STATEFUL SERVICES IN CONTAINERS OR: HOW I LEARNED TO STOP WORRYING AND LOVE THE CLOUD

MARKUS SOMMER

CERN OPENLAB

10TH AUGUST 2017

BACKGROUND

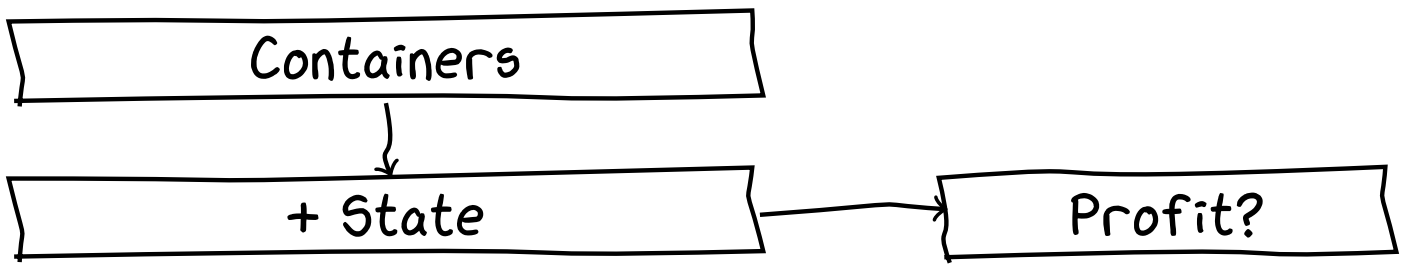
MODERN CLOUD-COMPUTING BASED ON COMPLETE VIRTUALISATION.

BOTH MACHINE AND CONTAINER VIRTUALISATION

PROBLEM

CONTAINER PARADIGM WELL SUITED FOR STATELESS APPLICATIONS

LESS WELL SUITED FOR STATEFUL APPLICATIONS



BUT WHY?

WHY NOT?

BUT NO, SERIOUSLY WHY?

STOP RUNNING SINGLE-PURPOSE SYSTEMS FOR EACH SERVICE

COMMON ADMINISTRATION & DEVELOPMENT FRAMEWORK

SOLUTION (?)

KUBERNETES

CONTAINER ORCHESTRATION

INCLUDES (BETA) SUPPORT FOR STATEFUL APPLICATIONS VIA "PERSISTENT VOLUMES" AND "STATEFULSETS"

DETAILS

PERSISTENT VOLUMES:

PROVIDE PERSISTENT STORAGE

DECOUPLED LIFECYCLES FOR DATA AND EXECUTORS

STATEFULSETS:

PROVIDE UNIQUE IDENTITIES TO PODS

BIND EXECUTOR TO DATA

DOES IT WORK, THOUGH?

SORT OF...

WHAT WORKS

STATEFULSETS WORK AS EXPECTED

SCALING

PERSISTENT VOLUMES
(PARTIALLY)

AUTOMATIC PROVISIONING OF PERSISTENT STORAGE
(WITH SOME MANUAL ADJUSTMENTS)

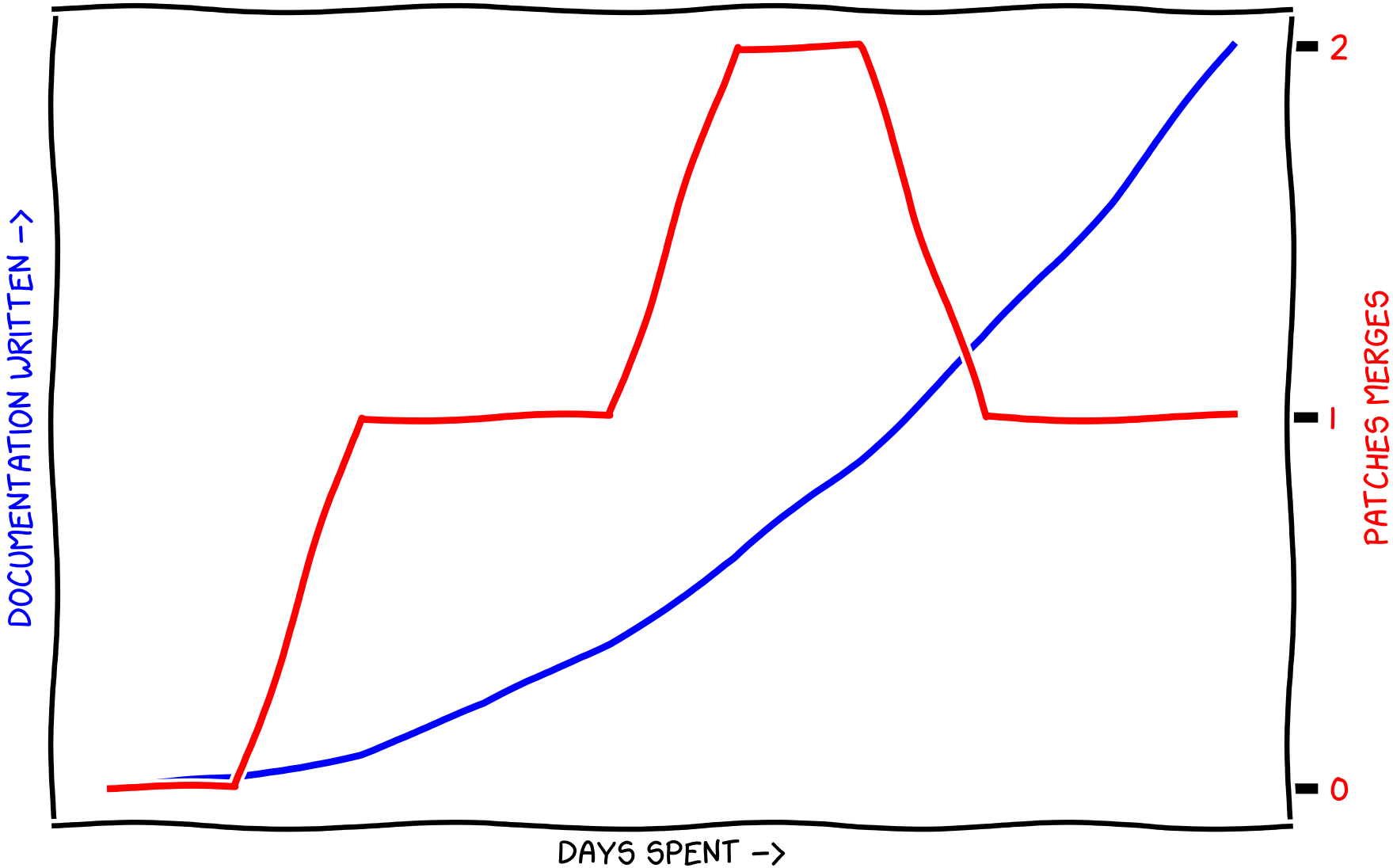
RESCHEDULING
(PARTIALLY)

WHAT DOESN'T

FEATURES RELYING ON THE OPENSTACK CLOUD-PROVIDER

PROPER RESCHEDULING OF APPLICATIONS DURING NODE OUTAGES

MY CONTRIBUTION



CONCLUSION

CLOUD-PROVIDER FOR OPENSTACK STILL UNSTABLE
(UNCLEAR IF IT'S REALLY BETTER ON GOOGLE COMPUTE ENGINE)

STATEFUL SERVICES NOT QUIET READY FOR PRODUCTION YET