**PMEM-CSI Setup**

@Master

git clone <https://github.com/intel/pmem-csi>

kubectl label node <node with pmem> storage=pmem

cd pmem-csi

curl -L https://pkg.cfssl.org/R1.2/cfssl\_linux-amd64 -o \_work/bin/cfssl --create-dirs

curl -L https://pkg.cfssl.org/R1.2/cfssljson\_linux-amd64 -o \_work/bin/cfssljson --create-dirs

chmod a+x \_work/bin/cfssl \_work/bin/cfssljson

export KUBCONFIG="/path/to/.kube/config"

export PATH="$PATH:$PWD/\_work/bin"

./test/setup-ca-kubernetes.sh

kubectl create -f deploy/kubernetes-1.17/pmem-csi-direct.yaml

*\*To use LVM mode:*

*kubectl create -f deploy/kubernetes-1.17/pmem-csi-lvm.yaml*

*lvm mode require 2 pre-steps: pmem-ns-init and pmem-vgm, the yaml file had already take care of these 2 steps.*

*\*\*When using LVM mode, csi will pre-create namespace and the amount of space to be used is determined by an argument: -useforfsdax=100 which means 100%(default) of PMEM will be used. To configure this option, add the argument in the pmem-csi-lvm.yaml file and set the value (0-100).*

(@pmem-csi-lvm.yaml)

apiVersion: apps/v1

kind: DaemonSet

metadata:

labels:

pmem-csi.intel.com/deployment: lvm-production

name: pmem-csi-node

namespace: default

spec:

selector:

matchLabels:

app: pmem-csi-node

pmem-csi.intel.com/deployment: lvm-production

template:

……………………..

initContainers:

- command:

- /usr/local/bin/pmem-ns-init

**- -usefordax=50 <- add this line to determine the percentage**

- -v=3

env:

- name: TERMINATION\_LOG\_PATH

value: /tmp/pmem-ns-init-termination-log

image: intel/pmem-csi-driver:canary

imagePullPolicy: Always

name: pmem-ns-init

\*\*\*After the creation of pvc, the pvc might show “Pending” status. Try to reduce the size in the pvc configuration files.

<https://github.com/intel/pmem-csi/issues/107>

[https://github.com/intel/pmem-csi/issues/107#issuecomment-447233685](https://github.com/intel/pmem-csi/issues/107%23issuecomment-447233685)

kubectl get nodes --show-labels

\*check whether the node with pmem is labeled:

pmem-csi.intel.com/node=<NODE-NAME>,storage=pmem

kubectl create -f deploy/kubernetes-1.17/pmem-storageclass-ext4.yaml

kubectl create -f deploy/ kubernetes-1.17/pmem-storageclass-xfs.yaml

kubectl create -f deploy/kubernetes-1.17/pmem-pvc.yaml

kubectl create –f deploy/kubernetes-1.17/pmem-app.yaml