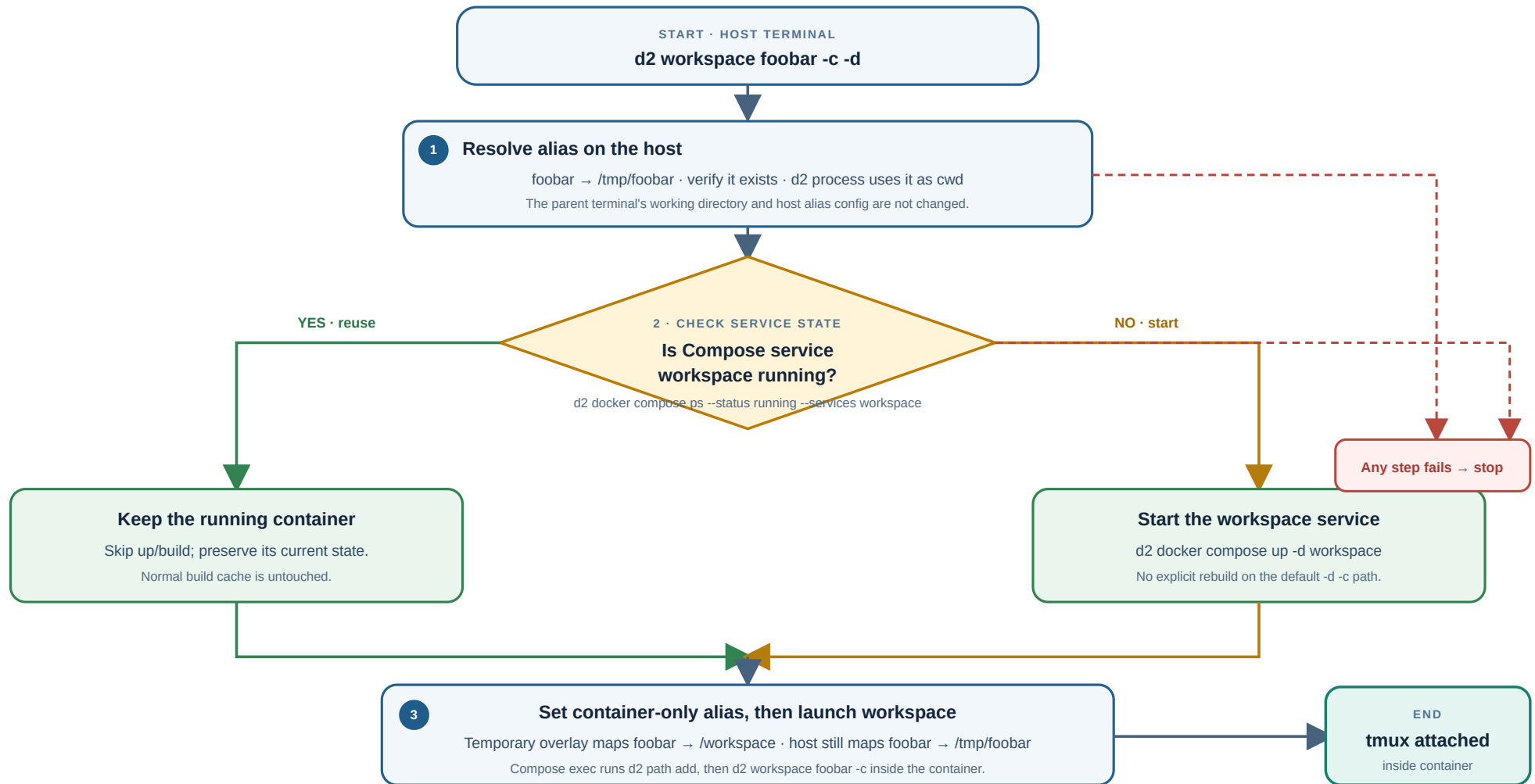


Problem 46 · Docker workspace

What happens when you run `d2 workspace foobar -c -d` and host alias `foobar` resolves to `/tmp/foobar`. This default path does not request a rebuild.



Required setup: the effective Compose config in `/tmp/foobar` defines service `workspace` and bind-mounts the project at `/workspace`. The command does not create that mount. Use `-C -d` to explicitly request a cached rebuild if stopped.

Command sequence and outcome

Compose commands are launched by the host-side d2 process from `/tmp/foobar`. Only the commands after `compose exec` run inside the container.

1 · Resolve the host alias

Host configuration resolves `foobar` to `/tmp/foobar`. The d2 process changes its own cwd there so Compose finds the project configuration. This does not change the parent shell directory.

3 · Start only when it is not running

```
d2 docker compose up -d workspace
```

Default `-d -c` does not request a rebuild. If you explicitly want a cached rebuild of a stopped service, use `d2 workspace foobar -C -d`, which runs `up -d --build workspace` and streams build output. A running service is reused even with `-C`.

5 · Start or attach to the inner workspace

```
d2 docker compose exec -it -e  
DEVELOPER_DASHBOARD_CONFIG_OVERLAY=/dev/shm/developer-  
dashboard-workspace-config workspace d2 workspace foobar -c
```

Inside the container, the alias resolves to `/workspace`; d2 creates or attaches to the `foobar` tmux session.

2 · Check whether workspace is running

```
d2 docker compose ps --status running --services workspace
```

If it reports workspace, reuse it. A failed probe is an error; startup must not guess.

4 · Register the alias inside the container

```
d2 docker compose exec -e  
DEVELOPER_DASHBOARD_CONFIG_OVERLAY=/dev/shm/developer-  
dashboard-workspace-config workspace d2 path add foobar  
/workspace
```

The overlay is container-side only. Host config remains `foobar` → `/tmp/foobar`.

What you get

Your terminal attaches to tmux running inside the Compose workspace container. With the required bind mount, edits under `/workspace` are the host files under `/tmp/foobar`. Detaching returns you to the original host shell directory.

Important: the workspace Compose service must already be configured and must mount the project at `/workspace`. The outer command runs on the host; only the two `compose exec` commands run inside the container. The final `exec` allocates a TTY so tmux attaches to your terminal. If tmux's default socket directory is not writable, the command selects a writable `TMUX_TMPDIR` (for example `/dev/shm`).