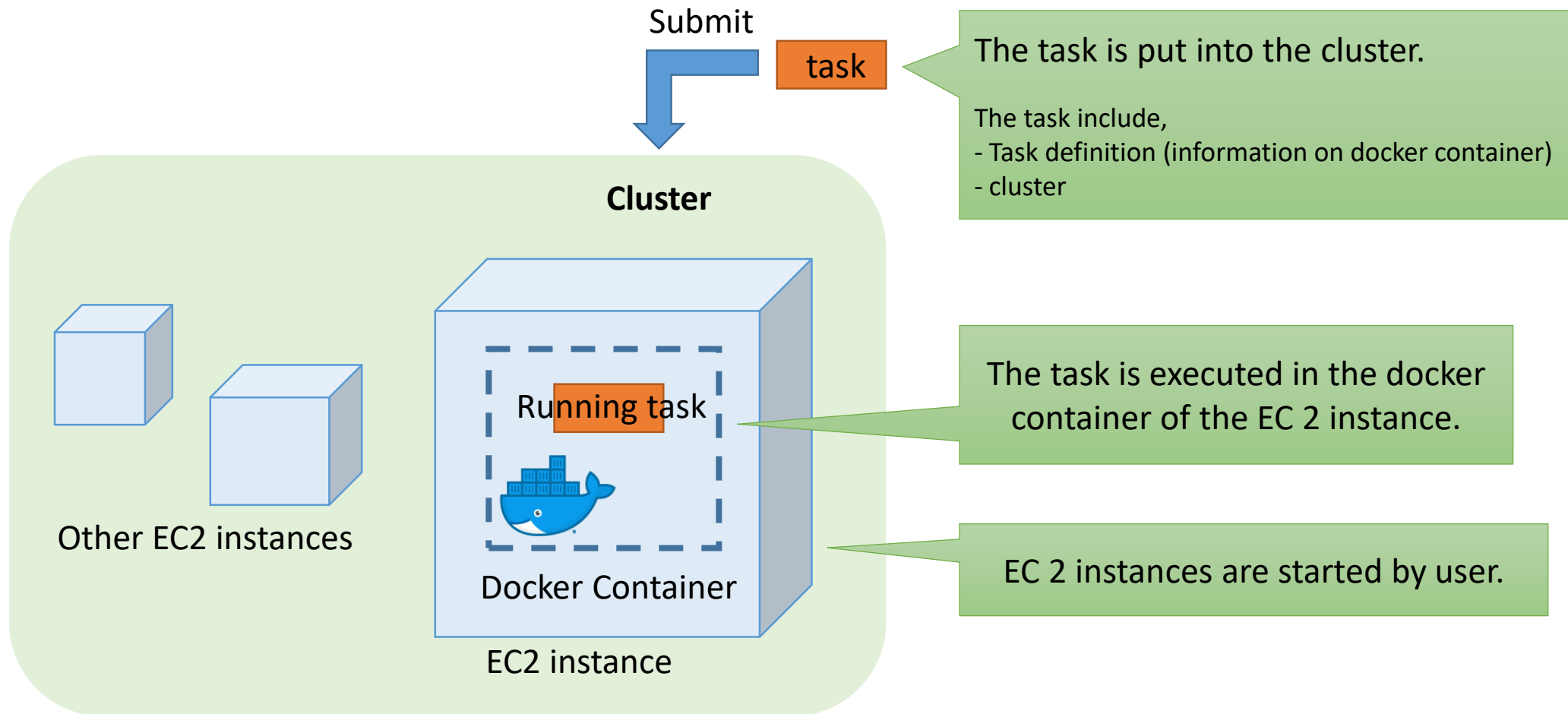


# Amazon Elastic Container Service (Amazon ECS)

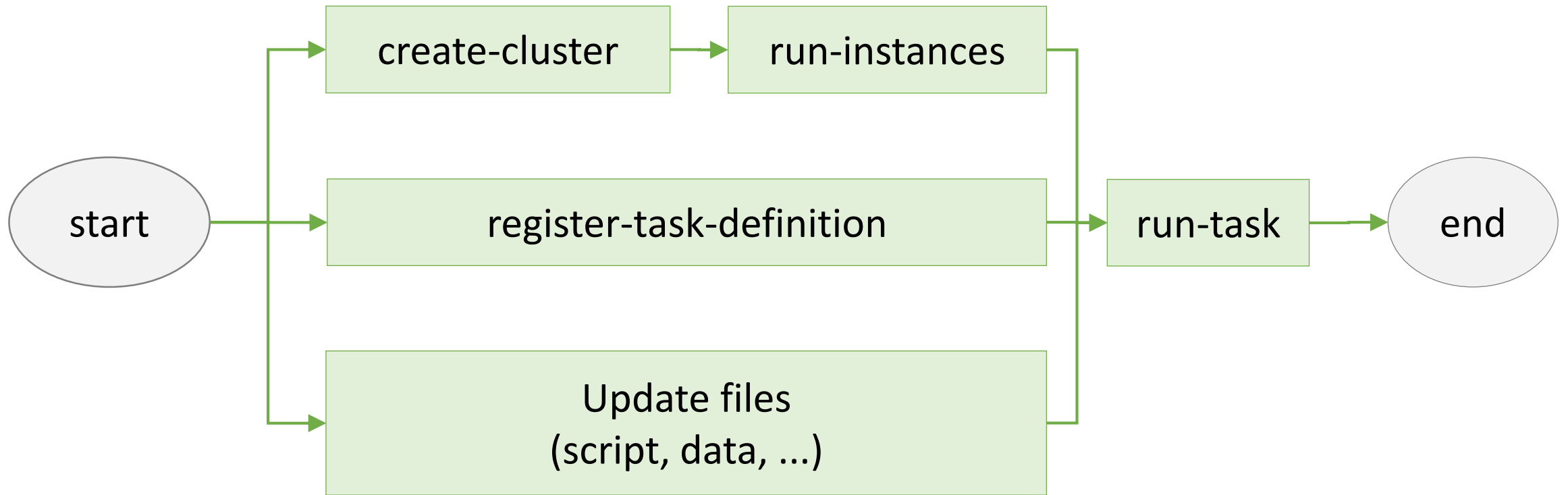
2018.4.2

Ai Okada

# Tasks and clusters in Amazon ECS

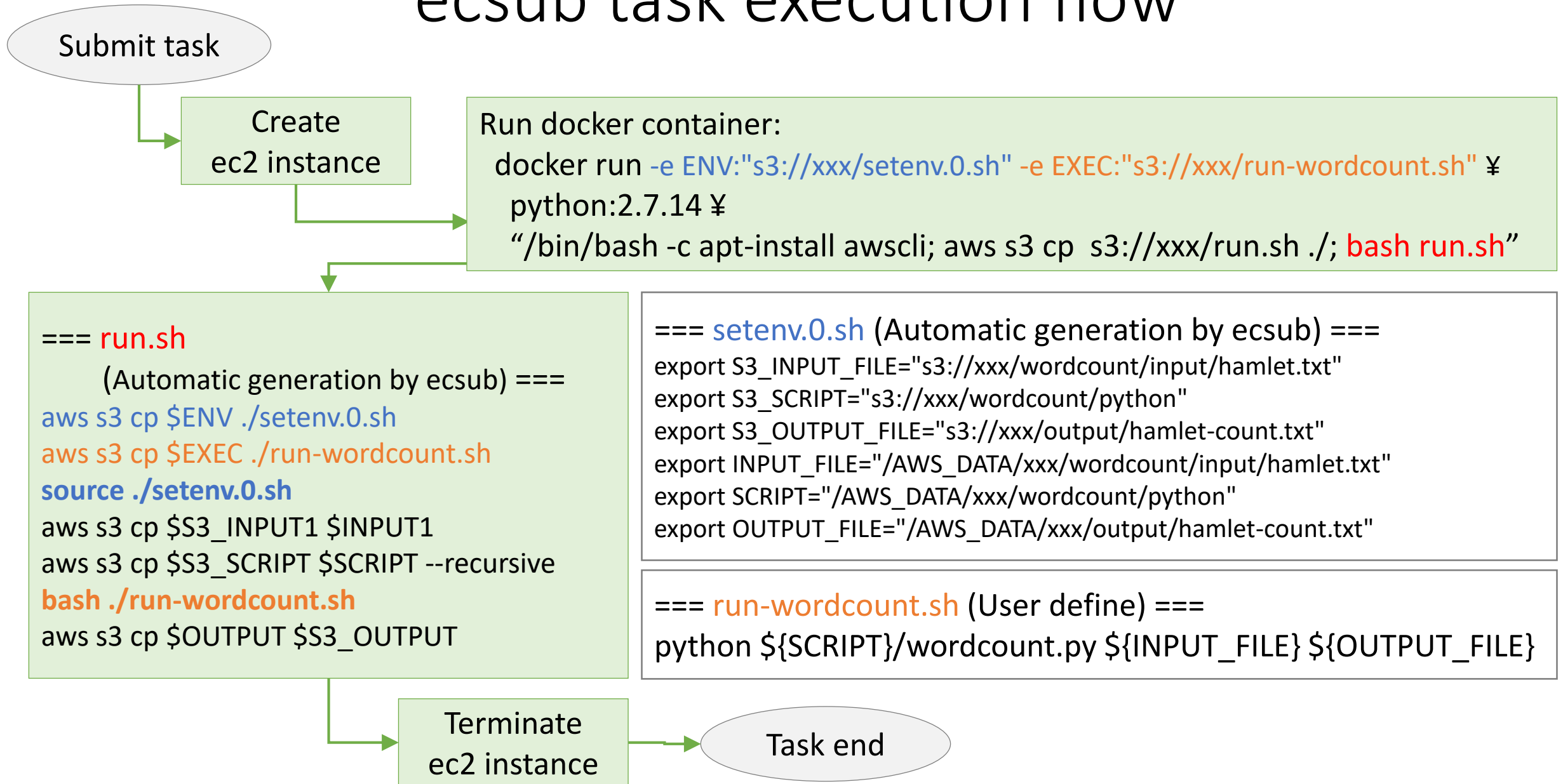


# AWS CLI: Command execution dependency



ecsub flow

# ecsub task execution flow



# ecsub data flow 1 - init

local directory

--tasks  
~/tasks.tsv

tasks.tsv

local directory  
or s3 bucket

--scripts  
~/run-wordcount.sh

run-  
wordcount.sh

s3 bucket

--aws-s3-bucket  
s3://bucket

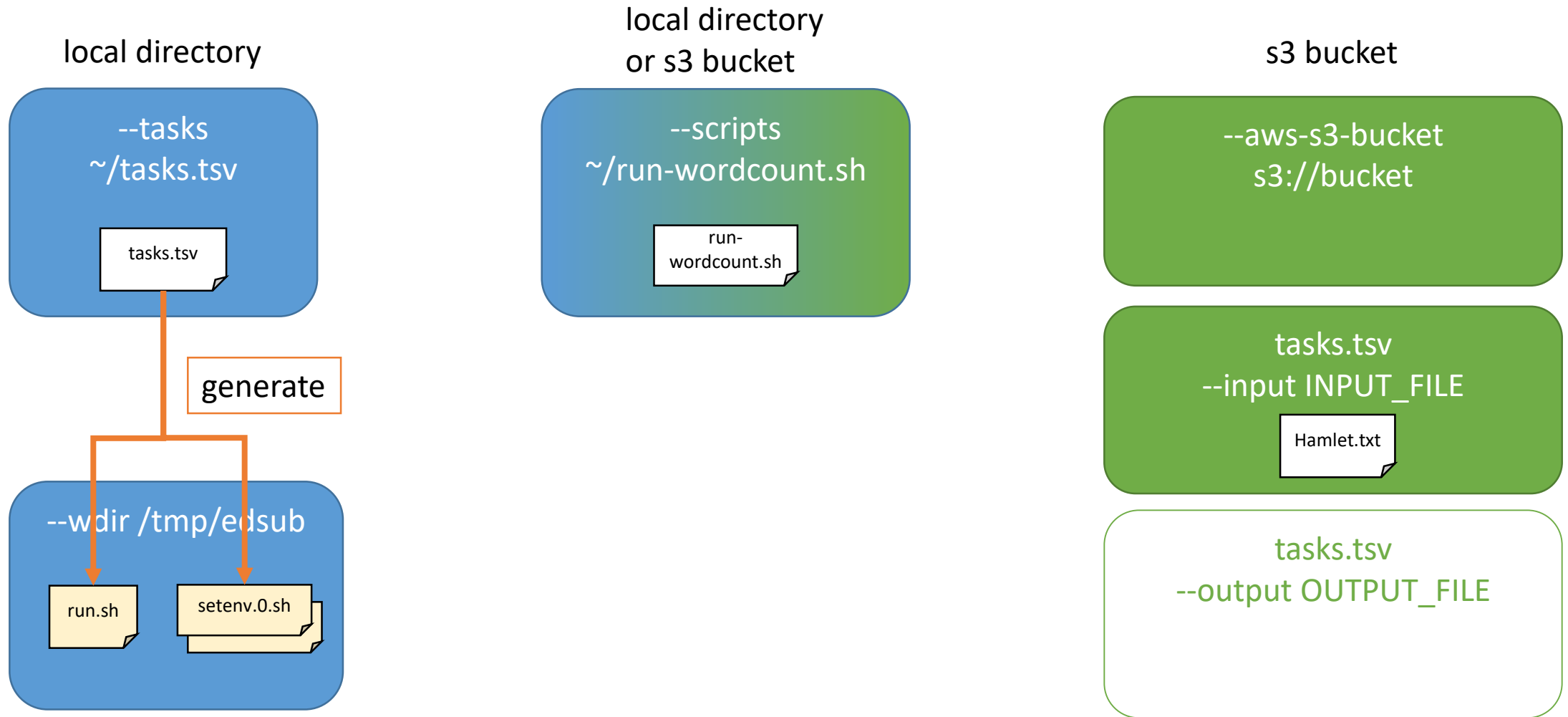
tasks.tsv  
--input INPUT\_FILE

Hamlet.txt

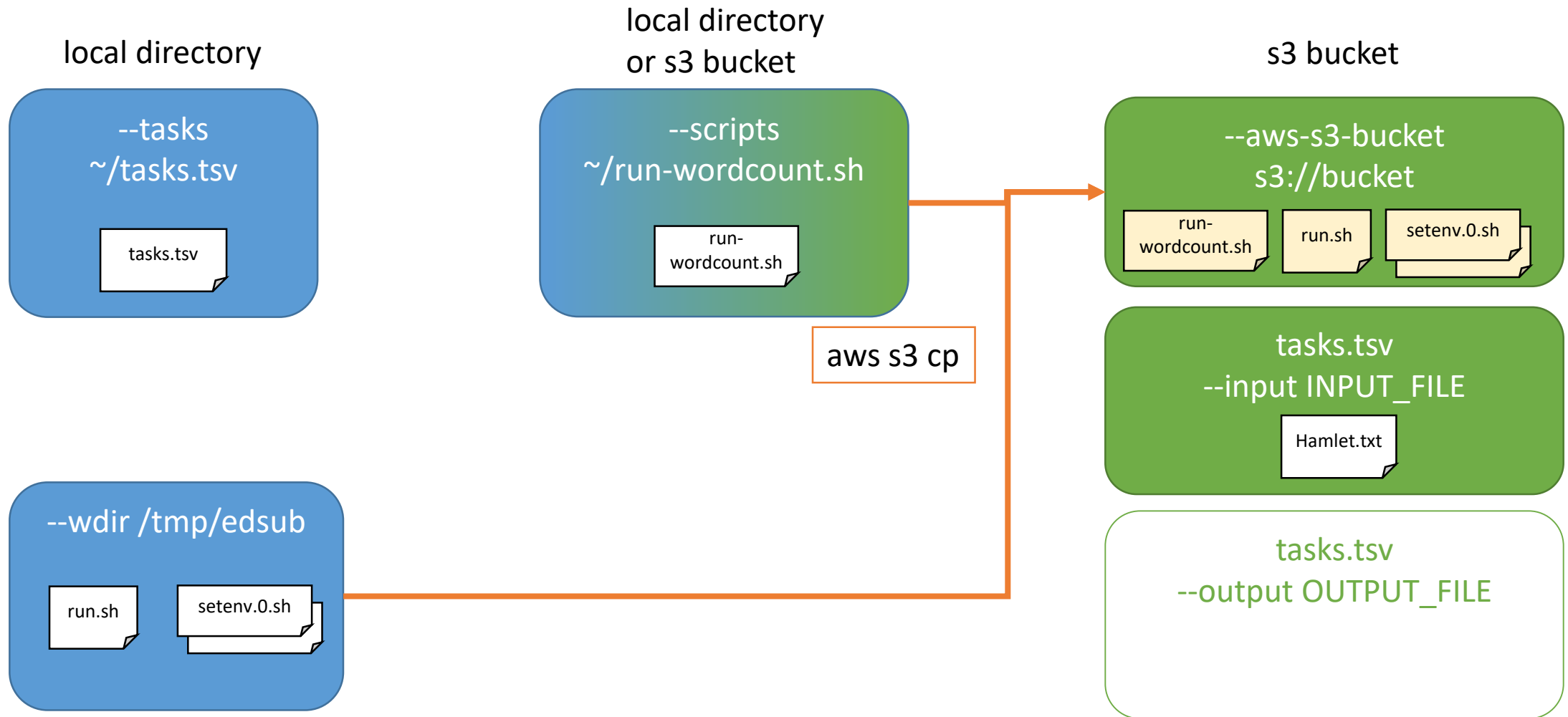
tasks.tsv  
--output OUTPUT\_FILE

--wdir /tmp/ecsub

# ecsub data flow 2 - generate scripts

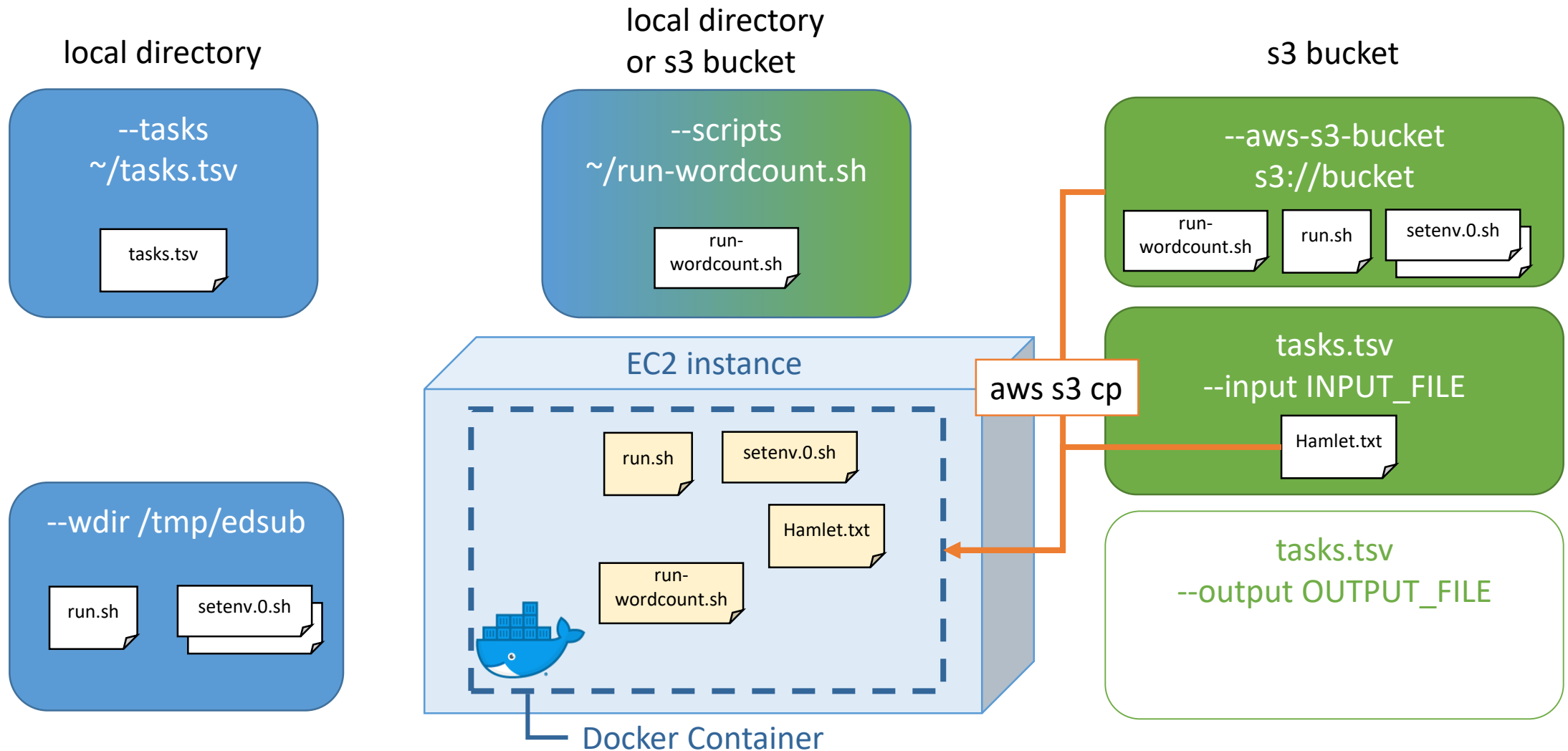


# ecsub data flow 3 - upload scripts to s3

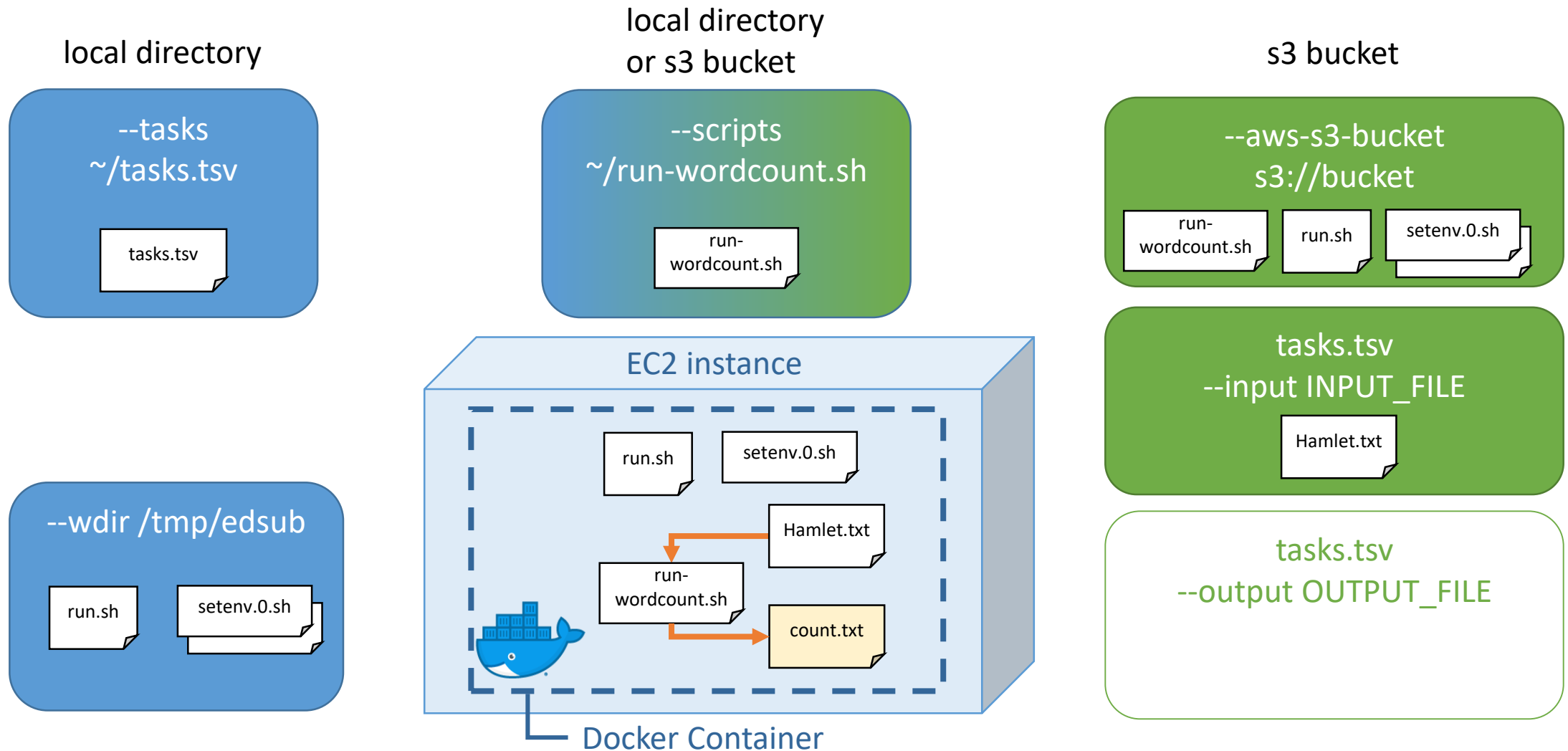




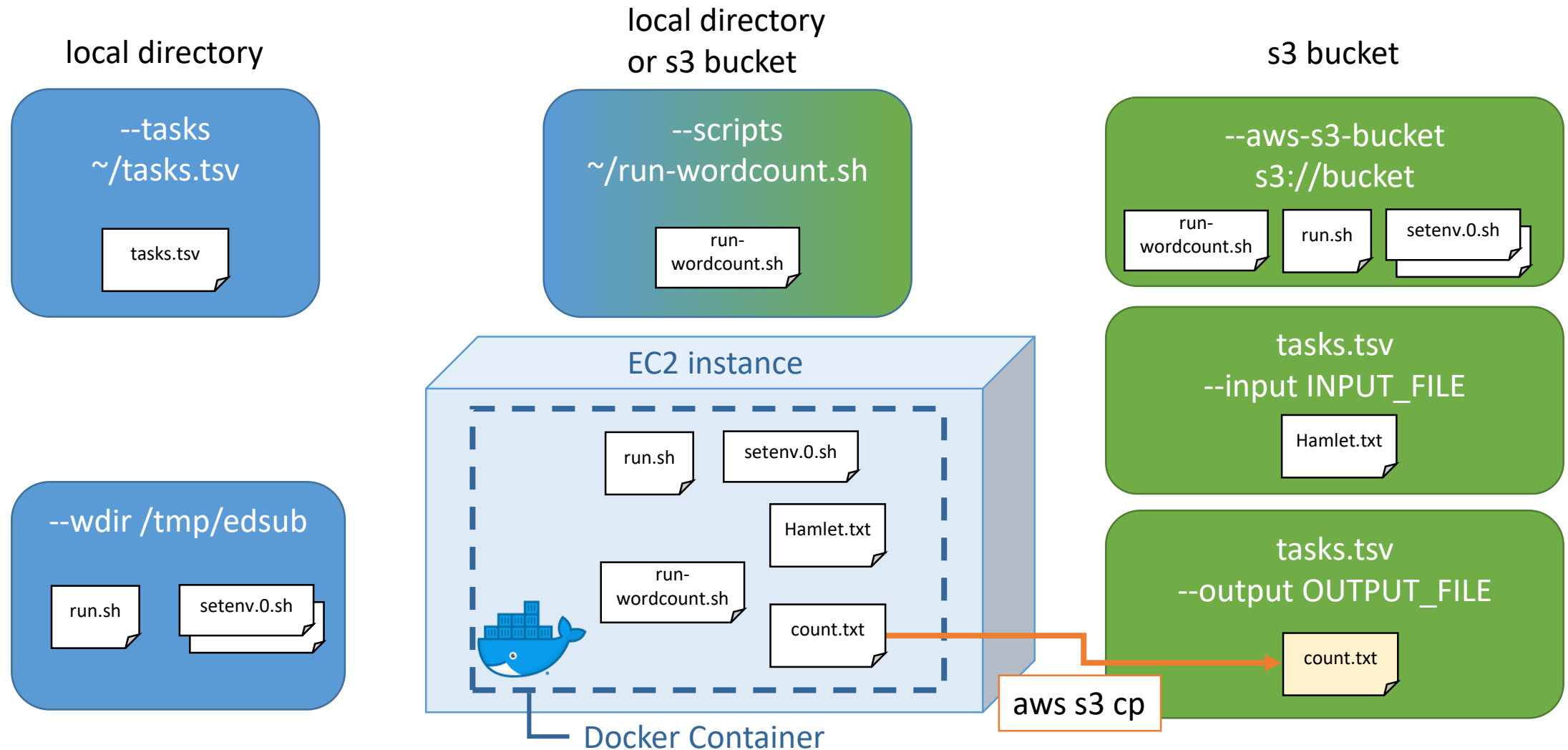
# ecsub data flow 4 - run docker container



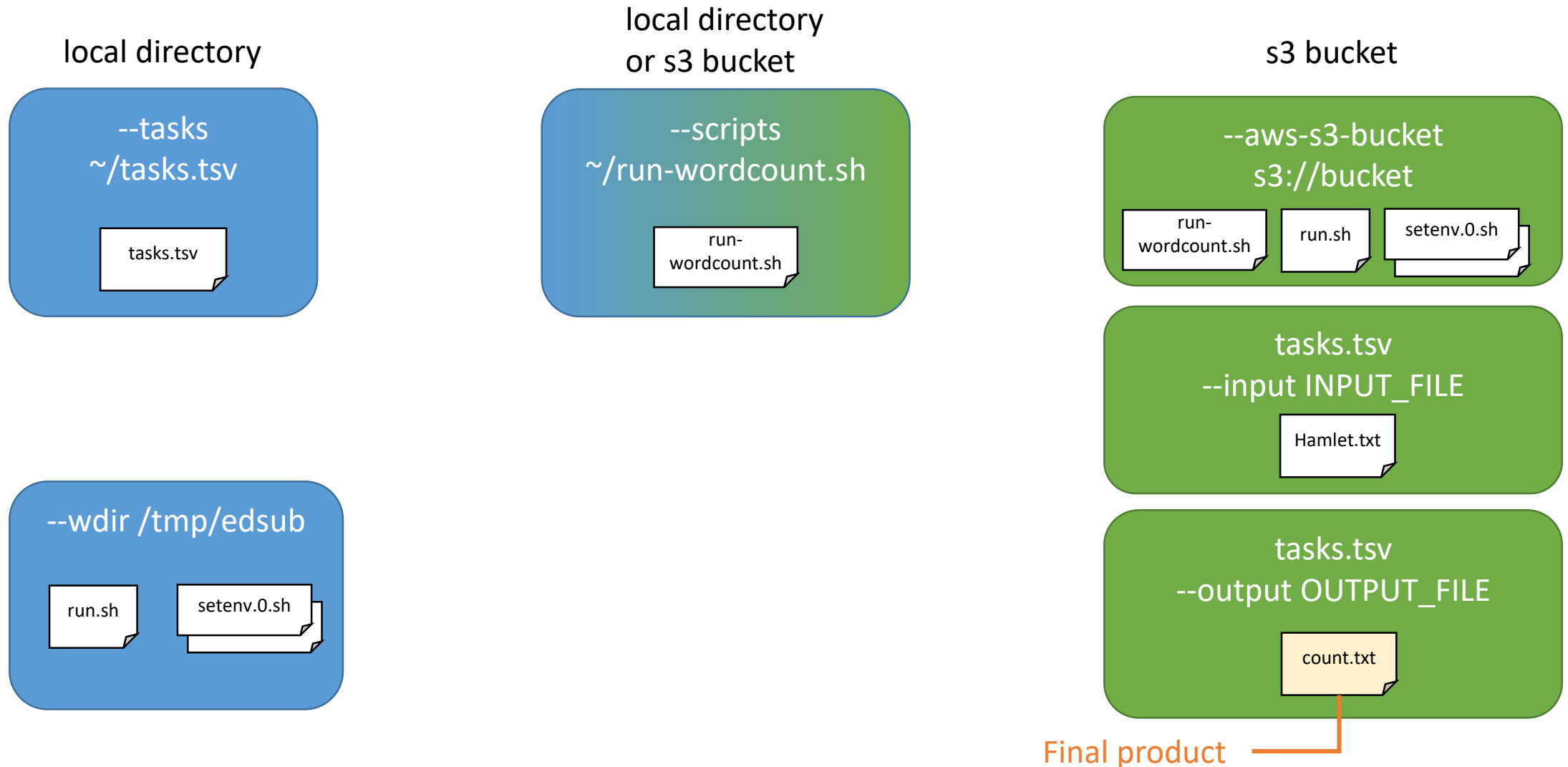
# ecsub data flow 5 - run task script



# ecsub data flow 6 - copy output-file to s3



# ecsub data flow 7 - terminate ec2 instance



ecsub tasks.tsv format

# format

- Separated with tab ("¥t")
- The first line is a header
  - --input [NAME] set s3 file-path, copy s3 -> container.
  - --input-recursive [NAME] set s3 directory-path, copy with recursive s3 -> container.
  - --output [NAME] set s3 file-path, copy container -> s3.
  - --output-recursive [NAME] set s3 directory-path, copy with recursive container -> s3.
  - --env [NAME] set environment variable.
- Not correspond to a comment line.

# example (./examples/tasks-wordcount.tsv)

--env NAME	--input INPUT_FILE	--input-recursive SCRIPT	--output OUTPUT_FILE
Hamlet	s3://ecsub-ohaio/wordcount/input/hamlet.txt	s3://ecsub-ohaio/wordcount/python	s3://ecsub-ohaio/output/hamlet-count.txt
Kinglear	s3://ecsub-ohaio/wordcount/input/kinglear.txt	s3://ecsub-ohaio/wordcount/python	s3://ecsub-ohaio/output/kinglear-count.txt