

A RESEARCH INTO CLUSTERS, CONTAINERS AND TESTING OF THEIR LOAD-BALANCING CAPABILITIES

CLUSTER COMPUTING

The clustering of computers has a long history that dates back to the 1960s as a solution to overcome the limitations of processing power and storage available at the time. Clustering inadvertently led to the early days of networks, a means of providing establishing communications between machines. The ARPANET project, the first “commodity-network” cluster, and would grow to become the internet as we know it today.



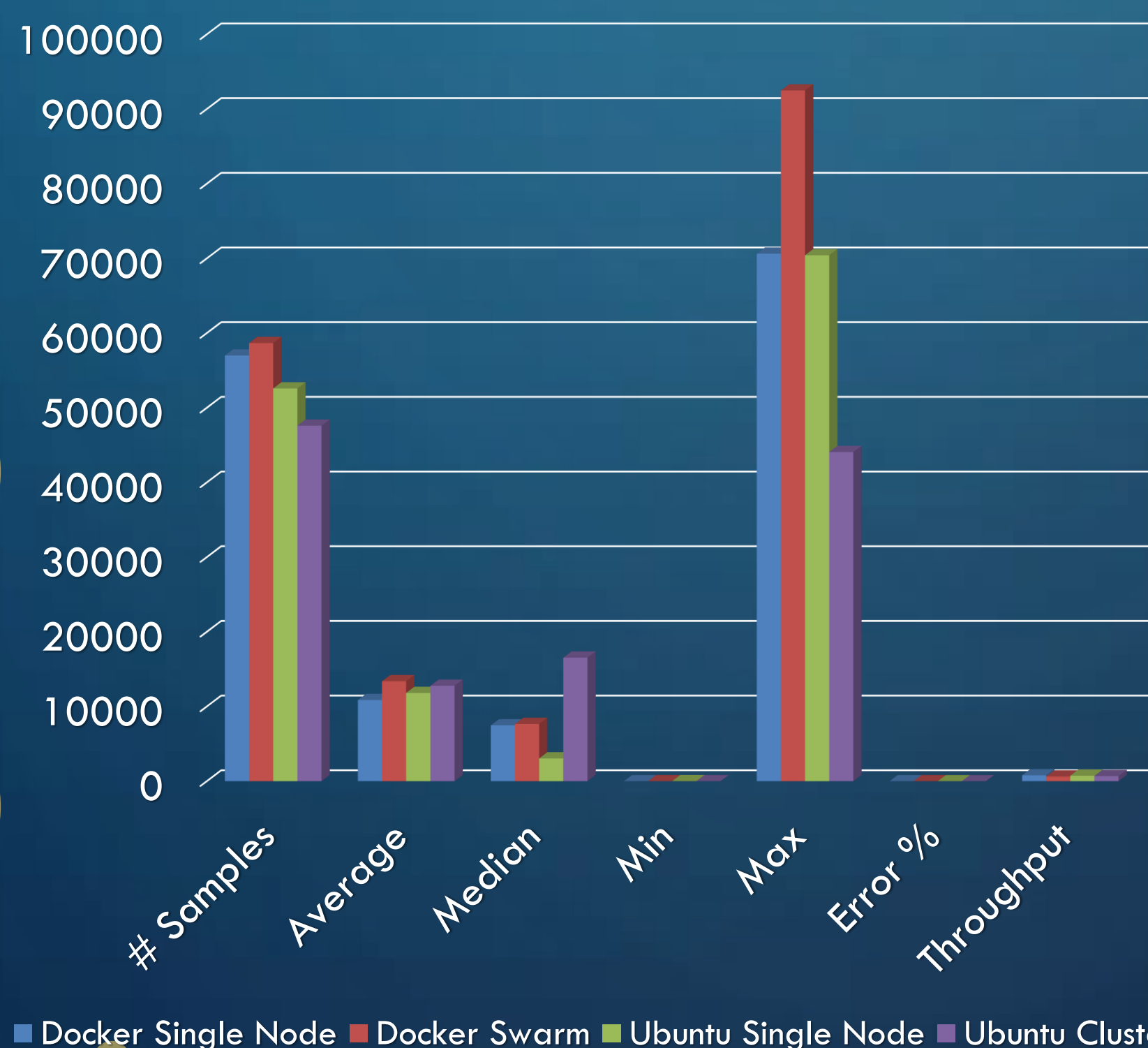
DOCKER

Docker, initially built off Linux Containers (LCX), was released in 2013 and is currently the most advanced and adopted container technology in the world.



LOAD-BALANCING TEST

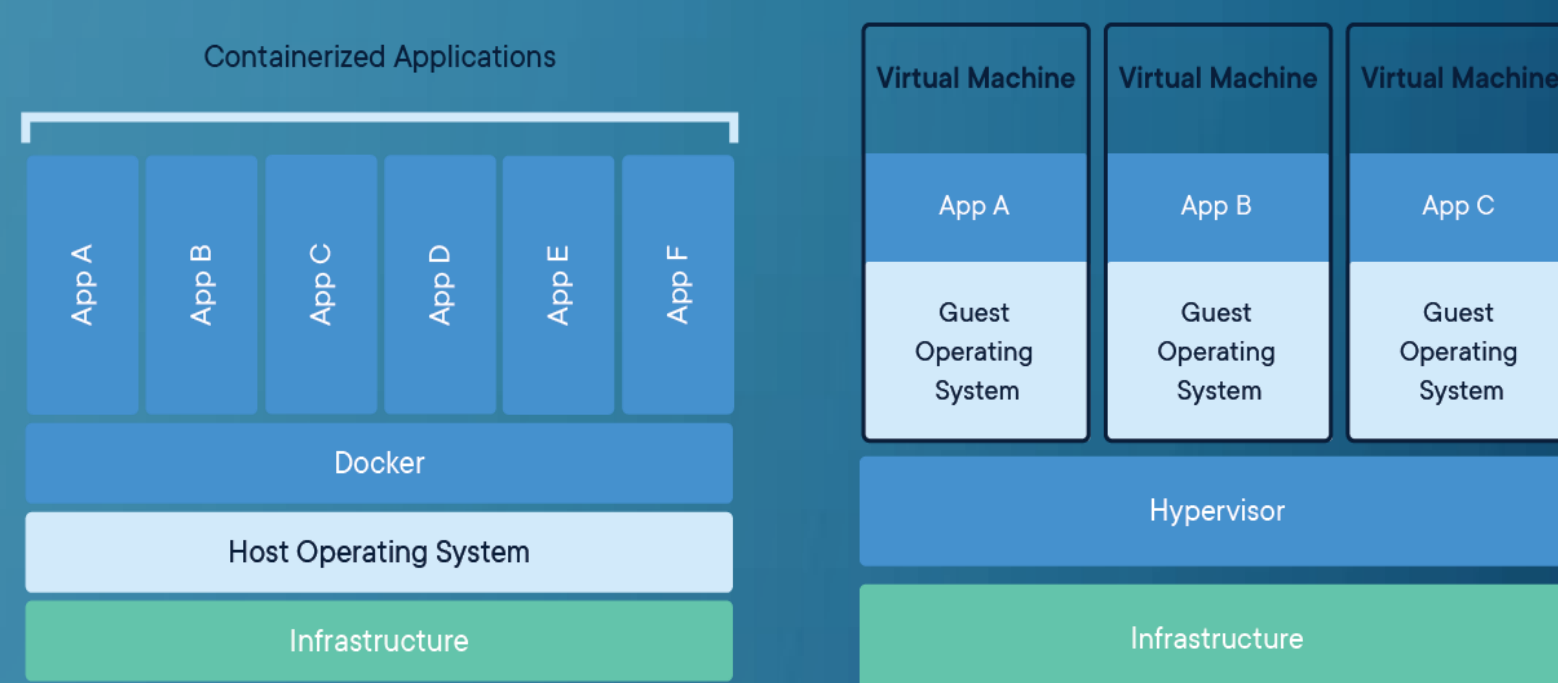
Label	# Samples	Average	Median	Min	Max	Error %	Throughput
Docker Single Node	57009	10875	7484	0	70638	58.94%	799.00491
Docker Swarm	58675	13390	7667	0	92487	47.47%	627.30529
Ubuntu Single Node	52594	11821	3055	4	70432	51.04%	737.57135
Ubuntu Cluster	47607	12781	16553	0	44073	68.09%	690.71731



CONTAINERS

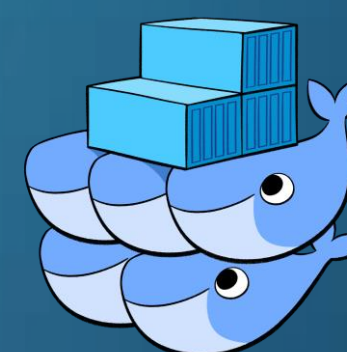
In 1979 with the release of Unix version 7, the implementation of “chroots” allowed the isolation of a single process, this was the first instance of containerisation, though this was forgotten about for over two decades.

Implemented in a version of FreeBSD OS, Jails would reanimate the forgotten process isolation system, bringing with it OS level virtualisation used in container tech today, a process that differs greatly from virtual computer technology.



DOCKER SWARM

Docker natively supports configuration of docker nodes into a cluster, known as Docker Swarm, providing the benefits of traditional computer clustering, and containers to support these functionalities.



The Apache JMeter HTTP request load testing provided an insight on how each configuration handles heavy traffic, though conclusively there was no clear winner.

Docker’s method of load-balancing incorporates the traditional spreading of traffic over the nodes and also creates containers to further accommodate traffic processing.

ID	NAME
hrh2sw7pyh63	webserver.1
qjyug3u21a6k	_ webserver.1
ud052cognmsb	_ webserver.1
175o8cbryuks	_ webserver.1
pfo7d38k3ime	_ webserver.1
dwgi79czg7eh	webserver.2
k0dtcua3i10l	_ webserver.2
42pwhvp74fgl	_ webserver.2
di8pauk9w3rz	_ webserver.2
wripg27poyoj	_ webserver.2
peuhv59qph40	webserver.3
lg5v9i0m8wlf	_ webserver.3
ng9kb36vxkhw	_ webserver.3
k6u1fvfrbszj	_ webserver.3
s85hg7i01tn	_ webserver.3