

Прототип пакетной обработки данных с использованием контейнеров

А.Баранов



ЛАБОРАТОРИЯ
ИНФОРМАЦИОННЫХ
ТЕХНОЛОГИЙ
имени М.Г. Мещерякова

Операционные задачи связанные с обслуживанием сервера

- Монтажные работы
- Установка и настройка ОС, Middleware
- Обновление ядра и ОС
- Ремонтные работы (hardware)

1 Сервер = N задач

K Серверов = K*N задач

Константа:

- Кол-во обслуживающего персонала
- Рост потребности в новых ресурсах

Предпосылки к созданию прототипа

- Как уменьшить объем операционных задач связанных с обслуживанием серверов?
- Как удовлетворить спрос в постоянной потребности в новых ресурсах?
- Необходимость в операционной модели: экономичность, уменьшение технического долга, масштабируемость, гибкость

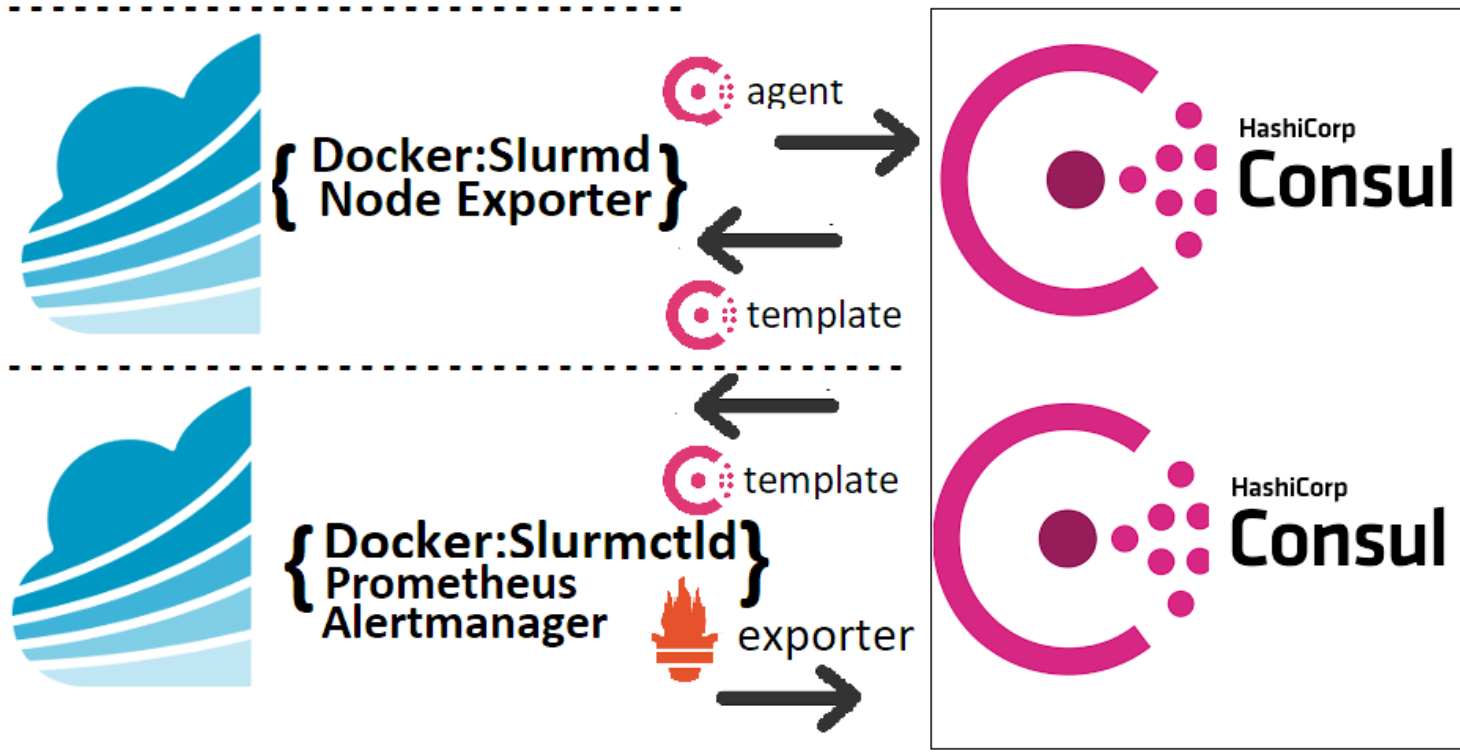
Middleware

- Opennebula (IAAS)
- Gitlab (CI,LFS)
- Prometheus/Alertmanager
(node-exporter)
- Terraform (terraform-
opennebula)
- Consul
(agent,template,discovery)
- Docker
- SLURM

Общая схема прототипа

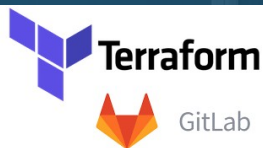


Slurmd [1...N]
Slurmctld

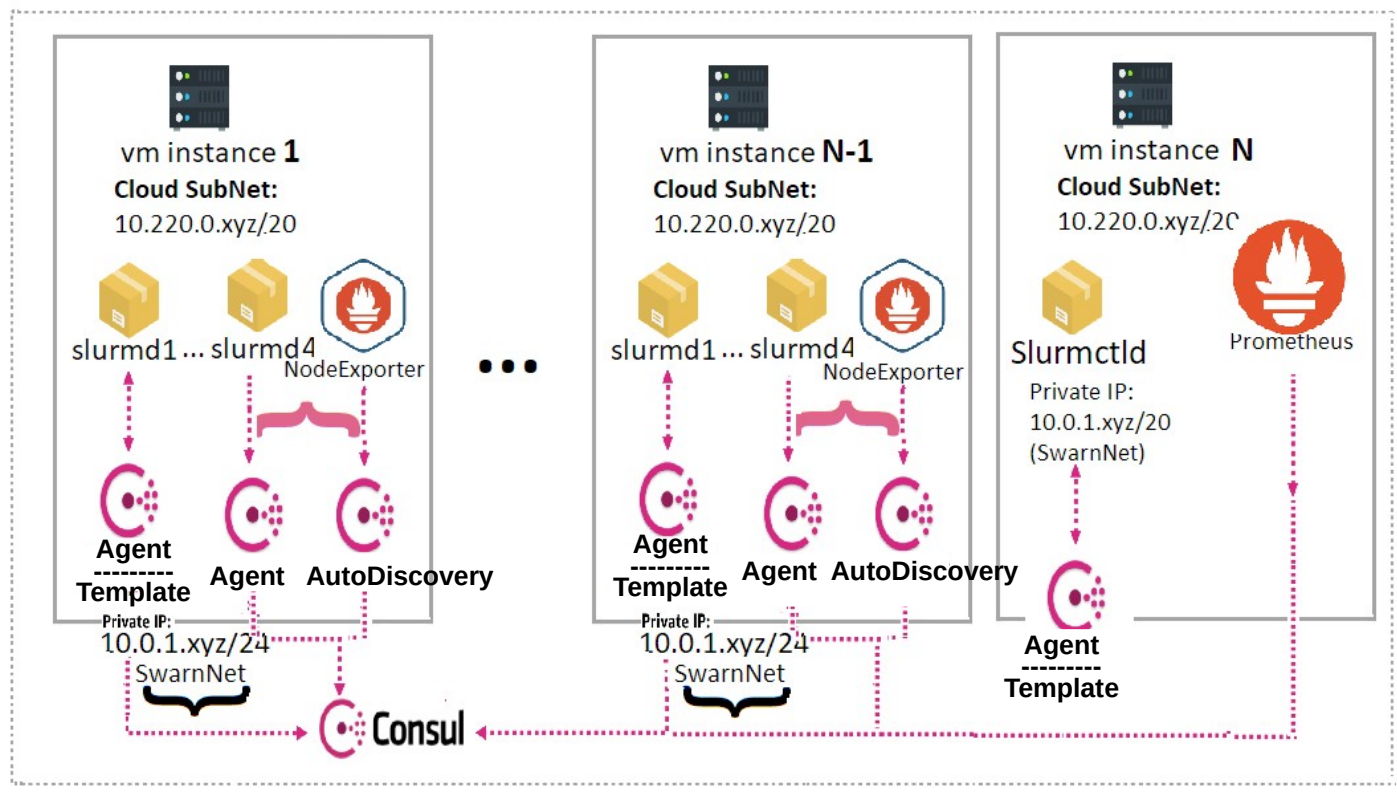


Начальная загрузка

1. Git push (Манифест Terraform)
2. Gitlab (CI):Terraform-opennebula
3. Создание VM:OS AlmaLinux8 + Docker (Opennebula)
4. Контекстуализация VM:
 - a. node-exporter+скрипт
 - b. Регистрация Docker node в кластере swarm (Network)
 - c. Запуск 4-х контейнеров Slurmd
 - d. Регистрация сервисов в Consul: slurmctld; slurmd; munged; node-exporter



Подключение вычислительных узлов



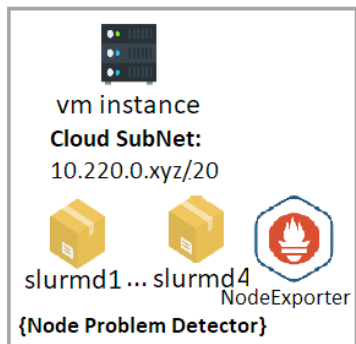
Drain (текущее состояние)

Необходимость
для выполнения
drain

Уведомление



Drain (проект)



```
# HELP container Metric read from /opt/node_exporter/textfile_collector/containers_id.prom
# TYPE container untyped
container{container1="b1b1c266a013",container2="77c15c60605e",container3="9c8e2b2617ee",container4="c7e2a136d6d7",state="drain"} 1
# HELP go_gc_duration_seconds A summary of the pause duration of garbage collection cycles.
```



Prometheus



AlertManager

```
- alert: SlurmAlarm
  expr: container{state="drain"} ==1
  for: 0s
  labels:
    group: slurm_alarm
  annotations:
    summary: "Containers {{ $labels.container }} has problem"
    description: "container has problem {{ $labels }}"
```



webhooks



```
[root@10-220-17-141 webhook]# docker exec -ti slurmctld sinfo
PARTITION AVAIL  TIMELIMIT  NODES  STATE MODELIST
all*      up      infinite    4    drain 9c8e2b2617ee,77c15c60605e,b1b1c266a013,c7e2a136d6d7
all*      up      infinite    4    idle  4f13f1cde25a,7db7bf00eb76,65aad010fd72,e34b02137806
odd       up      infinite    4    drain 9c8e2b2617ee,77c15c60605e,b1b1c266a013,c7e2a136d6d7
odd       up      infinite    1    idle  4f13f1cde25a
even      up      infinite    3    idle  7db7bf00eb76,65aad010fd72,e34b02137806
```

Обновление

- docker push git.jinr.ru:5005/telecast/for-share-containers/docker-slurmd:22.05.6

=====

- DUCC -Daemon that Unpacks Container Images into CernVM-FS (required: docker, stratum 0) | For use with Docker, DUCC will upload a so-called “thin image” to the registry for every converted image.
- cvmfs/graphdriver (last commit 6 years ago)

`$] docker run -it --volume /cvmfs:/cvmfs:shared slurmd ls -lna /cvmfs/containers.jinr.ru`

=====

- Apptainer(singularity): translating from OCI to SIF
[https://apptainer.org/docs/user/main/docker_and_oci.html#docker-and-oci:](https://apptainer.org/docs/user/main/docker_and_oci.html#docker-and-oci)
Apptainer aims for maximum compatibility with Docker, within the constraints on a runtime that is well suited for use on shared systems and especially in HPC environments.
- cvmfs-singularity-sync

Что нужно выполнить:

На данный момент прототип работает в облачной среде но,

- Миграция Docker to Apptainer
- Bare-Metal
- Тестирование производительности (HEPSPEC06):
Hardware vs Container

03.07.23

#GRID2023

прототип пакетной обработки данных с использованием контейнеров 11/12



Спасибо за внимание!

Вопросы?

А.Баранов

#GRID2023



ЛАБОРАТОРИЯ
ИНФОРМАЦИОННЫХ
ТЕХНОЛОГИЙ
имени М.Г. Мещерякова