



# gopdfprinter

---

[draft]

## Table of contents

---

|                           |    |
|---------------------------|----|
| 1. gopdfprinter           | 3  |
| 1.1 Architecture          | 3  |
| 2. Printing               | 4  |
| 2.1 Print reference       | 4  |
| 2.2 Operations            | 7  |
| 2.3 Hooks                 | 8  |
| 3. Métriques              | 9  |
| 4. Configuration          | 10 |
| 4.1 Authentication        | 10 |
| 4.2 Environment variables | 11 |
| 5. PDF Printer todo list  | 13 |

# 1. gopdfprinter

---

## 1.1 Architecture

---

- gopdfprinter api
- gopdfprinter worker
- queue: redis

Version PDF : [télécharger](#)

# 2. Printing

## 2.1 Print reference

Print are created sending a `POST` request to `/prints` endpoint; the full print object with print id is returned with 201 status.

```
{
  "name": "concatenated-filename.pdf",
  "src": [
    {
      "name": "filename.pdf",
      "url": "https://google.fr",
      "file": false
    }
  ],
  "operations": [
    "concatenate",
    "normalize",
    "upload"
  ],
  "hook": "https://echo.solidev.net/",
  "params": {
    "header_display": true,
    "footer_display": true,
    "margin_top": 80,
    "margin_bottom": 60,
    "header_template": "...",
    "footer_template": "...",
    "paper_format": "a2"
  }
}
```

### 2.1.1 `src` sources array

each source is an object with following fields :

Mandatory fields :

- `url` (**string**) : mandatory, url of page to be printed or zip object containing a index.html page to be printed

Optional fields :

- `name` (**string**) : output filename, defaults to print\_XXXX.pdf
- `file` (**bool**) : print as file (page or zip content is downloaded before printing as file://)
- `pages` (**string**) : pages interval, comma separated (example : `1-3, 2, 5`); all pages if empty or not given - NOT IMPLEMENTED
- `ref` (**string**) : source reference, not used for printing

### 2.1.2 `operations`

Array of operations

## 2.1.3 hook

Url of hook, receiving `POST` messages with the following json content :

```
{
  "print": "print uuid",
  "status": "current status",
  "progress": 100, // 0-100 progress
  "result": {
    "pages": [
      {"ref": "", "url": "result url"},
      {"ref": "", "url": "result url"},
    ],
    "zip": {"url": "zip result url"},
    "print": {"url": "concatenated result url"}
  }
}
```

## 2.1.4 params : print parameters

PDF print parameters can be given for each print, or globally :

### Global parameters

- **media** : media type to print (css media types, screen or print) (default: `print`, parameter : `printer.pdf.params.media`, environment : `GPD_PRINTER_PDF_PARAMS_MEDIA` )
- **background** : print backgrounds (true, false) (default : `true`, parameter : `printer.pdf.params.background`, environment : `GPD_PRINTER_PDF_PARAMS_BACKGROUND` )
- **scale** : print scale (floating number) (default : `1.0`, parameter : `printer.pdf.params.scale`, environment : `GPD_PRINTER_PDF_PARAMS_SCALE` )

### Paper parameters

- **orientation** : print paper orientation (portrait / landscape, any other value than portrait means landscape) (default : `portrait`, parameter: `printer.pdf.params.orientation`, environment: `GPD_PRINTER_PDF_PARAMS_ORIENATION` )
- **paper\_format** : print paper format name (A10 to 4A0, B10 to B0, C10 to C0) (default : `A4`, parameter: `printer.pdf.params.paper_format`, environment:

`GPD_PRINTER_PDF_PARAMS_PAPERFORMAT` ). If

`paper_width` and `paper_height` are provided and not null, the paper format is ignored.

- **paper\_width** : paper width **in millimeters** (default : `0`, parameter : `printer.pdf.params.paper_width`, environment: `GPD_PRINTER_PDF_PARAMS_PAPER_WIDTH` )
- **paper\_height** : paper height **in millimeters** (default : `0`, parameter : `printer.pdf.params.paper_height`, environment: `GPD_PRINTER_PDF_PARAMS_PAPER_HEIGHT` )

### Margins

- **margin\_top** : top margin in mm (default : 0, parameter : `printer.pdf.params.margin_top`, environment : `GPD_PRINTER_PDF_PARAMS_MARGIN_TOP` )
- **margin\_bottom** : bottom margin in mm (default : 0, parameter : `printer.pdf.params.margin_bottom`, environment : `GPD_PRINTER_PDF_PARAMS_MARGIN_BOTTOM` )
- **margin\_left** : left margin in mm (default : 0, parameter : `printer.pdf.params.margin_left`, environment : `GPD_PRINTER_PDF_PARAMS_MARGIN_LEFT` )
- **margin\_right** : right margin in mm (default : 0, parameter : `printer.pdf.params.margin_right`, environment : `GPD_PRINTER_PDF_PARAMS_MARGIN_RIGHT` )

### Header / footer

Those parameters have no default value (no header/footer display if nothing is provided in print query)

- **header\_display** : display header (true / false)
- **header\_template** : header html template
- **footer\_display** : display footer (true / false)
- **footer\_template** : footer html template

## Header &amp; footer tips &amp; tricks :

- pdf page header & footer don't load external resources : use base64 data for images, and some other tricks can be useful ([source](#))
- pdf page header & footer are not displayed if they intersect with page content : use `margin_top` and `margin_bottom` ([source](#))
- background colors are not displayed unless `html {  
-webkit-print-color-adjust: exact;}` is added in header style : ([source](#))
- chrome adds default padding to header & footer, to disable it, use style `#header, #footer { padding: 0 !important; }` : ([source](#))
- some css units may be scaled up / down (to be checked) : ([source](#))
- page number / totalPages and other special header/footer special css classes are not displayed, should be fixed soon ([source](#))

## Example footer / header :

```
<style>
  html {
    -webkit-print-color-adjust: exact;
  }
  #header, #footer {
    padding: 0 !important;
  }
  .pgn {
    width: 100vw;
    height: 1cm;
    background-color: #FF0000;
    border: 1px solid red;
    font-size: 5mm;
    text-align: center
  }
</style>
<div class='pgn'>
  <span class='pageNumber'>
    </span> of <span class='totalPages'>
    </span>
  RESERVED TO ME
</div>
```

## 2.2 Operations

---

- **pdf** print as pdf, always done, no need to provide this operation
- **referencesEach** : resolve page references for each source with file = true
- **references** : globally resolve page references on concatenated pdf (not implemented)
- **normalizeEach** : shrink and compress each pdf
- **normalize** : shrink and compress concatenated pdf
- **uploadEach** : upload each pdf
- **upload** : upload concatenated pdf
- **concatenate** : concatenate pdf
- **zip** : zip all pdf to a zip archive
- **uploadZip** : upload zip archive

## 2.3 Hooks

---

Messages are sent to hook urls using POST to hook url

The JSON payload have default fields :

- **print (string)**: job id
- **status (string)**: job status (string)
- **progress (string)**: job progress (integer)
- **result (string)** : job result

Job result fields are :

- **pages (array of result items)** : contains individual source result
- **zip (result item)** : contains zipped pages content result
- **print (result item)** : contains concatenated page result

A result items contains :

- **url** : result url
- other fields // TODO: normalize results fields

Other fields can be provided, but this is not used now.

# 3. Métriques

---

`gopdfprinter_queues_available_queue_count` `gopdfprinter_queues_running_queue_count`  
`gopdfprinter_queues_error_queue_count` `gopdfprinter_system_alive_worker_count`

# 4. Configuration

---

## 4.1 Authentication

---

### 4.1.1 Parameters

- **api.auth.type** :
  - `token` for token-based authentication
  - `none` for no authentication

For `token` mode authentication :

- **api.auth.token.token** : token value (default : `""` - empty token disable token auth)
- **api.auth.token.lookup** : location to find token in request (default : `header:x-token`)
  - `header:<name>`
  - `query:<name>`
  - `cookie:<name>`
  - `form:<name>`

## 4.2 Environment variables

- **GPD\_REDIS\_ADDRESS** : redis connection host:port (default: localhost:6379)
- **GPD\_REDIS\_PASSWORD** : redis password (default: empty)
- **GPD\_REDIS\_DB** : redis db number (default: 0)
- **GPD\_REDIS\_QUEUES\_AVAILABLE** : available queue name (default : available\_queue)
- **GPD\_REDIS\_QUEUES\_RUNNING** : running queue name (default : running\_queue)
- **GPD\_REDIS\_QUEUES\_ERROR** : error queue name (default : error\_queue)
- **GPD\_REDIS\_PRINTS\_PREFIX** : prints data prefix, a '.' is added before print id (default : prints)
- **GPD\_REDIS\_PRINTS\_TTL** : print data ttl in seconds (default: around 365 days)

### 4.2.1 API

- **GPD\_API\_HOST** : api host (default : 0.0.0.0)
- **GPD\_API\_PORT** : api port (default : 8000)

### 4.2.2 Authentication

- **GPD\_API\_AUTH\_TYPE** : authentication type
- **GPD\_API\_AUTH\_TOKEN\_TOKEN** : token value
- **GPD\_API\_AUTH\_TOKEN\_LOOKUP** : token lookup location

### 4.2.3 Task management

- **GPD\_PROCESSING\_TIMEOUT** : processing timeout in seconds (if a job stays longer in "processing", it is considered as dead and moved to error queue (default : 3600)
- **GPD\_ERROR\_TIMEOUT\$** : base time between retries (times square of retry count)
- **GPD\_ERROR\_MAXRETRY** : max number of retries before a print is reported as failed
- **GPD\_MONITORING\_KEYS\_ALIVE** : redis key for alive worker monitoring

### 4.2.4 Monitoring

- **GPD\_MONITORING\_SERVER\_PORTS\_FIRST** : prometheus metrics first port
- **GPD\_MONITORING\_SERVER\_PORTS\_LAST** : prometheus metrics last port
- **GPD\_REDIS\_KEYS\_MONITORING\_ALIVE** : redis key to count alive workers

### 4.2.5 Printing

- **GPD\_PRINTER\_PDF\_TIMEOUT** : pdf print timeout seconds (default 180)
- **GPD\_PRINTER\_TEMPDIR\_LOCATION** : temporary directory location (default : system temp dir)
- **GPD\_PRINTER\_TEMPDIR\_REMOVE** : remove print directory after print (default : false)

### 4.2.6 Uploading

- **GPD\_PRINTER\_UPLOAD\_DESTINATION** : s3 / local
- **GPD\_PRINTER\_UPLOAD\_S3\_KEY** : S3 access key
- **GPD\_PRINTER\_UPLOAD\_S3\_SECRET** : S3 secret key
- **GPD\_PRINTER\_UPLOAD\_S3\_REGION** : S3 region
- **GPD\_PRINTER\_UPLOAD\_S3\_BUCKET** : S3 bucket name
- **GPD\_PRINTER\_UPLOAD\_S3\_ENDPOINT** : S3 endpoint (blank for amazon defaults)
- **GPD\_PRINTER\_LOCAL\_URL** : base url (without slash) for local uploads
- **GPD\_PRINTER\_LOCAL\_PATH** : root dir for local upload

### Print progress weights

- **GPD\_PRINTER\_PROGRESS\_WEIGHT\_PDF** : progress pdf print weighting
- **GPD\_PRINTER\_PROGRESS\_REFERENCESEACH\_P** : progress pdf print weighting
- **GPD\_PRINTER\_PROGRESS\_NORMALIZEEACH\_P** : progress pdf print weighting

- **GPD\_PRINTER\_PROGRESS\_UPLOAD\_EACH\_PDF :** progress pdf print weighting
- **GPD\_PRINTER\_PROGRESS\_ZIP\_PDF :** progress pdf print weighting
- **GPD\_PRINTER\_PROGRESS\_REFERENCES\_PDF :** progress pdf print weighting
- **GPD\_PRINTER\_PROGRESS\_CONCATENATE\_PDF :** progress pdf print weighting
- **GPD\_PRINTER\_PROGRESS\_UPLOAD\_ZIP\_PDF :** progress pdf print weighting
- **GPD\_PRINTER\_PROGRESS\_NORMALIZE\_PDF :** progress pdf print weighting
- **GPD\_PRINTER\_PROGRESS\_UPLOAD\_PDF :** progress pdf print weighting
- **GPD\_PRINTER\_PROGRESS\_PING\_THROTTLE\_MS :** throttle progress pings milliseconds (default : 1500)
- **GPD\_PRINTER\_PDF\_PARAMS\_SCALE :** print scale as floating number
- **GPD\_PRINTER\_PDF\_PARAMS\_ORIENTATION :** portrait / landscape
- **GPD\_PRINTER\_PDF\_PARAMS\_PAPER\_FORMAT :** A10 to 4A0, B10 to B0, C10 to C0
- **GPD\_PRINTER\_PDF\_PARAMS\_PAPER\_HEIGHT :** paper height in mm
- **GPD\_PRINTER\_PDF\_PARAMS\_PAPER\_WIDTH :** paper width in mm
- **GPD\_PRINTER\_PDF\_PARAMS\_MARGIN\_TOP :** margin top in mm
- **GPD\_PRINTER\_PDF\_PARAMS\_MARGIN\_BOTTOM :** margin bottom in mm
- **GPD\_PRINTER\_PDF\_PARAMS\_MARGIN\_LEFT :** margin left in mm
- **GPD\_PRINTER\_PDF\_PARAMS\_MARGIN\_RIGHT :** margin right in mm

## Print parameters

- **GPD\_PRINTER\_PDF\_PARAMS\_MEDIA :** print / screen
- **GPD\_PRINTER\_PDF\_PARAMS\_BACKGROUND :** print background (true / false)

## 4.2.7 Logging

- **GPD\_LOGGING :** logging level (silent, quiet, verbose, debug, trace)

## 5. PDF Printer todo list

---

- add token authentication
- check zip and upload zip
- implement references
- document references and referencesEach
- document all job statuses
- check job errors management
- implement job retries
- implement job cancellation
- document print parameters, add parametrizable defaults
- add s3 base path
- add settings for logging levels
- add helm chart
- document and check helm config



<https://dev.solidev.net/solidev/gopdfprinter>