What does a job's 'instance count' and the 'host.processors=X' reservation mean with respect to machine cores?

Qube does not explicitly restrict a job to run on a particular core, it leaves that up to the applications to determine. If this is the case, then what do these terms used in Qube mean and how do they relate to machine cores?

Some terminology first:

- A **job instance** is a "copy" of a job running on a worker.
- A worker has **job slots** specified by worker_cpus; the default number is 1 slot per core on the worker.
- The instance's **reservation** string specifes (via "`host.processors=X`") how many worker slots each instance will reserve.

A job's reservations define how many jobs can 'fit' on a worker

A reservation of "`host.processors=4`" means that each instance will reserve 4 of the worker's job slots; 2 of these instances can run at the same time on an 8-slot worker.

A job's reservations **DO NOT** define how many threads the application being launched will run.

This is up to the application itself. If an application's default behavior is to run single-threaded, but the job is reserving 4 slots, 3 out of 4 cores will be unused. Many of Qube's submission interfaces have controls that will set the applications "thread count" and the job's "slot reservation" to the same value, so each job instance will run on as many cores as it has reserved.