

job property	handle	description
_id	read-only	CoreJob id taken from the Mongodb document in the CoreQueue.
_hash		md5 of CoreJob arguments
args	user-defined	arguments passed to the CoreJob
attempts	user-defined	maximum number of execution attempts after CoreJob failure
attempts_left	read-only	number of attempts left
author	user-defined	author(s) of the CoreJob
defer_max	user-defined	maximum number of seconds to defer the CoreJob
defer_time	user-defined	seconds to wait before restarting after a JobDeferred error
dependency	user-defined	list of CoreJobs which must be completed before execution starts
enqueued	read-only	dictionary of CoreJob information created at enqueueing
enqueued.at	read-only	datetime the CoreJob has been enqueued at
enqueued.host	read-only	name of the host the CoreJob has been enqueued at
enqueued.parent	read-only	id of the precursing CoreJob that has enqueued the current CoreJob
enqueued.user	read-only	name of the user that enqueued the CoreJob
error_time	user-defined	seconds to wait before restarting a CoreJob after failure
finished	read-only	datetime when the CoreJob finished with sucess, failure or deferral
force	user-defined	enforce CoreJob execution regardless of resources or dependencies
hidden	user-defined	hide the job from job listing
inactive_at	read-only	datetime when a deferring CoreJob turns inactive, derived from defer_max
killed	read-only	datetime a CoreJob has been killed
last_error	read-only	last stack trace in case of failure or error
locked	read-only	dictionary of CoreJob consumer information
locked.at	read-only	datetime when the CoreJob has been locked by a CoreWorker
locket.heartbeat	read-only	datetime of the last advertised CoreJob progress
locked.host	read-only	hostname of the CoreWorker that locked the CoreJob

job property	handle	description
locked.pid	read-only	process ID of the CoreWorkerProcess a CoreJob is executed with
locked.progress	read-only	last progress message of the CoreJob
locked.worker	read-only	name of the CoreWorker that locked the CoreJob
locked.user	read-only	name of the user running the CoreWorker that locked the CoreJob
max_parallel	user-defined	maximum amount of this CoreJob running on the same node
memory	user-defined	minimum memory required to launch that CoreJob
name	read-only	fully qualified name of the CoreJob
nodes	user-defined	nodes eligible to execute this CoreJob
nonstop	read-only	datetime when the CoreJob turns into a non-stopping job, determined by wall_time
priority	user-defined	priority of the CoreJob with 0 being the lowest priority possible
progress_intervall	user-defined	minimum ammount of time inbetween materialized progress() calls
project	read-only	project the CoreJob is running under, determined by package structure
python	user-defined	Python executable to be used for dedicated Python virtual environment
query_at	read-only	datetime to re-execute the CoreJob, derived from error_time or defer_time
removed	read-only	datetime of removal request
runtime	read-only	CoreJob execution runtime
schedule	user-defined	CoreJob schedule in crontab syntax
sources	read-only	set of sources processed by a CoreJob
started	read-only	datetime of CoreJob execution start
state	read-only	CoreJob state
tag	user-defined	tag the job with string argmuents
wall_at	read-only	datetime when a CoreJob turns to non-stopping
wall_time	user-defined	number of seconds before a running CoreJob turns into a non-stopping CoreJob
zombie	read-only	datetime when a CoreJob turned into state zombie due to lacking progress

CoreJob development

Smooth CoreJob operations

Informational properties