# EXPAND

## GOALS

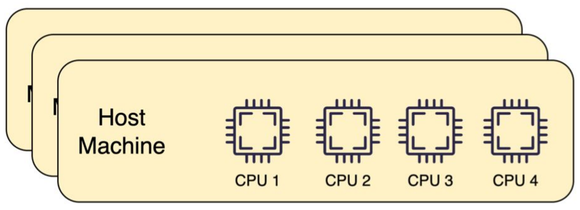
- Achieving end-to-end testing of a complex IoT platform with production level characteristics is extremely hard, with the core difficulty arising from the fact that test environments bear little resemblance to the production setup.
- In the EXPAND experiment, our primary objective has been the deployment of a customized load testing framework on a distributed realistic setup of IoT nodes to emulate our entire IoT platform with production level characteristics.

## CHALLENGES

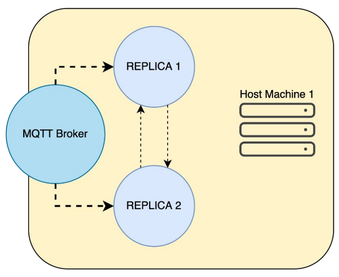
- The outcomes of the EXPAND experiment are crucial towards:
  - evaluating the performance, scalability and reliability of our entire IoT management platform
  - understanding its capacity limitations and
  - help in better planning our future infrastructure expansion

Through EXPAND experiment, we on:
- emulating the domX production environment
- emulating the traffic load of domX IoT devices
- assessing the platform performance under realistic workloads

## DEMO SETUP




Locust Distributed Mode



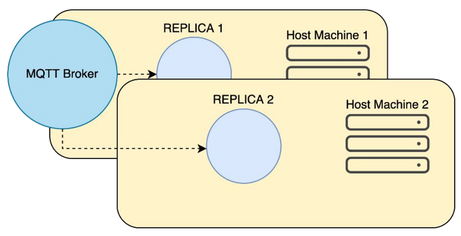

Replication on single VM

### VerneMQ & Docker settings:

- 0.2 CPUs resource constraints per VerneMQ replica
- 4 GB RAM per VerneMQ replica

### Locust settings:

- 21 individual topics with publish intervals and payload sizes of:
  - 2 sec with 200 bytes
  - 10 sec with 500 bytes
  - 60 sec with 700 bytes




Replication on multiple VMs

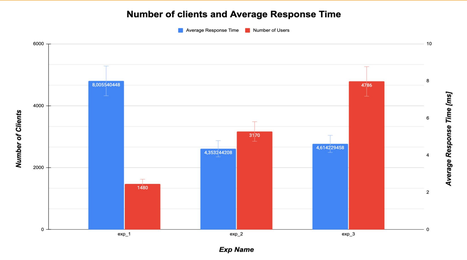
## RESULTS

### Performance analysis

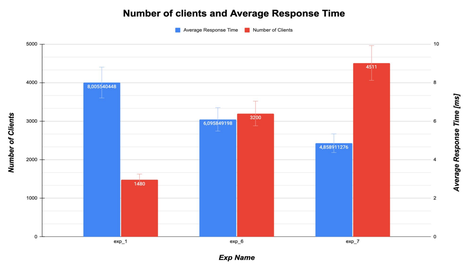
- **Linear** behavior on replica multiplication on *Single & Multiple VMs*
- **Non-Linear** results on increased CPU allocation

### Key parameters

- Client capacity
- Average response time
- CPU allocation
- Replication number

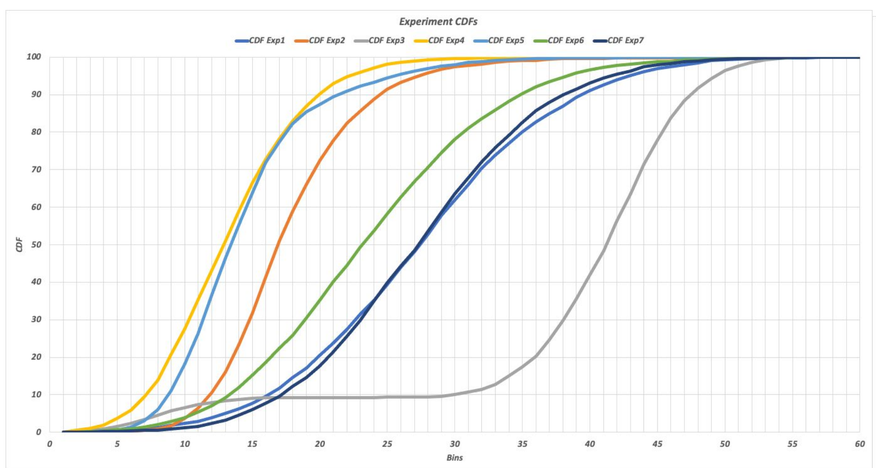




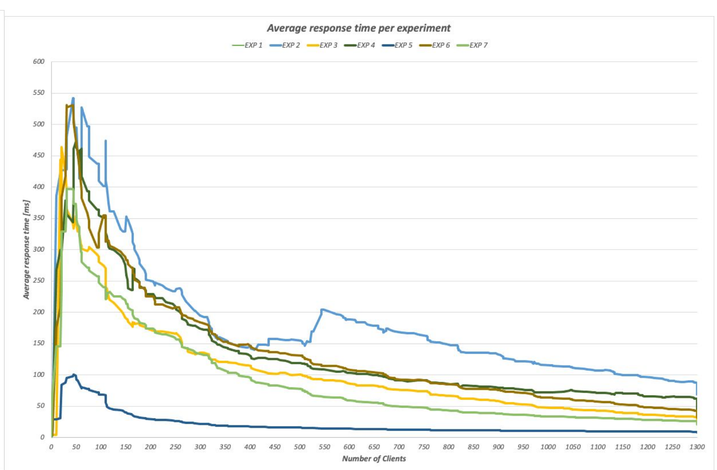



## MORE RESULTS

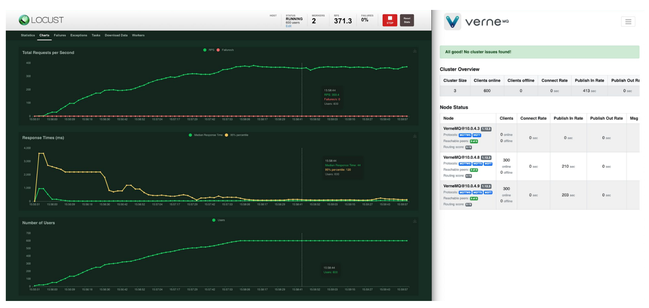
### Cumulative Distribution of Avg. Resp Time

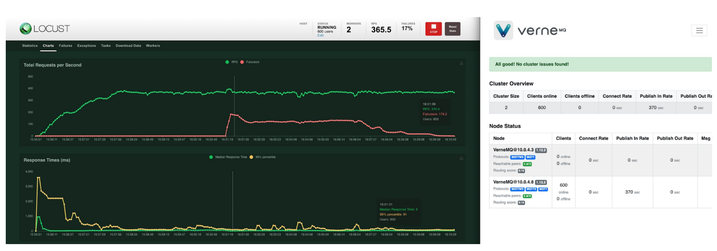



### Avg. Resp. Time per experiment




### Replicas on multi-VMs response, on enforced disconnection from cluster




## CONCLUSIONS

### Key findings

- Increased number of replicas on a single VM induces coordination overhead
- The use of 2 replicas is sufficient to mitigate unexpected server downtimes
- No significant RAM utilization is required
- CPU allocation can be dynamically scaled based on demand

## POST MORTEM

### Platform Upgrade Plan

- Reservation of 2 separate VMs
- deployment of 2 replicas on docker swarm & VerneMQ clustering
- Number of CPU increased based on load demand