerId=74578
Accept: application/json
Accept-Encoding: gzip
Authorization: Bearer ACCESS-TOKEN

HTTP/1.1 200
```

```
{
  "elements": [
    {
      "id": "211022095419A10001021",
      "processingId": 2,
      "amOrderId": 4836374,
      "status": 20,
      "type": 1,
      "amInvoiceNumber": 33,
      "amStatus": "OK",
      "originatorEkp": "0000000001",
      "submitterEkp": "0000000002",
      "inductionType": 2,
      "inductionDate": "2021-11-02T20:00:00.000Z",
      "depotName": "Darmstadt",
      "pickUpLocationStreet": "",
      "pickUpLocationHouseNumber": "",
      "pickUpLocationZip": "",
      "pickUpLocationCity": "",
      "pickUpLocationCountry": null
    },
    {...}
  ],
  "page": {
    "size": 2,
    "totalElements": 2,
    "totalPages": 1,
    "number": 0
  },
  "links": []
}
```

The result set includes all order records for the specified customer on this day. To find order records within a specific time frame, you can add further options to widen your search. Current filters include:

- **dateto**: select a specific start date. Format: 2022-06-01
- **datefrom**: a specific end date
- **processingid**: filter your records for the processing you created

E.g: GET (https://print-mailing-api.deutschepost.de/dispatchpreparation/orderrecords?customerId=74578&dateto=2022-06-01&datefrom=2021-08-01&processingid=1)

Change your order

You can change a created order in order management. Find the details of your change in the order records.

Example request and response for changing an order by adding an order record of type 'change':

```
PUT
https://print-mailing-api.deutschepost.de/dispatchpreparation/orderrecords?customerId=74578&processingid=1
Content-Type: application/json
Accept: application/json
Accept-Encoding: gzip
Authorization: Bearer ACCESS-TOKEN

{
  "customerId": 74578,
  "originatorEkp": "0000000001",
  "submitterEkp": "0000000002",
  "inductionDate": "2021-09-10T20:00:00Z"
  "induction": {
    "depotName": "Bautzen",
    "pickUpLocationStreet": "",
    "pickUpLocationHouseNumber": "",
    "pickUpLocationZip": "",
    "pickUpLocationCity": "",
    "pickUpLocationCountry": null
  }
}

HTTP/1.1 200 OK
{}
```

In this example, the induction depot name is changed from the previous example to the new value 'Bautzen'. A new order record is created with your data and the order is updated in AM. You can find the order record and access the details of your request afterwards.

Cancel your order

You can cancel the created order in AM. Thereafter, the order cannot be changed anymore. Find the details of your cancellation in the order records.

Example request and response for deleting an order by adding an order record of type 'change':

```
DELETE
https://print-mailing-api.deutschepost.de/dispatchpreparation/or
derrecords?customerId=74578&processingid=1
Content-Type: application/json
Accept: application/json
Authorization: Bearer ACCESS-TOKEN}

HTTP/1.1 200 OK
{}
```

Processings states

You can stepwise prepare input data for postage optimizations of your mailings. A so-called **processing** contains information about the mailing, production parameters, regions or postal codes and payment.

A processing exists in different states:

- **CREATED (10)**: Processing is saved as draft.
- **RELEASED (20)**: Processing is ready for execution and cannot be changed any longer.
- **GATHERING_MAILINGITEMS (22)**: For POSTWURFSPEZIAL only. Concrete zip codes for your target group selection are prepared. A processing will remain in this state for several hours.
- **SCHEDULED (25)**: Processing is picked for further processing.
- **RUNNING (30)**: Processing is currently running.
- **FINISHED (40)**: Processing is finished.

- **FAILED (50)**: Processing finished with an error.
- **ARCHIVED (90)**: Processing is archived 30 days after the last change. Existing induction documents are deleted. You can still see the processing's metadata.

Only a processing in state CREATED can be released. Furthermore, you cannot release a processing again, e.g. if it failed because of a configuration error.

Your processing should be in the FINISHED state, now. Be aware that this does not automatically imply a successful processing. If you get no result documents please have a look at the `log` resource where you will get detailed information about your processing.

FINISHED processings also do not implicitly mean an Order Creation in Deutsche Post Auftragsmanagement was successful. Please see Order Creations in AM.

Tip

It does however mean, that the resulting documents for the processing were generated

PROCESSING API - DIALOGPOST

Master data

Master data is used within your processings. You may add, update or delete you master data as needed. The next chapters explain each master data resource.

Contracts

You need a **valid access token** including the access right ROLE_DIS_ACCESS to continue (see this section).

Contracts define the basic payment and induction parameters for Dialogpost processings. You can get contracts either by using DV-Freimachung or you apply for a Postcard from Deutsche Post.

For the DV-Freimachung franking type you have to negotiate the terms of a contract with your DV franking consultant. For more information about DV-Freimachung please refer to the official Deutsche Post webpage for DV-Freimachung.

NOTE: Currently, using a Postcard contract it is only possible to process mailings with franking type Frankiervermerk/Frankierwelle.

For more information about a Postcard please refer to the official Deutsche Post webpage for Postcard

You have to add your contract data (for procedures 10 = DV-Freimachung for Dialogpost national or 25 = PostCard) to master data prior to processing executions using the `contracts` resource. Please note that in case you create a contract for payment by Postcard set subscriptionNumber to the last to digits of your Postcard number.

Example request and response to create a contract:

```
POST
https://print-mailing-api.deutschepost.de/dispatchpreparation/di
alogpost/contracts
Content-Type: application/json
Accept: application/json
Accept-Encoding: gzip
Authorization: Bearer ACCESS-TOKEN

{
  "customerId": 74578,
  "ownerEkp": "0000000001",
  "procedureNumber": "25",
  "subscriptionNumber": "01",
  "name": "My IT franking contract",
  "originatorEkp": "0000000002",
  "submitterEkp": "0000000003",
  "entitledToDeductPreTax": true,
  "payment": {
    "cardEkp": "0000000001",
    "cardNo": "2523001"
  },
  "inductionType": 2,
  "induction": {
    "depotName": "Bautzen"
  }
}

HTTP/1.1 201 Created
{
  "id": 47790,
  "ownerEkp": "0000000001",
  "procedureNumber": "25",
  "subscriptionNumber": "01",
  "name": "My IT franking contract",
  "originatorEkp": "0000000002",
  "submitterEkp": "0000000003",
  "entitledToDeductPreTax": true,
  "payment": {
    "cardEkp": "0000000001",
    "cardNo": "2523001"
  },
  "inductionType": 2,
  "induction": {
    "depotName": "Bautzen",
    "pickUpLocationStreet": null,
    "pickUpLocationHouseNumber": null,
    "pickUpLocationZip": null",
    "pickUpLocationCity": null,
    "pickUpLocationCountry": null
  }
}
```

The most important fields are:

- **ownerEkp**: The first 10 digits of the contract number.
- **payment**: Only set Deutsche Post Postcard payment information, if you don't use DV-Freimachung and using a Postcard for payment. Set null otherwise.
- **inductionType**: Choose if mailing items are picked up from a postal address (supply an address) or are carried to a Deutsche Post depot (pick a depot name from the `depotlookups` resource).

Working with processings

You need a **valid access token** including the access right ROLE_DIS_MANAGE_DIALOGPOST to continue. See section this section to learn more about access rights.

1. Create a processing
2. Add information about the mailing
3. Add regions or postal codes
4. Release your processing
5. Get the result of your processing

For the first 3 steps it is possible to delete your processing or mailing items. Only a processing in state CREATED can be released. Furthermore, you cannot release a processing again, e.g. if it failed because of a configuration error.

Create a processing

Example request and response:

```
POST
https://print-mailing-api.deutschepost.de/dispatchpreparation/di
alogpost/processings
Content-Type: application/json
Accept: application/json
Accept-Encoding: gzip
Authorization: Bearer ACCESS-TOKEN

{
  "customerId": 74578,
  "description": "Postcard Mailing 2036"
}

HTTP/1.1 201 Created
{
  "id": 612,
  "description": "Postcard Mailing 2036",
  "processingState": 10,
  "mailingItemsQuantity": 0,
  "mailingItemsErrors": null
}
```

Note

Use the returned `id` from the response for the following requests regarding this processing.

Add information about the mailing

The next step is to add some mailing information to your previously created processing.

Example request and response for Dialogpost:

```
POST https://print-mailing-api.deutschepost.de/dispatchpreparation/dialogpost/processings/612/mailing
Content-Type: application/json
Accept: application/json
Accept-Encoding: gzip
Authorization: Bearer ACCESS-TOKEN

{
  "inductionDate": "2030-09-17T13:14:50.704Z",
  "productionDefinition": 451,
  "submitterEkp": "0000000003",
  "originatorEkp": "0000000002",
  "zipCodeCheck": true,
  "definition": {
    "dialogpostDisabled": false,
    "productionInfoText": "string",
    "accountingInfoText": "string",
    "dataMatrixType": 2,
    "goGreen": false,
    "trackMatch": false,
    "variantProductionType": 1,
    "createDocumentTypes": [
      22,
      15
    ]
  },
  "payment": {
    "ownerEkp": "0000000001",
    "procedureNumber": "25",
    "subscriptionNumber": "02"
  }
  "customerId": 74578
}
```

```
HTTP/1.1 201 Created
{
  "inductionDate": "2030-09-17T13:14:50.704Z",
  "productionDefinition": 451,
  "submitterEkp": "0000000003",
  "originatorEkp": "0000000002",
  "zipCodeCheck": true,
  "definition": {
    "dialogpostDisabled": false,
    "productionInfoText": "string",
    "accountingInfoText": "string",
    "dataMatrixType": 2,
    "goGreen": false,
    "trackMatch": false,
    "variantProductionType": 1,
    "createDocumentTypes": [
      22,
      15
    ]
  },
  "payment": {
    "ownerEkp": "0000000001",
    "procedureNumber": "25",
    "subscriptionNumber": "02"
  },
  "customerId": 74578
}
```

The most important fields are:

- **productionDefinition**: Pick the id of one of the standard production definitions or create your own definitions.

- **ownerEkp, procedureNumber, subscriptionNumber**: A contract with these numbers has to exist.
- **variantProductionType**: Choose in which manner variants will be optimized, single stream, multi stream or per variant.

Add regions or postal codes

For example for Dialogpost the next step is to add a list of your concrete mailing items.

Theoretically possible is a total of 6 million items. However, we limit it for now by 200.000 items. If you want to add more than that, please contact it-csp@deutschepost.de one day ahead with your requested time-frame.

Example request to add a list of mailing items:

```
POST
https://print-mailing-api.deutschepost.de/dispatchpreparation/dialogpost/processings/612/mailingitems?customerId=74578
Content-Type: application/json
Accept: application/json
Accept-Encoding: gzip
Authorization: Bearer ACCESS-TOKEN

[
  {
    "id": 1,
    "zip": "64293"
  },
  {
    "id": 2,
    "zip": "02625"
  },
  {...}
]

HTTP/1.1 201 Created
{}
```

Release your processing

Finally, you start your processing. Depending on the complexity your processing will take seconds or minutes to complete. You may poll the `processing` resource until the state changes and results are present.

```
POST
https://print-mailing-api.deutschepost.de/dispatchpreparation/di
alogpost/processings/612/release?customerId=74578
Content-Type: application/json
Accept: application/json
Accept-Encoding: gzip
Authorization: Bearer ACCESS-TOKEN

HTTP/1.1 200
{
  "id": 612,
  "description": "Postcard Mailing 2036",
  "processingState": 20,
  "mailingItemsQuantity": 2,
  "mailingItemsErrors": null
}

GET
https://print-mailing-api.deutschepost.de/dispatchpreparation/di
alogpost/processings/612?customerId=74578
Accept: application/json
Accept-Encoding: gzip
Authorization: Bearer ACCESS-TOKEN

{
  "id": 612,
  "description": "Postcard Mailing 2036",
  "processingState": 40,
  "mailingItemsQuantity": 2,
  "mailingItemsErrors": null,
  "inputState": 92,
  "internalDataProcessingState": 91,
  "documentsCreationState": 91,
  "orderTransmissionState": 0,
  "orderResponseState": 0,
  "processingOutputState": 91
}
```

The six states within the response have id values that can range from 0 to 99. These values are further elaborated within the lookup resource. Each value contains a textual description of the id value. E.g. inputState: 1 equals "Started" and inputState: 91 equals "Finished".

Get the result of your processing

Your processing should be in the FINISHED state, now. Be aware that this does not automatically imply a successful processing. If you get no result documents please have a look at the `log` resource where you will get detailed information about your processing.

A successful processing produces three types of results

- **Postage information**: It will contain detailed information about postage and rebates of each variant (aka production definition) of a processing.
- **Mailing items**: Detailed information for every mailing item in your processing.
- **Induction & production documents**: A zip file, which contains all relevant documents for induction and fulfillment

Tip

Please set a correct **Accept-Encoding** http header for all your request, if applicable! The data amount returned, especially for mailing items, can be hugh. You save time and bandwidth on setting this http header.

The **included induction documents** are (depending on your choice for **createDocumentTypes** in the processing parameters):

- Einlieferungsbeleg
- Versandplan
- Bund-/Behälterzettel 1 Stück pro Blatt
- Bund-/Behälterzettel X Stück pro Blatt (schnittoptimiert)
- Packliste
- Palettenzettel

- Palettenliste
- Variantenliste (if variants exists)

For a failed processing no postage or mailing item results will be available!

Note

If you have problems understanding the error log, please contact it-csp@deutschepost.de with a mail subject "[#dis]" and the log file.

Example requests for getting the induction documents:

```
GET
https://print-mailing-api.deutschepost.de/dispatchpreparation/dialogpost/processings/612/results?customerId=74578
Accept: application/octet-stream, application/json, */*
Authorization: Bearer ACCESS-TOKEN

HTTP/1.1 200
[binary content as deflate compressed zip archive]
```

Example request for getting result postage information:

```
GET
https://print-mailing-api.deutschepost.de/dispatchpreparation/dialogpost/processings/612/results/postage?customerId=74578
Accept: application/json
Accept-Encoding: gzip
Authorization: Bearer ACCESS-TOKEN

HTTP/1.1 200
{
  "rebateFrankingTotal": 1000,
  "rebateProductionTotal": 5000,
  "postageTotal": 5000,
  "postageAdditionsTotal": 1000,
  "postageVatTotal": 950,
  "postageTotalGross": 5950,
  "postageToPay": 0,
  "variants": [
    {
      "variantName": "PostcardDL",
      "numberExtraItems": 50,
      "numberItemsTotal": 200,
      "postageAllItems": 7500,
      "postageVat": 950,
      "postageAdditionsTotal": 1000,
      "rebateExtraItems": 1000,
      "rebateFrankingTotal": 1000,
      "numberRealItems": 150,
      "numberItemsInContainersRebatedLr": 50,
      "numberItemsInContainersRebatedPlz": 50,
      "numberItemsInPalletsRebatedLz": 50,
      "numberItemsInPalletsRebatedLr": 50,
      "rebateContainersLr": 1000,
      "rebateContainersPlz": 1000,
      "rebatePalletsLr": 1000,
      "rebatePalletsLz": 1000,
      "rebatePercentageContainersLr": 3,
      "rebatePercentageContainersPlz": 4,
      "rebatePercentagePalletsLz": 5,
      "rebatePercentagePalletsLr": 6,
      "rebatePercentageExtraItems": 5,
      "rebateProductionTotal": 5000,
      "weightAllItemsInGram": 1500,
      "postageSingleItem": 28,
      "notEnabledForAutomation": true
    }
  ]
}
```

Likewise, you can request IT franking information as well as error messages if a mailing item could not be processed. Mailingitems contain information regarding production, such as the dmc, frankingId and inductionDate. Pallet and Container information is also provided. Their Ids determine which pallet/container a mailingitem has to be placed in. If a mailingitem is the last or first one to be placed in a specific pallet/container it will be flagged accordingly in firstOfPallet, firstOfContainer, lastOfPallet or lastOfContainer.

Example request for getting resulting mailing items:

```
GET
https://print-mailing-api.deutschepost.de/dispatchpreparation/di
alogpost/processings/612/results/mailingitems?customerId=74578
Accept: application/json
Accept-Encoding: gzip
Authorization: Bearer ACCESS-TOKEN

HTTP/1.1 200
[
 {
    "id": 150,
    "originalId": 1,
    "product": "Dialogpost Postkarte",
    "postage": 28,
    "inductionMonth": "2021-12",
    "dvfLine": "0001//0000001/29//50895",
    "inductionDate": "2021-12-14",
    "containerId": 1,
    "palletId": 15,
    "firstOfContainer": false,
    "lastOfContainer": true,
    "firstOfPallet": false,
    "lastOfPallet": true,
    "dmc": "44 45 41 08 30 01 66 6A 27 85 00",
    "inscription": "DIALOGPOST PREMIUMADRESS",
    "identification": "P",
    "frankingId": 01 42C1 ECC0 00 1000 0013,
    "zip": 12345,
    "errorMessages": []
  },
  {
    "id": 151,
    "originalId": 2,
    "product": null,
    "postage": null,
    "inductionMonth": null,
    "dvfLine": null,
    "inductionDate": null,
    "containerId": bull,
    "palletId": ,
    "firstOfContainer": null,
    "lastOfContainer": null,
    "firstOfPallet": null,
    "lastOfPallet": null,
    "dmc": null,
    "inscription": null,
    "identification": null,
    "frankingId": null,
    "zip": 12345,
    "errorMessages": [
      "Die Sendung 2 ist ungültig: Die PLZ 33336 ist ungültig."
    ]
  }, [...]
]
```

It is also possible to fetch a trace log for the processing which includes entries on different severity levels.

An example request for getting aforementioned trace log:

```
GET
https://print-mailing-api.deutschepost.de/dispatchpreparation/dialogpost/processings/612/log?customerId=74578
Accept: application/json
Accept-Encoding: gzip
Authorization: Bearer ACCESS-TOKEN

HTTP/1.1 200

{
    "infos": [
        {
            "code": null,
            "message": "Example Message Info",
            "severity": "INFO",
            "fieldName": null,
            "timestamp": "2022-01-01T12:57:37.303Z"
        }, [...]
    ],
    "warnings": [
        {
            "code": null,
            "message": "Example Message Warning",
            "severity": "WARN",
            "fieldName": null,
            "timestamp": "2022-01-01T12:57:37.303Z"
        }, [...]
    ],
    "errors": [
        {
            "code": null,
            "message": "Example Message Error",
            "severity": "INFO",
            "fieldName": null,
            "timestamp": "2022-01-01T12:57:37.303Z"
        }, [...]
    ]
}
```

Further processing details

Parameters allow for customizing your processing. In this chapter, you can find further examples.

Example request and response for Dialogpost processing with options:

```
POST https://print-mailing-api.deutschepost.de/dispatchpreparation/dialogpost/processings
Content-Type: application/json
Accept: application/json
Accept-Encoding: gzip
Authorization: Bearer ACCESS-TOKEN

{
  "customerId": 74578,
  "description": "Postcard Mailing 2036 - simulated",
  "simulation": true
}

HTTP/1.1 201 Created
```

```
{
  "id": 612,
  "description": "Postcard Mailing 2036 - simulated",
  "simulation": true,
  "processingState": 10,
  "mailingItemsQuantity": 0,
  "mailingItemsErrors": null
}
```

Example request and response for Dialogpost mailing with options:

```
POST
https://print-mailing-api.deutschepost.de/dispatchpreparation/di
alogpost/processings/612/mailing
Content-Type: application/json
Accept: application/json
Accept-Encoding: gzip
Authorization: Bearer ACCESS-TOKEN

{
  "zipCodeCheck": true,
  "inductionDate": "2030-09-17T13:14:50.704Z",
  "productionDefinition": 451,
  "originatorEkp": "0000000002",
  "submitterEkp": "0000000003",
  "payment": {
    "ownerEkp": "0000000001",
    "procedureNumber": "25",
    "subscriptionNumber": "02"
  },
  "definition": {
    "dialogpostDisabled": false,
    "productionInfoText": "My info text for production.",
    "accountingInfoText": "My info text for accounting.",
    "dataMatrixType": 2,
    "goGreen": true,
    "trackMatch": true,
    "createDocumentTypes": [22,23,34,35,9,24,26,25,15]
  },
  "induction": {
    "depotName": "Darmstadt"
  },
  "customerId": 74578
}

HTTP/1.1 201 Created
{ will return the stored data you posted }
```

Example request and response for Dialogpost mailing items with options:

```
POST
https://print-mailing-api.deutschepost.de/dispatchpreparation/di
alogpost/processings/612/mailingitems?customerId=74578
Content-Type: application/json
Accept: application/json
Accept-Encoding: gzip
Authorization: Bearer ACCESS-TOKEN

[
  {
    "id": 1,
    "zip": "64293",
    "productionDefinition": 451,
    "country": "DE",
    "infoTextAlphanumeric": "My info text alphanumeric",
    "infoTextHexadecimal": "My info text hexadecimal",
    "premiumAddressVariant": 1,
    "premiumAddressId": 1
  },
  {
    "id": 2,
    "zip": "64293",
    "productionDefinition": 452,
    "country": "DE",
    "infoTextAlphanumeric": "My info text alphanumeric",
    "infoTextHexadecimal": "My info text hexadecimal",
    "premiumAddressVariant": 2,
    "premiumAddressId": 2
  },
  {...}
]
```

```
HTTP/1.1 201 Created
{}
```

Setting Options

You can set quite a lot of options to control the optimization. For example:

- **simulation**: This parameter controls whether the optimization results in an official order in AM (Deutsche Post Auftragsmanagement). See `order records` resource.
- **dialogpostDisabled**: If you do not want to submit the mailing using Dialogpost (e.g. you want to speed up shipping) set this parameter to true.
- **dataMatrixType**: Set this to 0 if you do not want Data Matrix Codes to be created.
- **zipCodeCheck**: If set to true, German postal codes will be checked.
- **productionInfoText**: You can identify production documents with the value of this parameter.
- **accountingInfoText**: You can identify the accounting document with the value of this parameter.
- **createDocumentTypes**: Control, which resulting documents should be created. See lookup resource `DpDocumentType` for available values.

Variants

If you want to produce **production variants** you should define one production definition per variant and set this production definitions as a parameter in every mailing item. The production definition defined in the mailing will be the default production definition for all mailing items without a concrete definition.

An example you can find here example for variants

Overriding contract settings

While adding information about your mailing to your processing you have to set the reference to a specific contract by setting the values for "ownerEkp", "procedureNumber" and "subscriptionNumber". Doing this the information stored in the contract apply for the processing as well. However you can overwrite some of the values when setting the information about your mailing. That means the values will be different from your contract and will only apply for this specific processing.

These values are:

- depotName
- submitterEkp
- originatorEkp

An example you can find here example for overriding

Additional services

Premiumadress

With PREMIUMADRESS your mail items are checked by our mail carriers. Incorrect or wrong addresses are documented, compared against the latest databases, updated, and provided to you for download. You can choose from seven different product variants and always use PREMIUMADRESS exactly in the way you need it. For more information about PREMIUMADRESS please refer to PREMIUMADRESS web page

To use via the API you have to add information about your premiumadress ID (that is your contract ID) and the product variant you choose when adding you mailing items to your processing.

Here is an example how to do so: example for premiumadress

Track&Match

With Track&Match we will indicate how many mailing items are currently in the sorting and therefore will be very likely be dispatched the very next day. For more information about Track&Match please refer to Track&Match web page

Here is an example how to use Track&Match: example for Track&Match

GoGreen

With GoGreen, Deutsche Post offers a climate-neutral shipping option in Germany. The CO2 emissions generated during transport are offset by investing in internationally recognized climate protection projects. For more information about GoGreen please refer to GoGreen web page. Set the property goGreen to true in order to use this option.

This is an example how to activate GoGreen example for GoGreen

Please note that you need to have a GoGreen contract to use GoGreen. The flag is currently not supported - if you have a GoGreen contract all your mailing items will pay on your GoGreen contract.

PROCESSING API - POSTWURFSPEZIAL

Master data

Master data is used within your processings. You may add, update or delete you master data as needed. The next chapters explain each master data resource.

Default payment information

Please find the description in this section.

Default induction information

Please find the description in this section.

Working with processings

You need a **valid access token** including the access right ROLE_DIS_MANAGE_POSTWURFSPEZIAL to continue. See section this section to learn more about access rights.

1. Define your target group using the **Targeting API**
2. Create a processing using your target group selection id
3. Add information about the mailing

4. Release your processing
5. Wait until all concrete mailing items are prepared for your target group selection. **This will take up to one day!**
6. Get the result of your processing including all mailing items for your selection

Create a processing

Example request and response for processing creation:

```
POST
https://print-mailing-api.deutschepost.de/dispatchpreparation/po
stwurfspezial/processings
Content-Type: application/json
Accept: application/json
Accept-Encoding: gzip
Authorization: Bearer ACCESS-TOKEN

{
  "customerId": 74578,
  "description": "My POSTWURFSPEZIAL Postcard Mailing 2036",
  "simulation": true,
  "targetGroupSelectionId":
"221a304d-eabc-4c25-a2cb-c6fafceed64"
}

HTTP/1.1 201 Created
{
  "id": 212,
  "description": "My POSTWURFSPEZIAL Postcard Mailing 2036",
  "simulation": true,
  "targetGroupSelectionId":
"221a304d-eabc-4c25-a2cb-c6fafceed64",
  "processingState": 10,
  "processingType": 20,
  "mailingItemsQuantity": null,
  "mailingItemsErrors": null,
  "createdOn": "2022-01-17T12:45:34.000Z",
  "changedOn": null
}
```

Note

Use the returned `id` from the response for subsequent requests regarding this processing.

Add information about the mailing

The next step is to add some mailing information to your previously created processing.

Example request and response for POSTWURFSPEZIAL:

```
POST
https://print-mailing-api.deutschepost.de/dispatchpreparation/postwurfspezial/processings/212/mailing
Content-Type: application/json
Accept: application/json
Accept-Encoding: gzip
Authorization: Bearer ACCESS-TOKEN

{
  "customerId": 74578
  "inductionDate": "2023-09-17T13:14:50.704Z",
  "productionDefinition": 451,
  "originatorEkp": "0000000002",
  "submitterEkp": "0000000003",
  "definition": {
    "greeting": "To all households",
    "targetAudience": 0,
    "goGreen": true,
    "trackMatch": false,
    "accountingInfoText": "My info text for accounting.",
    "createDocumentTypes": [1,2,3,5]
  },
  "payment": {
    "payerEkp": "0000000001",
    "entitledToDeductPreTax": true,
    "paymentMethod": 3,
    "cardNo": "2502001",
    "iban": null,
    "bic": null,
    "bankName": null,
  },
  "induction": null,
  "contractualPartner": null
}
```

```
HTTP/1.1 200 OK
{ will return the stored data you posted }
```

The most important fields are:

- **payment**: Specific payment information is set for this processing. **NOTICE**: For POSTWURFSPEZIAL processings, you can only pay with Deutsche Post Postcard.
- **induction**: Default induction information is used here (which were set beforehand).
- **greeting**: You may add a greeting which is returned for every mailing item. It most probably will be the first address line.
- **trackMatch**: You can choose track and match feature for your mailings. Find the data matrix codes in the resulting mailing items.

Release your processing

Finally, you start your processing. A **basic validation** will check for obvious wrong mailing parameters and a not existing TGP selection id.

Note

If any validation error occurs, releasing the processing will be aborted and error codes will be reported synchronously.

Your processing should be in the state GATHERING_MAILINGITEMS ('22'), now. The concrete mailing items for a selection will be available **the next day at around 4am CET**. You may poll the `processing` resource until the state changes and results are present.

```
POST
https://print-mailing-api.deutschepost.de/dispatchpreparation/postwurfspezial/processings/212/release?customerId=74578
Content-Type: application/json
```

```
Accept: application/json
Accept-Encoding: gzip
Authorization: Bearer ACCESS-TOKEN

HTTP/1.1 200
  "id": 212,
  "description": "My POSTWURFSPEZIAL Postcard Mailing 2036",
  "simulation": true,
  "targetGroupSelectionId":
"221a304d-eabc-4c25-a2cb-c6fafceed64",
  "processingState": 22,
  "processingType": 20,
  "mailingItemsQuantity": "3658",
  "mailingItemsErrors": null,
  "createdOn": "2022-02-17T12:45:34.000Z",
  "changedOn": null
```

- **mailingItemsQuantity**: This is the calculated, expected amount of mailing items for your selection.

Get the result of your processing

Your processing should be in the FINISHED state, now. Be aware that this does not automatically imply a successful processing. If you get no result documents please have a look at the `log` resource where you will get detailed information about your processing.

```
GET
{dmsuite-requestexamples-url}/dispatchpreparation/postwurfspezia
l/processings/612/results?customerId={dmsuite-dispatchpreparatio
n-requestexamples-customerid}
Accept: application/octet-stream, application/json, */*
Accept-Encoding: gzip
Authorization: Bearer ACCESS-TOKEN

HTTP/1.1 200
[binary content as deflate compressed zip archive]
```

Example request for getting result postage information:

```
GET
https://print-mailing-api.deutschepost.de/dispatchpreparation/po
stwurfspezial/processings/612/results/postage?customerId=74578
Accept: application/json
Accept-Encoding: gzip
Authorization: Bearer ACCESS-TOKEN

HTTP/1.1 200
{
    "numberExtraItems": 50,
    "numberItemsTotal": 200,
    "postageAllItems": 7500,
    "postageAdditionsTotal": 1000,
    "numberRealItems": 150,
    "postageTotal": 5000,
    "postageVat": 950,
    "postageSingleItem": 28,
    "postageToPay": 107100,
    "postageExtraServicesTotal": 0,
}
```

Example request for getting resulting mailing items:

```
GET
https://print-mailing-api.deutschepost.de/dispatchpreparation/po
stwurfspezial/processings/612/results/mailingitems?customerId=74
578
Accept: application/json
Accept-Encoding: gzip
Authorization: Bearer ACCESS-TOKEN

HTTP/1.1 200
[ {
  "originalId" : 1,
  "id" : 1,
  "containerId" : 1,
  "palletId" : 0,
  "firstOfContainer" : true,
  "lastOfContainer" : null,
  "firstOfPallet" : true,
  "lastOfPallet" : null,
  "zip" : 24340,
  "greeting" : "Für Sie ganz persönlich",
  "street" : "Aschauer Landstr.",
  "houseNumber" : "2",
  "city" : "Altenhof",
  "dmc" : null
}, [...]
]
```

- **originalId**: This is the original id which was chosen by us. In future releases you will be able to upload your own mailing items. In this case, this will be chosen by you.
- **palletId, firstOfContainer, ..**: Use these production parameters to put your mailing item into the correct bucket during production.
- **dmc**: The data matrix code. Only provided, if you choose the track and match option.

Further processing details

Parameters allow for customizing your processing. In this chapter, you can find further examples.

- POSTCARD is the only possible payment method.

Variants

Variants are not supported for POSTWURFSPEZIAL.

Postwurfspezial production definitions

Please notice the special requirements for production definitions:

- The ProductionDefinitionOptimizationType of a production definition is always FABRICATION_EFFORT.

- Bundle production can be specified only for variants of 'GROSS' format.

Example request for a valid production definition:

```
POST
https://print-mailing-api.deutschepost.de/dispatchpreparation/productiondefinitions
Content-Type: application/json
Accept: application/json
Accept-Encoding: gzip
Authorization: Bearer ACCESS-TOKEN

{
  "name": "Postwurfspezial production definition",
  "customerId": 3453,
  "mailingItemType": 2,
  "lengthInDeciMm": 2200,
  "widthInDeciMm": 1050,
  "heightInDeciMm": 10,
  "weightInGram": 7,
  "notEnabledForAutomation": false,
  "productionDefinitionFabricationType": 1,
  "trays": {
    "trayType": 1,
    "maxNumberItemsPerContainer": 450,
    "numberItemsForTrayHalfFull": 228,
    "productionDefinitionOptimizationType": 2,
    "productionDefinitionNumberItemsDependentOptimizationType": null,
    "minNumberItemsForContainerLr": null,
    "minNumberItemsForContainerPlz": null
  },
  "bundles": null,
  "pallets": {
    "usePallets": true,
    "maxNumberMailingItemsPerPallet": 5000,
    "alwaysProducePallets": true,
    "traysStackable": false,
    "emptyWeightInGram": 22000,
    "maxNumberTraysPerPallet": 96,
    "maxNumberTraysPerLayer": 8,
    "productionDefinitionOptimizationType": 2,
    "minSavingToProduceExtraPallet": 0
  }
}
```

Additional services

Track&Match

Track&Match is supported for POSTWURFSPEZIAL. Enable it at the processing mailing configuration and find the resulting data matrix codes in the mailing items.

GoGreen

With GoGreen, Deutsche Post offers a climate-neutral shipping option in Germany. The CO2 emissions generated during transport are offset by investing in internationally recognized climate protection projects. For more information about GoGreen please refer to GoGreen web page. Set the property goGreen to true in order to use this option.

Please note that you need to have a GoGreen contract to use GoGreen. If you have a GoGreen contract all your mailing items will pay on your GoGreen contract.

Contractual Partner

In case you act as a reseller for an advertising customer, you can add the name and address of your contractual partner to the mailing. The data will be transferred to order management during order creation.

PROCESSING API - POSTAKTUELL

Master data

Master data is used within your processings. You may add, update or delete you master data as needed. The next chapters explain each master data resource.

Default payment information

Please find the description in this section.

Default induction information

Please find the description in this section.

Working with processings

You need a **valid access token** including the access right ROLE_DIS_MANAGE_POSTAKTUELL to continue. See section this section to learn more about access rights.

1. Define your target group using the **Targeting API**
2. Create a processing using your target group selection id
3. Add information about the mailing
4. Release your processing
5. Get the result of your processing including all mailing items for your selection

Create a processing

Example request and response for processing creation:

```
POST
https://print-mailing-api.deutschepost.de/dispatchpreparation/postaktuell/processings
Content-Type: application/json
Accept: application/json
Accept-Encoding: gzip
Authorization: Bearer ACCESS-TOKEN

{
  "customerId": 74578,
  "description": "My POSTAKTUELL Postcard Mailing 2036",
  "simulation": true,
  "targetGroupSelectionId":
"221a304d-eabc-4c25-a2cb-c6fafceed64"
}
```

```
HTTP/1.1 201 Created
{
  "id": 212,
  "description": "My POSTAKTUELL Postcard Mailing 2036",
  "simulation": true,
  "targetGroupSelectionId":
"221a304d-eabc-4c25-a2cb-c6fafceed64",
  "processingState": 10,
  "processingType": 20,
  "mailingItemsQuantity": null,
  "mailingItemsErrors": null,
  "createdOn": "2022-01-17T12:45:34.000Z",
  "changedOn": null
}
```

Note

Use the returned `id` from the response for subsequent requests regarding this processing.

Further information regarding creation:

- **description** cannot be empty. Please make sure to include a valid title for your mailing.

Add information about the mailing

The next step is to add some mailing information to your previously created processing.

Example request and response for Postaktuell:

```
POST
https://print-mailing-api.deutschepost.de/dispatchpreparation/po
staktuell/processings/212/mailing
Content-Type: application/json
Accept: application/json
Accept-Encoding: gzip
```

```
Authorization: Bearer ACCESS-TOKEN
```

```
{
  "customerId": 74578
  "inductionDate": "2023-09-17T13:14:50.704Z",
  "productionDefinition": 451,
  "originatorEkp": "0000000002",
  "submitterEkp": "0000000003",
  "definition": {
    "productVariant": 1,
    "selectionType": 1,
    "toleranceForLastBundle": 3,
    "roundDownToFull": 2,
    "accountingInfoText": "My info text for accounting.",
    "createDocumentTypes": [1,2,3,5]
  },
  "payment": {
    "payerEkp": "0000000001",
    "entitledToDeductPreTax": true,
    "paymentMethod": 3,
    "cardNo": "2502001",
    "iban": null,
    "bic": null,
    "bankName": null,
  },
  "induction": null,
  "contractualPartner": null
}
```

```
HTTP/1.1 200 OK
{ will return the stored data you posted }
```

The most important fields are:

- **payment**: Specific payment information is set for this processing.
- **induction**: Default induction information is used here (which were set beforehand). It most probably will be the first address line.

Release your processing

Finally, you start your processing. A **basic validation** will check for obvious wrong mailing parameters and a not existing TGP selection id.

Note

If any validation error occurs, releasing the processing will be aborted and error codes will be reported synchronously.

Depending on the complexity your processing will take seconds or minutes to complete. You may poll the `processing` resource until the state changes and results are present.

```
POST
https://print-mailing-api.deutschepost.de/dispatchpreparation/po
staktuell/processings/212/release?customerId=74578
Content-Type: application/json
Accept: application/json
Accept-Encoding: gzip
Authorization: Bearer ACCESS-TOKEN

HTTP/1.1 200
  "id": 212,
  "description": "My Postaktuell Test",
  "simulation": true,
  "targetGroupSelectionId":
"221a304d-eabc-4c25-a2cb-c6fafceed64",
  "processingState": 22,
  "processingType": 30,
  "mailingItemsQuantity": null",
  "mailingItemsErrors": null,
  "createdOn": "2022-04-19T10:11:22.000Z",
  "changedOn": null
```

Get the result of your processing

Your processing should be in the FINISHED state, now. Be aware that this does not automatically imply a successful processing. If you get no result documents please have a look at the `log` resource where you will get detailed information about your processing.

```
GET
{dmsuite-requestexamples-url}/dispatchpreparation/postaktuell/pr
ocessings/612/results?customerId={dmsuite-dispatchpreparation-re
questexamples-customerid}
Accept: application/octet-stream, application/json, */*
Accept-Encoding: gzip
Authorization: Bearer ACCESS-TOKEN

HTTP/1.1 200
[binary content as deflate compressed zip archive]
```

Example request for getting result postage information:

```
GET
https://print-mailing-api.deutschepost.de/dispatchpreparation/po
staktuell/processings/612/results/postage?customerId=74578
Accept: application/json
Accept-Encoding: gzip
Authorization: Bearer ACCESS-TOKEN

HTTP/1.1 200
{
    "numberPallets": 10,
    "postagesTariffA": {
      "numberItems": 1000,
      "costsPer1000": 2000,
      "costsTotal": 2000,
    },
    "postagesTariffB": {
      "numberItems": 1000,
      "costsPer1000": 2000,
      "costsTotal": 2000,
    },
    "postagesTariffsSum": {
      "numberItems": 2000,
      "costsPer1000": 2000,
      "costsTotal": 4000,
    },
    "postagePickUp": 7500,
    "additionalChargeForMinimumOrder": 1500,
    "additionalChargeZustellbezirksSelektion": 1500,
    "totalNet": 14500,
    "totalVat": 2755,
    "totalGross": 17255,
}
```

Note

For Postaktuell there are no mailingitems provided. Because every mailingitem has the same dimensions and does not contain address information, providing production information for postaktuell is redundant.

Additional Services

Contractual Partner

In case you act as a reseller for an advertising customer, you can add the name and address of your contractual partner to the mailing. The data will be transferred to order management during order creation.

FAQ

How do I receive my access data for the Print-Mailing API for Dispatch Preparation?

See Requesting access data for dispatch preparation.

Where do I get the technical specification?

You will find the technical specification for the Print-Mailing API for Dispatch Preparation in English at our developer portal.

Whom do I contact if I have any further technical questions?

You can reach us at it-csp@deutschepost.de.

Are addresses also transmitted to the Print-Mailing API when optimizing Dialogpost?

No, a continuous ID is allocated to the addresses, and only the IDs with the associated postal codes are transmitted to the service. The IDs and postal codes are restored following optimization. The addresses can then be attributed again by the ID.

Is it possible to transfer contracts for IT franking from other applications?

It is not possible to transfer existing IT contracts. As is the case for other applications, a new IT contract must be additionally created and/or agreed for the Print-Mailing API for Dispatch Preparation.

Can address data be transmitted in different formats?

No. The transmission of data through the Print-Mailing API for Dispatch Preparation must take place via an API request in json format.

Can order changes or cancellations be directly updated through an API entry in the OM (order management) system?

Yes. Through the Print-Mailing API for Dispatch Preparation, orders can be automatically created, changed or canceled.

What personal data are transferred to the print mailing services for target group planning and dispatch preparation (API and online service)?

For the posting of your mailings at one of our acceptance points, we require information on the sender, poster and, if applicable, deviating payer. Transferred are not just company data, but also the personal data of contact persons from a company. Specifically, this refers to the first and last names, e-mail address, phone number, fax number and the company name.

Are the complete recipient addresses transferred for a print mailing?

Recipient addresses and thus personal data are not transferred for the dispatch preparation of addressed mailings (Dialogpost); however, all recipient postal codes are transferred in order to create an optimized delivery sequence.

For what purpose and on what legal basis are the data processed?

The data are used solely for the fulfillment of the contract concluded with you. The legal basis is therefore Art. 6(1), letter b, of the General Data Protection Regulation (GDPR).

How are the data transmitted to our servers?

All communication with the print mailing services takes place with SSL encryption.

How long are the data retained?

Information pertaining to addresses and metadata (description, most recent processing status, number of shipments) is deleted 30 days after the last amendment of your data. System user data (e.g. first and last name, e-mail address, phone number) required to generate access data for the respective print mailing service are retained for the duration of the contractual relationship. These data are only deleted if the system user requests the deletion of their user account prior to the expiration of the contract.

Are data passed on to third parties?

No. No data are passed on to third parties. No data are transferred to third countries (countries outside the EEA).

What rights apply with regard to data processing?

Those impacted generally have the right to notification and to correction, deletion, restriction of processing, objection and data portability.

Who is responsible for data processing?

Deutsche Post AG, Charles-de-Gaulle Str. 20, 53113 Bonn, Germany

Whom can I contact if I have any questions about data protection?

a) Data Protection Officer of Deutsche Post AG: Gabriela Krader, 53250 Bonn, Germany datenschutz@dpdhl.com

b) Competent supervisory authority: The Federal Commissioner for Data Protection and Freedom of Information, Postfach 1468, 53004 Bonn, Germany

Support

Our extensive documentation and developer starter kits (Postman collections), as well as our connection and support team, are available to optimally assist you while you develop the API.

At the beginning of the process, we individually advise you on technical and process-related issues associated with the integration of our API into your business processes. We then support your developers throughout the entire development process and are available to answer questions pertaining to the required data sets. The final step is the acceptance of your development, whereby we verify the technically correct communication of your software with our API. After successful acceptance for the respective product (Postaktuell, Postwurfspezial, Dialogpost), you receive access to our live system.

Should you require assistance in connecting the Print-Mailing API for Dispatch Preparation, please contact our Support Team directly at the e-mail address it-csp@deutschepost.de. Enter "[#dis] your company name" in the e-mail subject header.

You will promptly receive a confirmation with a ticket number. Reply directly to that e-mail should you have questions or additional information. Do not change the subject header containing the ticket number so that we can correctly match further messages with your initial query.

Please use this e-mail address exclusively so that we can process your query as quickly as possible. Our Support Team will assist you with your questions and refer you to additional experts if needed. Please understand that we cannot answer any development-related questions (e.g. code generation in dedicated programming languages).

Approval Terms

For access to the live system of the Print-Mailing API for Dispatch Preparation, the following acceptance criteria are defined for each product (Postaktuell, Postwurfspezial, Dialogpost). It is essential that you contact our staff at it-csp@deutschepost.de prior to carrying out the tasks. Enter "[#dis] your company name / acceptance of access to live system" in the e-mail subject header. Our staff will contact you directly and coordinate the rest of the acceptance process with you.

Postaktuell

The tasks for the Deutsche Post product Postaktuell are described below. The results serve as an acceptance criterion for access to the live system for use of the Print-Mailing API.

The required access data will be sent to you by our IT Support staff once acceptance has been successful.

All orders must be created without transmission to the electric order management (OM) system. When using the OM system, contact IT Support to request a specific EKP combination.

Task

- Shipment dimensions:
- Length: 2,290 decimillimeters
- Width: 1,140 decimillimeters
- Thickness: 10 decimillimeters
- Weight: 10 grams
- Create contract with Postcard (procedureNumber=25) and posting office
- The predefined values must be used (see below).
- Create selection with the targeting API: the postal codes for Düsseldorf "40593," "40597," "40595," "40589," "40591," "40225," "40599," "40229"
- Create processing with the name "Abnahme Aufgabe Postaktuell"
- Start processing

Parameters/values to be defined

For the production sample:

- name: "Abnahme Postaktuell Kundenname"
- mailingItemType: 1
- lengthInDeciMm: see task
- widthInDeciMm: see task
- heightInDeciMm: see task
- weightInGram: see task
- productionDefinitionFabricationType: 2=Bund

bundles

- "maxNumberItemsPerContainer": 500,
- "productionDefinitionOptimizationType": 1

- "productionDefinitionNumberItemsDependentOptimizationType": not relevant for Postaktuell
- "minNumberItemsForContainerLr": not relevant for Postaktuell
- "minNumberItemsForContainerPlz": not relevant for Postaktuell

pallets

- usePallets: true
- maxNumberMailingItemsPerPallet: 60,000,
- alwaysProducePallets: true
- traysStackable: not relevant for Postaktuell
- emptyWeightInGram of pallet: 22,000
- maxNumberTraysPerPallet: not relevant for Postaktuell
- maxNumberTraysPerLayer: not relevant for Postaktuell
- productionDefinitionOptimizationType: not relevant for Postaktuell
- minSavingToProduceExtraPallet: not relevant for Postaktuell

For processing:

- "productVariant": 3
- "selectionType": 2
- "toleranceForLastBundle": 0
- "roundDownToFull": 100

```
emptyWeightInGram: 22,000
```

-
- "minNumberMailingItemsPerPallet": 10,000
- "maxGrossWeightPerPallet": 700,000
- "produceBundlesInPalletOrder": true
- "accountingInfoText": "My info text for accounting."
- "createDocumentTypes": 1,2,3,4

Postwurfspezial

The tasks for the Deutsche Post product Postwurfspezial are described below. The results serve as an acceptance criterion for access to the live system for use of the Print-Mailing API.

The required access data will be sent to you by our IT Support staff once acceptance has been successful.

All orders must be created without transmission to the order management (OM) system. When using the OM system, contact IT Support to request a specific EKP combination.

Task

- Shipment dimensions:
- Length: 2,200 decimillimeters
- Width: 1,100 decimillimeters
- Thickness: 20 decimillimeters
- Weight: 15 grams
- Create contract with Postcard (procedureNumber=25) and posting office
- The predefined values must be used (see below).
- Create selection with the targeting API: the postal codes for Düsseldorf "40593," "40597," "40595," "40589," "40591," "40225," "40599," "40229" with purchasing power = 5 (on top of average)
- Create processing with the name "Abnahme Aufgabe Postwurfspezial"
- Start processing

Parameters/values to be defined

For the production sample:

- name: "Abnahme Postwurfspezial Kundenname"
- mailingItemType: 1 (1=Briefsendung, 2=Postkarte, 3=Büchersendung, 4=Warensendung, 5=Infocard, 6=Katalog, 8=Sachet, 10=Warenprobe)
- lengthInDeciMm: see task
- widthInDeciMm: see task
- heightInDeciMm: see task
- notEnabledForAutomation: false (true=nicht automationsfähig, false=automationsfähig)
- dialogpostKoop: false (if true the mail qualtity will be increased to 100.000)

- unwrapped: false (true=ohne Umhüllung, false=mit Umhüllung)
- weightInGram: see task
- productionDefinitionFabricationType: 1 (1=Behälter, 2=Bund)

trays

- trayType: 1 (1=Größe 1, 2= Größe 2, 3= Größe1 quick trays, 4= Größe2 quick trays)
- maxNumberItemsPerContainer: 200 (maximum number of items per tray)
- numberItemsForTrayHalfFull: 100 (number of items "halb voll" / discount-eligible weight)
- productionDefinitionOptimizationType: 2 (1=Entgelt minimieren, 2= Fertigungsaufwand minimieren, 3= Aufwandsart abhängig von Sendungsmenge, bei Paletten: nach Fertigungsrabatten fertigen)
- productionDefinitionNumberItemsDependentOptimizationType: 1 (1= LR-Behälter Mindestfüllmenge wie LR-Behälter (minNumberItemsForContainerLr), 2= Keine PLZ-Behälter fertigen, 3= PLZ-Behälter abhängig von Sendungsmenge fertigen (minNumberItemsForContainerPlz))
- minNumberItemsForContainerLr: 0
- minNumberItemsForContainerPlz: 0

pallets

- usePallets: true
- alwaysProducePallets: true
- traysStackable: false
- emptyWeightInGram of pallet: 22,000
- maxNumberTraysPerPallet: 84
- maxNumberTraysPerLayer: 7
- productionDefinitionOptimizationType: 2
- minSavingToProduceExtraPallet: 0

For processing:

- entitledToDeductPreTax: true
- gogreen: false

- trackmatch: false
- accountingInfoText: "Acceptance of Postwurfspezial"
- greeting: "To the residents"
- targetAudience: 0
- createDocumentTypes: 1,2,3,4,5

Dialogpost

The tasks for the Deutsche Post product Dialogpost are described below. The results serve as an acceptance criterion for access to the live system for use of the Print-Mailing API.

The required access data will be sent to you by our IT Support staff once acceptance has been successful.

The tasks must be created in simulation mode.

Task 1 (simple mailing without variants)

Shipment dimensions:

- Length: 2,200 decimillimeters
- Width: 1,100 decimillimeters
- Thickness: 25 decimillimeters
- Weight: 18 grams

Steps:

- Create contract with Postcard (procedureNumber=25) or IT franking (procedureNumber=10) and create posting office
- The predefined values must be used (see below).
- Create processing with the name "Abnahme Aufgabe1 Kundenname"
- Add 7,754 addresses with the data set provided
- Start processing

Task 2 (mailing with 3 variants)

Shipment dimensions:

- Length: 2,200 decimillimeters
- Width: 1,100 decimillimeters
- Thickness: 25 decimillimeters
- Variant A weight: 20 grams
- Variant B weight: 18 grams
- Variant C weight: 16 grams

Steps:

- Create contract with (procedureNumber=25) or IT franking (procedureNumber=10) and create pickup
- The predefined values must be used (see below). Three variants each with different weights must be taken into account. A separate production sample must be created for each variant, each with a different shipment weight.
- Create processing with the name "Abnahme Aufgabe2 Kundenname"
- Add 19,678 addresses with the data set provided
- Start processing

Parameters/values to be defined

For the contracts:

- entitledToDeductPreTax: true

For the production sample:

- name: "Abnahme Aufgabe 1/2 Kundenname"
- mailingItemType: 1 (1=Briefsendung, 2=Postkarte, 3=Büchersendung, 4=Warensendung, 5=Infocard, 6=Katalog, 8=Sachet, 10=Warenprobe)
- lengthInDeciMm: see task
- widthInDeciMm: see task
- heightInDeciMm: see task
- notEnabledForAutomation: false (true=nicht automationsfähig, false=automationsfähig)

- dialogpostKoop: false (true= es wird automatisch auf 100.000 Sendungen aufgezahlt, sollte die Sendungsmenge nicht ausreichen, false=es findet keine Aufzahlung statt)
- unwrapped: false (true=ohne Umhüllung, false=mit Umhüllung)
- weightInGram: see task
- productionDefinitionFabricationType: 1 (1=Behälter, 2=Bund)

trays

- trayType: 1 (1=Größe 1, 2= Größe 2, 3= Größe1 quick trays, 4= Größe2 quick trays)
- maxNumberItemsPerContainer: 160 (maximum number of items per tray)
- numberItemsForTrayHalfFull: 80 (number of items "halb voll" / discount-eligible weight)
- productionDefinitionOptimizationType: 1 (1=Entgelt minimieren, 2= Fertigungsaufwand minimieren, 3= Aufwandsart abhängig von Sendungsmenge, bei Paletten: nach Fertigungsrabatten fertigen)
- productionDefinitionNumberItemsDependentOptimizationType: 1 (1= LR-Behälter Mindestfüllmenge wie LR-Behälter (minNumberItemsForContainerLr), 2= Keine PLZ-Behälter fertigen, 3= PLZ-Behälter abhängig von Sendungsmenge fertigen (minNumberItemsForContainerPlz))
- minNumberItemsForContainerLr: 0
- minNumberItemsForContainerPlz: 0

pallets

- usePallets: true
- alwaysProducePallets: false
- traysStackable: false
- emptyWeightInGram of pallet: 22,000
- maxNumberTraysPerPallet: 84
- maxNumberTraysPerLayer: 7
- productionDefinitionOptimizationType: 1
- minSavingToProduceExtraPallet: 0

For processing:

- zipCodeCheck: true
- dialogpostDisabled: false
- dataMatrixType: 2
- gogreen: false
- trackmatch: false
- variantProductionType: 1
- createDocumentTypes: 22, 23, 34, 35, 9, 24, 26, 25, 15

Legal Terms

Legal Terms for the use of and/or access to the Print Mailing Dispatch Preparation API (Print Mailing Dispatch API)

These Legal Terms do not replace and/or modify the dispatch conditions for the products Dialogpost, Dialogpost schwer, Postwurfspezial and Postaktuell pursuant to the respectively valid service brochures and associated general terms and conditions (GT&C) and the Terms of Use for the Order Management System of Deutsche Post AG, as well as of Deutsche Post Direkt or any other mail and/or shipment services agreement, which govern Your Deutsche Post or Deutsche Post Direkt shipments and mailings.

Only PDF documents created directly from the Print Mailing Dispatch API may be used for posting. You are not permitted to program your own posting lists.

We offer no warranty with respect to the contents and results that can be obtained from the use of the API (hereinafter also referred to as “Service”), or the accuracy and reliability of the information received within the Service, and we do not guarantee that the quality of the information received in connection with the Service will match your or your customers’ expectations. In particular, the information in the data may differ from the shipment quantities, weights and formats actually posted and therefore simply reflects a status before the shipment was posted.

In the event that you would like to offer the Services to your customers as a sales agent on behalf of and by order of Deutsche Post AG or its affiliated companies, you shall ensure that

- you inform your customers about the nature and scope of the services performed by Deutsche Post AG or its affiliated companies adequately and in compliance with the law. The relevant service descriptions are available at https://www.deutschepost.de/de/p/print-mailing/downloads.html.

- your customers are notified of and accept the contractual terms (general terms and conditions) for the respective services provided by Deutsche Post AG or its affiliated companies adequately and in compliance with the law. The relevant contractual terms are available at https://www.deutschepost.de/de/a/agb.html.
- customer payments take place in compliance with the relevant legal and regulatory requirements (e.g. with the inclusion of your customers' PostCard, involvement of a payment service provider).

You may use the data collected via the Print Mailing Dispatch API for the purposes specified in these Legal Terms, the General Developer Portal Terms of Use, the API descriptions and only as permissible according to the relevant laws.

Additional Information

GSA data

Postage optimization and dispatch preparation for Dialogpost

If you want to program the postage optimization and dispatch preparation for Dialogpost yourself rather than using the Dialogpost Manager or our API, you will require the file "GSA-Sequenzen BZ bundesweit*.csv". With the help of this file attached, you are able to integrate the GSA sorting into your postage optimization program.

Postman Collection

In addition to the documentation and the comprehensive open API definition, we offer you three Postman collections with examples of the most important API resources/methods. There is a collection with sample requests for each product (Dialogpost, Postwurfspezial, Postaktuell) and a single environment file. The collections are suited for quickly testing the API and its main functions.

Before you import the collections in Postman and try out the API requests, please send our Support Team an API credentials request. API credentials (access data) are required to test the API. In our Postman collection user guide, you will learn how to install Postman, import the collections and begin testing the API.

The Postman collections are available here in the Download Area.

Reference Docs

Download API Spec

the developer - Website

Send email to the developer

Servers

https://print-mailing-api.deutschepost.de/dispatchpreparation - Production environmenthttps://uat.print-mailing-api-test.deutschepost.de/dispatchpreparation - Test environment

Addresses

Create, Update Or Get Addresses Use In Processings. You Need An Access Token With ROLE_DIS_ACCESS To Access This Resource.

Endpoints

PUT

/addresses/{id}

Update an existing address.

DELETE

/addresses/{id}

Delete address.

GET

/addresses

Find addresses which can be referenced by ekp in processing.

POST

/addresses

Add a new address. EKP must be different from other addresses for this customerId.

Postaktuell Processing Results

Collect Results And Access Information Logs.You Need An Access Token With ROLE_DIS_ACCESS And ROLE_DIS_MANAGE_POSTAKTUELL To Access This Resource.

Endpoints

`GET`

`/postaktuell/processings/{id}/results`

Get processing result documents.

`GET`

`/postaktuell/processings/{id}/results/postage`

Get postage result information.

`GET`

`/postaktuell/processings/{id}/log`

Retrieve a trace log with info and error messages.

Lookups

Lists Static Request Oder Response Values Of Other API Resources

Endpoints

`GET`

`/dispatchpreparationlookups`

Retrieve all lookups for this service.

GET

/dispatchpreparationerrorcodelookups

Retrieve all error code lookups for this service.

GET

/depotlookups

Retrieve all Deutsche Post depot names.

DIALOGPOST Processing Results

Collect Results And Access Information Logs.You Need An Access Token With ROLE_DIS_ACCESS And ROLE_DIS_MANAGE_DIALOGPOST To Access This Resource.

Endpoints

GET

/dialogpost/processings/{id}/results

Get processing result documents.

GET

/dialogpost/processings/{id}/results/postage

Get postage result information.

GET

/dialogpost/processings/{id}/results/mailingitems

Get resulting mailing items.

GET

/dialogpost/processings/{id}/log

Retrieve a trace log with info and error messages.

Cost Calculation

Cost Calculation For DIALOGPOST, POSTWURFSPEZIAL And POSTAKTUELL.

Endpoints

`POST`

`/postwurfspezial/simplecostcalculation`

Calculate costs without a list of mailing items.

`POST`

`/postaktuell/simplecostcalculation`

Create a cost calculation.

`POST`

`/dialogpost/standardcostcalculation`

Calculate costs with a list of mailing items.

`POST`

`/dialogpost/simplecostcalculation`

Calculate costs without a list of mailing items.

Order Records (Deprecated)

Manage All Order Records Which Were Sent To The Deutsche Post Auftragsmanagement.You Need An Access Token With ROLE_DIS_ACCESS To Access This Resource.

Endpoints

PUT

~~/orderrecords/{id}~~

Creates a new order record entry of type 'Change' and update the order details in order management (AM).

DELETE

~~/orderrecords/{id}~~

Cancel an order record within Deutsche Post Auftragsmanagement.

Productiondefinitions

Create, Update Or Get Your Production Definitions. You Need An Access Token With ROLE_DIS_ACCESS To Access This Resource.

Endpoints

GET

`/productiondefinitions/{id}`

Get a specific production definition.

PUT

`/productiondefinitions/{id}`

Update an existing production definition.

DELETE

`/productiondefinitions/{id}`

Delete an existing production definition.

GET

`/productiondefinitions`

Get all production definitions.

POST

/productiondefinitions

Add a new production definition.

DIALOGPOST Contracts

Create, Update Or Get Your DIALOGPOST Contracts.You Need An Access Token With ROLE_DIS_ACCESS To Access This Resource.

Endpoints

GET

/dialogpost/contracts/{id}

Get a specific contract.

PUT

/dialogpost/contracts/{id}

Update an existing contract based on information you negotiated with your DV franking consultant.

DELETE

/dialogpost/contracts/{id}

Delete a contract.

GET

/dialogpost/contracts

Get all contracts.

POST

`/dialogpost/contracts`

Add a new contract based on information you negotiated with your DV franking consultant.

DIALOGPOST Processings

Create, Update And Release Processings Or Track Processing States.You Need An Access Token With ROLE_DIS_ACCESS And ROLE_DIS_MANAGE_DIALOGPOST To Access This Resource.

Endpoints

`GET`

`/dialogpost/processings`

Get an overview of your processings.

`POST`

`/dialogpost/processings`

Create a new processing as a draft.

`POST`

`/dialogpost/processings/{id}/release`

Release a processing on basis of the current draft.

`POST`

`/dialogpost/processings/{id}/mailingitems`

Create mailing items.

`DELETE`

`/dialogpost/processings/{id}/mailingitems`

Delete mailing items.

GET

/dialogpost/processings/{id}/mailing

Get information about the mailing.

POST

/dialogpost/processings/{id}/mailing

Add information about the mailing.

GET

/dialogpost/processings/{id}

Get detail information about a processing.

DELETE

/dialogpost/processings/{id}

Delete a processing draft.

Default Induction

Manage Induction Information For POSTWURFSPEZIAL And POSTAKTUELL Processings.You Need An Access Token With ROLE_DIS_ACCESS To Access This Resource.

Endpoints

GET

/defaultinduction

Get default induction information for a customer.

PUT

/defaultinduction

Update default induction information for a customer.

`POST`

`/defaultinduction`

Define default induction information for a customer.

`DELETE`

`/defaultinduction`

Delete default induction information for a customer.

POSTWURFSPEZIAL Processings

Create, Update And Release Processings. Track Processing States And Collect Results.You Need An Access Token With ROLE_DIS_ACCESS And ROLE_DIS_MANAGE_POSTWURFSPEZIAL To Access This Resource.

Endpoints

`GET`

`/postwurfspezial/processings/{id}`

Get detail information about a processing.

`PUT`

`/postwurfspezial/processings/{id}`

Update an existing processing draft.

`DELETE`

`/postwurfspezial/processings/{id}`

Delete a processing draft.

`GET`

`/postwurfspezial/processings`

Get an overview of your processings.

`POST`

`/postwurfspezial/processings`

Create a new processing as a draft.

`POST`

`/postwurfspezial/processings/{id}/release`

Release a processing on basis of the current draft.

`GET`

`/postwurfspezial/processings/{id}/mailing`

Get information about the mailing.

`POST`

`/postwurfspezial/processings/{id}/mailing`

Add information about the mailing.

POSTAKTUELL Processings

Create, Update And Release Processings. Track Processing States And Collect Results.You Need An Access Token With ROLE_DIS_ACCESS And ROLE_DIS_MANAGE_POSTAKTUELL To Access This Resource.

Endpoints

`GET`

`/postaktuell/processings/{id}`

Get detail information about a processing.

PUT

`/postaktuell/processings/{id}`

Update an existing processing draft.

DELETE

`/postaktuell/processings/{id}`

Delete a processing draft.

POST

`/postaktuell/processings`

Create a new processing as a draft.

POST

`/postaktuell/processings/{id}/release`

Release a processing on basis of the current draft.

GET

`/postaktuell/processings/{id}/mailing`

Get information about the mailing.

POST

`/postaktuell/processings/{id}/mailing`

Add information about the mailing.

POSTWURFSPEZIAL Processing Results

Collect Results And Access Information Logs.You Need An Access Token With ROLE_DIS_ACCESS And ROLE_DIS_MANAGE_POSTWURFSPEZIAL To Access This Resource.

Endpoints

GET

/postwurfspezial/processings/{id}/results

Get processing result documents.

GET

/postwurfspezial/processings/{id}/results/postage

Get postage result information.

GET

/postwurfspezial/processings/{id}/results/mailingitems

Get resulting mailing items.

GET

/postwurfspezial/processings/{id}/log

Retrieve a trace log with info and error messages.

Order Records

Manage Order Information (Order Records) Which Were Sent To The Deutsche Post Auftragsmanagement.You Need An Access Token With ROLE_DIS_ACCESS To Access This Resource.

Endpoints

GET

/orderrecords

Fetches a list of order records.

`PUT`

`/orderrecords`

Updates an order in order management (AM) and creates a record entry of type 'Change'.

`DELETE`

`/orderrecords`

Cancel an order within Deutsche Post Auftragsmanagement.

Default Payment

Manage Payment Information For POSTWURFSPEZIAL And POSTAKTUELL Processings.You Need An Access Token With ROLE_DIS_ACCESS To Access This Resource.

Endpoints

`GET`

`/defaultpayment`

Get default payment information for a customer.

`PUT`

`/defaultpayment`

Update default payment information for a customer.

`POST`

`/defaultpayment`

Define default payment information for a customer.

`DELETE`

`/defaultpayment`

Delete default payment information for a customer.

Downloads

By downloading any file, you accept our **Terms of Use**

Open API Specification File
YAML

Einrichtungsauftrag_Dispatchpreparation_2022_12_1.docx
DOCX - 19.28 KB

Postman-Collection-User-Guide-EN_0.pdf
PDF - 544.1 KB

Environment-REF-Versandvorbereitung.postman_environment_0.json
JSON - 1.9 KB

VersandvorbereitungAPI-DialogPost-13Sept2022.postman_collection_1.json
JSON - 422.72 KB

VersandvorbereitungAPI-Postaktuell-13Sept2022.postman_collection_0.json
JSON - 29.36 KB

VersandvorbereitungAPI-Postwurfspezial-13Sept2022.postman_collection_0.json
JSON - 30.21 KB

GSA-Sequenzen BZ bundesweit 2022-12-01.csv
CSV - 37.96 KB