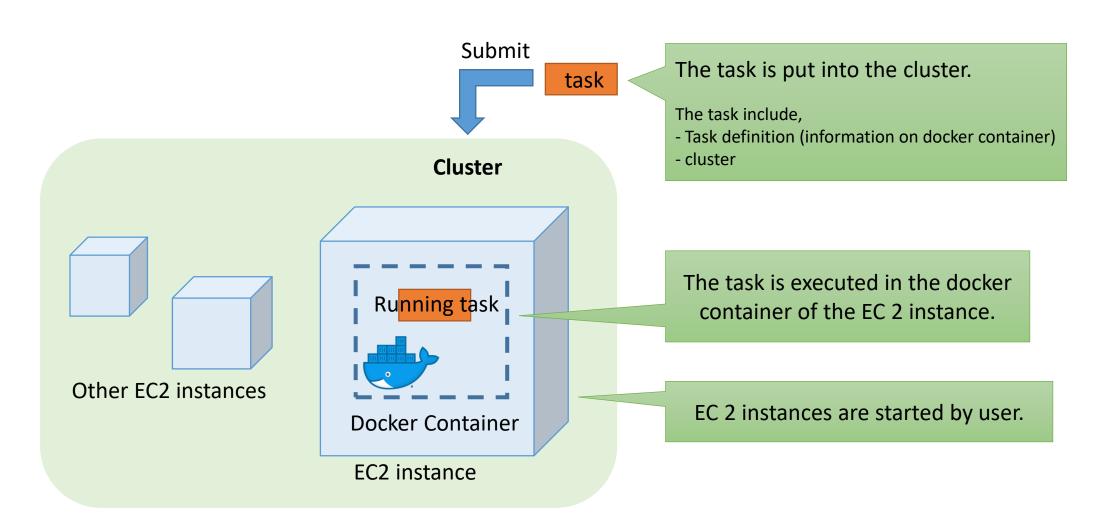
# Amazon Elastic Container Service (Amazon ECS)

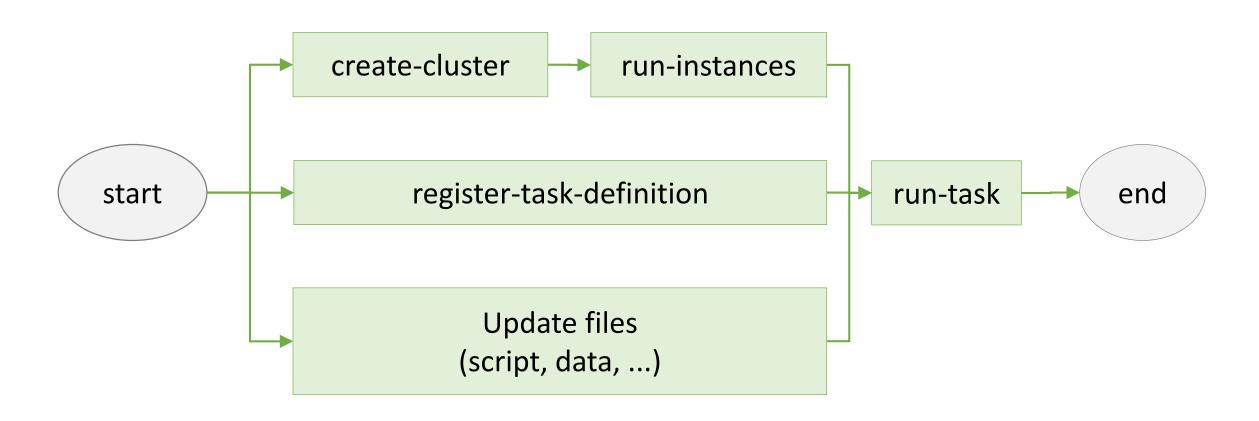
2018.4.2

Ai Okada

#### Tasks and clusters in Amazon ECS



## AWS CLI: Command execution dependency



# ecsub flow

### ecsub task execution flow

Submit task

Create ec2 instance

Run docker container:

docker run -e ENV:"s3://xxx/setenv.0.sh" -e EXEC:"s3://xxx/run-wordcount.sh" ¥ python:2.7.14 ¥

"/bin/bash -c apt-install awscli; aws s3 cp s3://xxx/run.sh ./; bash run.sh"

```
=== run.sh

(Automatic generation by ecsub) ===
aws s3 cp $ENV ./env.sh
aws s3 cp $EXEC ./exec.sh
source ./env.sh
aws s3 cp $S3_INPUT1 $INPUT1
aws s3 cp $S3_SCRIPT $SCRIPT --recursive
bash ./exec.sh
aws s3 cp $OUTPUT $S3_OUTPUT
```

=== setenv.0.sh (Automatic generation by ecsub) ===
export S3\_INPUT\_FILE="s3://xxx/wordcount/input/hamlet.txt"
export S3\_SCRIPT="s3://xxx/wordcount/python"
export S3\_OUTPUT\_FILE="s3://xxx/output/hamlet-count.txt"
export INPUT\_FILE="/AWS\_DATA/xxx/wordcount/input/hamlet.txt"
export SCRIPT="/AWS\_DATA/xxx/wordcount/python"
export OUTPUT\_FILE="/AWS\_DATA/xxx/output/hamlet-count.txt"

```
=== run-wordcount.sh (User define) ===
python ${SCRIPT}/wordcount.py ${INPUT_FILE} ${OUTPUT_FILE}
```

Terminate ec2 instance Task end

## ecsub data flow 1 - init

local directory

--tasks ~/tasks.tsv tasks.tsv local directory or s3 bucket

--scripts ~/run-wordcount.sh

wordcount.sh

--wdir /tmp/edsub

s3 bucket

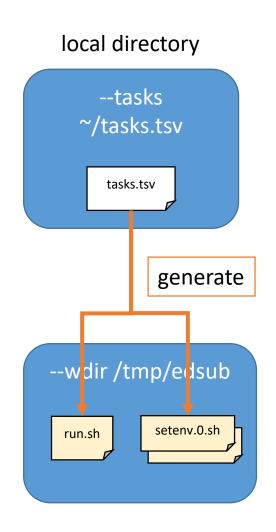
--aws-s3-bucket s3://bucket

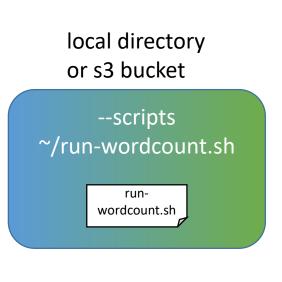
tasks.tsv --input INPUT\_FILE

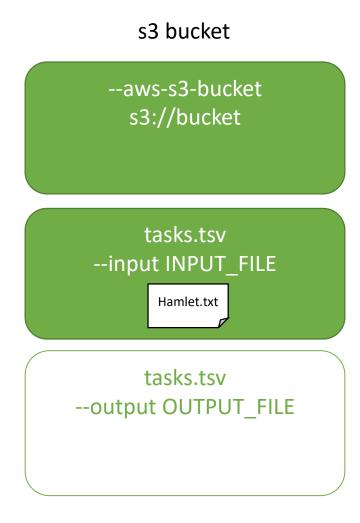
Hamlet.txt

tasks.tsv
--output OUTPUT\_FILE

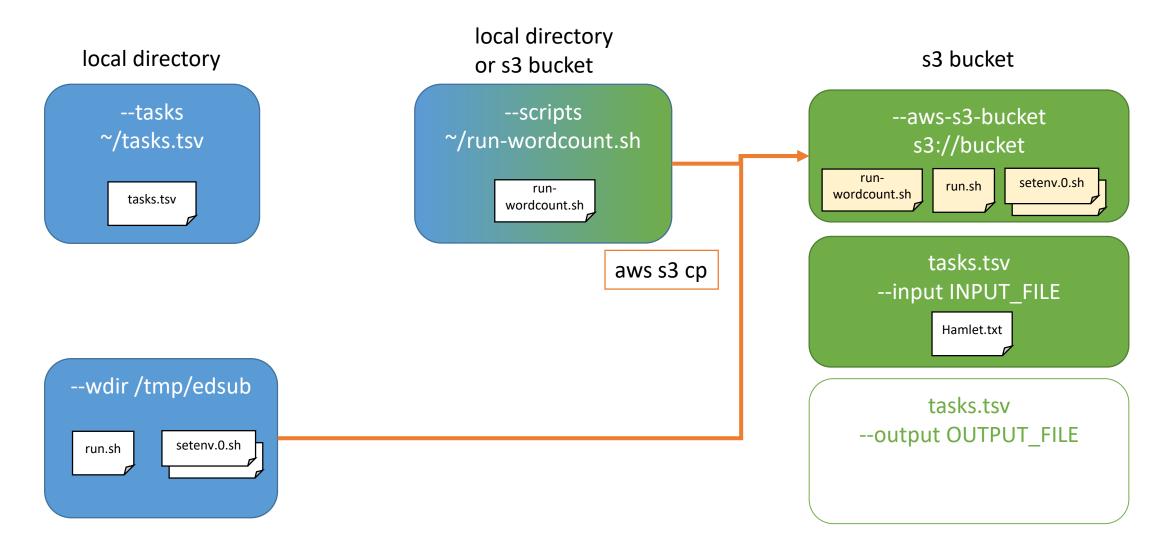
## ecsub data flow 2 - generate scripts



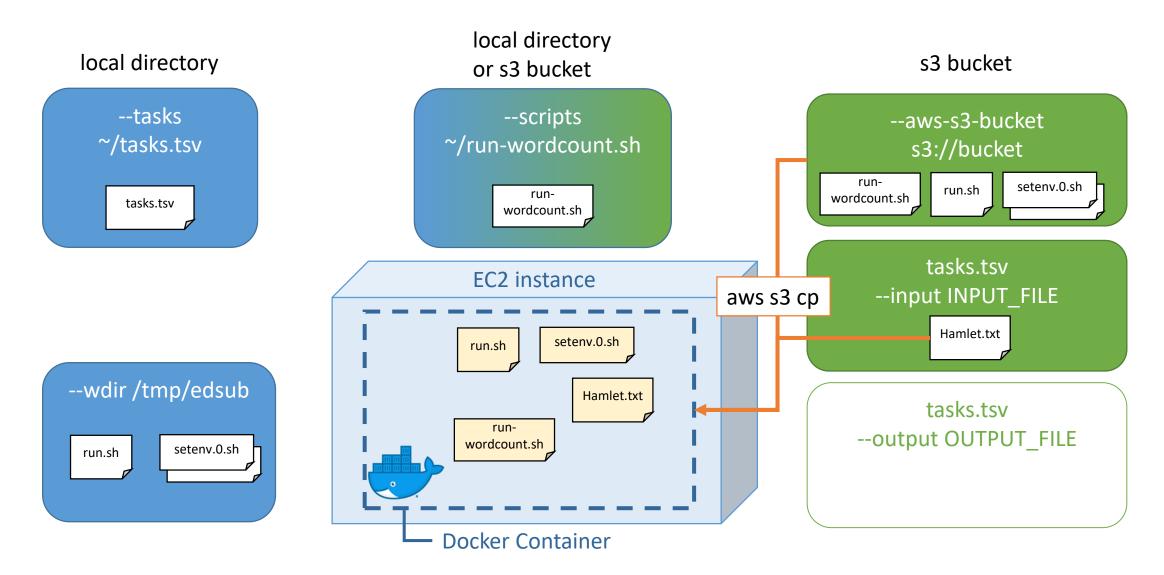




## ecsub data flow 3 - upload scripts to s3

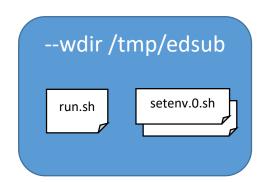


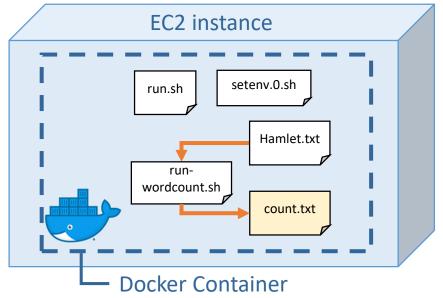
## ecsub data flow 4 - run docker container



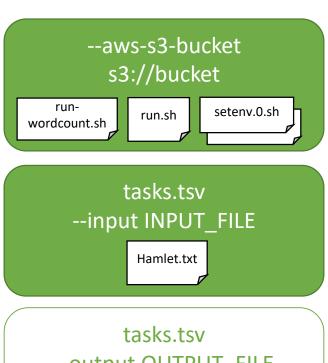
## ecsub data flow 5 - run task script

local directory local directory or s3 bucket --tasks --scripts ~/run-wordcount.sh ~/tasks.tsv runtasks.tsv wordcount.sh EC2 instance



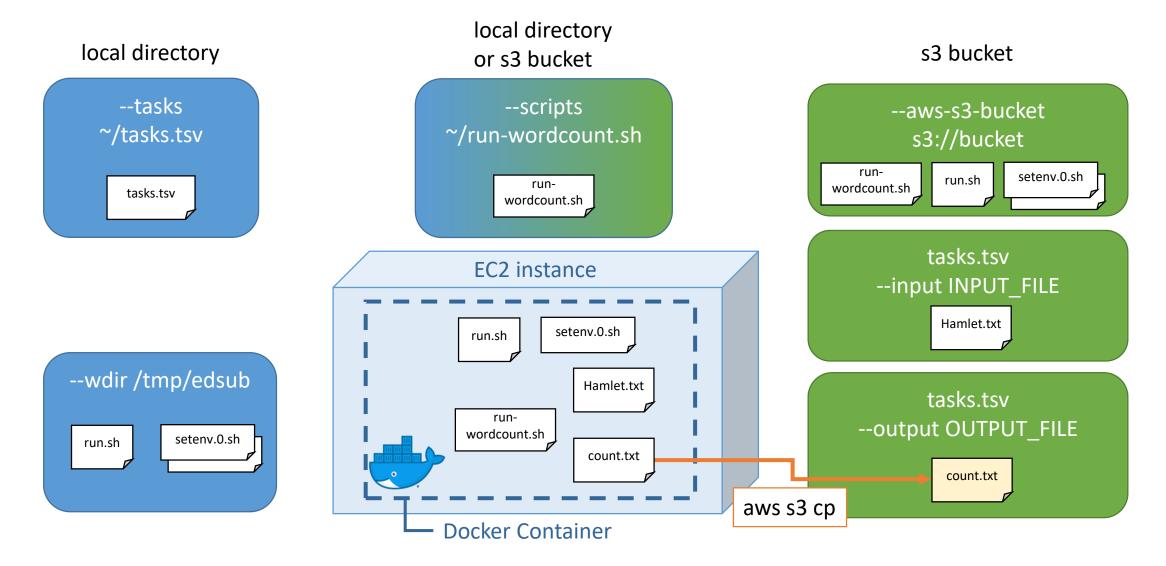


s3 bucket



--output OUTPUT FILE

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



#### ecsub data flow 7 - terminate ec2 instance

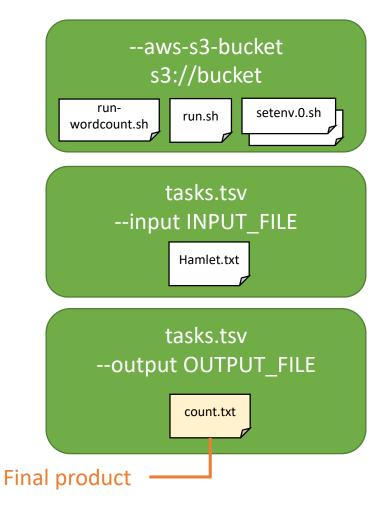
local directory

--tasks ~/tasks.tsv local directory or s3 bucket

--scripts ~/run-wordcount.sh

--wdir /tmp/edsub

s3 bucket



# ecsub tasks.tsv format

### format

- Separated with tab ("¥t")
- The first line is a header
  - --input [NAME] set s3 file-path, copy s3 to container.
  - --input-recursive [NAME] set s3 directory-path, copy s3 to container recursive.
  - --output [NAME] set s3 file-path, copy container to s3.
  - --output-recursive [NAME] set s3 directory-path, copy container to s3 recursive.
  - --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