

# Atomic Run behaviour

Proposed behaviour for “atomic run” to handle (a) both interactive and background images; (b) supplying an optional command to run or not; (c) handling container not present, stopped, or already running.

Case 1: “atomic run <image>” with no additional command specified

	No container	Container stopped	Container running
Interactive image <sup>1</sup>	Run with LABEL RUN Dan> If no LABEL RUN, run CMD, if no CMD run /bin/sh?	Start with docker start -ai <sup>2</sup>	Exec default CMD <sup>3</sup> docker exec -ti <sup>2,4</sup>
Background image <sup>1</sup>	Run with LABEL RUN Dan>If no LABEL RUN, run cmd, else error? [sct] If there's no LABEL RUN then we cannot distinguish this from the case above	Start with docker start	Noop

Case 2: “atomic run <image> <cmd>”

	No container	Container stopped	Container running
Interactive image	Run with LABEL RUN <sup>5</sup> Exec new command docker exec -ti Dan> If no label run, just use standard run to execute CMD, If no CMD shouldn't we just do docker run -ti cmd [sct] No point running CMD if we've been asked to run something else. +1 on running cmd	Start with docker start <sup>5</sup> Exec new command docker exec -ti <sup>8</sup>	Exec new command docker exec -ti <sup>8</sup>
Background image	Run with LABEL RUN <sup>6</sup> Dan > If no LABEL RUN, just run with standard run command. [sct] +1 Exec new command docker exec -ti	Start with docker start <sup>7</sup> Exec new command docker exec -ti	Exec new command docker exec -ti <sup>8</sup>

Notes:

- 1) Determine interactive vs daemon via “docker inspect” on an existing container. If attributes Config.AttachStdin, Config.AttachStdout and Config.AttachStderr are all true, container is interactive, else it is background/daemon (ie. started with “docker run -d”)
- 2) Make “-i” conditional on Config.OpenStdin?

