

---

# **kliko Documentation**

*Release 0.4*

**Gijs Molenaar**

March 14, 2016



<b>1</b>	<b>Introduction</b>	<b>3</b>
1.1	installation . . . . .	3
1.2	Getting started . . . . .	3
1.3	Contributing . . . . .	3
<b>2</b>	<b>Terminology</b>	<b>5</b>
2.1	Kliko . . . . .	5
2.2	Kliko image . . . . .	5
2.3	Kliko container . . . . .	5
2.4	The kliko.yml file . . . . .	5
2.5	The parameters.json file . . . . .	5
<b>3</b>	<b>The specification</b>	<b>7</b>
3.1	The kliko.yml file . . . . .	7
3.2	An example kliko.yml file . . . . .	9
<b>4</b>	<b>Usage</b>	<b>23</b>
4.1	Validation . . . . .	23
4.2	Docker . . . . .	24
4.3	Django . . . . .	24
<b>5</b>	<b>Indices and tables</b>	<b>25</b>
	<b>Python Module Index</b>	<b>27</b>



Contents:



---

## Introduction

---

KLIKO is a specification, validator and parser for the Scientific Compute Container specification. KLIKO is written in Python.

### 1.1 installation

From the source folder:

```
$ python setup.py install
```

or from pypi:

```
$ pip install kliko
```

### 1.2 Getting started

To get started you should:

- Create a Docker container from your application
- Add or modify a script in the container that can parse and use a `parameters.json` file.
- Add a `kliko.yml` file to the root of the container which defines the valid fields in the parameters file.
- You can validate your kliko file with the `kliko-validate.py` script installed by the kliko Python library.

### 1.3 Contributing

Contributions are more than welcome! If you experience any problems let us know in the bug tracker. We accept patches in the form of github pull requests. Please make sure your code works with python 2 and python3, and is pep8 compatible. Also make sure the test suit actually passes all tests. We use docker in some of the tests so you need to have that installed and configured.





---

## Terminology

---

### 2.1 Kliko

A specification which defines constraints on a Docker container to aid in the scheduling of scientific compute tasks. It is also a Python library that can be used to check if a container confirms the specification.

### 2.2 Kliko image

A Docker image conforming to the kliko specification. An image is an ordered collection of root filesystem changes and the corresponding execution parameters for use within a container runtime. Images are read-only.

### 2.3 Kliko container

A container is an active (or inactive if exited) stateful instantiation of an image.

Read more about Docker terminology in the [Docker glossary](#).

### 2.4 The kliko.yml file

A yaml formatted file conforming to the Kliko specification that defines the parameters a Kliko container is expecting. This is the file you want to create and add to your Docker image if you want to create a Kliko container.

### 2.5 The parameters.json file

A json encoded structure that contains all the parameter values for your compute task. This file is presented to your container at runtime by the container runner, for example RODRIGUES or Nextflow. The valid fields are defined by the Kliko image container and are defined in the `kliko.yml` file.



---

## The specification

---

- Kliko is based on standard docker containers
- Container must have a CMD specified, which would be the main program of the container. It should not require arguments.
- logging should be written to STDOUT and STDERR.
- We define two types of compute containers, split IO and joined IO containers.
- For split IO Input files will be mounted read only into `/input`. Output file should be written to `/output`, which will be mounted by the host.
- For joined IO containers input & output is the `/work` folder which will be mounted RW.
- parameters for the computation will be given when the container is run in the form of a file in json format called `parameters.json` in `/input`
- Which parameters the container will accept should be defined in a yaml file `/kliko.yml`
- The `kliko.yml` file should follow the schema defined in `kliko/schema.yml`.
- an example parameters definition file can be found in `examples/form.yml`
- fields with type `file` will enable supply of custom input files. these will be put in the `/input` folder.

### 3.1 The kliko.yml file

The kliko file should be in YAML format and has these required fields:

#### 3.1.1 schema\_version

The version of the kliko specification. note that this is independent of the versioning of the Kliko library.

#### 3.1.2 name

Name of the kliko image. For example `radioastro/simulator`.

#### 3.1.3 description

A more detailed description of the image.

### 3.1.4 author

Who made the container.

### 3.1.5 email

email adres of the author.

### 3.1.6 url

Where to find the specific kliko project on the web.

### 3.1.7 io

Which IO mode to use, could be `join` or `split`. For `split` IO Input files will be mounted read only into `/input`. Output file should be written to `/output`, which will be mounted by the host. For joined IO containers input & output is the `/work` folder which will be mounted RW.

### 3.1.8 Sections

The paramaters are grouped in sections. Sections are just lists of fields.

### 3.1.9 fields

A section consists of a list of fields.

### 3.1.10 field

each field has 2 obligatory keys, a `name` and a `type`. Name is a short reference to the field which needs to be unique. This will be the name for internal reference. The `type` defines the `type` of the field and can be one of `choice`, `char`, `float`, `file`, `bool` or `int`.

**Optional keys are:**

- **initial:** supply a initial (default) value for a field
- **max\_length:** define a maximum length in case of string type
- **choices:** define a list of choices in case of a choice field. The choices should be a mapping
- **label:** **The label used for representing the field to the end user. If no label is given the name of the field is used.**
- **required:** Indicates if the field is required or optional
- **help\_text:** An optional help text that is presented to the end user next to the field.

## 3.2 An example kliko.yml file

Below is an example kliko file.

```

1  schema_version: 1
2  name: simulator form
3  description: simulator form
4  container: skasa/simulator
5  author: Gijs Molenaar
6  email: gijsmolenaar@gmail.com
7  url: http://github.com/ska-sa/rodrigues/
8  io: split
9
10 sections:
11   -
12     name: telescope
13     description: Observatory
14     fields:
15       -
16         name: name
17         type: char
18         initial: new simulation
19         max_length: 200
20       -
21         name: observatory
22         type: choice
23         choices:
24           MeerKAT: MeerKAT
25           Kat-7: Kat-7
26           JVLA-A: JVLA-A
27         initial: MeerKAT
28       -
29         name: output
30         type: choice
31         choices:
32           Image: Image
33           Visibilities: Visibilities
34         initial: Image
35         label: Output type
36       -
37         name: sefd
38         type: float
39         label: SEFD
40         required: False
41         help_text: System defaults will be used if left blank
42     -
43     name: sky
44     description: Sky model
45     fields:
46       -
47         name: sky_type
48         type: choice
49         choices:
50           Tigger-LSM: Tigger-LSM
51           ASCII: ASCII
52           FITS: FITS
53           KATALOG: KATALOG
54       -

```

```
55     name: sky_model
56     type: file
57     required: False
58     -
59     name: katalog_id
60     type: choice
61     label: KATALOG
62     required: False
63     initial: nvss6deg
64     choices:
65         nvss6deg: nvss6deg
66         scubedldeg: scubedldeg
67         3c147_field: 3c147_field
68         3c147_field_no_core: 3c147_field_no_core
69     -
70     name: radius
71     type: float
72     label: Radius
73     initial: 0.5
74     required: False
75     help_text: Radius of degrees
76     -
77     name: fluxrange
78     type: char
79     label: Flux range
80     initial: 0.001-1
81     max_length: 32
82     help_text: Flux range in Jy
83     -
84     name: ms_dec
85     type: char
86     label: Declination
87     initial: -30d0m0s
88     help_text: in dms
89     -
90     name: ms_ra
91     type: char
92     label: Right ascension
93     initial: 0h0m0s
94     help_text: in hms
95     max_length: 32
96     -
97     name: add_noise
98     type: bool
99     initial: True
100    required: False
101    -
102    name: vis_noise_std
103    type: float
104    label: Visibility noise std
105    initial: 0
106    help_text: Generates from SEFD if 0
107    -
108    name: observation
109    description: Observation setup
110    fields:
111        -
112        name: ms_synthesis
```

```

113     type: float
114     label: Synthesis time
115     initial: 0.25
116     help_text: in hours
117     -
118     name: ms_dtime
119     type: int
120     label: Integration time
121     initial: 10
122     help_text: in seconds
123     -
124     name: ms_freq0
125     type: float
126     label: Start frequency
127     initial: 700
128     help_text: in MHz
129     -
130     name: ms_dfreq
131     type: float
132     label: Channel width
133     initial: 50e3
134     help_text: in kHz
135     -
136     name: ms_nchan
137     type: int
138     label: Channels
139     initial: 1
140     help_text: Number of frequency channels
141
142     -
143     name: imaging
144     description: imaging settings
145     fields:
146         -
147             name: use_default_im
148             type: bool
149             label: Use default imaging settings
150             initial: True
151             required: False
152         -
153             name: imager
154             type: choice
155             label: Imager
156             initial: LW
157             choices:
158                 LWIMAGER: LWIMAGER
159                 WSCLEAN: WSCLEAN
160                 CASA: CASA
161         -
162             name: im_npix
163             type: int
164             label: Image size
165             initial: 512
166             help_text: in pixels
167         -
168             name: im_cellsize
169             type: float
170             label: Pixel size

```

```

171     help_text: in arcseconds
172     initial: 1
173     -
174     name: im_weight
175     type: choice
176     label: uv-Weighting
177     choices:
178         Natural: Natural
179         Uniform: Uniform
180         Briggs: Briggs
181     initial: N
182     -
183     name: im_robust
184     type: float
185     label: Robust
186     initial: 0
187     help_text: Briggs robust parameter
188     -
189     name: im_weight_fov
190     type: float
191     label: Weight FoV
192     help_text: in arcminutes
193     required: False
194     -
195     name: im_wprojplanes
196     type: int
197     label: w-Projection planes
198     initial: 0
199     -
200     name: im_mode
201     type: choice
202     label: Imaging mode
203     choices:
204         channel: channel
205         mfs: mfs
206         velocity: velocity
207         frequency: frequency
208     initial: channel
209     -
210     name: channelise
211     type: int
212     label: Image channelise
213     initial: 0
214     help_text: 0 means average all, 1 per channel, etc.
215     -
216     name: im_stokes
217     type: char
218     label: Stokes
219     initial: I
220     max_length: 4
221
222
223
224     -
225     name: lwimager
226     description: LWIMAGER deconvolution settings
227     fields:
228         -

```



```
229     name: lwimager
230     type: bool
231     label: Deconvolve with me!
232     required: False
233   -
234     name: lwimager_niter
235     type: int
236     label: NITER
237     initial: 1000
238   -
239     name: lwimager_gain
240     type: float
241     label: Loop gain
242     initial: 0.1
243   -
244     name: lwimager_threshold
245     type: float
246     label: Clean Threshold
247     initial: 0
248     help_text: In Jy
249   -
250   -
251     name: lwimager_sigmalevel
252     type: float
253     label: Clean sigma level
254     initial: 0
255     help_text: In sigma above noise
256   -
257     name: lwimager_operation
258     type: choice
259     label: Clean algorithm
260     choices:
261       csclean: csclean
262       hogbom: hogbom
263       clark: clark
264       multiscale: multiscale
265     initial: csclean
266   -
267     name: lwimager_cyclefactor
268     type: float
269     label: Cycle factor
270     initial: 1.5
271   -
272     name: lwimager_cyclespeedup
273     type: float
274     label: Cycle speed up
275     initial: -1
276   -
277     name: lwimager_stoppointmode
278     type: float
279     label: Stop point mode
280     initial: -1
281   -
282     name: lwimager_nscales
283     type: int
284     label: Scales for MS clean
285     initial: 4
286   -
```

```
287     name: lwimager_uservector
288     type: char
289     label: Clean scales
290     required: False
291     help_text: Comma seperated scales in pixels
292     max_length: 64
293
294 -
295     name: wsclean
296     description: WSCLEAN deconvolution settings
297     fields:
298     -
299         name: wsclean
300         type: bool
301         label: Deconvolve with me!
302         required: False
303     -
304         name: wsclean_niter
305         type: int
306         label: NITER
307         initial: 1000
308     -
309         name: wsclean_gain
310         type: float
311         label: Minor loop gain
312         initial: 0.1
313     -
314         name: wsclean_mgain
315         type: float
316         label: Major loop gain
317         initial: 0.9
318     -
319         name: wsclean_threshold
320         type: float
321         label: Clean Threshold
322         initial: 0
323         help_text: In Jy
324     -
325         name: wsclean_sigmalevel
326         type: float
327         label: Clean sigma level
328         initial: 0
329         help_text: In sigma above noise
330     -
331         name: wsclean_joinpolarizations
332         type: bool
333         label: Join polarizations
334         required: False
335     -
336         name: wsclean_joinchannels
337         type: bool
338         label: Join channels
339         required: False
340     -
341         name: wsclean_multiscale
342         type: bool
343         label: Multiscale clean
344         required: False
```

```

345     -
346       name: wsclean_multiscale_dash_threshold_dash_bias
347       type: float
348       label: Multi scale threshold bias
349       initial: 0.7
350     -
351       name: wsclean_multiscale_dash_scale_dash_bias
352       type: float
353       label: Multi scale bias
354       initial: 0.6
355     -
356       name: wsclean_cleanborder
357       type: float
358       label: Clean border
359       initial: 5
360       help_text: In percentage of image width/height
361     -
362       name: wsclean_smallpsf
363       type: bool
364       label: Small PSF
365       required: False
366       help_text: Resize the psf to speed up minor clean iterations
367     -
368       name: wsclean_nonnegative
369       type: bool
370       label: No negative
371       required: False
372       help_text: Do not allow negative components during cleaning
373     -
374       name: wsclean_stopnegative
375       type: bool
376       label: Stop on negative
377       required: False
378     -
379       name: wsclean_beamsize
380       type: char
381       label: Restoring beam size
382       required: False
383       help_text: In arcseconds
384       max_length: 32
385   -
386     name: casa
387     description: CASA deconvolution settings
388     fields:
389       -
390         name: casa
391         type: bool
392         label: Deconvolve with me!
393         required: False
394       -
395         name: casa_niter
396         type: int
397         label: NITER
398         initial: 1000
399       -
400         name: casa_threshold
401         type: float
402

```

```
403     label: Threshold
404     initial: 0
405     -
406     name: casa_sigmalevel
407     type: float
408     label: Clean sigma level
409     initial: 0
410     help_text: In sigma above noise
411     -
412     name: casa_gain
413     type: float
414     label: Loop Gain
415     initial: 0.1
416     help_text: Clean Loop gain
417     -
418     name: casa_psfmode
419     type: choice
420     label: PSF mode
421     initial: clark
422     choices:
423         clark: clark
424         clarkstokes: clarkstokes
425         hogbom: hogbom
426     -
427     name: casa_imagermode
428     type: choice
429     label: Imager mode
430     required: False
431     choices:
432         csclean: csclean
433         mosiac: mosiac
434     initial: csclean
435     -
436     name: casa_gridmode
437     type: choice
438     label: Grid mode
439     required: False
440     choices:
441         widefield: widefield
442         aprojection: aprojection
443     initial: widefield
444     help_text: A-projection only works JVLA
445     -
446     name: casa_nterms
447     type: int
448     label: NTERMS
449     initial: 1
450     help_text: See CASA clean task
451     -
452     name: casa_reffreq
453     type: float
454     label: MFS ref Frequency
455     required: False
456     help_text: in MHz
457     -
458     name: casa_multiscale
459     type: char
460
```

```

461     label: Multiscale
462     required: False
463     help_text: Deconvolution scales in pixels
464     max_length: 200
465     -
466     name: casa_negcomponent
467     type: float
468     label: Negative Components
469     initial: -1
470     help_text: See CASA clean task
471     -
472     name: casa_smallscalebias
473     type: float
474     label: Small scale bias
475     initial: 0.6
476     help_text: See CASA clean task
477     -
478     name: casa_restoringbeam
479     type: char
480     label: Restoring beam
481     required: False
482     max_length: 32
483     -
484     name: casa_cyclefactor
485     type: float
486     label: Cycle factor
487     initial: 1.5
488     -
489     name: casa_cyclespeedup
490     type: int
491     label: Cycle speed up
492     initial: -1
493
494
495     -
496     name: moresane
497     description: MORESANE deconvolution settings
498     fields:
499
500         -
501             name: moresane
502             type: bool
503             label: Deconvolve with me!
504             required: False
505         -
506             name: moresane_scalecount
507             type: int
508             label: Scale count
509             required: False
510             help_text: See MORESANE help
511         -
512             name: moresane_subregion
513             type: int
514             label: Sub region
515             required: False
516             help_text: Inner region to deconvolve in pixels
517         -
518             name: moresane_startscale

```

```
519     type: int
520     label: Start scale
521     initial: 1
522     -
523     name: moresane_stopscale
524     type: int
525     label: Stop scale
526     initial: 20
527     -
528     name: moresane_sigmalevel
529     type: float
530     label: Threshold level
531     initial: 3
532     help_text: in sigma above noise
533     -
534     name: moresane_loopgain
535     type: float
536     label: Loop gain
537     initial: 0.1
538     -
539     name: moresane_tolerance
540     type: float
541     label: Tolerance
542     initial: .75
543     -
544     name: moresane_accuracy
545     type: float
546     label: Accuracy
547     initial: 1e-6
548     -
549     name: moresane_majorloopmiter
550     type: int
551     label: Major loop iterations
552     initial: 100
553     -
554     name: moresane_minorloopmiter
555     type: int
556     label: Minor loop iterations
557     initial: 50
558     -
559     name: moresane_convmode
560     type: choice
561     label: Convolution mode
562     initial: circular
563     choices:
564         circular: circular
565         linear: linear
566     -
567     name: moresane_enforcepositivity
568     type: bool
569     label: Enforce Positivity
570     required: False
571     -
572     name: moresane_edgesuppression
573     type: bool
574     label: Edge Suppression
575     required: False
576
```

```
577     -
578       name: moresane_edgeoffset
579       type: int
580       label: Edge offset
581       initial: 0
582     -
583       name: moresane_mfs
584       type: bool
585       label: MFS map
586       required: False
587       help_text: Comes with an spi map
588     -
589       name: moresane_spi_dash_sigmalevel
590       type: float
591       label: spi threshold level
592       initial: 10
593       help_text: In sigma above noise
594
```

Loading a Kliko container with the previous kliko file is loaded up in RODRIGUES will result in the form below:

## Create new simulation

**Observatory**

Name

Observatory

Output type

SEFD

System defaults will be used if left blank

**Sky model**

Sky type

Sky model  No file chosen

KATALOG

Radius   
Radius of degrees

Flux range   
Flux range in Jy

Declination   
in dms

Right ascension   
in hms

Add noise

Visibility noise std   
Generates from SEFD if 0

**Observation setup**

Synthesis time   
in hours

Integration time   
in seconds

Start frequency   
in MHz

Channel width   
in kHz

Channels   
Number of frequency channels

**imaging settings**

Use default imaging settings

Imager

Image size   
in pixels

Pixel size   
in arcseconds

uv-Weighting

Robust   
Briggs robust parameter

Weight FoV

w-Projection planes

Imaging mode

Image channelise

0 means average all, 1 per channel, etc.

Stokes

**LWIMAGER deconvolution settings**

Deconvolve with me!

NITER

Loop gain

Clean Threshold   
In Jy

Clean sigma level   
In sigma above noise

Clean algorithm

Cycle factor

Cycle speed up

Stop point mode

Scales for MS clean

Clean scales

Comma separated scales in pixels

**WSCLEAN deconvolution settings**

Deconvolve with me!

NITER

Minor loop gain

Major loop gain

Clean Threshold   
In Jy

Clean sigma level   
In sigma above noise

Join polarizations

Join channels

Multiscale clean

Multi scale threshold bias

Multi scale bias

Clean border   
In percentage of image width/height

Small PSF   
Resize the psf to speed up minor clean iterations

No negative   
Do not allow negative components during cleaning

Stop on negative

Restoring beam size   
In arcseconds

**CASA deconvolution settings**

Deconvolve with me!

NITER

Threshold

Clean sigma level   
In sigma above noise

Loop Gain   
Clean Loop gain

PSF mode

Imager mode

Grid mode   
A-projection only works JVLA

NTERMS   
See CASA clean task

MFS ref Frequency   
in MHz

Multiscale   
Deconvolution scales in pixels

Negative Components   
See CASA clean task

Small scale bias   
See CASA clean task

Restoring beam

Cycle factor

Cycle speed up

**MORESANE deconvolution settings**

Deconvolve with me!

Scale count   
See MORESANE help

Sub region   
Inner region to deconvolve in pixels

Start scale

Stop scale

Threshold level   
in sigma above noise

Loop gain

Tolerance

Accuracy

Major loop iterations

Minor loop iterations

Convolution mode

Enforce Positivity

Edge Suppression

Edge offset

MFS map   
Comes with an spi map

spi threshold level   
In sigma above noise



Processing this form will result in the following parameters.json file which is presented to the Kliko container on runtime:

```

1 {
2   "wsclean_threshold": 0.0,
3   "im_stokes": "I",
4   "casa_smallscalebias": 0.6,
5   "casa_threshold": 0.0,
6   "sky_model": null,
7   "casa_niter": 1000,
8   "wsclean_sigmalevel": 0.0,
9   "im_cellsize": 1.0,
10  "im_robust": 0.0,
11  "moresane_scalecount": null,
12  "casa_reffreq": null,
13  "casa_multiscale": "",
14  "lwimager_threshold": 0.0,
15  "casa_cyclefactor": 1.5,
16  "wsclean_cleanborder": 5.0,
17  "lwimager_nscales": 4,
18  "im_npix": 512,
19  "casa_cyclespeedup": -1,
20  "moresane_edgesuppression": false,
21  "im_weight": "initial",
22  "wsclean_beamsize": "",
23  "radius": 0.5,
24  "wsclean_gain": 0.1,
25  "im_weight_fov": null,
26  "im_wprojplanes": 0,
27  "ms_ra": "0h0m0s",
28  "moresane_accuracy": 1e-06,
29  "ms_freq0": 700.0,
30  "lwimager_cyclefactor": 1.5,
31  "wsclean_multiscale_dash_threshold_dash_bias": 0.7,
32  "lwimager_niter": 1000,
33  "moresane_spi_dash_sigmalevel": 10.0,
34  "ms_nchan": 1,
35  "wsclean_niter": 1000,
36  "im_mode": "channel",
37  "moresane_majorloopmiter": 100,
38  "casa_gridmode": "widefield",
39  "fluxrange": "0.001-1",
40  "moresane_mfs": false,
41  "name": "new simulation",
42  "channelise": 0,
43  "moresane_subregion": null,
44  "lwimager_operation": "csclean",
45  "wsclean_multiscale": false,
46  "casa_psfmode": "clark",
47  "moresane_tolerance": 0.75,
48  "lwimager": false,
49  "lwimager_uservector": "",
50  "lwimager_stoppointmode": -1.0,
51  "casa_sigmalevel": 0.0,
52  "sefd": null,
53  "lwimager_gain": 0.1,
54  "moresane_edgeoffset": 0,
55  "moresane_loopgain": 0.1,
56  "katalog_id": "nvss6deg",

```

```
57 "wsclean_joinpolarizations": false,  
58 "ms_dfreq": 50000.0,  
59 "output": "Image",  
60 "wsclean": false,  
61 "wsclean_nonnegative": false,  
62 "wsclean_stopnegative": false,  
63 "moresane_convmode": "circular",  
64 "ms_synthesis": 0.25,  
65 "casa_negcomponent": -1.0,  
66 "casa_imagermode": "csclean",  
67 "ms_dec": "-30d0m0s",  
68 "moresane_minorloopmiter": 50,  
69 "use_default_im": true,  
70 "casa_nterms": 1,  
71 "ms_dtime": 10,  
72 "moresane_enforcepositivity": false,  
73 "moresane": false,  
74 "add_noise": true,  
75 "wsclean_multiscale_dash_scale_dash_bias": 0.6,  
76 "moresane_sigmalevel": 3.0,  
77 "moresane_stopscale": 20,  
78 "casa_restoringbeam": "",  
79 "casa": false,  
80 "wsclean_mgain": 0.9,  
81 "wsclean_joinchannels": false,  
82 "imager": "CASA",  
83 "vis_noise_std": 0.0,  
84 "moresane_startscale": 1,  
85 "lwimager_cyclespeedup": -1.0,  
86 "lwimager_sigmalevel": 0.0,  
87 "observatory": "MeerKAT",  
88 "wsclean_smallpsf": false,  
89 "sky_type": "Tigger-LSM",  
90 "casa_gain": 0.1  
91 }
```

## 4.1 Validation

Kliko and parameter validation related functions.

`kliko.validate.convert_to_parameters_schema` (*kliko*)

Convert a kliko schema into a validator for the parameters generated with a kliko schema.

**Parameters** `kliko` (*str*) – a kliko definition

**Returns** A structure for a pykwalify validator

`kliko.validate.validate` (*kliko\_file*='/*kliko.yml*', *paramaters\_file*='/*parameters.json*')

Validate the kliko and paramaters file and parse the parameters file. Should be run inside the Kliko container.

**Parameters**

- `kliko_file` (*str*) – Path to a kliko file
- `paramaters_file` (*str*) – path to a parameters file

**Returns** The validated and parsed paramaters file

`kliko.validate.validate_kliko` (*kliko*, *version*=2)

validate a kliko yaml string

**Parameters** `kliko` – a parsed kliko object

**Returns** a (nested) kliko structure

**Return type** dict

**Raises** an exception if the string can't be parsed or is not in the following the Kliko schema

`kliko.validate.validate_parameters` (*parameters*, *kliko*)

validate a set of parameters given a kliko definition

**Parameters**

- `parameters` (*dict*) – A structure that should follow the given kliko structure
- `kliko` (*dict*) – A nested dict which defines the valid parameters in Kliko format

**Returns** the parsed parameters

**Return type** str

**Raises** an exception if the string can't be parsed or is not in the defining valid parameters

## 4.2 Docker

Helper functions for using Kliko in combination with Docker

`kliko.docker.extract_params` (*docker\_client*, *image\_name*)

**Parameters**

- **docker\_client** (*docker.docker.Client*) – a docker client object
- **image\_name** (*str*) – name of the image to use for kliko.yml extraction

**Returns** content of the param schema

**Return type** `str`

## 4.3 Django

---

## Indices and tables

---

- `genindex`
- `modindex`
- `search`



**k**

`kliko.docker`, 24

`kliko.validate`, 23





## C

`convert_to_parameters_schema()` (in module `kliko.validate`), 23

## E

`extract_params()` (in module `kliko.docker`), 24

## K

`kliko.docker` (module), 24

`kliko.validate` (module), 23

## V

`validate()` (in module `kliko.validate`), 23

`validate_kliko()` (in module `kliko.validate`), 23

`validate_parameters()` (in module `kliko.validate`), 23