ENFOCUS

PITSTOP LIBRARY CONTAINER

Installation Guide

05 - 2024

Contents

1. Important notes

As of version 2415, we are offering the endpoint “/preflight” as a replacement for “/job”. However, we will keep /job as a valid endpoint for several releases to ensure backwards compatibility. That said, we firmly encourage you to consider the “/job” endpoint deprecated and start replacing any “/job” call to a “/preflight” call. Note that apart from the endpoint name itself, nothing changes regarding sending parameters.

2. What is the PitStop Library Container?

Put 25 years of PDF preflight and correction knowhow in your cloud. Validate PDF files, using PitStop Library, as they are submitted.

OEM vendors, print service providers, ad agencies, marketing content creators, and any other manufacturer that wants to put PDF preflight and/or high-quality and color-consistent images generated from PDF files in their web-to-print, management information system, online proofing solutions, or any other customer portal solution, being it for B2B or B2C purposes, can access the power of PitStop via REST API. Provided as a Docker image, PitStop Library Container makes integration easier and future-forward by providing a cloud-ready preflight solution that works on any platform of your choice, being it private cloud or using any available cloud service. It's also possible to add a preflight service to your internal processes and by deploying the service on premise.

Control over job queuing, scalability, and distribution is in the hands of the developer.

Preflight Profiles and Action Lists created with PitStop Pro and PitStop Server can be used with the PitStop Library Container, keeping uniformity across print production.

PitStop Library Container is a perfect tool for:

- OEMs with the need to put PDF preflight in their cloud solutions
- W2P and MIS vendors wanting to incorporate preflight
- Print businesses with home-brewed or off the shelf W2P systems
- Publications that receive print ads from multiple sources

Benefits:

- Preflight PDFs during the job onboarding process
- Capture errors before PDFs hits production, saving time and money
- Fix errors to avoid delays and provide an even higher level of service
- Provide instant feedback on preflight results
- Preflight reports can be generated in PDF, XML, and JSON
- Unrestricted scaling
- Spread load by running multiple preflight processing tasks in parallel on a single server or load-balance between multiple servers
- Run on any platform and OS of your choice
- Create high-quality and color-consistent images from PDF files

The PitStop Library Container supports the following languages: English, German, Spanish, French, Italian, Dutch, Japanese, Portuguese, Chinese, and Polish.

For more information about the Docker technology, refer to *https://www.docker.com/resources/what-container/*

3. Getting the Docker image

3.1. Docker Desktop

To install the Docker container on your host machine, you need Docker Desktop, which can be downloaded from *https://www.docker.com/get-started/*

Docker Desktop is available for Mac, Windows, and Linux.

If you want to run PitStop Library Container in a cloud environment, see *further*.

3.2. AWS CLI

The PitStop Library Container is available through an AWS ECR repository. More information on AWS ECR is available at *https://aws.amazon.com/ecr/*

To get the image, you need to install the AWS CLI environment, which is available at *https://docs.aws.amazon.com/cli/latest/userguide/getting-started-install.html*

If the AWS CLI environment is installed, open the console and make sure both Docker Desktop and AWS CLI are installed correctly by checking the version:

```
>> docker --version
```

```
>> aws --version
```

If everything is installed correctly, it shows the corresponding version.

3.3. Adding credentials

Note: The Docker Desktop application must be up and running before continuing.

Then add the credentials to AWS CLI by entering the following command:

```
>> aws configure
```

It will ask for the **AWS Access Key ID** and the **AWS Secret Access Key**, which are provided by Enfocus.

It will also ask for the **Default region name**, where you can enter *eu-west-1* and the **Default output format**, which can be left empty (just press enter).

Then log in to AWS by entering this command:

```
>> aws ecr get-login-password --region eu-west-1 | docker login --username AWS --
password-stdin 118129972018.dkr.ecr.eu-west-1.amazonaws.com
```

If the credentials were entered correctly, then this command will show: *Login Succeeded*.

3.4. Downloading the image locally

To download the latest version of the image from the AWS ECR repository, use the following command:

```
>> docker pull
118129972018.dkr.ecr.eu-west-1.amazonaws.com/enfocus-plc-preflight-worker:latest
```

Now it starts downloading, extracting, and installing the Docker image. You can also see in the Docker Desktop application that the image is now available.

3.5. Using Google Container Registry

Google Container Registry provides secure, private Docker repository storage on Google Cloud Platform (GCP). You can use gcloud to push repositories to your registry; then you can pull repositories using an HTTP endpoint from any machine, whether it's a Google Compute Engine instance or your own hardware.

If you want to store the Docker image in your own Google Container Registry, proceed as described below.

3.5.1. Download the image locally

For more details, refer to *Downloading the image locally* on page 7.

3.5.2. Get the GCloud command line tool

You must install the GCloud command line tool.

To get the installer, go to *https://cloud.google.com/sdk/docs/install* and follow the instructions.

3.5.3. Use GCloud tool to connect

```
$gcloud auth configure-docker

$ docker tag 118129972018.dkr.ecr.eu-west-1.amazonaws.com/enfocus-plc-preflight-worker
 gcr.io/[YOURGOOGLEPROJECTID]/118129972018.dkr.ecr.eu-west-1.amazonaws.com/enfocus-
plc-preflight-worker
```

```
$ docker push gcr.io/[YOURGOOGLEPROJECTID]/118129972018.dkr.ecr.eu-
west-1.amazonaws.com/enfocus-plc-preflight-worker
```

3.5.4. Run the image

Go to your Google Project website. Then switch to the Google project website and go to your Google Container Registry for your project. Here you should see the image and if you click on it you will come to this level where you can use "Deploy to cloud run".

Onced configured, run it.

3.5.5. Edit your Yml file

Open the Yml file for and add the following options below the “containers” section:

```
containers:

      - image: >-

          gcr.io/project/118129972018.dkr.ecr.eu-west-1.amazonaws.com/enfocus-plc-
preflight-worker:latest

        ports:

        - name: http1

          containerPort: 3000

        env:

        - name: COM_ENFOCUS_CLOUD_EQUIPMENT_TOKEN

          value: XXXXXXXX

        - name: COM_ENFOCUS_CLOUD_ACCOUNT_NUMBER

          value: XXXXXXXX

        - name: COM_ENFOCUS_CLOUD_PLC_REGION

          value: eu

        - name: COM_ENFOCUS_CLOUD_SUBSCRIPTION_TYPE_ID

          value: X

        startupProbe:

          timeoutSeconds: 240

          periodSeconds: 240

          failureThreshold: 1

          tcpSocket:

            port: 3000
```

3.6. Downloading a specific version

If, for some reason, you want to use a previous version of the PLC, this can be done with the following command:

```
>> docker pull
118129972018.dkr.ecr.eu-west-1.amazonaws.com/enfocus-plc-preflightworker:2303
```

So you can just replace "latest" with the specific version you want to download. Examples of supported versions are: 22.0, 2303, 2319, 2349 and 2403.

Note: As of 2023 the version number is formatted as follows: YYWW where YY is the number of the year and WW is the week number. So 2319 means the version that is released in week 19 of 2023.

4. Configuring the PitStop Library Container

After downloading the Docker image, and before you can run it, you need to configure it correctly. For the configuration, we make use of the file. This file contains some parameters and environment variables that need to be adjusted according to your specific situation. The docker-compose.yml file is provided by Enfocus and can be edited.

More generic information on how this file is built, can be found at *https://docs.docker.com/compose/compose-file/*

An example file is shipped along with the documentation (see *the appendix of this document*).

The file looks like this:

```
version: '3.2'
        services:
        worker-service:
        image: <image name>
        container_name: node-worker
        ports:
        - 3000:3000
        volumes:
        - ~/.aws/credentials:/root/.aws/credentials:ro
        - /var/run/docker.sock:/var/run/docker.sock
        - ~/local_dir:/root/dir_within_docker:ro
        - ~/output_dir_local:/root/dir_output_within_docker:
        restart: always
        environment:
        - <some environment variables>
```

The different properties are:

- The version property contains the Docker compose version.
- The image property should contain the full image name as you see it in your Docker Desktop.
- The container_name property can be whatever value. It just gives a name to the container.
- The ports property defines on what port the Docker image is listening for incoming API calls.

Note: While the first port may be set to any value you need to use for your server, the second one MUST remain 3000 for the Docker container. In other words, you can set ports this way: {any port value}:3000

- The restart property defines what needs to happen when the Docker container closes unexpectedly. The always value means that it will automatically restart when it was closed unexpectedly. You can leave out this property, but it is recommended to keep it as such.
- The volumes property contains mounted drives that will be accessible from within the Docker container. The example above is an example of the AWS credentials files that can be used to verify access to a AWS SQS queue.
- The environment property contains all environment variables explained below. Note that the signature is as follows: ENVIRONMENT_VARIABLE=VALUE

Environment variable	Default value	Description
COM_ENFOCUS_CLOUD_PREFLIGHT_JOB_MANAGERS_MAX_COUNT		

Environment variable	Default value	Description
	3	The number of workers that will be started with the Docker container. This should not be more than the number of cores.
COM_ENFOCUS_CLOUD_PREFLIGHT_PROCESSOR_URL_PORTS		
	<empty>	Comma separated field that contains port numbers that will be assigned to the workers. Typically, this can be left empty. Only in exceptional cases where there are specific port requirements, this should be added.
COM_ENFOCUS_CLOUD_PREFLIGHT_SQS_NAME		
	<empty>	The name of the AWS SQS queue to pull jobs from (see Queuing for PLC).
COM_ENFOCUS_CLOUD_PREFLIGHT_SQS_LOCAL_ENDPOINT		
	<empty>	Set required value to configure the local SQS setup.
COM_ENFOCUS_CLOUD_PREFLIGHT_SQS_PROFILE		
	<empty>	Set required value to define local SQS queue name if above env is configured.
COM_ENFOCUS_CLOUD_PREFLIGHT_SQS_ACCESS_KEY		
	<empty>	Set required value to define the access key for SQS queue.
COM_ENFOCUS_CLOUD_PREFLIGHT_SQS_SECRET_KEY		
	<empty>	Set required value to define the secret key for SQS queue.
COM_ENFOCUS_CLOUD_AWS_REGION		
	eu-west-1	This is the region where the SQS queue is configured.
COM_ENFOCUS_CLOUD_PLC_REGION		
	eu	This is the region of the Enfocus License server. Can be either eu or us, depending on your location.
COM_ENFOCUS_CLOUD_EQUIPMENT_TOKEN		
	<empty>	This is the so-called equipment token that is used for accessing your license on our licensing server. This token is provided by Enfocus and is uniquely linked to your account. This token is a secret, which should never be shared externally.
COM_ENFOCUS_CLOUD_ACCOUNT_NUMBER		
	<empty>	This is your account number. Every customer has a unique account number, which is provided by Enfocus.
COM_ENFOCUS_CLOUD_SUBSCRIPTION_TYPE_ID		

Environment variable	Default value	Description
	ENF-PLC-PLCTR	This is the subscription type identifier that is provided by Enfocus. This contains a fixed value for the PitStop Library Container.
COM_ENFOCUS_CLOUD_FONT_LICENSE_CHECK		
	false	This is a Boolean value that you can set to true to skip the font license check. Remember that you are responsible for complying with the font license restrictions! If you enable the Skip Font License Check, the following message will be displayed in the logs: `Some fonts may be licensed for view and print only. Enfocus disclaims any liability for any use of the fonts which is not in accordance with the licensing agreement between yourself and the print vendor.`
COM_ENFOCUS_CLOUD_PROGRESS_TIMEOUT		
	30000	Set a value in ms to extend delay before a timeout event aborts an ongoing job.

Important: To run the PitStop Library Container, there should be a connection to the Enfocus license server. At start-up, the PitStop Library Container checks connectivity and will only continue if a connection can be made.

5. Running the PitStop Library Container

After the image has been downloaded to your local environment, and the necessary configuration options have been set up and saved in your docker-compose.yml file, you can run the Docker software by changing the working directory to the one where your docker-compose.yml file is located and run the following command:

```
>> docker-compose up
```

If everything is set correctly, then the above command must confirm the container has been created and that the number of workers you set in the yml file are up and ready to process files:

```
 ✔ Container node-worker  Created
Attaching to node-worker
node-worker  | {
node-worker  |   worker: 'PLC-PREFLIGHT',
node-worker  |   version: '2343',
node-worker  |   service: 'WorkerManager',
node-worker  |   level: 'info',
node-worker  |   message: 'Initializing workers',
node-worker  |   timestamp: '2023-11-23T08:53:58.356Z'
node-worker  | }
node-worker  | {
node-worker  |   worker: 'PLC-PREFLIGHT',
node-worker  |   version: '2343',
node-worker  |   service: 'com.enfocus.preflight-worker.server',
node-worker  |   level: 'info',
node-worker  |   message: 'Listening on port 3002',
```

5.1. Health check

You can check if the Docker instance is still actively running by doing a health check. There are 2 methods that you can use to check if the instance is OK.

The first method is a version check. This method always gives you the same response as shown below.

Request URL (GET):

```
/version.json
```

Example response:

```
{
  "name": "PLC-PREFLIGHT",
  "org": "com.enfocus.cloud",
  "version": "XXXX"
}
```

Where XXXX stands for the current version, for example: 2403.

The second method is a check to see how many workers there are, and what their status is, as shown below.

Request URL (GET):

```
/isAlive
```

Example response:


```
[
  {
    "workerInstance": 1,
    "processing": false,
    "alive": true
  },
  {
    "workerInstance": 2,
    "processing": false,
    "alive": true
  },
  {
    "workerInstance": 3,
    "processing": false,
    "alive": true
  }
]
```

6. Understanding PitStop PDF editing technology

The PitStop Library is a very powerful technology that lets users preflight and modify PDF Files. Many people would immediately think to Enfocus PitStop Pro, the Adobe Acrobat plugin. But that's just one of the various products that rely on the PitStop Library, including PitStop Server, the SDK and of course, the PitStop Library Container.

The main advantage is that in all those products, preflighting or editing PDF files, all use the same concepts and tools. Let's have a look.

6.1. Preflighting

Preflighting a file consists in checking if a PDF file complies to your print production requirements. A very typical example is to verify that a font is not missing or that there are no images below a certain resolution. But those are only small examples, there is indeed many parameters that you can check as well.

Running a preflight with the PitStop Library requires a Preflight Profile. This is a list that contains all the checks you have defined, and that PitStop will use to analyze (and possibly fix) the PDF document.

In PitStop Pro, you can work with a UI for creating and using Preflight Profiles. But when it comes to the PitStop Library Container, you need a Preflight Profile file. Those have the "ppp" extension like for example "my_preflight_profile.ppp".

You can reference those ppp files with the following parameter:

```
"profileURL": "/some/location/my_preflight_profile.ppp",
```

If you want to learn more about how to build and export a Preflight Profile that you can use in the PitStop Library Container, we recommend you look at *the PitStop Quick Start Guide and the PitStop Preflight Checks manual on our website*.

The purpose of running a preflight is generally to get a "report" in return. You can get different types of reports such as PDF, XML or JSON reports. It's likely that you will want to get a JSON report for your PitStop Library Container integration, but you can combine several types of reports if you need (PDF+JSON+XML, JSON+XML...).

A report lists all the checks that were controlled and when any issue is found, it's also reported. Note that for PDF Reports, it's also possible to customize them, using your own company logo and colors, a/o adjust the level of details in the report.

To learn more about creating customized reports, we recommend you take a look at *the PitStop Report Template manual on our website*.

6.2. Editing a file

While Preflight Profiles mainly focus on checking/fixing a file to make it compliant with the production requirements, the PitStop Library also allows you to simply modify a PDF file. Such modifications can consist in adding content, changing page boxes...Possibilities are countless.

Again, PitStop Pro offers UI tools inside Adobe Acrobat to select objects and edit their properties. You can run Action Lists as well to apply multiple modifications to a PDF in one operation. A good analogy is a macro in Microsoft Excel where you can automate the execution of some actions.

But when it comes to the PitStop Library Container, you will need a file for the Action List to be executed. Those files have the “eal” extension like in “my_action_list.eal”.

You can call those EAL files with the following parameter:

```
"actionListURLs": ["/some/location/my_action_list.eal"]
```

The main difference here, is that you need to set an array of URLs. Indeed, you can run one to many Action Lists in one PLC call. By contrast, you can only run one Preflight Profile.

To learn more about creating Action Lists, we recommend you take a look at *the PitStop Pro Reference Guide and the PitStop Action Manual on our website*.

6.3. Using variables

The PitStop libray hosts a powerful feature, named “Smart Variables”. Variables can help maximizing the use of both Preflight Profiles and Action Lists with the PitStop Library Container.

A typical example is when you want to re-use a Preflight Profile for multiple different cases. Say that you want to check for an image resolution: if you use static values like for ex: 300ppi, it will only check for this value. But if your product is a wide-format banner, you may need a lower value.

Hence come the variables. You can set your Preflight Profile to use a “variable” for the resolution check. In other words, you tell PitStop: “I need a resolution check” but I will provide the exact value at execution time. That way, you can change the value of the variable on the fly and be sure PitStop will always check for the expected value.

To learn more about adding variables to Action Lists and Preflight Profiles, we recommend you to take a look at the *Smart Preflight* chapter in the *the PitStop Pro Reference Guide on our website.*

Adding variables requires the use of a Variable Set. This is a “container” for your variables. Later on, when you need to place a variable, PitStop needs to know in which Variable Set this variable is found. If you target the wrong Variable Set, PitStop may fail finding the expected variable and to proceed, or possibly use a wrong value.

In PitStop Pro and Server UI, it’s easy to create and access Variable Sets and their variables. But when it comes to the PitStop Library Container, you will need a “Variable Set” file. Those files have the extension “evs” like in “my_variable_set.evs”.

You can call those EVS files with the following parameter:

```
"variableSetURL": "/some/location/my_variable_set.evs"
```

6.4. Updating Variable Sets

Once Variables are placed and Variable Sets created, the question is how can you update the internal values? For example, to define a variable “productType” to either a “flyer”, “banner” or “businessCards” to name a few.

There are two ways to update the Variable Set file (.evs). The first one is to rely on a job ticket. A job ticket is typically an XML file where job data resides. You can reference those Job Ticket calling the following argument:

```
"jobTicketURL": "/some/location/my_job_ticket.xml"
```

That said, you still need to link the variable to the location of the value in the job ticket. For example, if your XML file is like

```
<job productType="Brochure" .../>
```

You want your variable to pick the value of the productType attribute. That is typically done in PitStop Server that offers a specific UI where you can do those associations. Then, all you need is to reference the job ticket and PitStop will automatically update the Variable Sets and its inner variables.

If you don’t have PitStop Server, then you need to create a Variable Set yourself that you can pass to the PitStop Library Container. Good news is that an “evs” file is actually an XML file. So it’s technically easy to create such a file. You could start from a default Variable Set (one that was manually created in PitStop Pro for example), and later tweak that file to change a value of a specific variable.

While this approach can definitely work, we encourage you to take advantage of the NPM package that we provide for free on GitHub. It’s a Class we designed to ease the integration of the PitStop Library Container in Node.js.

It’s then very easy to define a Variable Set on the fly and quickly output it as a evs file.

6.5. More info needed?

This concludes this quick tour of the most critical PitStop Library elements. During your integration journey with the PitStop Library Container, you may discover other topics like Color Management, Flattening, Fonts management…

Remember that we provide an extensive explanation in the Annexes of every single property that you can use with the PitStop Library Container. In case of doubt, have a look at those or please consult the existing online manuals. If unspite of our efforts, something remains unclear, feel free to let us know. We will then take actions to improve our documentation.

7. To queue or not to queue

There are 2 ways to start a job, either by direct access to the API or by queuing jobs and let them be pulled by the Docker image. Both have their advantages, and you should choose the best approach for your application.

Direct access is used when jobs need immediate processing. This is the fastest way, but if all workers are busy processing other jobs, new jobs will fail to start, and you will get an error message in the response when doing an API call. If you need more workers, you will have to scale the Docker image to be able to process enough in parallel. Disadvantage is that you need to set up scaling and this comes with a hardware cost. Second disadvantage is that you will have to deal with unprocessed jobs and build a mechanism to start them again.

Queuing is typically used when you either have high volume or unpredictable peak moments, and don't necessarily need to have immediate response. With this approach, the hardware is used in the most efficient way, and it can cope perfectly with peak moments. All jobs are just added to the queue, and they are pulled from the queue and processed one after the other. If you see that your queue keeps growing, then it's a good practice to start scaling. Even with this approach, you can track progress, but as long as the job is in the queue, you don't see any progress.

Note: When the request is sent directy to the endpoint (/preflight or /pdf2image), the PLC will return a job ID. When you send requests to a queue, the job ID value will be replaced by the Queue ID value when the job is processed by the worker.

As version 2349 and above merges services into one container, you need to set the jobType parameter to distinguish between a preflight or an image generation request:

- For pdf2image requests, you need to add the parameter "jobType":"pdf2image"
- For preflight requests, you need to add the parameter "jobType":"preflight"

In sections 5.1 and 5.2, we provide guidance for using local queues with ElasticMQ and LocalStack. Following those guidelines, you should be able to send requests to a local queue.

Which service should you prefer? We tend to think that using ElasticMQ may be a better option. First, there is no need to create queue explicitly. And you don't have to run two images. But feel free to explore the both of them and evaluate the one you are the most comfortable with.

7.1. Setting a local queue using ElasticMQ

ElasticMQ is a message queue system, offering an actor-based Scala and an SQS-compatible REST (query) interface.

ElasticMQ follows the semantics of SQS. Messages are received by polling the queue. When a message is received, it is blocked for a specified amount of time (the visibility timeout). If the message isn't deleted during that time, it will be again available for delivery. Moreover, queues and messages can be configured to always deliver messages with a delay.

The focus in SQS (and ElasticMQ) is to make sure that the messages are delivered. It may happen, however, that a message is delivered twice (if, for example, a client dies after receiving a message and processing it, but before deleting). That's why clients of ElasticMQ(and Amazon SQS) should be idempotent.

As ElasticMQ implements a subset of the SQS query (REST) interface, it is a great SQS alternative both for testing purposes (ElasticMQ is easily embeddable) and for creating systems which work both within and outside of the Amazon infrastructure.

More info: https://github.com/softwaremill/elasticmq

7.1.1. Set the yml file

Add the following to the yml file:

```
depends_on:
      - elasticmq
elasticmq:
 image: softwaremill/elasticmq-native:latest   //Image name we are taking from docker
    ports:                                                          //ports to expose

      - "9324:9324"
      - "9325:9325"
    stdin_open: true
    volumes:
      - ~/custom-test.conf:/opt/elasticmq.conf    // Mount the volume which has
 configuration file to create queue: "path-to-file/file-name.conf"
```

7.1.2. Create the conf file

Create a conf file (in our case custom-test.conf) with the following content:

```
node-address {
    protocol = http
    host = "*"
    port = 9324
    context-path = ""
}

rest-sqs {
    enabled = true
    bind-port = 9324
    bind-hostname = "0.0.0.0"
    // Possible values: relaxed, strict
    sqs-limits = strict
}

queues {
    PLC {                                                  // queue name
        defaultVisibilityTimeout = 10 minutes
        delay = 5 seconds
        receiveMessageWait = 0 seconds
        fifo=true
}
```

7.1.3. Add environment variables

Add the following environment variables to your yml file:

- COM_ENFOCUS_CLOUD_PREFLIGHT_SQS_LOCAL_ENDPOINT=http://elasticmq:9324

- COM_ENFOCUS_CLOUD_PREFLIGHT_SQS_NAME= PLC.fifo # name of your SQS bucket
- COM_ENFOCUS_CLOUD_AWS_REGION=localhost

7.1.4. Start PLC

Start PLC. If everything was set properly, the following message should appear:

7.1.5. Send requests to the queue

Send a get request to the URL:

http://localhost:9324/000000000000/PLC.fifo?Action=SendMessage&MessageBody={{job1}}&MessageGroupId={{randomMessageGroupId}}&MessageDedupl

Where:

- job1 is the JSON data of the request
- MessageGroupid and MessageDeduplicationId fields are mantatory in elasticMQ

7.2. Setting a local queue using LocalStack

LocalStack is a cloud service emulator that runs in a single container on your laptop or in your CI environment. With LocalStack, you can run your AWS applications or Lambdas entirely on your local machine without connecting to a remote cloud provider! Whether you are testing complex CDK applications or Terraform configurations, or just beginning to learn about AWS services, LocalStack helps speed up and simplify your testing and development workflow.

LocalStack supports a growing number of AWS services, like AWS Lambda, S3, Dynamodb, Kinesis, SQS, SNS, and many more!

More info: *https://www.localstack.cloud/*

7.2.1. Set the yml file

Add the following to the yml file:

```
depends_on:
      - localstack
 localstack:
    image: localstack/localstack:latest  // Image name
```

```
    ports:
      - "4566:4566"  # LocalStack SQS port
    environment:
      - SERVICES=sqs
      - DEBUG=1
```

7.2.2. Set the yml file for queuing

Create another yml file with the following content:

```
version: '3.8'
services:
  localstack:
    image: localstack/localstack:latest
    ports:
      - "4566:4566"  # LocalStack SQS port
    environment:
      - SERVICES=sqs
      - DEBUG=1
```

Next:

1. Start the container.
2. Open another command prompt for creating queue (Make sure aws-cli is installed).
3. aws --endpoint-url *http://localhost:4566* sqs create-queue --queue-name PLC

7.2.3. Add environment variables

Add the following Environment variables to your yml file:

- COM_ENFOCUS_CLOUD_PREFLIGHT_SQS_LOCAL_ENDPOINT=*http://localstack:9324*
- COM_ENFOCUS_CLOUD_PREFLIGHT_SQS_NAME= PLC # name of your SQS bucket
- COM_ENFOCUS_CLOUD_AWS_REGION=eu-west-1 // Provide AWS region ,it is not localhost!

7.2.4. Start PLC

Start PLC. If everything was set properly, the following message should appear:

```
docker-worker-1  | {
docker-worker-1  |   worker: 'PLC',
docker-worker-1  |   version: '2403',
docker-worker-1  |   service: 'SQS Service',
docker-worker-1  |   level: 'info',
docker-worker-1  |   message: 'Connected to SQS: http://sqs.eu-west-1.localhost.localstack.cloud:4566/000000000000/PLC',
docker-worker-1  |   timestamp: '2024-02-01T09:25:33.157Z'
docker-worker-1  | }
```

7.2.5. Send requests to the queue

Send a get request to the URL:

http://sqs.eu-west-1.localhost.localstack.cloud:4566/000000000000/PLC?Action=SendMessage&MessageBody={{job1}}

Where, job1 is the JSON data of the request.

8. Processing files

Once the Docker image runs on your local system, you can access the API Swagger documentation in your browser on the following location:

http://localhost:3000/api-docs

Note that 3000 is the default port. If you have chosen a different port, then change the URL accordingly.

8.1. Start a job

In order to preflight a file, you need to use the /preflight endpoint using a POST request. The parameters are passed in the body like above.

Note: If you are interested in generating images with PLC, you need to use a specific endpoint and appropriate parameters. For more information, refer to *Generating images* on page 28.

Request URL (POST):

```
/preflight
```

Example of some parameters passed in the Body (for more details, see the Swagger Documentation):

```
{
  "jobStatusURL": "http://localhost:8080/result",
  "reportProgress":true,
  "abortingTypeValue": "5",
  "progressTimeoutValue: "100",
  "inputFileURL": "url",
  "profileURL": "url",
  "actionListURLs": "string",
  "variableSetURL": "url",
  "allowFixes":false,
  "jobTicketURL": "url",
  "extraFontsFolderURL": "url",
  "reportTemplate": {
      "configFileURL": "string",
      "templateFileURL": "string"
     },
  "outputFixedFileURL": "url",
  "reportLanguage": "en-US",
  "maxItemsPerCategory": "100",
  "maxNumOccurrencesPerItem": "100",
  "reportURLs": {
       "JSON": "string",
       "XML": "string",
       "PDF": "string"
     }
}
```

Note: For more information about the parameters, refer to *Parameters overview /preflight parameters* on page 43.

Input files (inputFileURL, profileURL, actionListURLs, variableSetURL, jobticketURL, extraFontsFolderURL, reportTemplate) should be available through URLs that are accessible for the Docker container. Be careful with spaces. Depending on the webserver, you should replace spaces with the URL encoded value of %20. It's also possible to work with local files if you mount a folder in the volumes section of the docker-compose.yml file.

Output files (reportURLs, outputFixedFileURL) will be stored on the given location with a HTTP PUT command. Typically, pre-signed URLs are used for that, but any URL supporting PUT operations will work fine. You can also write files to a local folder (see *Using local files instead of URLs* on page 23).

The **jobStatusURL** will be used to post the result of the job. This is a webhook that can be used as trigger for further processing.

actionListURLs is a JSON array of URLs (using [] and a comma (,) as separator).

In version 2349 and above, a new parameter can be used: **allowFixes**. This parameter allows you, when set to false, to disallow fixes as set in a Preflight Profile. In other words, even if a Preflight Profile requires a fix (like a spot to CMYK modification), that fix won't apply if you set the allowFixes parameter to false.

Note: By default, allowFixes is set to true meaning that a fix when set is always applied.

More information about the parameters can be found in the Swagger documentation when the image is running locally: *http://localhost:3000/api-docs*

When adding the job, you will get a response as follows:

```
{
    "id": "b5014a3b-e89a-4792-b269-6808e3180f26",
    "message": "Job received "
}
```

The id can be used in the */progress* API call and the */abort* API call.

Note: Some parameters can be optional or mandatory depending on the context. For example, if you ask for a report which is XML or JSON, you won't need to set a PDF Report Template Path. However, this will become mandatory if you require a PDF Report to be produced. In case of doubt, please refer to the Swagger Documentation.

8.2. Using local files instead of URLs

To prevent Preflight Profiles or Action Lists to be downloaded every time, it is possible to work with local files. First you need to mount them by adding the following in the volumes section of your docker-compose.yml file:

```
[code]
    volumes:
    - ~/.aws/credentials:/root/.aws/credentials:ro
    - /var/run/docker.sock:/var/run/docker.sock
    - ~/myuser/plc/my_input_folder:/root/input/
    - ~/myuser/plc/my_output_folder:/root/output/
[/code]
```

where *~/myuser/plc/my_input_folder* and *~/myuser/plc/my_output_folder* are the folders in your host system, and */root/input* and */root/output* are the folders inside the Docker container.

Once you have added this and restarted the PLC, you can start using files through this mechanism.

Your parameters could look like this:

```
profileURL:/root/input/my_preflight_profile.ppp
actionListURLs:["/root/input/my_action_list.eal"]
outputFixedFileURL: "/root/output/my_pdf.pdf"
```

Note that you also can combine this with URLs. Typically your input PDF file can be served through URL, while your Preflight Profile (.ppp) and Action Lists (.eal) can be served through local file.

It's also possible to use the <JOBID> parameter in your local file path, for example:

```
outputFixedFileURL: "/root/output/<JOBID>/my_pdf.pdf"
```

This will create a folder in */root/output/* with the job ID and the *my_pdf.pdf* file will be created inside that folder.

Note that output files are being overwritten, so don't use the same filenames for your *outputFixedFileURL* or your *reportURLs*.

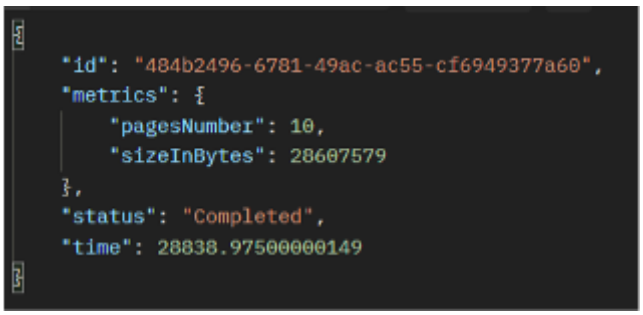
8.3. Progress of a job

With this API call, you can track the progress of a job, using /preflight or /pdf2image endpoint.

Request URL (GET):

```
/progress/<job_id>
```

The job_id is the ID you get as response from the job API call.



Note: The "Completed" status is temporary. When a new job has been started, the previous job is deleted, and you will get the same response as if a job is not found.

When a request fails, it will both output the error message to the PLC logs, but also to the job status URL (jobStatusURL). See the following screenshot for example:

```
{
  id: 'fda5196f-c36d-48a6-87b3-f7b402d618db',
  status: 'Failed',
  errorMessage: 'Failed to download input https://enfocus-plc-rocket.s3.eu-west-1.amazonaws.com/Input%20PDF/10page.pdf?A
WSAccessKeyId=AKIARXAIYKMZDUO3BOP5&Expires=1701845258&Signature=5P%2BN2WnkuCijZE2vSPHWUF3rlxU%3D file due to incorrect U
RL.',
  jobErrorStatus: 'DOWNLOAD_FILE_FROM_S3_ERROR',
  time: 1826.8361000004224
}
```


8.4. Get progress updates

Instead of requesting progress updates, you can ask to receive progress updates by defining a webhook URL in the jobStatusURL parameter.

Example:

```
"jobStatusURL": "http://localhost:8080/result,
"reportProgress": true
```

This is particularly helpful when using queuing as you can't anticipate the exact moment the request will be processed or when you want to show a progress bar in a UI.

You can adapt the frequency of those updates with the "progressMinFraction" parameter:

```
"progressMiniFraction": 0.05
```

The value can vary from 0 to 1. The lower the number, the more frequent the updates will be. The optimal value depends here on your material and requirements. You may want to adjust this parameter later on.

Once the job has been submitted, PLC will send progress messages to the jobStatusURL using POST and with a JSON message in the body:


```
{
  currentPage: 0,
  fraction: 0.17737671,
  totalPages: 1,
  type: 'processingPages',
  id: '015371f3-ac64-4c51-9690-bfecfa193177'
}
```


"id" refers to the ID of the job, either the ID returned by submitting a job to the /preflight endpoint, or the ID returned by submitting a job to an SQS queue.

"type" refers to the activity PLC is busy with. The possible values for /preflight are:

- initializing
- processingPages
- processingReport
- savingJsonReport
- savingXmlReport
- savingPdfReport
- idle

- other

“fraction” gives insight on the current progress of a certain type of activity.

“currentPage” and **“totalPages”** are self-explanatory.

The last update you will receive when the job is actually completed has this structure:

```
{
  id: '015371f3-ac64-4c51-9690-bfecfa193177',
  metrics: { pagesNumber: 6, sizeInBytes: 124716 },
  time: 3966.219419002533
}
```

While the “id” parameter is still present, all previous ones ("type", "fraction"...) are replaced by two new parameters:

- **"metrics"**: a JSON structure that gives info on the file size in bytes and the page count.
- **"time"**: indicates the task execution duration in milliseconds

Note: When using the /pdf2image endpoint, the progress updates only display one message on completion. The “status” parameter will indicate “completed”. We are working at providing a more consistent progress mechanism between /preflight and /pdf2image endpoints for progress updates.

```
{
  id: '015371f3-ac64-4c51-9690-bfecfa193177',
  metrics: { pagesNumber: 6, sizeInBytes: 124716 },
  status: 'completed',
  time: 3966.219419002533
}
```

8.5. Abort a job

If a job takes too long or seems to be stuck, you can always abort the job with the abort API call to free up resources. This call needs the job ID that you got from from the /preflight or /pdf2image API call.

Request URL (GET):

```
/abort/<job_id>
```

If the job is found, you get the following response:

```
{
    "message": "Currently processing job will be aborted."
}
```

If a job is not found, then you get an error response:

```
{
    "error": "No job found with id cce2ef34-e77a-40c0-af06-823a43022a53"
}
```

Note: The"Completed" status is temporary. When a new job has been started, the previous job is deleted, and you will get the same response as if a job is not found.

9. Generating images

PLC offers the possibility to generate images from your PDF files ensuring high quality and color consistency. As for preflighting, the parameters are passed in the body of the POST call.

Request URL (POST):

```
/pdf2image
```

Example parameters in the body:

```
{
  "inputFileURL": "url",
  "reference": "abc",
  "prerenderActionListURLs": "[url]",
  "restrictingActionListURL": "url",
  "variableSetURL": "url",
  "extraFontsFolderURL": "[url]",
  "jobStatusURL": "http://localhost:8080/result",
  "imageProperties": {
    "type": "JPEG",
    "qualityJPEG": "Medium",
    "progressiveJPEG": false,
    "interlacedPNG": false,
    "colorSpace": "DeviceRGB",
    "backgroundColor": [
      0,
      100,
      50
    ]
  },
  "embedICCProfile": false,
  "pageRange": "1-5",
  "includeEmptyPages": false,
  "exportOption": "Composite",
  "renderArea": {
    "rectangle": {
      "minX": 5.00008,
      "minY": 4.00008,
      "maxX": 10.00008,
      "maxY": 8.00008
    },
    "pageBox": "TrimBox"
  },
  "imageResolution": 72,
  "imageSize": {
    "width": 300,
    "height": 300
  },
  "antiAliasing": false,
  "output": {
    "targetLocation": "url"
  }
}
```

Note: For more information about the parameters, refer to *Parameters overview / pdf2image* on page 54.

Input files (inputFileURL, prerenderActionListURLs, restrictingActionListURL, variableSetURL, jobStatusURL, extraFontsFolderURL) should be available through URLs that are accessible for the Docker container. Be careful with spaces. Depending on the webserver, you should replace spaces with the URL encoded value of %20. It's also possible to work with local files if you mount a folder in the volumes section of the docker-compose.yml file.

Output parameter is a zip file that contains all the generated images. It will be stored in the given location with a HTTP PUT command. Typically, pre-signed URLs are used for that, but any URL supporting PUT operations will work fine. You can also write files to a local folder (see *Using local files instead of URLs*).

The **jobStatusURL** will be used to post the result of the job. This is a webhook that can be used as trigger for further processing. For more information, refer to *Progress of a job* on page 24.

prerenderActionListURLs is a JSON array of URLs (using [] and a comma (,) as separator).

More information about the parameters can be found in the Swagger documentation when the image is running locally: *http://localhost:3000/pdf2image/api-docs/*

When adding the job, you will get a response as follows:

```
{
    "id": "b5014a3b-e89a-4792-b269-6808e3180f26",
    "message": "Job received "
}
```

The id can be used in the */pdf2image/progress* API call and the */pdf2image/abort* API call.

Note: All of the methods (progress, abort...) detailed in *Processing files* on page 22 also apply to the /pdf2image endpoint. We only isolated pdf2image for a better understanding of the specific body parameters.

10. Preflightthenimage

The 2415 version of the PLC brings a new endpoint, named “/preflightthenimage”. Calling this method will ensure that both the preflight and the image generation are run alongside.

It will first operate the preflight of the file, then the image generation. If the Preflight fails due to a damaged PDF file for example, the image is not generated. Note that the number of transactions logged will vary from 2 (when all requests were fine) to 0 (if none of the requests work).

To use this method, define your JSON Body request like this example below:

```
{
  "inputFileURL": "url",
  "outputFixedFileURL": "url",
  "reference": "string",
  "profileURL": "url",
  "actionListURLs": "[url]",
  "variableSetURL": "url",
  "jobTicketURL": "url",
  "extraFontsFolderURL": "[url]",
  "jobStatusURL": "http://localhost:8080/result",
  "allowFixes": true,
  "reportProgress": true,
  "progressMinFraction": 0.05,
  "reportURLs": {
    "JSON": "string",
    "XML": "string",
    "PDF": "string"
  },
  "reportTemplate": {
    "configFileURL": "string",
    "templateFileURL": "string"
  },
  "reportLanguage": "enUS",
  "maxItemsPerCategory": 100,
  "maxNumOccurrencesPerItem": 100,
  "colorManagement": {
    "images": {
      "sourceProfiles": {
        "profileGray": {
          "url": "presigned url of Generic Enfocus gray.icm"
        },
        "profileRGB": {
          "url": "presigned url of Generic Enfocus RGB.icm"
        },
        "profileCMYK": {
          "url": "presigned url of Generic Enfocus CMYK.icm"
        },
        "profileLabPath": {
          "url": "presigned url of Generic Enfocus Lab.icm"
        },
        "intentOverrides": false
      },
      "targetProfiles": {
        "profileGray": {
          "url": "presigned url of Generic Enfocus gray.icm"
        },
        "profileRGB": {
```

```
          "url": "presigned url of Generic Enfocus RGB.icm"
        },
        "profileCMYK": {
          "url": "presigned url of Generic Enfocus CMYK.icm"
        },
        "profileLabPath": {
          "url": "presigned url of Generic Enfocus Lab.icm"
        },
        "intentOverrides": false
      },
      "renderingIntent": "objectDefined"
    },
    "blackPointCompensation": false
  },
  "flattening": {
    "rasterToVectorRatio": 100,
    "lineArtAndTextResolution": 1200,
    "gradientAndMeshResolution": 300,
    "textToOutlines": false,
    "strokesToOutlines": false,
    "clipComplexRegions": false,
    "preserveOverprint": true,
    "blendingColorSpace": {
      "path": "url",
      "name": "string"
    },
    "removeICCProfile": true,
    "recompressImages": {
      "colorImage": {
        "format": "JPEG",
        "quality": "4bit"
      },
      "grayscaleImage": {
        "format": "JPEG",
        "quality": "4bit"
      },
      "oneBitImage": {
        "format": "CCITT Group 3"
      },
      "asciiFilter": ""
    }
  },
  "restrictingActionListURL": "url",
  "imageProperties": {
    "type": "JPEG",
    "qualityJPEG": "Medium",
    "progressiveJPEG": false,
    "interlacedPNG": false,
    "colorSpace": "DeviceRGB",
    "backgroundColor": [
      0,
      100,
      50
    ]
  },
  "embedICCProfile": false,
  "pageRange": "1-5",
  "includeEmptyPages": false,
  "exportOption": "Composite",
  "renderArea": {
    "rectangle": {
      "minX": 5.00008,
      "minY": 4.00008,
      "maxX": 10.00008,
      "maxY": 8.00008
    },
    "pageBox": "TrimBox"
  },
  "imageResolution": 72,
  "imageSize": {
    "width": 300,
    "height": 300
  },
  "antiAliasing": false,
  "output": {
    "targetLocation": "url"
  }
```

```
}
```

To avoid conflicts, we introduce a new property:

```
{
  "output": {
    "preflightedFileURL": "url",
     "renderededFileURL":"url"
  }
```

But you can keep on using the output.targetLocation for /pdf2image and outputFixedFileURL for /job.

11. Debugging

When running the PitStop Library Container via command line, the logs are shown in the command line interface. However, you can also check the logs with the following command:

```
>> docker logs --details <container-name>
```

The container name is defined in the docker-compose.yml file.

12. PLC Node Libraries

If you are familiar with Node.js, we kindly encourage you to visit this GitHub repository:

https://github.com/EnfocusSW/pitstop-library-container

There we provide a set of useful classes for working with the PLC using Node.js. We may as well add more resources in the future. So don't hesitate in bookmarking this place and suggest additions.

This is an example of code you can find and reuse on our GitHub repository:

```
import { PLCJobOptions } from "pitstop-library-container/plc-job";

//create the job options and submit to the queue
const jobOptions: PLCJobOptions = {
  reference: "a string that can be used to identify the job",
  inputFileURL: "URL to the input file",
  actionListURLs: ["URL to Action List 1", "URL to Action List 2"],
  reportURLs: { JSON: "URL to JSON report" },
  outputFixedFileURL: "URL to preflighted file",
  reportProgress: false,
  reportLanguage: "itIT",
};
```

13. Deployment, queuing and scaling

Control over job queuing, scalability and distribution of Docker instances on one or multiple servers is in the hands of the developer. Various readily available tools can be utilized to help you with your implementation.

13.1. Running on AWS

See: *https://docker-curriculum.com/#docker-on-aws*

13.2. Running on Azure

See: *https://learn.microsoft.com/en-us/training/modules/run-docker-with-azure-container-instances/*

13.3. Running on Google Cloud

See: *https://cloud.google.com/run/docs/deploying*

13.4. AWS SQS

SQS is a queueing mechanism from AWS that allows jobs to be placed in a queue at any volume to prevent jobs from being lost. PitStop Library Container has native support for SQS and can be configured easily.

First, you need to create a credentials file. On Windows, this can be done by creating a file named: c:\users\<username>\.aws\credentials

Edit this file with Notepad and put the following in the file:

```
[default]
aws_access_key_id=<your_aws_access_key_id>
aws_secret_access_key=<your_aws_secret_access_key>
region=<your_region>
```

The region can be *eu-west-1* for instance.

Second, you need to configure the PitStop Library Container so that it reads jobs from the AWS SQS queue. You can do this by editing your docker-compose.yml file and adding the following environment variables:

```
- COM_ENFOCUS_CLOUD_PREFLIGHT_SQS_NAME=<name_of_your_queue>
```

```
- COM_ENFOCUS_CLOUD_AWS_REGION=<region_of_your_queue>
```

You need to add these variables to the environment section.

Also, you need to link your AWS credentials file in the docker-compose.yml file by adding this line to the volumes section:

```
- ~/.aws/credentials:/root/.aws/credentials:ro
```

This makes sure that the PitStop Library Container can access the AWS SQS queue.

Third, you need to send jobs to the SQS queue. For this, you need to make use of the AWS SDK. Documentation can be found here: *https://docs.aws.amazon.com/index.html* (search for AWS SDK, and you will find the documentation for multiple programming languages).

Below is an example for PHP, but you can use other languages to do this as well.

```
// This loads the AWS SDK. Make sure to download it and put it in this dir.
require 'aws/aws-autoloader.php';

use Aws\Sqs\SqsClient;
use Aws\Exception\AwsException;

$sharedConfig = array(
    'profile' => 'default',
    'region' => 'eu-west-1',  // don't forget to change this to your region!
    'version' => 'latest',
     'aws_access_key_id' => '...',    // fill in your credentials!
     'aws_secret_access_key' => '...'  // fill in your credentials!
);

$client = new SqsClient($sharedConfig);

try {
     $params = array(
          'DelaySeconds' => 10,
          'MessageBody' => '
                {
                "reference": "...",// fill in the correct variables
                "inputFileURL": "...",
                "profileURL": "...",
                "outputFixedFileURL": "...",
                "reportURLs": {
                           "JSON":"..."
                     }
                }
                ',
          'QueueUrl' => '<your queue url>'// the url of your sqs queue
     );
     $result = $client->sendMessage($params);
     var_dump($result);

} catch (AwsException $e) {
    // output error message if fails
     echo $e->getMessage();
}
```

More information on how to set up an SQS queue and send jobs can be found at *https://aws.amazon.com/sqs/*

13.5. Docker Swarm

With Docker Swarm, you can set up scaling.

More information can be found at *https://docs.docker.com/engine/swarm/*

Note: You need the correct license to allow for scaling.

13.6. Adjusting Workers Count dynamically

Until PLC 2409, the only solution for increasing or reducing the number of workers is to shut down the instance(s), edit the yml file and restart the instance.

To remove this hassle, we are thrilled to introduce a new method for dynamic allocation of workers. Now, you can adjust the number of workers with a single API Call with the following method:

```
{{url}}//ChangeWorkersCount

and set the body request to the following:
{
"count":5
}
```

The request returns an immediate answer. If you set a count greater than the number of current workers, then workers should be added instantaneously. If you set a count lower than the number of current workers, then the PLC will close the necessary amount of workers when they are not busy anymore.

In other words, if you have 6 workers and 4 workers idle, and you ask to set count to 2, then all 4 idle workers are closed immediately.

In the case that among those 6 workers, 5 are busy, PLC will close 1 immediately and will later close 3 more. It's important to remind this as a call to the /alive method may report a number of workers different than the one expected.

Remember that the main worker will never be closed even when idle.

When a change of workers is required to better adapt to production peaks, call the following endpoint:

```
{{url}}//ChangeWorkersCount
```

and set the body request to the following:

```
{
 "count":5
}
```

Note that some rules apply:

1. The master instance cannot be stopped.
2. If a worker is currently processing a request, it cannot be stopped.

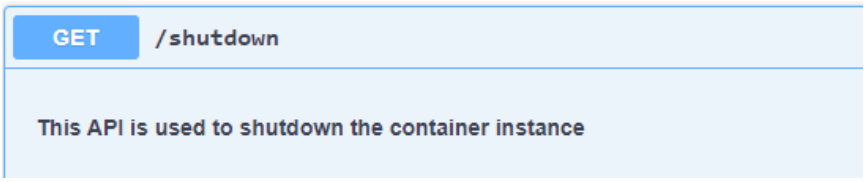
Therefore, the number of alive workers after the call may remain higher than the expected number. For example, considering that 4 workers are running, with 3 currently processing

requests: if an attempt is made to reduce the worker count to 2, it will only stop the one worker that is free.

However, if the worker is the master instance, no worker will be decreased (you will always have a minimum of one alive worker). We will improve this feature in a future release to warrant that the exact required number of workers is found after API call.

13.7. Disabling an instance remotely

Starting PLC 2421, you can shut down an instance remotely using the /shutdown method.



For example, if you started an instance on a server of which the URL is *https://my.server:12345*, you can disable it calling *https://my.server:12345/shutdown*

The following json message should be returned:

```
{
  "statusCode": 202,
  "message": "Request received for shutting down the PLC instance"
 }
```

If you need to disable multiple instances, then you need to call the /shutdown method on each instance. Note that this is an asynchronous call: the PLC will first wait until any ongoing process comes to an end before shutting down the instance.

To monitor when the instance is actually shut down, you may need to call the /isAlive method until it returns an error.

Note: PLEASE check that you added the following line under “volumes” in your yml file (see *Configuring the PitStop Library Container* on page 10)

```
- /var/run/docker.sock:/var/run/docker.sock
```

This is required to properly handle the shutdown of instances!

14. Resource constraints

By default, a container has no resource constraints and can use as much of a given resource as the host's kernel scheduler allows.

However, you can configure the resources as you wish. More information can be found at *https://docs.docker.com/config/containers/resource_constraints/*

14.1. Statistics

To help monitoring PLC health, statistics info has been added in the logs messages when a request is executed:

```
"cpu":          "CPU utilization in percentage(Example: if it is 1 , 1% of CPU is consumed)",
"memory":       "Memory utilization in bytes",
"InstanceId":   "The id of the instance",
"JobID":        "The id of the job",
"TimeConsumed": "Total processing time to complete the request"
```

15. Annex

15.1. Example files

Available for download:

- An example *docker-compose.yml file* that you can use. You still need to fill in your credentials and region.
- A *postman collection file* that you can use to test.

15.2. Hello World(s)

To help you get a quick up and running use case, we happily provide some simple use cases. Following those steps must deliver the expected results and help you debugging any issues you would face when starting working with PLC on your own files and set-up.

Note that we don't cover any of the technical steps to download PLC image, install Docker Desktop and configure the YAML file for running PLC. This was already discussed earlier in this manual. If your PLC can't run at this stage, be sure to review the installation process before you try running those examples.

Sample files are provided for convenience. You can download them *here*.

Ensure that you get those files to reproduce those examples:

Note: We consider all files to be local and accessible through YAML configuration like this:

```
/Some/local/Folder:/root/
```

The following examples are provided to help you getting the PLC to work. If you could run those samples and get the expected results, you know that everything should now work properly. Keep in mind that many parameters can be set on top of what we use. Find the full list in the *annex*.

15.2.1. Get a simple preflight JSON report

The idea here is to run a simple preflight and get a report back.

Required files:

- Preflight Profile: “my_preflight_profile.ppp”
- Sample PDF File: “my_pdf_sample.pdf”

The following JSON body must result in:

- The processed document in the output folder
- The JSON report in the output folder

Send the following body to the /preflight endpoint:

```
{
    "reference": "abc123",
    "inputFileURL": "/root/examples/input/my_pdf_sample.pdf",
    "profileURL": "/root/examples/01_simple_preflight/resources/
my_preflight_profile.ppp",
    "outputFixedFileURL": "/root/examples/output/my_pdf_sample_preflighted.pdf",
    "maxItemsPerCategory": "100",
    "maxNumOccurencesPerItem": "100",
    "reportLanguage": "en-US",
    "reportURLs": {
        "JSON": "/root/examples/output/my_json_report.json"
    }
}
```

15.2.2. Get a PDF preflight report

The idea here is to return a PDF report using a default Report template:

Required Files:

- Preflight Profile: “my_preflight_profile.ppp”
- Sample PDF File: “my_pdf_sample.pdf”
- PDF Report prefs: “PitStopReportDefault.prefs”
- PDF Report template: “PitStopReportDefault.pdf”

The following JSON body must result in:

- The processed document in the output folder
- The PDF report in the output folder

Send the following body to the /preflight endpoint:

```
{
    "reference": "abc123",
    "inputFileURL": "/root/examples/input/my_pdf_sample.pdf",
    "profileURL": "/root/examples/01_simple_preflight/resources/
my_preflight_profile.ppp",
    "outputFixedFileURL": "/root/examples/output/my_pdf_sample_preflighted.pdf",
    "maxItemsPerCategory": "100",
    "maxNumOccurencesPerItem": "100",
    "reportLanguage": "en-US",
    "reportTemplate": {
```

```
        "configFileURL": "/examples/02_PDF_DEFAULT_REPORT/templates/Default/
PitStopReportDefault.prefs",
        "templateFileURL": "/examples/02_PDF_DEFAULT_REPORT/templates/Default/
PitStopReportDefault.pdf"
    },
    "reportURLs": {
        "PDF": "/root/examples/output/my_json_report.pdf"
    }
}
```

15.2.3. Get a customized PDF report

The idea here is to return a PDF report using a customized report template:

Note: All info on customizing a PDF report can be found in *the PitStop Report Templates manual*.

Required Files:

- Preflight Profile: “my_preflight_profile.ppp”
- Sample PDF File: “my_pdf_sample.pdf”
- PDF Report prefs: “MyCompanyReport.prefs”
- PDF Report template: “MyCompanyReport.pdf”

The following JSON body must result in:

- The processed document in the output folder
- The Customized PDF report in the output folder

Send the following body to the /preflight endpoint:

```
{
    "reference": "abc123",
    "inputFileURL": "/root/examples/input/my_pdf_sample.pdf",
    "profileURL": "/root/examples/01_simple_preflight/resources/
my_preflight_profile.ppp",
    "outputFixedFileURL": "/root/examples/output/my_pdf_sample_preflighted.pdf",
    "maxItemsPerCategory": "100",
    "maxNumOccurencesPerItem": "100",
    "reportLanguage": "en-US",
    "reportURLs": {
        "JSON": "/root/examples/output/my_json_report.json",
        "PDF": "/root/examples/output/my_json_report.pdf"
    },
    "reportTemplate": {
    "configFileURL": "/root/examples/03_PDF_CUSTOM_REPORT/templates/Custom/
MyCompanyReport.prefs",
    "templateFileURL": "/root/examples/03_PDF_CUSTOM_REPORT/templates/Custom/
MyCompanyReport.pdf"
  }
}
```

15.2.4. Convert document to Gray

The idea here is to convert a PDF to gray using an Action List:

Required Files:

- Action List: “convert_to_gray.eal”
- Sample PDF File: “my_pdf_sample.pdf”

The following JSON body must result in:

- The gray converted document in the output folder

Send the following body to the /preflight endpoint:

```
{
"reference": "abc123",
"inputFileURL": "/root/examples/input/my_pdf_sample.pdf",
"outputFixedFileURL": "/root/examples/output/my_pdf_sample_converted_to_gray.pdf",
"actionListURLs":["/root/examples/04_CONVERT_TO_GRAY/convert_to_gray.eal"]
}
```

15.2.5. Generate a JPG file

The idea here is to generate a JPG from the supplied PDF.

Required Files:

- ·Sample PDF File: “my_pdf_sample.pdf”

The following JSON body must result in:

- A zip file found in the output folder containing the generated image.

Send the following body to the /pdf2image endpoint:

```
{
  "inputFileURL": "/root/examples/input/my_pdf_sample.pdf",
  "reference": "JOB123",
  "imageProperties": {
    "type": "JPEG",
    "qualityJPEG": "Medium",
    "progressiveJPEG": false,
    "colorSpace": "DeviceRGB",
    "pageRange": "1"
  },
  "imageSize": {
    "width": 600
  },
  "output": {
    "targetLocation": "/root/examples/output/images.zip"
  }
}
```

15.3. Parameters overview /preflight parameters

Parameter & Type	Required?	Description
inputFileURL		
String	Required	The URL of the input file to be processed *Example: inputFileURL:“/some/location/input.pdf”*
outputFixedFileURL		
String	Required	The URL of the output file once processed

Parameter & Type	Required?	Description
		Example: outputFixedFileURL:"/some/location/output.pdf"
reference		
String	Required	Any value used as a reference *Example: reference:"JOB1334"*
profileURL		
String	If no Action Lists are set in the "actionListURLs" parameter	The URL of the Preflight Profile file to be used for preflighting the file *Example: profileURL:"/some/location/preflight.ppp"*
actionListURLs		
Array	Required if no Preflight Profile is defined	An array of strings that define URLs of Action Lists to be used *Example: actionListURLs:["/some/location/action.eal"]*
variableSetURL		
String	If Action Lists or a Preflight Profile is defined using variables. Note that errors will only be thrown at execution time.	The URL of the Variables Set file *Example: variableSetURL:"/some/location/variables.evs"*
jobTicketURL		
String	If Action Lists or a Preflight Profile is defined using variables linked to a Job Ticket. Note that errors will only be thrown at execution time.	The URL of the XML Job Ticket file. *Example: jobTicketURL:"/some/location/jobticket.xml"*
extraFontsFolderURL		
Array	Optional	An array containing URLs where extra fonts can be searched for during fonts related operations such as embedding fonts *Example: extraFontsFolderURL:["/some/location/fonts"]*
jobStatusURL		
String	Required if "reportProgress" is set to true	The URL that will receive progress messages from PLC during execution *Example: jobStatusURL:"http://localhost:8080/result"*
allowFixes		
Boolean	Optional, true by default	Set to false to prevent any automatic fixes when preflighting a file

Parameter & Type	Required?	Description
		Example: allowFixes:false
reportProgress		
Boolean	Optional, false by default	Set to true if the PLC must report progress during the job execution *Example: reportProgress:true*
progressMinFraction		
Number	Required if "reportProgress" is set to true	Set a value between 0 and 1 to set the frequency for sending progress messages. 0.5 would mean having a message at the middle of the job execution, 1 means when job is completed *Example: progressMinFraction:0.25*
reportURLs		
JSON	Required if a Preflight Profile is defined in the "profileURL"parameter	A JSON object that contains at least one report type set among JSON, XML a/o PDF *Example:* *reportURLs:{* *...* *}*
reportURLs/*JSON*		
String	Optional	The URL where to save the JSON report file *Example: JSON:"/some/location/report.json"*
reportURLs/*XML*		
String	Optional	The URL where to save the XML report file *Example: XML:"/some/location/report.xml"*
reportURLs/*PDF*		
String	Optional	The URL where to save the PDF report file *Example: PDF:"/some/location/report.pdf"*
reportTemplate		
JSON	Required if a Preflight Profile is defined in the "profileURL"parameter AND a PDF report file is requested in the reportURLs/PDF parameter	A JSON object that contains two properties: • configFileURL • templateFileURL *Example:* *reportTemplate: {* *...*

Parameter & Type	Required?	Description
		},
reportTemplate/*configFileURL*		
String	Required if a preflight profile is defined in the “profileURL”parameter AND a PDF report file is requested in the reportURLs/PDF parameter	The URL to the config file *Example: configFileURL: "/some/location/template.prefs"*
reportTemplate/*templateFileURL*		
String	Required if a Preflight profile is defined in the “profileURL”parameter AND a PDF report file is requested in the reportURLs/PDF parameter	The URL to the PDF template file *Example: templateFileURL: "/some/location/template.pdf"*
reportLanguage		
String	Required if a Preflight Profile is defined in the “profileURL” parameter	The language that should be used for the report. The attribute is a 4-character string in the form 'enUS', 'frFR', 'deDE',... English is always available *Example: reportLanguage:'deDE'*
maxItemsPerCategory		
Number	Optional	A number that defines the maximum number of items reported per category. Default is 100 *Example: maxItemsPerCategory:50*
maxNumOccurrencesPerItem		
Number	Optional	A number that defines the maximum number of occurences reported per Item. Default is 100 *Example: maxNumOccurrencesPerItem:80*
colorManagement		
JSON	Optional	A JSON object that contains color managements settings *Example:* *colorManagement:{* *...* *}*
colorManagement/*images*		

Parameter & Type	Required?	Description
JSON	Optional	A JSON object that contains color managements settings for images specifically *Example:* *Images:{* *...* *}*
colorManagement/images/*sourceProfiles*		
JSON	Optional	A JSON object that contains properties for defining the source ICC profiles *Example:* *sourceProfiles:{* *...* *}*
colorManagement/images/sourceProfiles/*profileGray*		
JSON	Optional	A JSON object that contains a "url" property *Example:* *profileGray:{* *...* *}*
colorManagement/images/sourceProfiles/profileGray/*url*		
String	Optional	The URL to the ICC profile for gray images *Example: url:"/some/location/gray.icc"*
colorManagement/images/sourceProfiles/*profileRGB*		
JSON	Optional	A JSON object that contains a "url" property *Example:* *profileRGB:{* *...* *}*
colorManagement/images/sourceProfiles/profileRGB/*url*		
String	Optional	The URL to the ICC profile for RGB images *Example: url:"/some/location/rgb.icc"*
colorManagement/images/sourceProfiles/profileCMYK		
JSON	Optional	A JSON object that contains a "url" property

Parameter & Type	Required?	Description
		Example: *{* *...* *}*
colorManagement/images/sourceProfiles/profileCMYK/*url*		
String	Optional	The URL to the ICC profile for CMYK images *Example: url:"/some/location/cmyk.icc"*
colorManagement/images/sourceProfiles/*profileLabPath*		
JSON	Optional	A JSON object that contains a "url" property *Example: profileLabPath:{* *...* *}*
colorManagement/images/sourceProfiles/profileLabPath/*url*		
String	Optional	The URL to the ICC profile for LAB images *Example: url:"/some/location/lab.icc"*
colorManagement/images/sourceProfiles/*intentOverrides*		
Boolean	Optional, false by default	Set to true if the outputIntent set in the input file takes precedence over sourceProfiles settings. *Example: intentOverrides:true*
colorManagement/images/*targetProfiles*		
JSON	Optional	A JSON object that contains properties for defining the source ICC profiles *Example:* *targetProfiles:{* *...* *}*
colorManagement/images/targetProfiles/*profileGray*		
JSON	Optional	A JSON object that contains a "url" property *Example:* *profileGray:{* *...* *}*
colorManagement/images/targetProfiles/profileGray/*url*		

Parameter & Type	Required?	Description
String	Optional	The URL to the ICC profile for gray images *Example: url:"/some/location/gray.icc"*
colorManagement/images/targetProfiles/*profileRGB*		
JSON	Optional	A JSON object that contains a "url" property *Example:* *profileRGB:{* *...* *}*
colorManagement/images/targetProfiles/profileRGB/*url*		
String	Optional	The URL to the ICC profile for RGB images *Example: url:"/some/location/rgb.icc"*
colorManagement/images/targetProfiles/*profileCMYK*		
JSON	Optional	A JSON object that contains a "url" property *Example:* *{* *...* *}*
colorManagement/images/targetProfiles/profileCMYK/*url*		
String	Optional	The URL to the ICC profile for CMYK images *Example: url:"/some/location/cmyk.icc"*
colorManagement/images/targetProfiles/*profileLabPath*		
JSON	Optional	A JSON object that contains a "url" property *Example:* *profileLabPath:{* *...* *}*
colorManagement/images/targetProfiles/profileLabPath/*url*		
String	Optional	The url to the ICC profile for LAB images *Example: url:"/some/location/lab.icc"*
colorManagement/images/targetProfiles/*intentOverrides*		
Boolean	Optional, false by default	Set to true if the OutputIntent set in the input file takes precedence over sourceProfiles settings.

Parameter & Type	Required?	Description
		Example: intentOverrides:true
colorManagement/images/*renderingIntent*		
String	Optional	The output rendering intent; one of: • **ObjectDefined** (default): the rendering intent of the object itself, as shown on the Prepress tab of the PitStop Inspector • **Relative**: in-gamut colors are reproduced exactly; out-of-gamut colors are mapped to the nearest value within the reproducible gamut; corrections are made for the output medium's white point. • **Absolute**: in-gamut colors are reproduced exactly; out-of-gamut colors are mapped to the nearest value within the reproducible gamut; no correction is made for the output medium's white point. • **Saturation**: colors are represented in a manner that preserves or emphasizes saturation. • **Perceptual**: colors are represented in a manner that provides a pleasing perceptual appearance. *Example: renderingIntent:"ObjectDefined"*
colorManagement/*blackPointCompensation*		
Boolean	Optional, default is false	Set to true to put Black Point compensation on. *Example: blackPointCompensation:false*
flattening		
JSON	Optional	Set options for flattening PDF files *Example:* *Flattening:{* *...* *}*
flattening/*rasterToVectorRatio*		
Number	Optional	The raster/vector balance: a value from 0 to 100. This controls the degree of rasterization of complex areas in the document. If set to 100, no areas in the document will get rasterized for performance reasons (although some areas may still be rasterized for other reasons). The default is 100. *Example: rasterToVectorRatio:100*
flattening/*lineArtAndTextResolution*		

Parameter & Type	Required?	Description
Number	Optional	The resolution for rasterizing vectors, text, and gradients. This is a value from 1 to 9600. The default is 1200. *Example: lineArtAndTextResolution:1200*
flattening/*gradientAndMeshResolution*		
Number	Optional	The resolution for gradients and Illustrator mesh objects after rasterization because of flattening. This is a value from 1 to 1200. The default is 300. *Example: gradientAndMeshResolution:300*
flattening/*textToOutlines*		
Boolean	Optional	If present and true, all strokes are converted to filled paths. *Example: textToOutlines:true*
flattening/*strokesToOutlines*		
Boolean	Optional	If present and true, all text is converted to outlines and all type glyph information is discarded. *Example: strokesToOutlines:true*
flattening/*clipComplexRegions*		
Boolean	Optional	If not present or true, this option ensures that the boundaries between vector artwork and rasterized artwork fall along object paths. This option reduces stitching artifacts as a result of a part of an object being rasterized while another part of the object remains in vector form. *Example: clipComplexRegions:false*
flattening/*preserveOverprint*		
Boolean	Optional	If not present or true, overprint is preserved where possible. If this option is disabled, all overprint in the document will be simulated by blending the colors of overprinted objects. *Example: preserveOverprint:true*
flattening/*blendingColorSpace*		
JSON	Optional	A JSON object that defines settings for the blending color space to be used for flattening. *Example:* *blendingColorSpace:{* *...* *}*

Parameter & Type	Required?	Description
flattening/blendingColorSpace/*path*		
String	Required if the blendingColorSpace is defined	The path to a valid ICC profile to be used for blending. *Example: path:"/some/location/blending.icc"*
flattening/blendingColorSpace/*name*		
String	Required if the blendingColorSpace is defined	One of: Current (the default): use the current blending color space if present: • None: don't use a blending color space • DeviceGray • DeviceRGB • DeviceCMYK • ICCProfile: A custom ICC Profile *Example: name:"DeviceGray"*
flattening/*removeICCProfile*		
Boolean	Optional	If not present or true, the ICC profiles are removed after flattening. *Example: removeICCProfile:true*
flattening/*recompressImages*		
JSON	Optional	A JSON object to define settings for images recompression *Example:* *recompressImages:{* *...* *}*
flattening/recompressImages/*colorImage*		
JSON	Optional	A JSON object to define settings for compressing color images *Example:* *colorImage:{* *...* *}*
flattening/recompressImages/colorImage/*format*		
String	Required if recompressing color images is defined	One of the following values: • ZIP • JPEG

Parameter & Type	Required?	Description
		Example: format:"JPEG"
flattening/recompressImages/colorImage/*quality*		
String	Required if recompressing color images is defined	One of the following: When ZIP was defined in the format: • 4bit • 8bit When JPEG was defined in the format: • Minimum • Low • Medium • High *Example: quality:"Minimum"*
flattening/recompressImages/*grayscaleImage*		
JSON	Optional	A JSON object to define settings for compressing grayscale images *Example:* *grayscaleImage:{* *...* *}*
flattening/recompressImages/grayscaleImage/*format*		
JSON	Required if recompressing grayscale images is defined	One of the following values: • ZIP • JPEG *Example: format:"JPEG"*
flattening/recompressImages/grayscaleImage/*quality*		
String	Required if recompressing grayscale images is defined	One of the following: When ZIP was defined in the format: • 4bit • 8bit When JPEG was defined in the format: • Minimum • Low • Medium • High *Example: quality:"Minimum"*

Parameter & Type	Required?	Description
flattening/recompressImages/*oneBitImage*		
JSON	Optional	A JSON object to define settings for compressing 1bit images *Example:* *oneBitImage:{* *...* *}*
flattening/recompressImages/oneBitImage/*format*		
String	Required if recompressing 1bit images is defined	One of the following values: • ZIP • CCITTGroup3 • CCITTGroup4 • RunLength *Example: format:"CCITTGroup3"*
flattening/recompressImages/*asciiFilter*		
String	Optional	One of the following values: • None • ASCIIHex • ASCII85 *Example: asciiFilter:"ASCIIHex"*

15.4. Parameters overview /pdf2image

Parameter & Type	Required?	Description
inputFileURL		
String	Required	The URL of the input file to be processed *Example: inputFileURL:"/some/location/input.pdf"*
reference		
String	Optional	Any value used as a reference *Example: reference: "JOB1334"*
prerenderActionListURLs		
Array	Optional	A list of Action List URLs to be run before generating images. *Example:*

Parameter & Type	Required?	Description
		prerenderActionListURLs:[*“/some.location/prerender.eal”* *]*
restrictingActionListURL		
String	Optional	The path to an Action List used as a restriction. *Example: restrictingActionListURL:”restriction.eal”*
variableSetURL		
String	If defined Action Lists or a Preflight Profile is defined using variables. Note that errors will only be thrown at execution time.	The URL of the Variables Set file *Example: “/some/location/variables.evs”*
extraFontsFolderURL		
Array	Optional	An array containing URLs where extra fonts can be searched for during fonts related operations such as embedding fonts. *Example: extraFontsFolderURL:[“/some/location/fonts”]*
jobStatusURL		
String	Required if “reportProgress” is set to true.	The URL that will receive progress messages from PLC during execution. *Example: jobStatusURL:“http://localhost:8080/result”*
imageProperties		
JSON	Json	A JSON object used to define properties for image generation *Example:* *imageProperties:{* *...* *}*
imageProperties/type		
String	Optional	One of the following: • PNG • JPEG *Example: type:“JPEG”*
imageProperties/*qualityJPEG*		

Parameter & Type	Required?	Description
String	Optional but only applies if imageProperties/type is JPEG.	One of the following: • Minimum • Low • Medium (default) • High • Maximum *Example: qualityJPEG:"Low"*
imageProperties/*progressiveJPEG*		
Boolean	Optional but only applies if imageProperties/type is JPEG.	Set to true to generate a progressive JPEG file. *Example: progressiveJPEG:true*
imageProperties/*interlacedPNG*		
Boolean	Optional but only applies if imageProperties/type is PNG.	Set to true to generate an interlaced PNG file. *Example: interlacedPNG:false*
imageProperties/*colorSpace*		
String	Optional	One of the following: • DeviceGray • DeviceRGB • DeviceCMYK *Example: colorSpace:"DeviceRGB"*
imageProperties/*backgroundColor*		
Array	Optional	An array of one to four numbers depending on the color space. • If DeviceGray is used, you only need one value from 0 to 100 ([50]). • If DeviceRGB is used, you need 3 numbers from 0 to 255 ([120,36,48]). The first number is the Red channel, the second the Green channel, the last one the Blue channel. • If DeviceCMYK is used, you need 4 numbers from 0 to 100 ([0,25,61,11]). The first one is the Cyan channel, the second one is the Magenta channel, the third one is the Yellow channel and the last one is for the Black channel. *Example: backgroundColor:[0,0,0]*
embedICCProfile		
Boolean	Optional	Set to true to get the ICC profile embedded into the resulting image. *Example: embedICCProfile:true*

Parameter & Type	Required?	Description
includeEmptyPages		
Boolean	Optional	Set to false to discard blank pages from image generation. *Example: includeEmptyPages:true*
pageRange		
String	Optional	A string that is a valid range for pages (ex: 1 or 2-3). *Example: pageRange:"1,3,6-8"*
exportOption		
String	Optional	One of the following: • Composite (default): renders the page as a whole • Separations: renders the various separations as separate images *Example: exportOption:"Separations"*
renderArea		
JSON	Optional	A JSON object that defines the rendering area. If defined, the object must contain either a rectangle or a pageBox property. *Example:* *renderArea:{* *...* *}*
renderArea/*rectangle*		
JSON	Optional	A JSON object that includes four properties: minX, minY, maxX and maxY. *Example:* *rectangle:{* *...* *}*
renderArea/rectangle/**minX**		
Number	Required if rectangle property is defined.	A number that represents the x value for the lower left corner of the rectangle used for rendering the image. *Example: minX:5.425*
renderArea/rectangle/*minY*		

Parameter & Type	Required?	Description
Number	Required if rectangle property is defined.	A number that represents the y value for the lower left corner of the rectangle used for rendering the image. *Example: minY:3.1*
renderArea/rectangle/*maxX*		
Number	Required if rectangle property is defined.	A number that represents the x value for the top right corner of the rectangle used for rendering the image. *Example: maxX:5.425*
renderArea/rectangle/*maxY*		
Number	Required if rectangle property is defined.	A number that represents the y value for the top right corner of the rectangle used for rendering the image. *Example: maxY:3.1*
renderArea/*pageBox*		
String	Required if the rectangle property is not defined but a renderArea is defined.	One of the following: • TrimBox • MediaBox • BleedBox • ArtBox • CropBox *Example: pageBox:"TrimBox"*
imageResolution		
Number	Optional	A number for the resulting image resolution. If a resolution is set, setting an imageSize is not necessary. *Example: imageResolution:144*
imageSize		
JSON	Optional	A JSON object that contains a width a/or a width property. If imageSize is set, setting image resolution is not necessary. *Example:* *imageSize:{* ... *}*
imageSize/*width*		
Number	Required if imageSize parameter is set.	A number that represents the expected image width in pixels. *Example: width:360*

Parameter & Type	Required?	Description
imageSize/*height*		
Number	Required if imageSize parameter is set.	A number that represents the expected image height in pixels. *Example: height:360*
antialiasing		
Boolean	Optional	Set to true to enable antialiasing. *Example: antialiasing:true*
output		
JSON	Required	A JSON object that contains a targetLocation property. *Example:* *output:{* *...* *}*
output/*targetLocation*		
String	Required	The URL of the folder where the resulting ZIP file that contains generated images will be saved. *Example: targetLocation:"/some/location"*

15.5. Errors troubleshooting

For an overview of the main errors, please download the errors troubleshooting guide *here*.

16. Third-Party License Information

```
    This product includes lzw-ab.

    Copyright (c) David Bryant
    All rights reserved.

    Redistribution and use in source and binary forms, with or without
    modification, are permitted provided that the following conditions are met:

    * Redistributions of source code must retain the above copyright notice,
      this list of conditions and the following disclaimer.
    * Redistributions in binary form must reproduce the above copyright notice,
      this list of conditions and the following disclaimer in the
      documentation and/or other materials provided with the distribution.
    * Neither the name of Conifer Software nor the names of its contributors
      may be used to endorse or promote products derived from this software
      without specific prior written permission.

    THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
    AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
    IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
    ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR
    ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
    DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
    SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
    CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
    OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
    OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

------------------------------------------------------------

This product includes Botan.

      Copyright (C) 1999-2022 The Botan Authors
      All rights reserved.

      Redistribution and use in source and binary forms, with or without
      modification, are permitted provided that the following conditions are met:

      1. Redistributions of source code must retain the above copyright notice,
         this list of conditions, and the following disclaimer.

      2. Redistributions in binary form must reproduce the above copyright
         notice, this list of conditions, and the following disclaimer in the
         documentation and/or other materials provided with the distribution.

      THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
      AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
      IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
      ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE
      LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
      CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
      SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
      INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
      CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
      ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
      POSSIBILITY OF SUCH DAMAGE.

------------------------------------------------------------

This product includes ICC Profiles.

    Some ICC Profiles were created by FFEI Ltd. (www.ffei.co.uk) using Fujifilm
    ColourKit Profiler Suite (www.colourprofiling.com)

------------------------------------------------------------

This product includes ICC Profiles.
```

```
    Some ICC profiles are copyright (C) by European Color Initiative, www.eci.org

------------------------------------------------------------

This product includes ICC Profiles.

    Some ICC profiles are copyright (C) of WAN-IFRA, www.wan-ifra.org

------------------------------------------------------------

This product includes ICC Profiles.

    Some ICC profiles are copyright (C) IDEAlliance(R). G7(R), GRACol(R) and
    SWOP(R) are all registered trademarks of IDEAlliance(C).

------------------------------------------------------------

This product includes PANTONE Color Libraries.

    PANTONE© and other Pantone trademarks are the property of Pantone LLC.
    Pantone is a wholly owned subsidiary of X-Rite, Incorporated.

------------------------------------------------------------

This product includes curl.

   COPYRIGHT AND PERMISSION NOTICE

   Copyright (c) 1996 - 2023, Daniel Stenberg, <daniel@haxx.se>, and many
   contributors, see the THANKS file.

   All rights reserved.

   Permission to use, copy, modify, and distribute this software for any purpose
   with or without fee is hereby granted, provided that the above copyright
   notice and this permission notice appear in all copies.

   THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
   IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
   FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN
   NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM,
   DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR
   OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE
   OR OTHER DEALINGS IN THE SOFTWARE.

   Except as contained in this notice, the name of a copyright holder shall not
   be used in advertising or otherwise to promote the sale, use or other dealings
   in this Software without prior written authorization of the copyright holder.

------------------------------------------------------------

This product includes LibTIFF.

      Copyright (c) 1988-1997 Sam Leffler
      Copyright (c) 1991-1997 Silicon Graphics, Inc.

      Permission to use, copy, modify, distribute, and sell this software and
      its documentation for any purpose is hereby granted without fee, provided
      that (i) the above copyright notices and this permission notice appear in
      all copies of the software and related documentation, and (ii) the names of
      Sam Leffler and Silicon Graphics may not be used in any advertising or
      publicity relating to the software without the specific, prior written
      permission of Sam Leffler and Silicon Graphics.

      THE SOFTWARE IS PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND,
      EXPRESS, IMPLIED OR OTHERWISE, INCLUDING WITHOUT LIMITATION, ANY
      WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

      IN NO EVENT SHALL SAM LEFFLER OR SILICON GRAPHICS BE LIABLE FOR
      ANY SPECIAL, INCIDENTAL, INDIRECT OR CONSEQUENTIAL DAMAGES OF ANY KIND,
      OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS,
      WHETHER OR NOT ADVISED OF THE POSSIBILITY OF DAMAGE, AND ON ANY THEORY OF
      LIABILITY, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE
      OF THIS SOFTWARE.

------------------------------------------------------------
```

```
This product includes FreeType.

    Portions of this software are copyright (C) 2014 The FreeType
    Project (www.freetype.org) licensed under the Freetype License.
    All rights reserved.

------------------------------------------------------------

This product includes ICU.

      Copyright (c) 1991-2023 Unicode, Inc. All rights reserved.
      Distributed under the Terms of Use in https://www.unicode.org/copyright.html.

      Permission is hereby granted, free of charge, to any person obtaining
      a copy of the Unicode data files and any associated documentation
      (the "Data Files") or Unicode software and any associated documentation
      (the "Software") to deal in the Data Files or Software
      without restriction, including without limitation the rights to use,
      copy, modify, merge, publish, distribute, and/or sell copies of
      the Data Files or Software, and to permit persons to whom the Data Files
      or Software are furnished to do so, provided that either
      (a) this copyright and permission notice appear with all copies
      of the Data Files or Software, or
      (b) this copyright and permission notice appear in associated
      Documentation.

      THE DATA FILES AND SOFTWARE ARE PROVIDED "AS IS", WITHOUT WARRANTY OF
      ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE
      WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
      NONINFRINGEMENT OF THIRD PARTY RIGHTS.
      IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS
      NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL
      DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE,
      DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER
      TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR
      PERFORMANCE OF THE DATA FILES OR SOFTWARE.

      ---------------------------------------------------------------------

      Chinese/Japanese Word Break Dictionary Data (cjdict.txt)

       #     The Google Chrome software developed by Google is licensed under
       # the BSD license. Other software included in this distribution is
       # provided under other licenses, as set forth below.
       #
       #  The BSD License
       #  http://opensource.org/licenses/bsd-license.php
       #  Copyright (C) 2006-2008, Google Inc.
       #
       #  All rights reserved.
       #
       #  Redistribution and use in source and binary forms, with or without
       # modification, are permitted provided that the following conditions are met:
       #
       #  Redistributions of source code must retain the above copyright notice,
       # this list of conditions and the following disclaimer.
       #  Redistributions in binary form must reproduce the above
       # copyright notice, this list of conditions and the following
       # disclaimer in the documentation and/or other materials provided with
       # the distribution.
       #  Neither the name of  Google Inc. nor the names of its
       # contributors may be used to endorse or promote products derived from
       # this software without specific prior written permission.
       #
       #
       #  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND
       # CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES,
       # INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
       # MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
       # DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
       # LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
       # CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
       # SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR
       # BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF
       # LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
       # NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
```

```
# SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
#
#
#  The word list in cjdict.txt are generated by combining three word lists
# listed below with further processing for compound word breaking. The
# frequency is generated with an iterative training against Google web
# corpora.
#
#  * Libtabe (Chinese)
#    - https://sourceforge.net/project/?group_id=1519
#    - Its license terms and conditions are shown below.
#
#  * IPADIC (Japanese)
#    - http://chasen.aist-nara.ac.jp/chasen/distribution.html
#    - Its license terms and conditions are shown below.
#
#  ---------COPYING.libtabe ---- BEGIN--------------------
#
#  /*
#   * Copyright (c) 1999 TaBE Project.
#   * Copyright (c) 1999 Pai-Hsiang Hsiao.
#   * All rights reserved.
#   *
#   * Redistribution and use in source and binary forms, with or without
#   * modification, are permitted provided that the following conditions
#   * are met:
#   *
#   * . Redistributions of source code must retain the above copyright
#   *   notice, this list of conditions and the following disclaimer.
#   * . Redistributions in binary form must reproduce the above copyright
#   *   notice, this list of conditions and the following disclaimer in
#   *   the documentation and/or other materials provided with the
#   *   distribution.
#   * . Neither the name of the TaBE Project nor the names of its
#   *   contributors may be used to endorse or promote products derived
#   *   from this software without specific prior written permission.
#   *
#   * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
#   * "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
#   * LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS
#   * FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE
#   * REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
#   * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
#   * (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
#   * SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
#   * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
#   * STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
#   * ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
#   * OF THE POSSIBILITY OF SUCH DAMAGE.
#   */
#
#  /*
#   * Copyright (c) 1999 Computer Systems and Communication Lab,
#   *                    Institute of Information Science, Academia
#       *                    Sinica. All rights reserved.
#   *
#   * Redistribution and use in source and binary forms, with or without
#   * modification, are permitted provided that the following conditions
#   * are met:
#   *
#   * . Redistributions of source code must retain the above copyright
#   *   notice, this list of conditions and the following disclaimer.
#   * . Redistributions in binary form must reproduce the above copyright
#   *   notice, this list of conditions and the following disclaimer in
#   *   the documentation and/or other materials provided with the
#   *   distribution.
#   * . Neither the name of the Computer Systems and Communication Lab
#   *   nor the names of its contributors may be used to endorse or
#   *   promote products derived from this software without specific
#   *   prior written permission.
#   *
#   * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
#   * "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
#   * LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS
#   * FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE
#   * REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
#   * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
```

```
#   * (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
#   * SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
#   * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
#   * STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
#   * ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
#   * OF THE POSSIBILITY OF SUCH DAMAGE.
#   */
#
#  Copyright 1996 Chih-Hao Tsai @ Beckman Institute,
#      University of Illinois
#  c-tsai4@uiuc.edu  http://casper.beckman.uiuc.edu/~c-tsai4
#
#  ---------------COPYING.libtabe-----END--------------------------------
#
#
#  ---------------COPYING.ipadic-----BEGIN-------------------------------
#
#  Copyright 2000, 2001, 2002, 2003 Nara Institute of Science
#  and Technology.  All Rights Reserved.
#
#  Use, reproduction, and distribution of this software is permitted.
#  Any copy of this software, whether in its original form or modified,
#  must include both the above copyright notice and the following
#  paragraphs.
#
#  Nara Institute of Science and Technology (NAIST),
#  the copyright holders, disclaims all warranties with regard to this
#  software, including all implied warranties of merchantability and
#  fitness, in no event shall NAIST be liable for
#  any special, indirect or consequential damages or any damages
#  whatsoever resulting from loss of use, data or profits, whether in an
#  action of contract, negligence or other tortuous action, arising out
#  of or in connection with the use or performance of this software.
#
#  A large portion of the dictionary entries
#  originate from ICOT Free Software.  The following conditions for ICOT
#  Free Software applies to the current dictionary as well.
#
#  Each User may also freely distribute the Program, whether in its
#  original form or modified, to any third party or parties, PROVIDED
#  that the provisions of Section 3 ("NO WARRANTY") will ALWAYS appear
#  on, or be attached to, the Program, which is distributed substantially
#  in the same form as set out herein and that such intended
#  distribution, if actually made, will neither violate or otherwise
#  contravene any of the laws and regulations of the countries having
#  jurisdiction over the User or the intended distribution itself.
#
#  NO WARRANTY
#
#  The program was produced on an experimental basis in the course of the
#  research and development conducted during the project and is provided
#  to users as so produced on an experimental basis.  Accordingly, the
#  program is provided without any warranty whatsoever, whether express,
#  implied, statutory or otherwise.  The term "warranty" used herein
#  includes, but is not limited to, any warranty of the quality,
#  performance, merchantability and fitness for a particular purpose of
#  the program and the nonexistence of any infringement or violation of
#  any right of any third party.
#
#  Each user of the program will agree and understand, and be deemed to
#  have agreed and understood, that there is no warranty whatsoever for
#  the program and, accordingly, the entire risk arising from or
#  otherwise connected with the program is assumed by the user.
#
#  Therefore, neither ICOT, the copyright holder, or any other
#  organization that participated in or was otherwise related to the
#  development of the program and their respective officials, directors,
#  officers and other employees shall be held liable for any and all
#  damages, including, without limitation, general, special, incidental
#  and consequential damages, arising out of or otherwise in connection
#  with the use or inability to use the program or any product, material
#  or result produced or otherwise obtained by using the program,
#  regardless of whether they have been advised of, or otherwise had
#  knowledge of, the possibility of such damages at any time during the
#  project or thereafter.  Each user will be deemed to have agreed to the
#  foregoing by his or her commencement of use of the program.  The term
#  "use" as used herein includes, but is not limited to, the use,
```

```
#  modification, copying and distribution of the program and the
#  production of secondary products from the program.
#
#  In the case where the program, whether in its original form or
#  modified, was distributed or delivered to or received by a user from
#  any person, organization or entity other than ICOT, unless it makes or
#  grants independently of ICOT any specific warranty to the user in
#  writing, such person, organization or entity, will also be exempted
#  from and not be held liable to the user for any such damages as noted
#  above as far as the program is concerned.
#
#  ---------------COPYING.ipadic-----END---------------------------------

-------------------------------------------------------------------

Lao Word Break Dictionary Data (laodict.txt)

# Copyright (C) 2016 and later: Unicode, Inc. and others.
# License & terms of use: http://www.unicode.org/copyright.html
# Copyright (c) 2015 International Business Machines Corporation
# and others. All Rights Reserved.
#
# Project: https://github.com/rober42539/lao-dictionary
# Dictionary: https://github.com/rober42539/lao-dictionary/laodict.txt
# License: https://github.com/rober42539/lao-dictionary/LICENSE.txt
#          (copied below)
#
# This file is derived from the above dictionary version of Nov 22, 2020
#  --------------------------------------------------------------------
#  Copyright (C) 2013 Brian Eugene Wilson, Robert Martin Campbell.
#  All rights reserved.
#
#  Redistribution and use in source and binary forms, with or without
#  modification, are permitted provided that the following conditions are met:
#
#  Redistributions of source code must retain the above copyright notice, this
#  list of conditions and the following disclaimer. Redistributions in binary
#  form must reproduce the above copyright notice, this list of conditions and
#  the following disclaimer in the documentation and/or other materials
#  provided with the distribution.
#
# THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
# "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
# LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS
# FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE
# COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT,
# INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
# (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
# SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
# HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
# STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
# ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
# OF THE POSSIBILITY OF SUCH DAMAGE.
#  --------------------------------------------------------------------------

-------------------------------------------------------------------

Burmese Word Break Dictionary Data (burmesedict.txt)

#  Copyright (c) 2014 International Business Machines Corporation
#  and others. All Rights Reserved.
#
#  This list is part of a project hosted at:
#    github.com/kanyawtech/myanmar-karen-word-lists
#
#  --------------------------------------------------------------------------
#  Copyright (c) 2013, LeRoy Benjamin Sharon
#  All rights reserved.
#
#  Redistribution and use in source and binary forms, with or without
#  modification, are permitted provided that the following conditions
#  are met: Redistributions of source code must retain the above
#  copyright notice, this list of conditions and the following
#  disclaimer.  Redistributions in binary form must reproduce the
#  above copyright notice, this list of conditions and the following
#  disclaimer in the documentation and/or other materials provided
#  with the distribution.
```

```
 #
 #    Neither the name Myanmar Karen Word Lists, nor the names of its
 #    contributors may be used to endorse or promote products derived
 #    from this software without specific prior written permission.
 #
 #  THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND
 #  CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES,
 #  INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
 #  MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
 #  DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS
 #  BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
 #  EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED
 #  TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
 #  DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON
 #  ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR
 #  TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF
 #  THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
 #  SUCH DAMAGE.
 #  ------------------------------------------------------------------------------

-------------------------------------------------------------------

Time Zone Database

  ICU uses the public domain data and code derived from Time Zone
Database for its time zone support. The ownership of the TZ database
is explained in BCP 175: Procedure for Maintaining the Time Zone
Database section 7.

 # 7.  Database Ownership
 #
 #    The TZ database itself is not an IETF Contribution or an IETF
 #    document.  Rather it is a pre-existing and regularly updated work
 #    that is in the public domain, and is intended to remain in the
 #    public domain.  Therefore, BCPs 78 [RFC5378] and 79 [RFC3979] do
 #    not apply to the TZ Database or contributions that individuals make
 #    to it.  Should any claims be made and substantiated against the TZ
 #    Database, the organization that is providing the IANA
 #    Considerations defined in this RFC, under the memorandum of
 #    understanding with the IETF, currently ICANN, may act in accordance
 #    with all competent court orders.  No ownership claims will be made
 #    by ICANN or the IETF Trust on the database or the code.  Any person
 #    making a contribution to the database or code waives all rights to
 #    future claims in that contribution or in the TZ Database.

-------------------------------------------------------------------

Google double-conversion

Copyright 2006-2011, the V8 project authors. All rights reserved.
Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

    * Redistributions of source code must retain the above copyright
      notice, this list of conditions and the following disclaimer.
    * Redistributions in binary form must reproduce the above
      copyright notice, this list of conditions and the following
      disclaimer in the documentation and/or other materials provided
      with the distribution.
    * Neither the name of Google Inc. nor the names of its
      contributors may be used to endorse or promote products derived
      from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

-------------------------------------------------------------------
```

```
File: aclocal.m4 (only for ICU4C)
Section: pkg.m4 - Macros to locate and utilise pkg-config.


Copyright (c) 2004 Scott James Remnant <scott@netsplit.com>.
Copyright (c) 2012-2015 Dan Nicholson <dbn.lists@gmail.com>

This program is free software; you can redistribute it and/or modify
it under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful, but
WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program; if not, write to the Free Software
Foundation, Inc., 59 Temple Place - Suite 330, Boston, MA
02111-1307, USA.

As a special exception to the GNU General Public License, if you
distribute this file as part of a program that contains a
configuration script generated by Autoconf, you may include it under
the same distribution terms that you use for the rest of that
program.


(The condition for the exception is fulfilled because
ICU4C includes a configuration script generated by Autoconf,
namely the `configure` script.)

------------------------------------------------------------------------

File: config.guess (only for ICU4C)


This file is free software; you can redistribute it and/or modify it
under the terms of the GNU General Public License as published by
the Free Software Foundation; either version 3 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful, but
WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the GNU
General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program; if not, see <https://www.gnu.org/licenses/>.

As a special exception to the GNU General Public License, if you
distribute this file as part of a program that contains a
configuration script generated by Autoconf, you may include it under
the same distribution terms that you use for the rest of that
program.  This Exception is an additional permission under section 7
of the GNU General Public License, version 3 ("GPLv3").


(The condition for the exception is fulfilled because
ICU4C includes a configuration script generated by Autoconf,
namely the `configure` script.)

------------------------------------------------------------------------

File: install-sh (only for ICU4C)


Copyright 1991 by the Massachusetts Institute of Technology

Permission to use, copy, modify, distribute, and sell this software and its
documentation for any purpose is hereby granted without fee, provided that
the above copyright notice appear in all copies and that both that
copyright notice and this permission notice appear in supporting
documentation, and that the name of M.I.T. not be used in advertising or
publicity pertaining to distribution of the software without specific,
```

```
      HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
      STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
      ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
      OF THE POSSIBILITY OF SUCH DAMAGE.

-----------------------------------------------------------

This product includes OpenSSL.

      Copyright (C) 1995-1998 Eric Young (eay@cryptsoft.com)
      All rights reserved.

      Redistribution and use in source and binary forms, with or without
      modification, are permitted provided that the following conditions
      are met:
      1. Redistributions of source code must retain the copyright
         notice, this list of conditions and the following disclaimer.
      2. Redistributions in binary form must reproduce the above copyright
         notice, this list of conditions and the following disclaimer in the
         documentation and/or other materials provided with the distribution.
      3. All advertising materials mentioning features or use of this software
         must display the following acknowledgement:
         "This product includes cryptographic software written by
         Eric Young (eay@cryptsoft.com)"
         The word 'cryptographic' can be left out if the rouines from the library
         being used are not cryptographic related :-).
      4. If you include any Windows specific code (or a derivative thereof) from
         the apps directory (application code) you must include an acknowledgement:
         "This product includes software written by Tim Hudson (tjh@cryptsoft.com)"

      THIS SOFTWARE IS PROVIDED BY ERIC YOUNG ``AS IS'' AND
      ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
      IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
      ARE DISCLAIMED.  IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
      FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
      DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
      OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
      HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
      LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
      OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
      SUCH DAMAGE.

-----------------------------------------------------------

This product includes Potrace.

      Copyright (c) 2001-2019 Icosasoft Software Inc. All rights reserved.

-----------------------------------------------------------

This product includes Qt.

      The software uses Qt, licensed under LGPL v3. The Qt Toolkit is Copyright (C)
 2019 The Qt Company Ltd.

      Portions of this software are copyright (C) 2006-2015 The FreeType Project
 (www.freetype.org). All rights reserved.

      Copyright (C) 1991-2011, Thomas G. Lane, Guido Vollbeding.
      This software is based in part on the work of the Independent JPEG Group.

      Secure Hash Algorithm SHA-3 - brg_endian
      Copyright (c) 1998-2013, Brian Gladman, Worcester, UK. All rights reserved.

      LICENSE TERMS

      The redistribution and use of this software (with or without changes)
      is allowed without the payment of fees or royalties provided that:

       1. source code distributions include the above copyright notice, this
          list of conditions and the following disclaimer;

       2. binary distributions include the above copyright notice, this list
          of conditions and the following disclaimer in their documentation;

       3. the name of the copyright holder is not used to endorse products
          built using this software without specific written permission.
```

```
      DISCLAIMER

      This software is provided 'as is' with no explicit or implied warranties
      in respect of its properties, including, but not limited to, correctness
      and/or fitness for purpose.

---------------------------------------------------------------

This product includes QtSingleApplication.

        Copyright (C) 2013 Digia Plc and/or its subsidiary(-ies).
        Contact: http://www.qt-project.org/legal

        This file is part of the Qt Solutions component.

        You may use this file under the terms of the BSD license as follows:

        Redistribution and use in source and binary forms, with or without
        modification, are permitted provided that the following conditions are
        met:
          * Redistributions of source code must retain the above copyright
            notice, this list of conditions and the following disclaimer.
          * Redistributions in binary form must reproduce the above copyright
            notice, this list of conditions and the following disclaimer in
            the documentation and/or other materials provided with the
            distribution.
          * Neither the name of Digia Plc and its Subsidiary(-ies) nor the names
            of its contributors may be used to endorse or promote products derived
            from this software without specific prior written permission.


        THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
        "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
        LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
        A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
        OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
        SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
        LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
        DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
        THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
        (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
        OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.


---------------------------------------------------------------

This product includes XMP Toolkit.
Copyright (c) 2020, Adobe
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this
   list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice,
   this list of conditions and the following disclaimer in the documentation
   and/or other materials provided with the distribution.

3. Neither the name of the copyright holder nor the names of its
   contributors may be used to endorse or promote products derived from
   this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE
FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
---------------------------------------------------------------
```

17. Copyrights