

# jobgen

## Job File Generator

John Marshall – 2019-07-23

# What Is jobgen?

A tool which accepts directives for a virtualized queueing system and generates directives for a target queueing system.

# Features

- Powerful configuration
  - Key-value
  - Cascading
  - Types
  - Constraints
  - Traceable
- Extensible via hooks (Python)
- Support for multiple queueing systems (Gridengine, PBS, PBSPro, Slurm)
- Generate documentation from configuration
- Portable: Python2, can use outside of Dorval

# Standard Settings

- `request.env`
- `request.errpath`
- `request.joinouterr`
- `request.mail`
- `request.mailopts`
- `request.name`
- `request.outpath`
- `request.project`
- `request.queue`
- `request.rerun`
- `request.shell`
- `request.wallclock`
- `request.chunk.*.memory`
- `request.chunk.*.ncores`
- `request.chunk.*.nslots`

# jobgen Usage – Shortcuts

- `-c <name>=<value>` – Request chunk settings  
`request.chunk.0.<name>=<value>`
- `-H <name>=<value>` – Request hint setting  
`request.hint.<name>=<value>`
- `-k <name>=<value>` – Full key setting  
`<name>=<value>`
- `-r <name>=<value>` – Request setting  
`request.<name>=<value>`
- `-v <name>=<value>` – Request environment variable  
`request.env.<name>=<value>`

# Example - Hello

- ```
#!/bin/bash
#JGEN -r name=hello
#JGEN -r queue=dev
#JGEN -c nslots=1
#JGEN -c ncores=1
#JGEN -c memory=4G
#JGEN -c image=dfo/dfo_default_ubuntu-18.04-amd64_latest
#JGEN -r wallclock=5:00
echo "hello"
```

# Generate

- gpssc1 - Gridengine  
    `jobgen -p gpssc1 -j hello.jgen`
- ppp1 – PBSPro  
    `jobgen -p ppp1 -j hello.jgen`

# Generated – Info

- Part of generated results
- Gridengine

```
# ---- jobgen - start
# ---- jobgen -- timestamp (2019-07-17 22:27:07)
# ---- jobgen -- generator (jobgen.generators.gridengine.GridengineGPSCGenerator)
# ---- jobgen -- syshooks (check_memory_tmpfs) (None) (None)
# ---- jobgen -- hooks (bump_wallclock) (None) (None)
```

- PBSPro

```
# ---- jobgen - start
# ---- jobgen -- timestamp (2019-07-17 22:32:30)
# ---- jobgen - generator (jobgen.generators.pbspro.PBSProPPPGenerator)
# ---- jobgen -- syshooks (check_memory_tmpfs) (None) (None)
# ---- jobgen -- hooks (None) (None) (None)
```



# Generated – gpssc1 Gridengine

```
#!/bin/bash
echo "hello"

# ...
#$ -j n
#$ -N hello
#$ -S /bin/bash
#$ -v JOBGEN_JOINOUTERR=false
#$ -v JOBGEN_NAME=hello
#$ -v JOBGEN_NSLOTS=1
#$ -v JOBGEN_QUEUE=dev
#$ -v JOBGEN_SHELL=/bin/bash
#$ -v JOBGEN_SLOT_IMAGE=dfo/dfo_default_ubuntu-18.04-amd64_latest
#$ -v JOBGEN_SLOT_MEMORY=4294967296B
#$ -v JOBGEN_SLOT_NCORES=1
#$ -v JOBGEN_SLOT_TMPFS=524288000B
#$ -v JOBGEN_WALLCLOCK=5:00
#$ -pe dev 1
#$ -l h_rt=0:05:00
#$ -l res_image=dfo/dfo_default_ubuntu-18.04-amd64_latest
#$ -l res_mem=4096
#$ -l res_cpus=1
#$ -l res_tmpfs=500
#
# ---- jobgen -- end
```

# Generated – ppp1 PBSPro

```
#!/bin/bash
# ...
#PBS -N hello
#PBS -S /bin/bash
#PBS -v JOBGEN_JOINOUTERR=false
#PBS -v JOBGEN_NAME=hello
#PBS -v JOBGEN_NSLOTS=1
#PBS -v JOBGEN_QUEUE=dev
#PBS -v JOBGEN_SHELL=/bin/bash
#PBS -v JOBGEN_SLOT_IMAGE=dfo/dfo_default_ubuntu-18.04-amd64_latest
#PBS -v JOBGEN_SLOT_MEMORY=4294967296B
#PBS -v JOBGEN_SLOT_NCORES=1
#PBS -v JOBGEN_WALLCLOCK=5:00
#PBS -q development
#PBS -l walltime=0:05:00
#PBS -l select=1:ncpus=1:mem=4096M:res_image=dfo/dfo_default_ubuntu-18.04-amd64_latest
#
# ---- jobgen - end

echo "hello"
```

# JOBGEN Environment Variables

- Many by default
- Correspond to many standard settings
- Configurable
- E.g.,
  - JOBGEN\_ERRPATH
  - JOBGEN\_MAIL
  - JOBGEN\_NAME
  - JOBGEN\_NSLOTS
  - JOBGEN\_SLOT\_MEMORY
  - JOBGEN\_SLOT\_NCORES
  - JOBGEN\_OUTPATH
  - JOBGEN\_QUEUE
  - JOBGEN\_WALLCLOCK

# Validation Examples

- `jobgen -p gpssc1 -j hello.jgen -c memory=1G`  
error: failed to load request settings: key  
(request.chunk.0.memory) value (1073741824) below minimum  
(2621440000)
- `jobgen -p gpssc1 -j hello.jgen -c memory=120G`  
error: failed to load request settings: key  
(request.chunk.0.memory) value (128849018880) exceeds  
maximum (104857600000)

# Validation Examples

- `jobgen -p gpssc1 -j hello.jgen -c nslots=0`  
error: failed to load request settings: key (request.chunk.0.nslots) value (0) below minimum (1)
- `jobgen -p gpssc1 -j hello.jgen -c ncores=20`  
error: failed to load request settings: key (request.chunk.0.ncores) value (20) exceeds maximum (16)
- `jobgen -p gpssc1 -j hello.jgen -r wallclock=10`  
error: failed to load request settings: key (request.wallclock) value (10) below minimum (30)

# Configuration – Profile

- INI-style (sections, key=value, comments)
- Files (system, user)
- Directives (job file, command line)
- Settings combined / overlaid on top of each other (last wins!)
- Sections maintained

# Profiles Load Order

- `etc/jobgen/profiles/base.conf`
- `etc/jobgen/profiles/site.conf`
- `etc/jobgen/profiles/<profname>.conf`
- `~/ .jobgen/profiles/base.conf`
- `~/ .jobgen/profiles/site.conf`
- `~/ .jobgen/profiles/<profname>.conf`
- job file directives
- command line directives

# Configuration – Settings

- key = value
- Organized by sections
- Meta
  - Type
  - Constraint
  - Information
- Non-meta
  - Values
  - Constrained by meta settings



# Setting Namespaces

- `info` – Informational
- `generator` – Generator
- `meta` – Metadata
- `qs.<qstype>` - Queueing system specific
- `request` – Request

# Request Info

- Flattened profiles using cascading method
  - Select first instance from section in cascade order
- Cascade order (most to least significant) by section
  - directives
  - user.<username>
  - group.<groupname>
  - queue.<qname>
  - qs.<qsname>
  - default

# Cascade Example

- Settings (flattened)

```
[directives]
request.chunk.0.nslots = 4
request.queue = dev

[user.jdm]
request.mail = jdm@abc.xyz
request.chunk.0.image = ub1804

[queue.dev]
request.chunk.0.memory = 4G

[default]
request.chunk.0.tmpfs = 500M
request.chunk.0.nslots = 1
request.name = unset
```

- Final values

- request.chunk.0.image = ub1804
- request.chunk.0.memory = 4G
- request.chunk.0.nslots = 4
- request.chunk.0.tmpfs = 500M
- request.mail = jdm@abc.xyz
- request.name = unset

# Meta Information

- Provides information about values for keys
- Format
  - `meta.<mtype>.<key> = <mvalue>`
- Where:
  - `<mtype>` – meta information being set
  - `<key>` – key affected
  - `<mvalue>` – value for meta setting
- Default for all non-meta settings
  - `meta.type.<key> = string`

# Meta Value Type (meta.type)

- boolean – true, false, yes, y, n
- float – floating point number
- integer – integer number
- memory – integer with suffix (B, K, M, G, T)
- namelist
- string – without quotes, single single/double quotes, triple single/double quotes; default
- stringmatch – string must match from a list of strings
- stringregexp – string must match a regular expression

# Meta Type (meta.<mtype>)

- Specific meaning determined by meta.type setting
- Meta types
  - csvvalues – comma-separated values (string)
  - description – text description
  - max – max value (integer, float, memory ,time)
  - min – min value (integer, float, memory, time)
  - regexp – regular expression string (string)

# Meta Examples

- `meta.type.request.joinouterr = boolean`
- `meta.type.request.name = string`
- `meta.type.request.chunk.0.memory = memory`
- `meta.type.request.chunk.*.gputype = stringmatch`
- `meta.type.request.wallclock = time`

# Meta Examples

- `meta.type.request.memory = memory`  
`meta.min.request.memory = 1G`  
`meta.max.request.memory = 1000000M`
- `meta.type.request.chunk.*.gputype = stringmatch`  
`meta.csvvalues.request.chunk.*.gputype =`  
`"k20", "k80", "v100"`
- `meta.type.request.wallclock = time`  
`meta.min.request.wallclock = 60`  
`meta.max.request.wallclock = 3:00:00`



# Gridmon Apps

- Job Launch for jobgen
  - Launch a job from a jobgen job file
  - Test impact of directive changes
  - View generated job file for target queueing system
  - <https://gridmon.science.gc.ca/apps/joblauncher>
- jobgen Debugger
  - Useful to inspect profile and meta information when loaded
  - <https://gridmon.science.gc.ca/apps/jobgenapp>

# portal Pages

- Generated with jgendoc from profiles
- For each cell/cluster
- System information
- Queue information

# portal Page Example

Dashboard / Home 

 Edit  Save for later  Watching  Share 

## jgendoc: gpsec2

Created by John, last modified just a moment ago

### Jobgen: gpsec2

#### System

|                        |                                                                                                 |
|------------------------|-------------------------------------------------------------------------------------------------|
| <b>Name</b>            | gpsec2                                                                                          |
| <b>Location</b>        | Dorval, Quebec                                                                                  |
| <b>Description</b>     | General Purpose Science Cluster 2                                                               |
| <b>Support Contact</b> | ssc.hpcoptimizationsservice-serviceoptimisationchp.spc@canada.ca                                |
| <b>Support URL</b>     | <a href="https://expl.info/display/SCIDOCS/gpsec2">https://expl.info/display/SCIDOCS/gpsec2</a> |

#### Queues

##### dev

| Key Type | Key                    | Description                        | Type    | Value | Constraints                                                                                            |
|----------|------------------------|------------------------------------|---------|-------|--------------------------------------------------------------------------------------------------------|
| standard | request.chunk.*.memory | Memory per slot for chunk          | memory  |       | <ul style="list-style-type: none"><li>keytype=standard</li><li>max=100000M</li><li>min=2500M</li></ul> |
| standard | request.chunk.*.ncores | Number of cores per slot for chunk | integer |       | <ul style="list-style-type: none"><li>keytype=standard</li><li>max=24</li><li>min=1</li></ul>          |

# Profile File Syntax Highlighting

- Highlight
  - Meta settings
  - Non-meta settings
  - Section names
  - Keys
  - Values
- Editors
  - Sublime Text 3
  - vim
- Viewers
  - Confluence

# Syntax Highlighting – Sublime Text

```
1 | # syntax=jobgenconf
2 | #
3 | # gp scl.conf
4 | #
5 |
6 | [default]
7 | info.support.url = "https://portal.science.gc.ca/confluence/display/SCIDOCs/gpscl"
8 | info.system.description = "General Purpose Science Cluster 1"
9 | info.system.name = "gpscl"
10 |
11 | meta.min.request.chunk.*.memory = 2500M
12 | request.chunk.0.memory = 2500M
13 | request.chunk.0.ncores = 1
14 | request.chunk.0.nslots = 1
15 | request.chunk.0.tmpfs = 500M
16 | request.qs = gridengine-gpsc
17 | request.queue = dev
18 | request.wallclock = 60
19 |
20 | [queue.dev]
21 | meta.max.request.chunk.*.memory = 100000M
22 | meta.max.request.chunk.*.ncores = 16
23 | qs.gridengine.request.pe = dev
24 |
```

# Syntax Highlighting – vim

```
# syntax=jobgenconf
#
# ppp1.conf
#

[default]
info.support.url = "https://portal.science.gc.ca/confluence/display/SCIDOCS/ppp1"
info.system.description = "Pre and Post Processing Cluster 1"
info.system.location = "(HPCR) Dorval, Quebec"
info.system.name = "ppp1"

request.qs = pbspro-ppp
request.queue = dev
request.chunk.0.nslots = 1
request.chunk.0.ncores = 1

[queue.dev]
meta.max.request.chunk.*.memory = 200000M
meta.max.request.chunk.*.ncores = 44
meta.max.request.chunk.*.tmpfs = 200000M
qs.pbs.request.queue = development

[queue.prod]
meta.max.request.chunk.*.memory = 200000M
meta.max.request.chunk.*.ncores = 44
meta.max.request.chunk.*.tmpfs = 200000M
qs.pbs.request.queue = production
```

# Syntax Highlighting – Confluence

```
# syntax=jobgenconf
#
# gpvc1.conf
#

[default]
meta.min.request.chunk.*.memory = 2500M
request.chunk.0.memory = 2500M
request.chunk.0.ncores = 1
request.chunk.0.nslots = 1
request.chunk.0.tmpfs = 500M
request.qs = gridengine-gpsc
request.queue = dev
request.wallclock = 60

[queue.dev]
meta.max.request.chunk.*.memory = 100000M
meta.max.request.chunk.*.ncores = 16
qs.gridengine.pe = dev
```