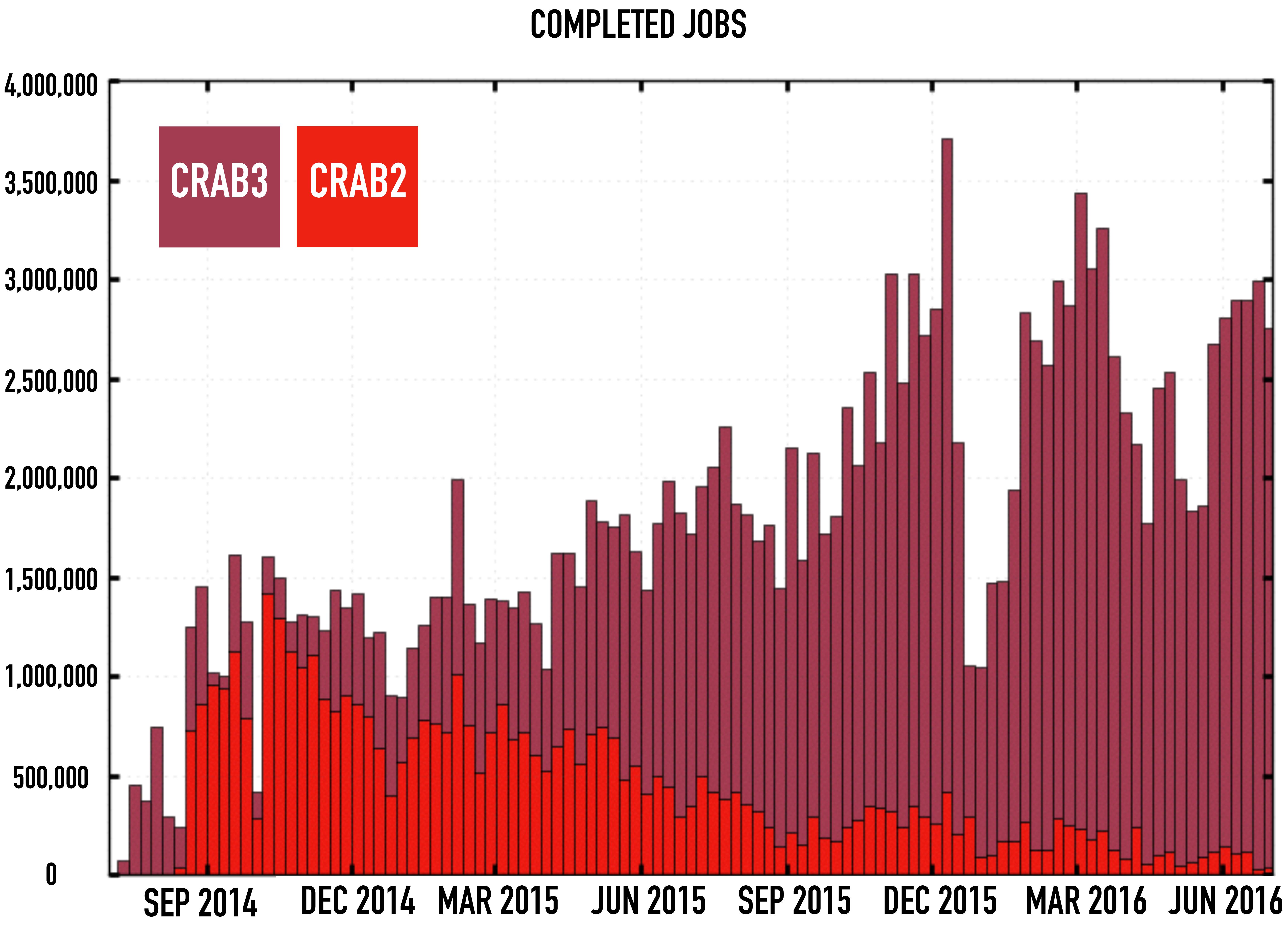
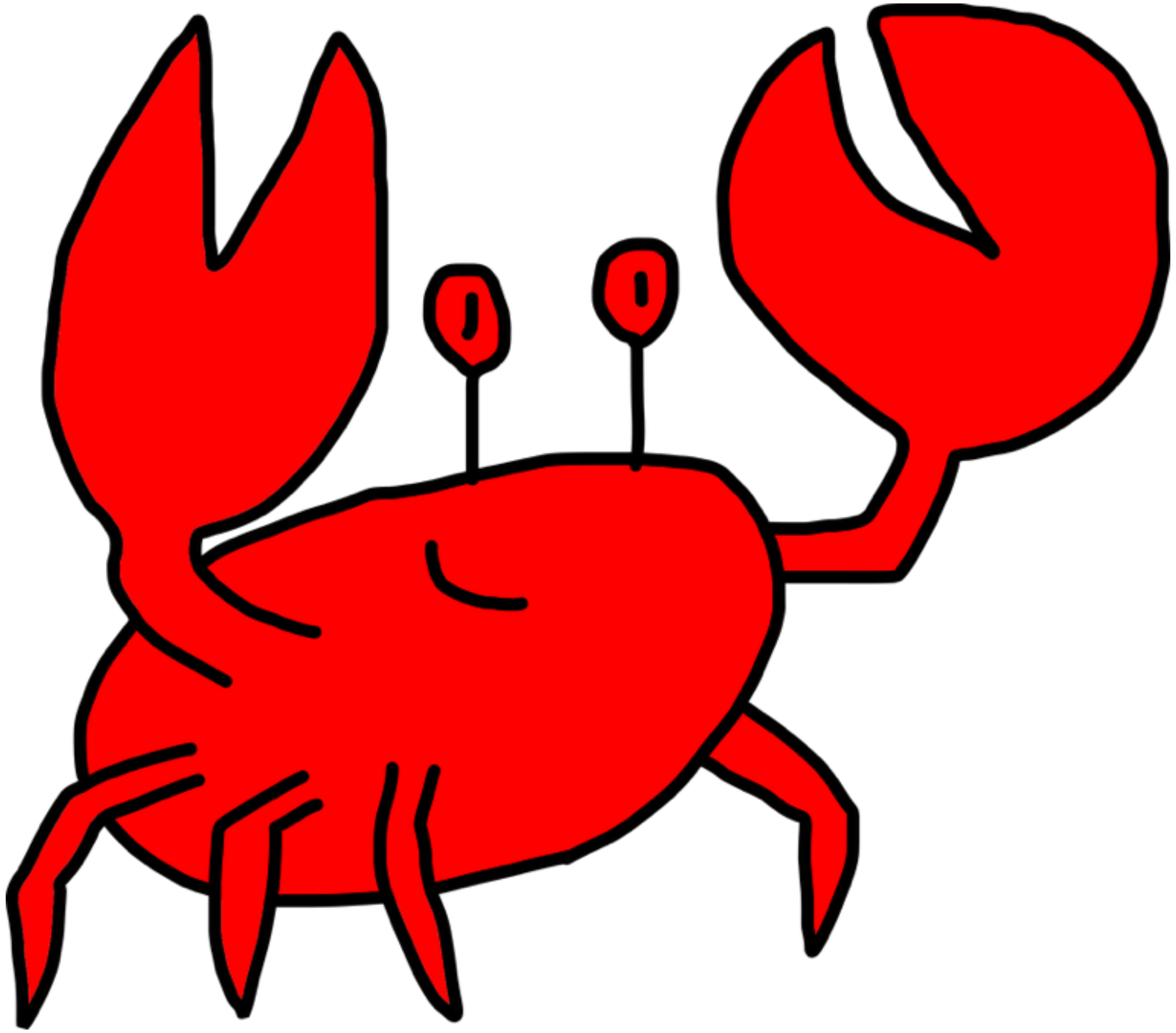
MATTHIAS WOLF, MARCO MASCHERONI, STEFANO BELFORTE, ERIC VAANDERING, JOSE HERNANDEZ, ANNA WOODARD, BRIAN BOCKELMAN

USING DAGMAN IN CRAB3 TO IMPROVE TASK SPLITTING FOR CMS USERS

WHAT IS CRAB3?

- Hundreds of physicists regularly submit analysis jobs to the Grid using a tool called CRAB3
- Architecture: lightweight user client and CRAB3 server which accepts user requests (“tasks”)
- CRAB3 manages ~3 million jobs per week





CONVENTIONAL USER WORKFLOW

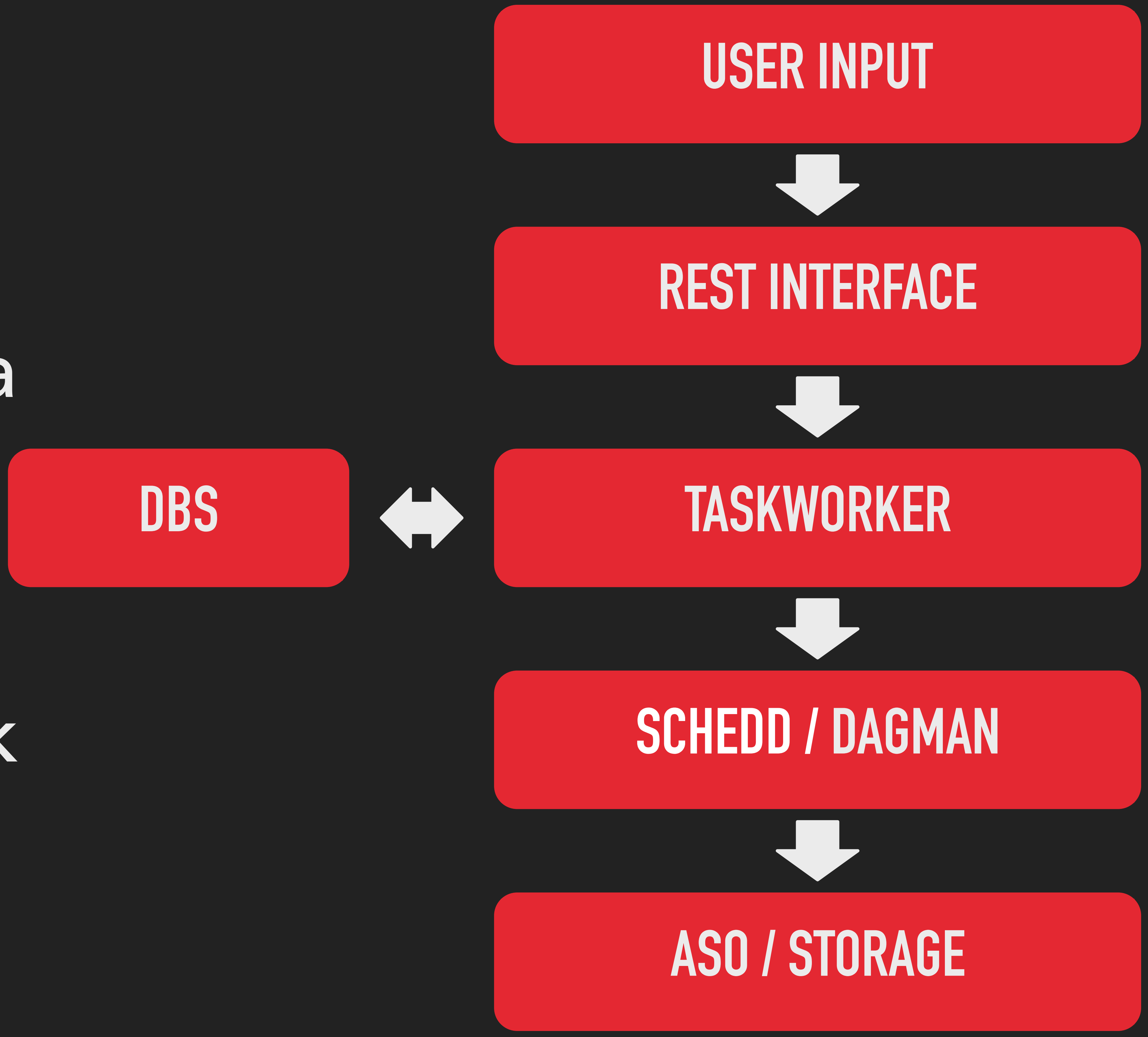
SPECIFICATION OF A CRAB3 ANALYSIS TASK

- User responsible for specifying:
  - Dataset (group of files) to process
  - Code to run
  - "Splitting parameters", i.e., how many units each job should process
    - Units = files, lumi sections, or events

CRAB3 SERVER'S HANDLING OF THE TASK

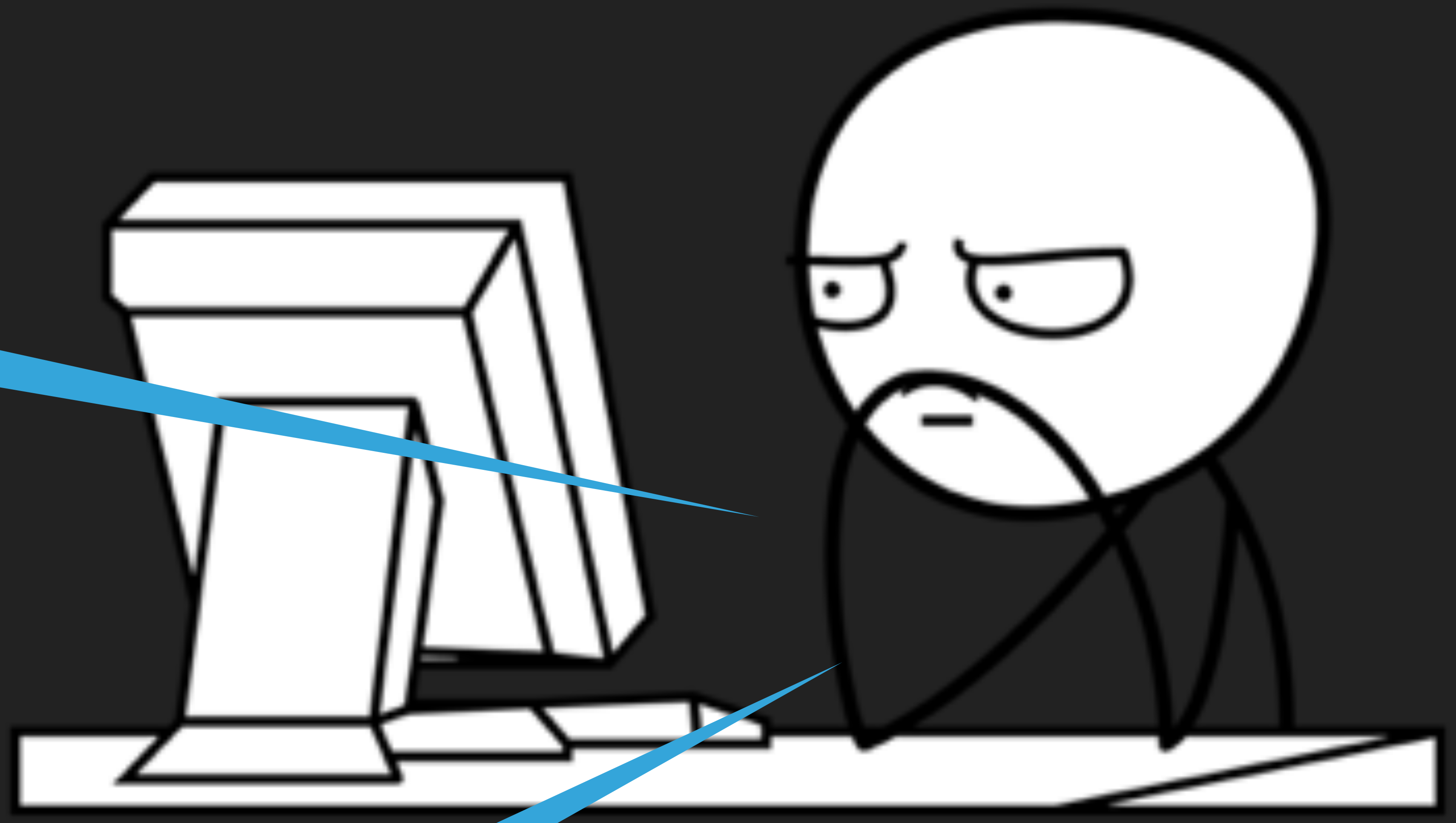
Server initializes task and:

- Pulls dataset metadata in from the CMS dataset bookkeeping catalog (DBS)
- Splits input dataset into jobs using metadata + splitting parameters
- Creates a single-depth DAG:
  - DAG=Directed Acyclic Graph, keeps track of dependencies between jobs
  - Single depth=no interdependency between jobs
- DAG submitted to scheduler for processing



SPLITTING IS HARD FOR USERS

WHY DID YOU SUBMIT 10,000 FIVE MINUTE JOBS?!?



CRAB3 OPERATOR

UHM….

HOW LONG DOES YOUR CODE TAKE TO RUN PER EVENT? HOW BIG IS YOUR DATASET?

EHM….

PHYSICIST

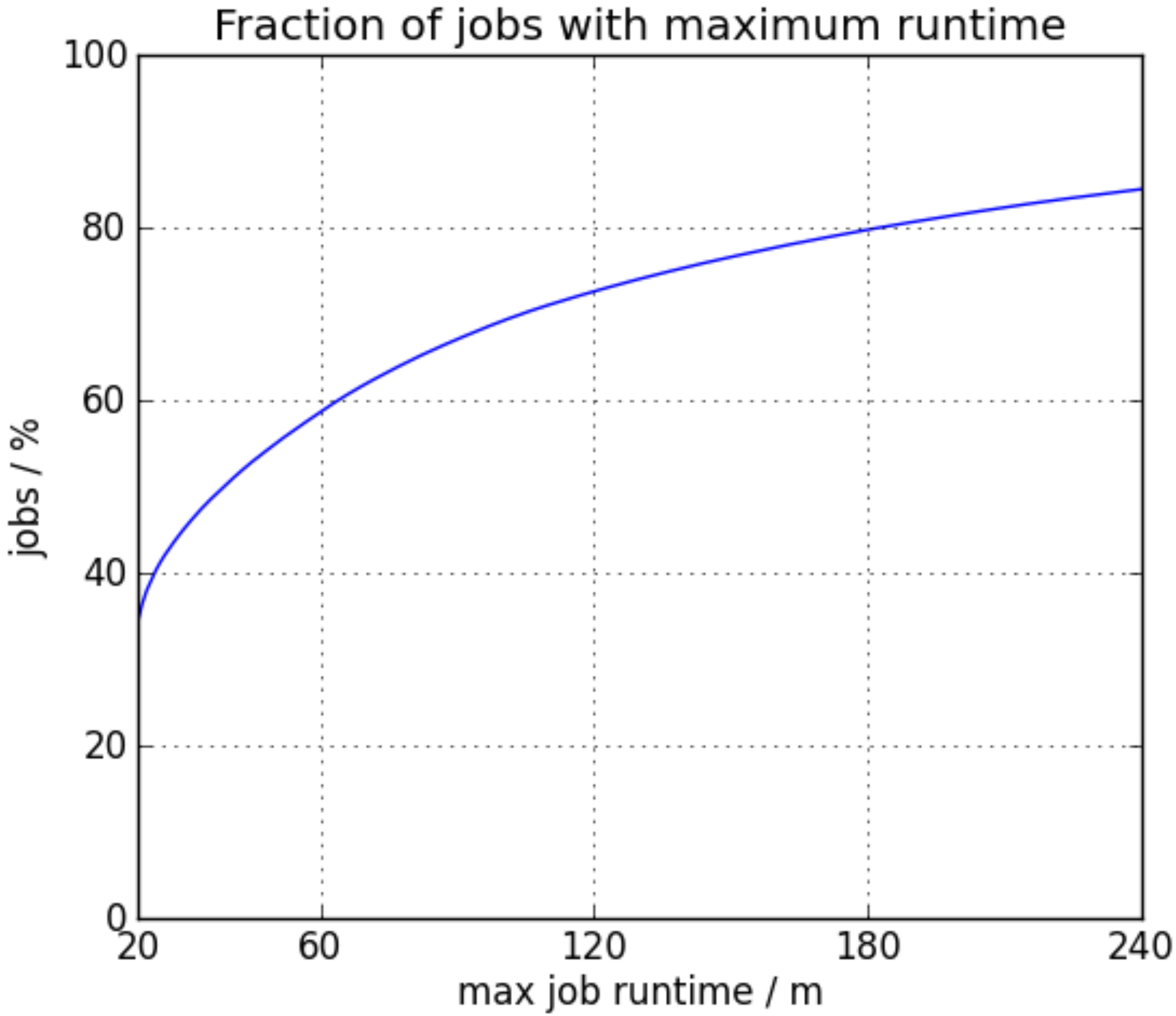
SPLITTING IS HARD FOR USERS…

- Difficult for users to provide optimal task splitting parameters
  - Code runtime may differ for each:
    - Code iteration
    - Dataset (different event complexity)
- Tendency for user to choose parameters which create many short jobs
  - Hard limit for jobs per task is quite high (10,000 jobs)
  - Users can get away with memory leaks with short jobs

…AND THEY DO IT POORLY!

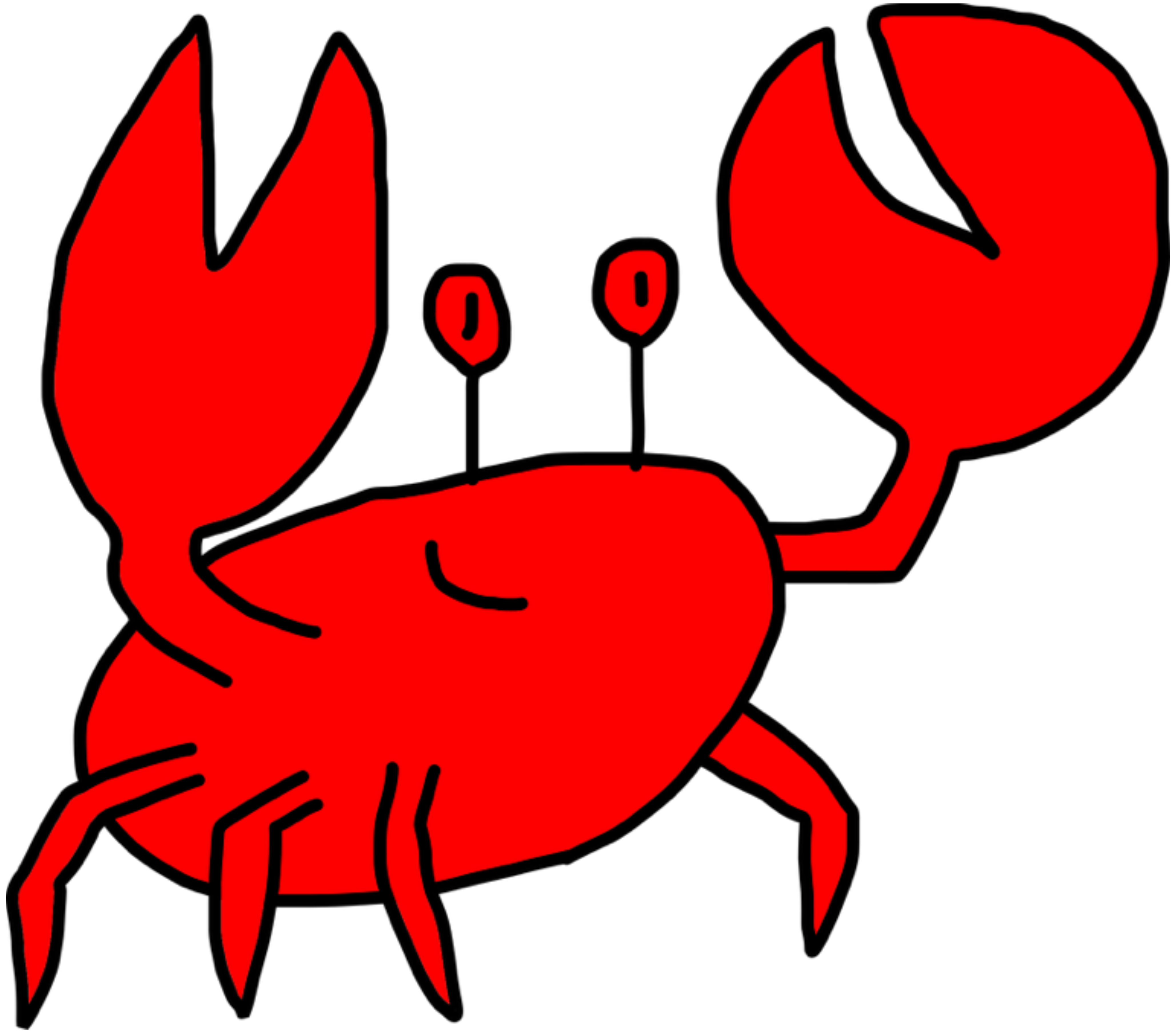
TOO MANY SHORT JOBS

- Almost 50% of the jobs have a runtime of ~40 minutes!
- Large number of small jobs causes excessive load on the schedulers and other central components.



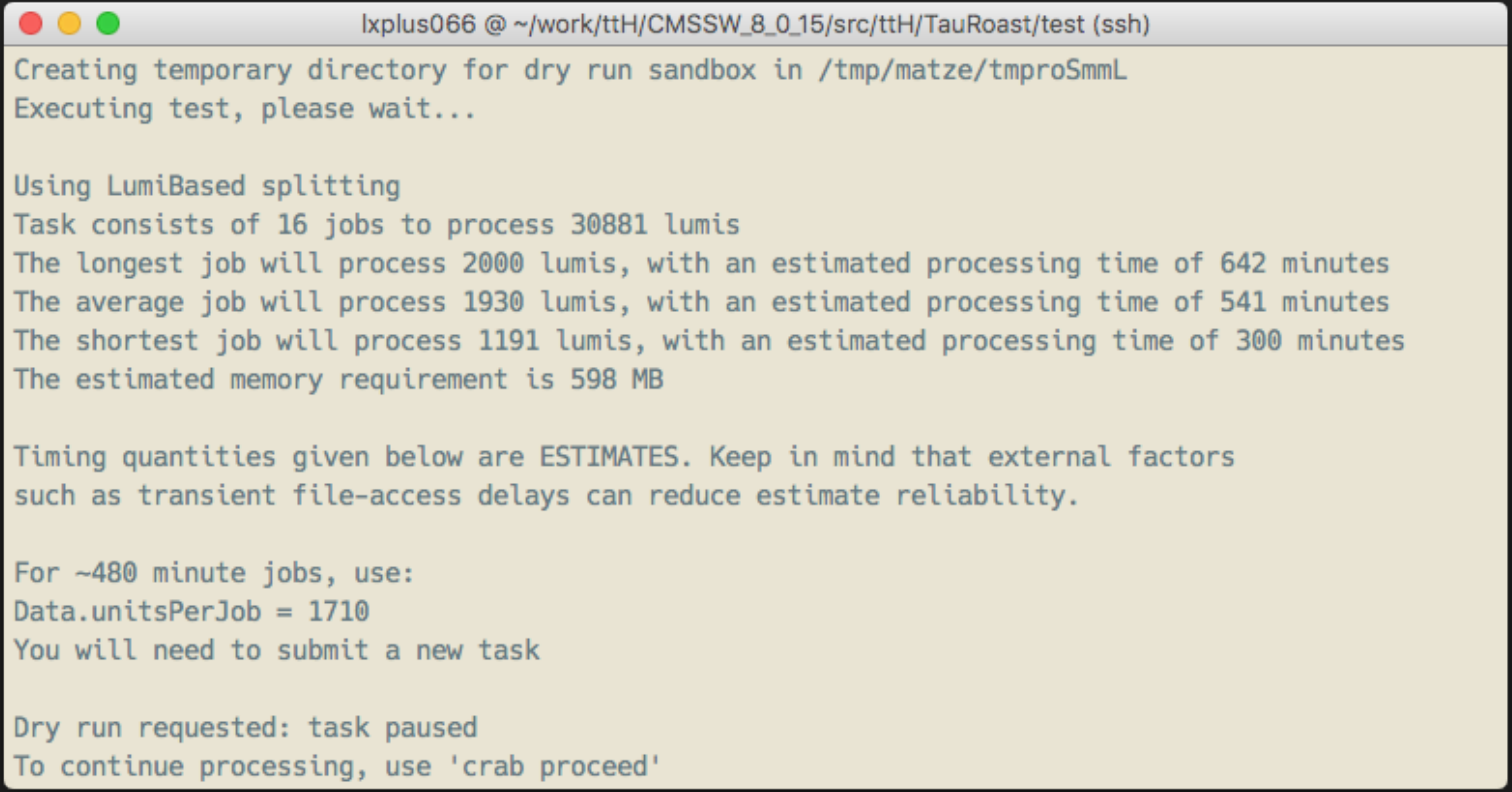
THERE’S NO ADVANCED PHYSICS IN SPLITTING.

CAN CRAB3 DO A BETTER JOB THAN THE USERS?



DRY-RUN FOR BETTER SPLITTING

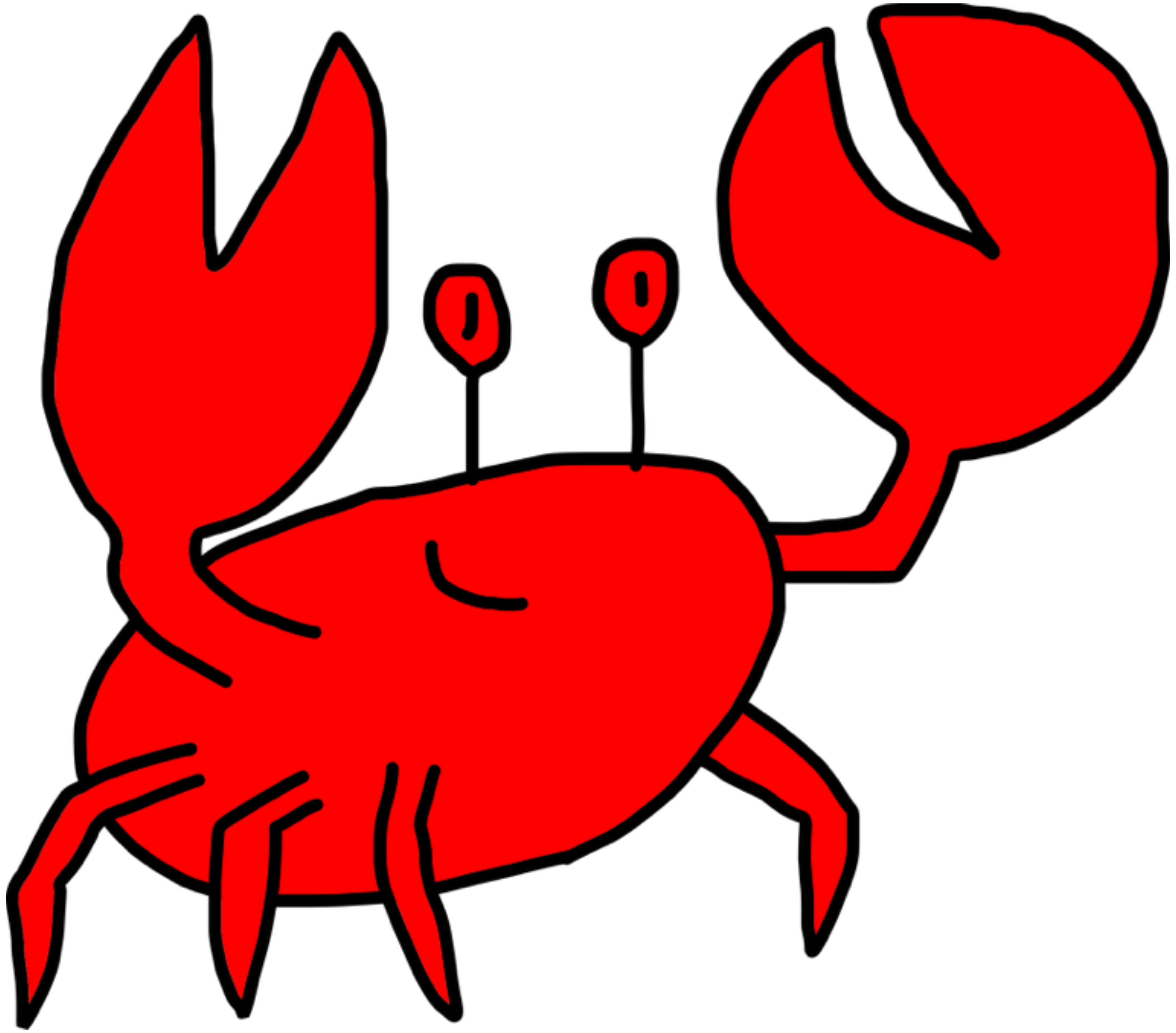
IMPROVING THE SPLITTING EXPERIENCE: DRY-RUN



lxplus066 @ ~/work/ttH/CMSSW_8_0_15/src/ttH/TauRoast/test (ssh)

```
Creating temporary directory for dry run sandbox in /tmp/matze/tmproSmmL
Executing test, please wait...

Using LumiBased splitting
Task consists of 16 jobs to process 30881 lumis
The longest job will process 2000 lumis, with an estimated processing time of 642 minutes
The average job will process 1930 lumis, with an estimated processing time of 541 minutes
The shortest job will process 1191 lumis, with an estimated processing time of 300 minutes
The estimated memory requirement is 598 MB

Timing quantities given below are ESTIMATES. Keep in mind that external factors
such as transient file-access delays can reduce estimate reliability.

For ~480 minute jobs, use:
Data.unitsPerJob = 1710
You will need to submit a new task

Dry run requested: task paused
To continue processing, use 'crab proceed'
```

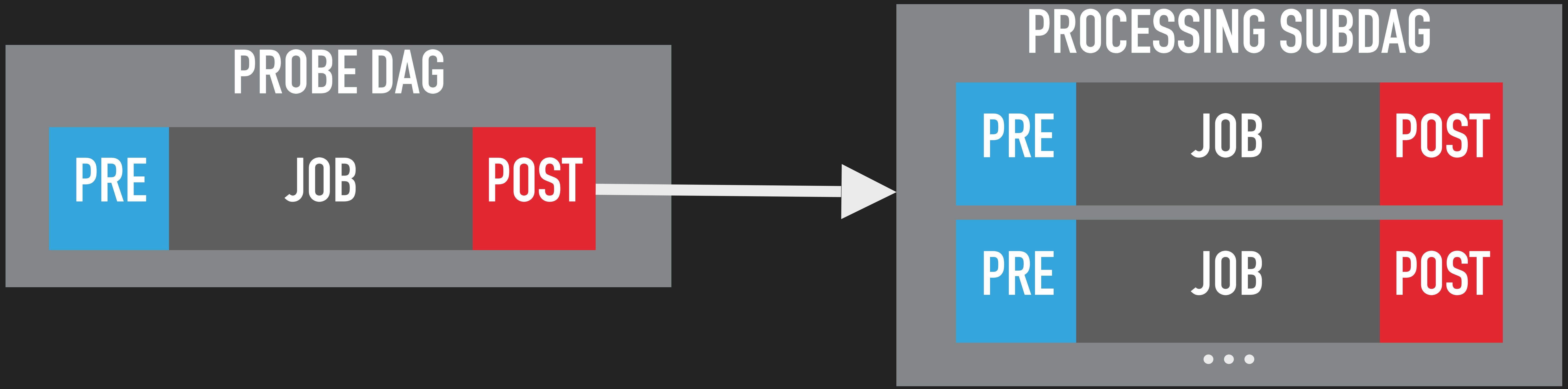
Run a small test job locally to provide timing estimate and recommend splitting parameters!



AUTOMATIC SPLITTING

USING DAGMAN TO DO THE SPLITTING AUTOMATICALLY

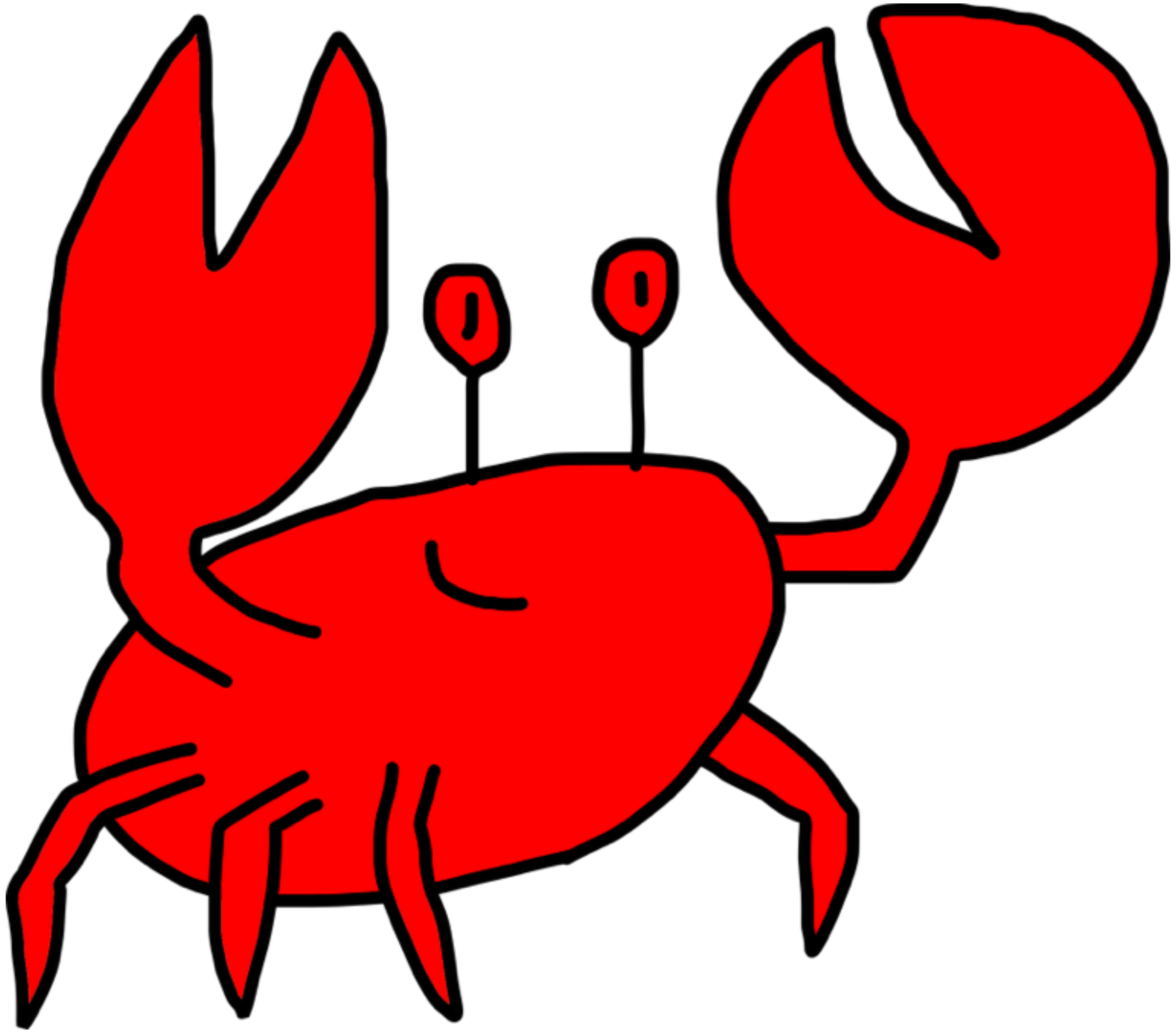
- TaskWorker sends a single node DAG (“probe” for timing)
- The probe’s post processing step then:
  - Performs splitting
  - Creates a SubDAG with processing jobs



BENEFITS

For users– removes guesswork: only need to configure desired runtime

For CRAB3 operations– allows enforcement of a minimum job runtime



TAIL-SPLITTING FOR FASTER TASK COMPLETION

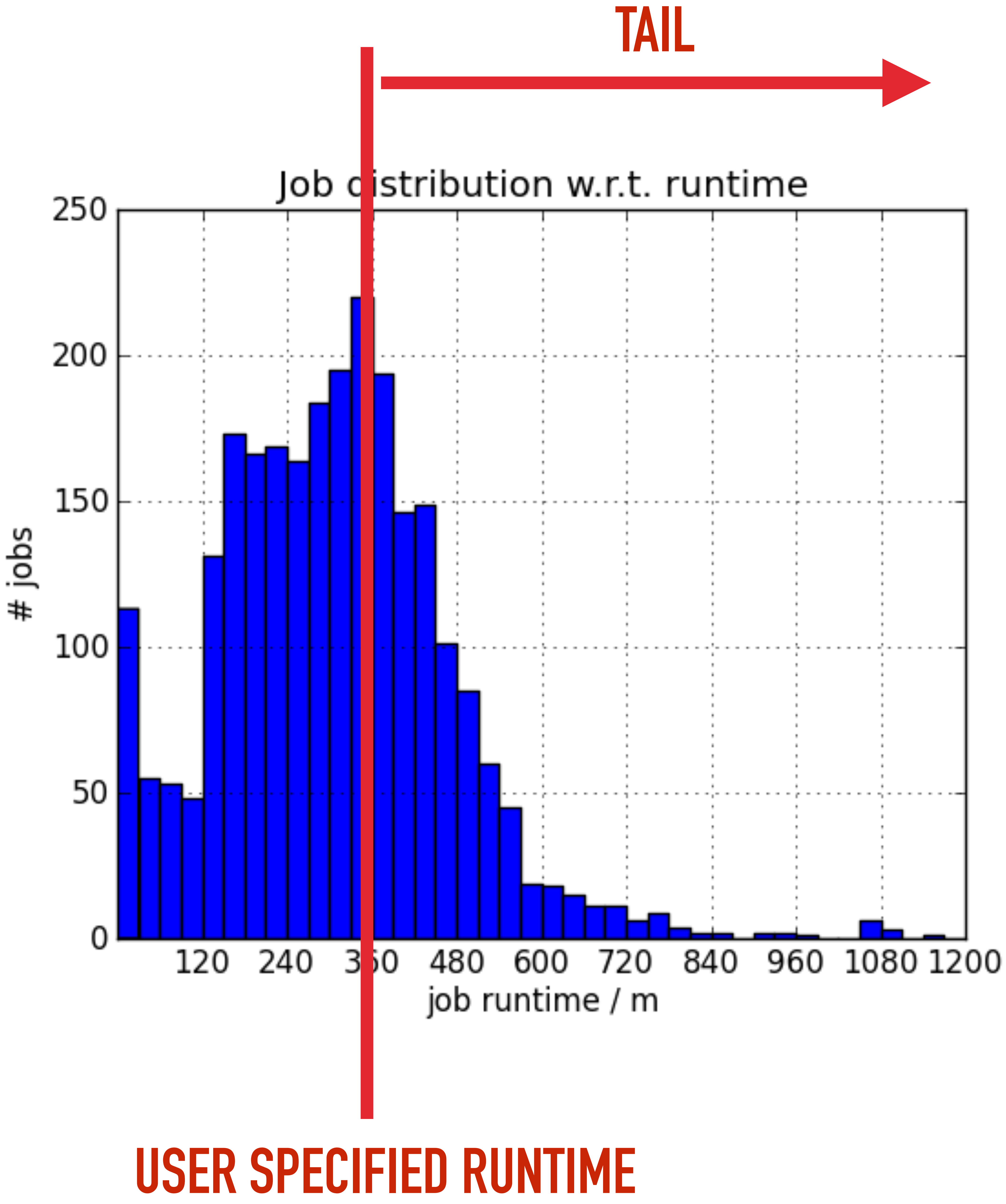
CONSTRAINING TASK RUNTIME

Not all machines/sites equal: some jobs still take more time than expected.

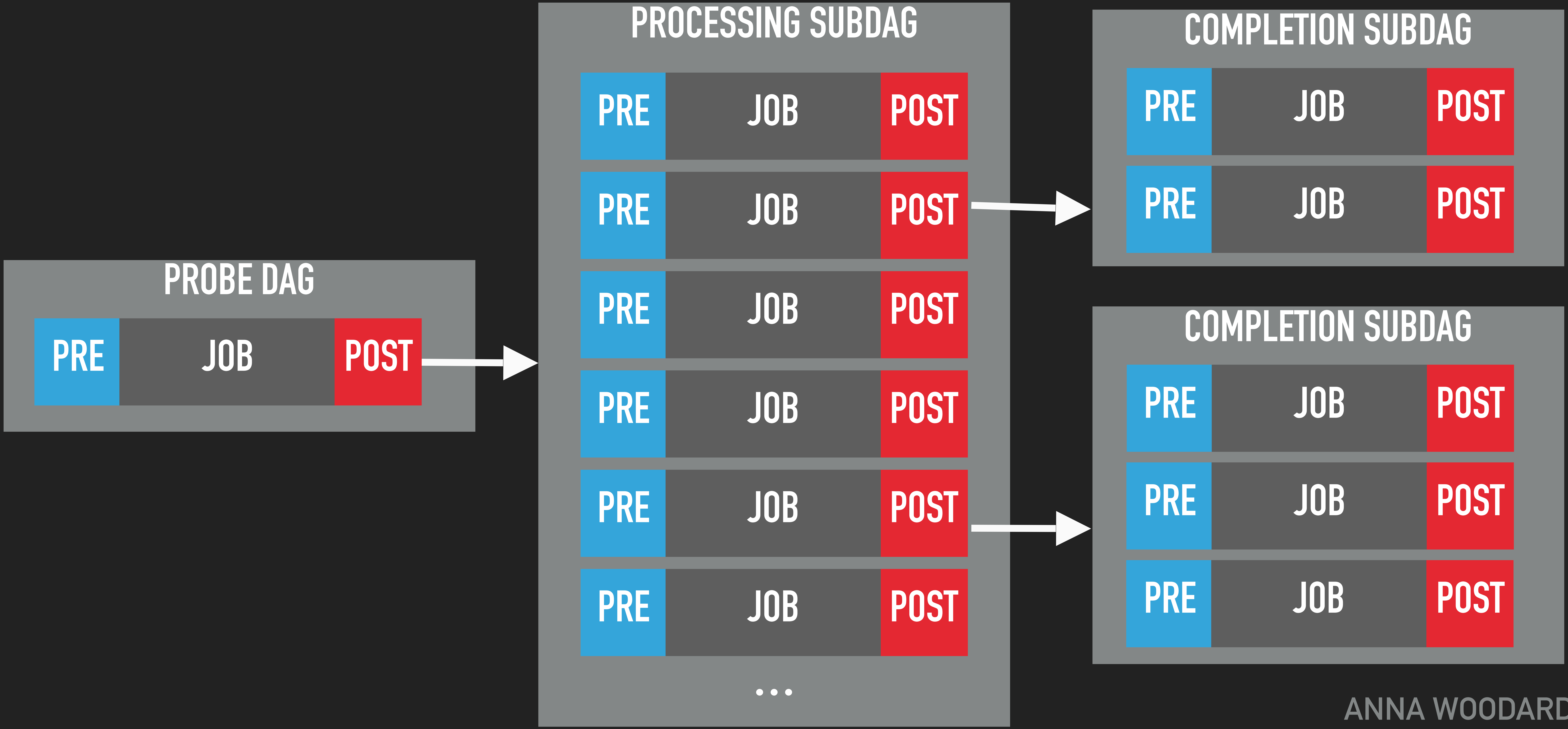
- Since CMSSW_7_2_0: Can configure runtime limitations

New possibility:

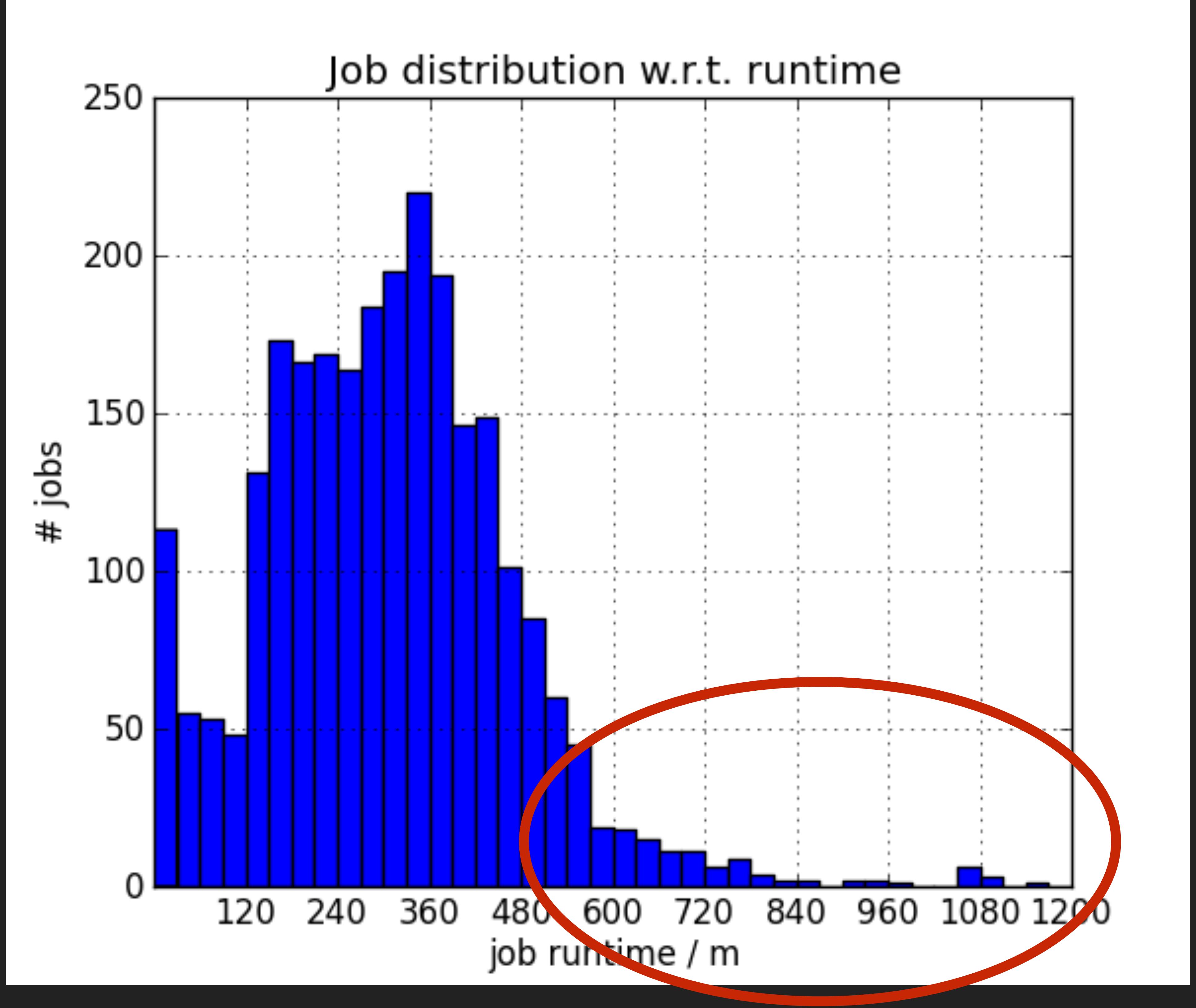
- Use this to cut off tails in the runtime distribution!
- Resplit unfinished work to create subDAGs with small jobs which run in parallel
- Faster task completion!



TAIL-SPLITTING FOR FASTER TASK COMPLETION

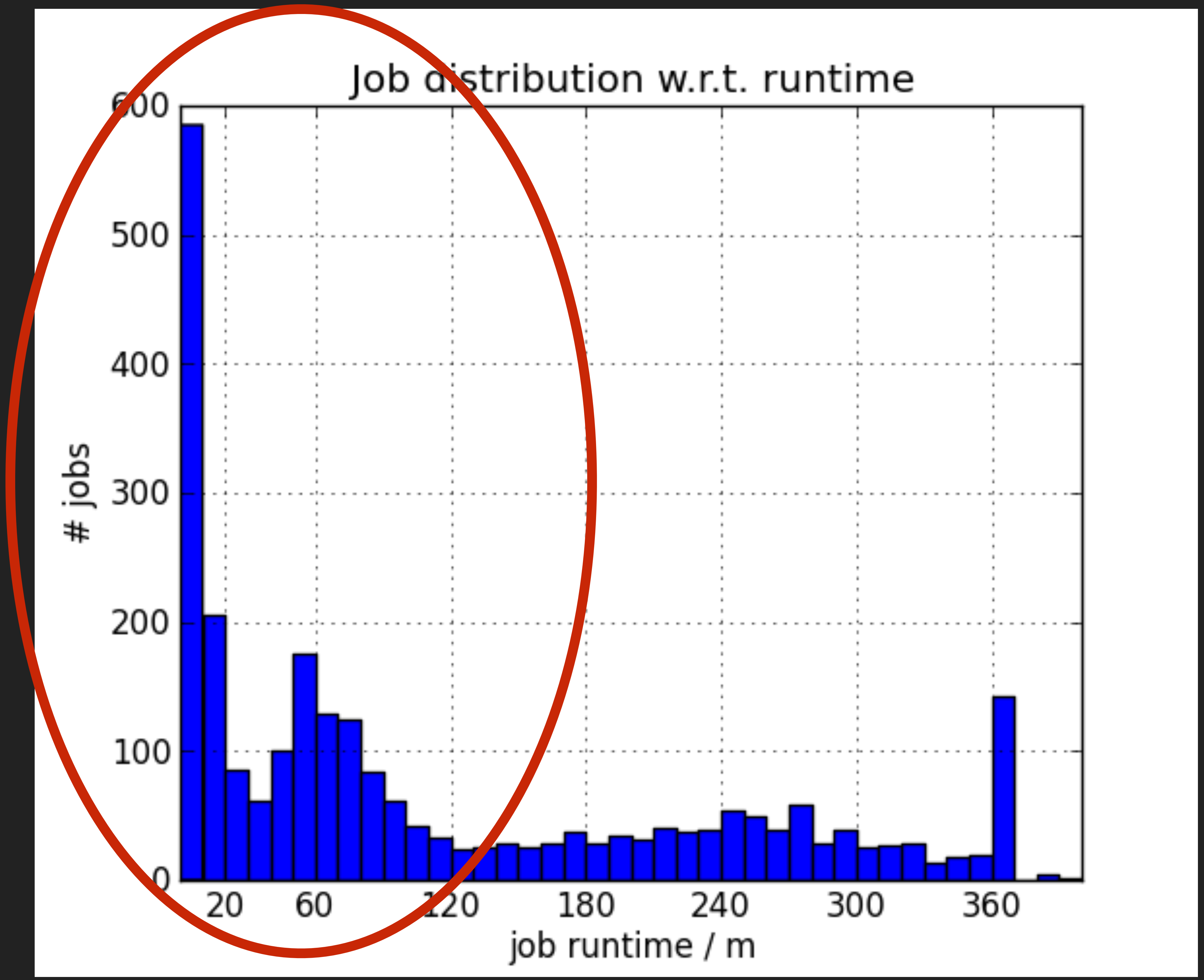


TAIL-SPLITTING FOR FASTER TASK COMPLETION



TWENTY HOURS TO COMPLETE!

CONVERT TAIL INTO SMALLER JOBS



CHALLENGES

- SubDAGs for the tail-splitting jobs increase resource consumption

CONCLUSIONS

- We have implemented automatic task splitting
  - Will be included in the October CRAB3 release
  - Tail-splitting feature is currently disabled, will be enabled pending more tweaks to address resource consumption