
KubeCon
CloudNativeCon
OPEN SOURCE SUMMIT
China 2019

Node Feature Discovery

NFD and My Adventure in the Cloud Native Project Jungle

Markus Lehtonen

Agenda

- My Journey
- Resource Management in Kubernetes
- Node Feature Discovery (NFD)
- Demo
- Future of NFD

Introduction

Markus Lehtonen

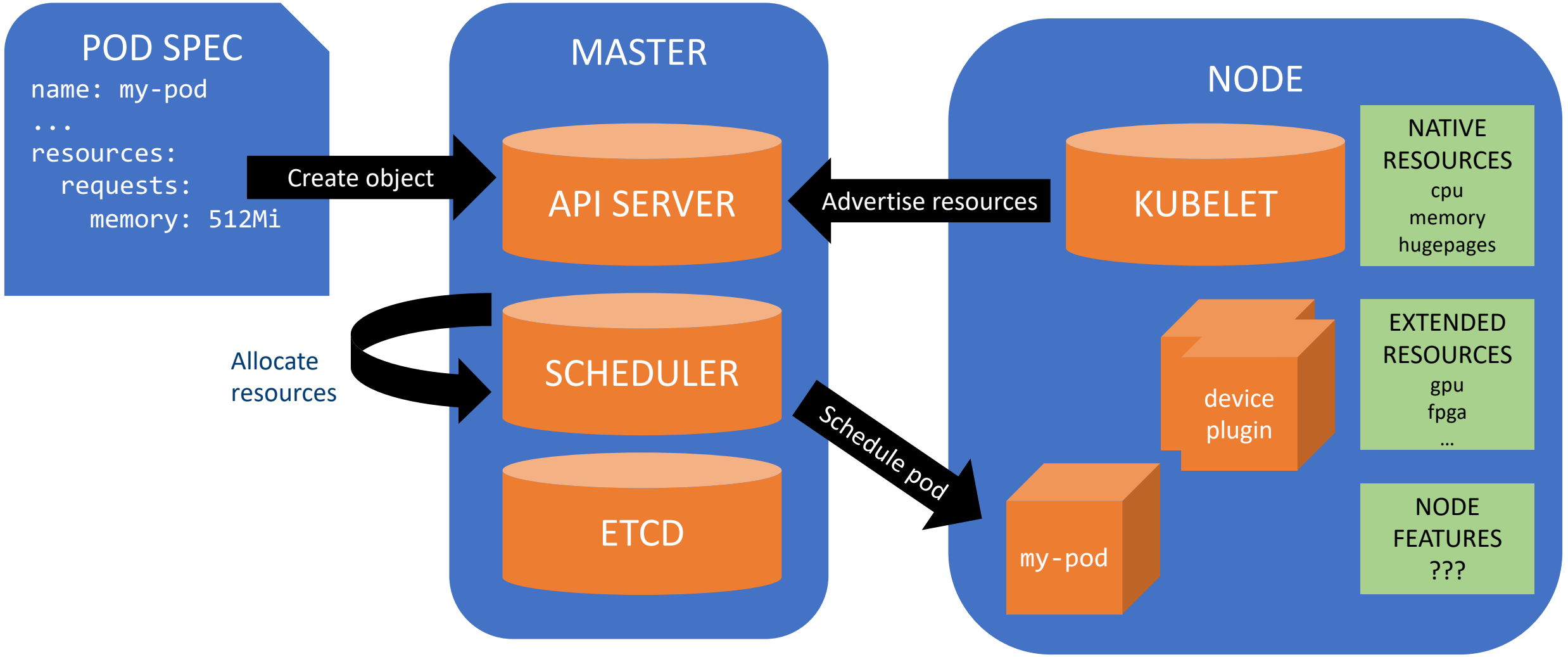
Cloud Software Engineer, Intel

@marquiz

How I Got Involved

- Background in embedded
- Hopped in the K8s wagon in 2018
- NFD needed care
- Fun and welcoming year

Resource Management in K8s



What About Node Features?

WHAT

- Platform capabilities
- Non-allocatable, ”unlimited” resources

WHY

- Heterogenous clusters
- Strict workload requirements
- Workload performance improvements

What About Node Features?

HOW

Node Feature Discovery – NFD

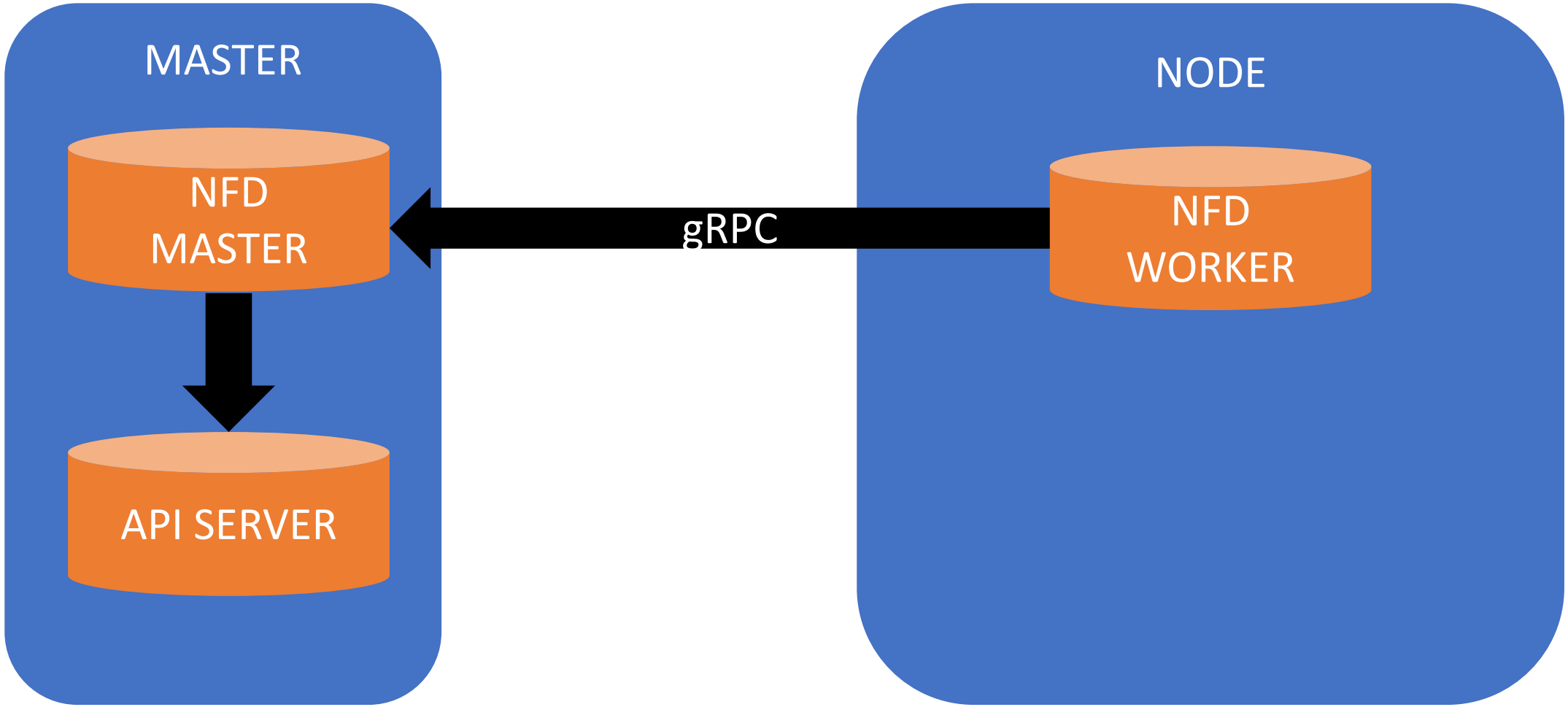
- Fills the gap
- Running one instance per Node
- Advertise features as Node labels

Node labels can be used in workload spec

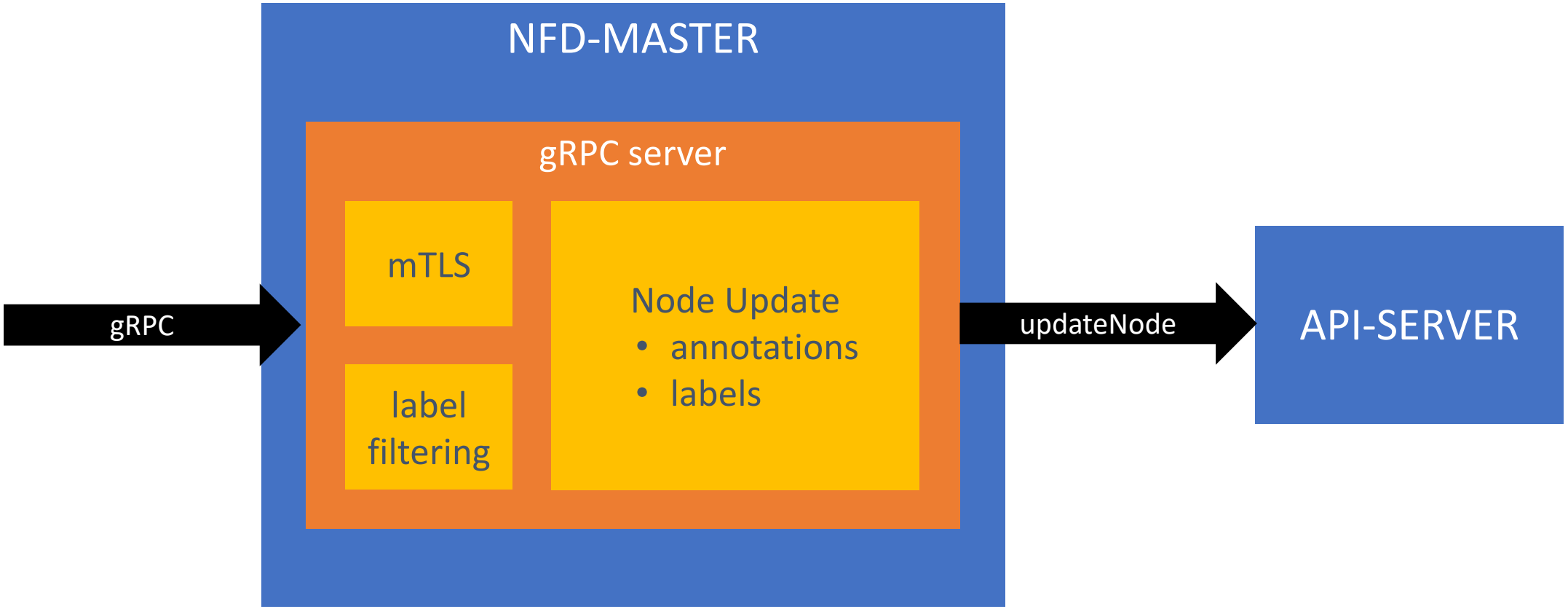
Node Feature Discovery

- Sponsored by SIG Node
- Under Kubernetes-SIGs in Github
  - github.com/kubernetes-sigs/node-feature-discovery

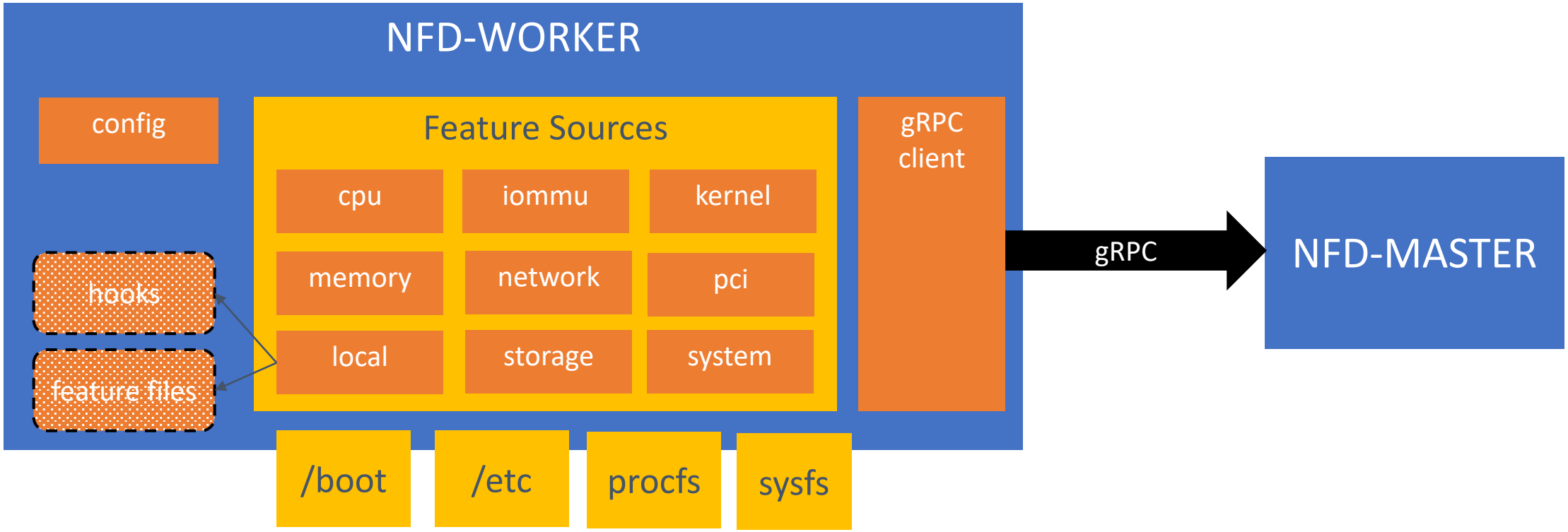
NFD – Architecture



NFD-Master


NFD-MASTER
gRPC server
mTLS
label filtering
Node Update
• annotations
• labels
gRPC
updateNode
API-SERVER

NFD-Worker


NFD-WORKER
config
Feature Sources
cpu
iommu
kernel
memory
network
pci
local
storage
system
gRPC client
hooks
feature files
/boot
/etc
procfs
sysfs
gRPC
NFD-MASTER

Feature Sources

Discovery organized into a hierarchy of sources

HW SOURCES	
CPU	PCI
IOMMU	Storage
Memory	Network

OTHER SOURCES
Kernel
System
Local (custom hooks)

Node feature labels

```
feature.node.k8s.io/<source name>-<feature name>[.<attribute name>]=<value>
```

Re-discovery/re-label every 60s (by default)

Feature Sources – CPU

Feature Name	Attribute	Description
cpuid	<cpuid flag>	CPU capability is supported
hardware_multithreading	*n/a*	Hardware multithreading, such as Intel® HTT, enabled
power	sst_bf.enabled	Intel® SST-BF (Intel Speed Select Technology - Base frequency) enabled
pstate	turbo	Turbo freq. are enabled in pstate driver
rdt	RDTMON RDTCMT RDTMBM RDTL3CA RDTL2CA RDTMBA	Intel® RDT Monitoring Technology Intel® Cache Monitoring Intel® Memory Bandwidth Monitoring Intel® L3 Cache Allocation Technology Intel® L2 Cache Allocation Technology Intel® Memory Bandwidth Allocation Technology

Feature Sources – Kernel

Feature Name	Attribute	Description
config	<option name>	Kernel config option is enabled • true for bool/tristate options (set 'y' or 'm') • config value for other options (str, int, …) Defaults: NO_HZ, NO_HZ_IDLE, NO_HZ_FULL, PREEMPT
selinux	enabled	Selinux is enabled on the node
version	full major minor revision	Full kernel version (e.g. '4.5.6-7-g123abcde') First component of the kernel version (e.g. '4') Second component of the kernel version (e.g. '5') Third component of the kernel version (e.g. '6')

Configurable:

- Kconfig file to read
- Kconfig options to discover

Feature Sources – System

Feature Name	Attribute	Description
os_release	ID	Operating system identifier (e.g. ’centos’)
	VERSION_ID	Operating system version identifier (e.g. ’7.6')
	VERSION_ID.major	First component of the OS version id (e.g. ’7')
	VERSION_ID.minor	Second component of the OS version id (e.g. ’6')

Feature Sources – Memory

Feature Name	Attribute	Description
numa	*n/a*	Multiple memory nodes i.e. NUMA architecture
nv	present	NVDIMM device(s) are present
	dax	NVDIMM DAX mode regions are present

Feature Sources – PCI

Feature Name	Attribute	Description
<device label>	present	PCI device is detected. Defaults: GPU and accelerator cards are detected

<device label> is composed of raw PCI IDs, separated by underscores.

Configurable:

- Fields that <device_label> contains
- Device classes that are detected

Feature Sources – Network

Feature Name	Attribute	Description
sriov	capable	Single Root Input/Output Virtualization (SR-IOV) enabled Network Interface Card(s) present
	configured	SR-IOV virtual functions have been configured

Feature Sources – IOMMU, Storage

IOMMU FEATURES

Feature Name	Attribute	Description
enabled	*n/a*	An IOMMU is present and enabled in the kernel

STORAGE FEATURES

Feature Name	Attribute	Description
nonrotationaldisk	*n/a*	Non-rotational disk, like SSD, is present in the node

Feature Sources – Local

- User-specific feature detection
- Custom feature sources in a pluggable way
- Create new labels / override existing labels
- Two mechanisms
  - Hooks
  - Feature files

Feature Sources – Local Hooks

- Execute files from
  `/etc/kubernetes/node-feature-discovery/source.d/`
- Stdout is turned into labels
  `[[<label ns>]/<source name>-]<feature name>[=<value>]`
- stderr directed to NFD logs

STDOUT FROM `my_source`

```
my_bool
my_non_bool=myvalue
/override_src-my_feature_1
/override_src-my_feature_2=123
my.namespace/value=k8s
```

NODE LABELS CREATED

```
feature.node.kubernetes.io/my-source-my_bool=true
feature.node.kubernetes.io/my-source-my_non_bool=myvalue
feature.node.kubernetes.io/override_src-my_feature_1=true
feature.node.kubernetes.io/override_src-my_feature_2=123
my.namespace/value=k8s
```

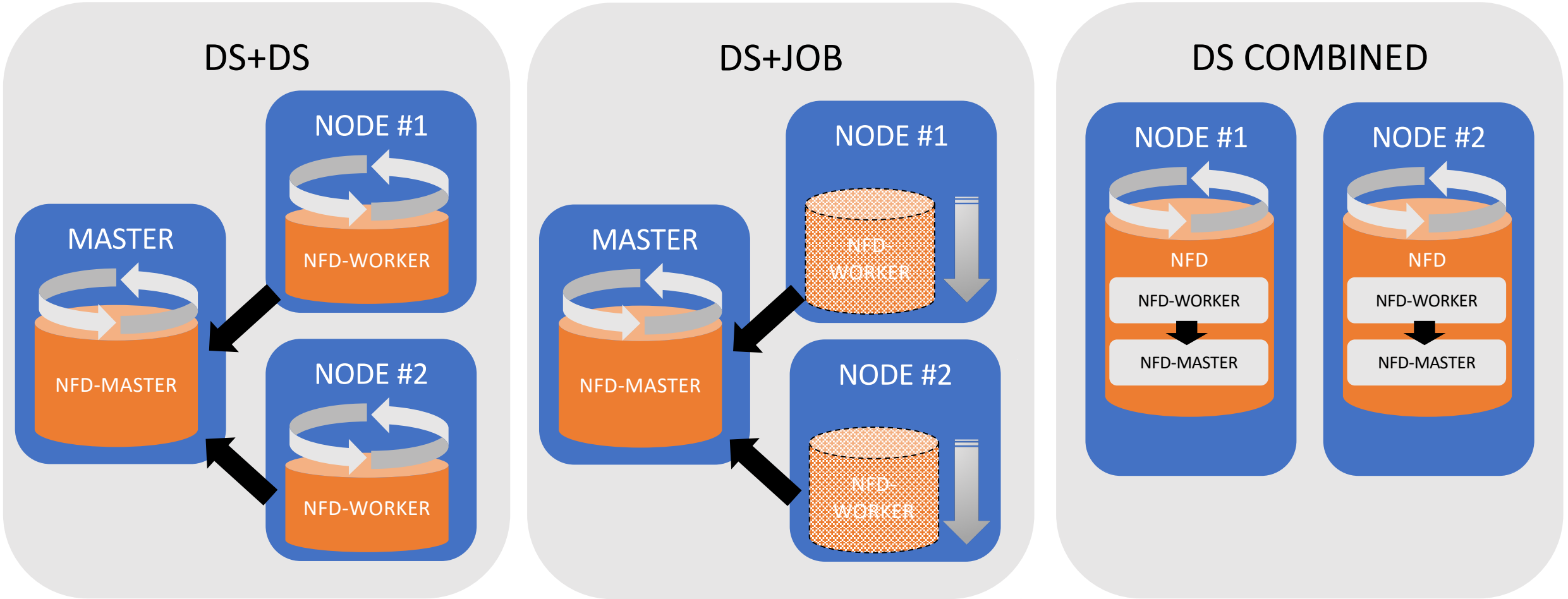
Feature Sources – Feature Files

- Read files from
  `/etc/kubernetes/node-feature-discovery/features.d/`
- File content is turned into labels
- Format is similar to hooks
  `[[<label ns>]/<source name>-]<feature name>[=<value>]`

Configuration Options

- nfd-worker has (optional) configuration file
- Three configurable sources
  - cpu
  - kernel
  - pci

NFD Deployment



Workload Deployment – nodeSelector


```
nodeSelector:
    <label>: <value>
```

```
apiVersion: v1
kind: Pod
metadata:
  name: my-pod
spec:
  containers:
  - image: k8s.gcr.io/pause
    name: pause
  nodeSelector:
    feature.node.kubernetes.io/cpu-pstate.turbo: 'true'
```

Workload Deployment – nodeAffinity


```
nodeAffinity:
    requiredDuringSchedulingIgnoredDuringExecution:
      nodeSelectorTerms:
      - matchExpressions:
        - key: <label>
          operator: {In|NotIn|Exists|DoesNotExist|Gt|Lt}
          values: [<list of values>]

    preferredDuringSchedulingIgnoredDuringExecution:
    - weight: <integer>
      preference:
        matchExpressions:
        - key: <label>
          operator: {In|NotIn|Exists|DoesNotExist|Gt|Lt}
          values: [<list of values>]
```

Workload Deployment – nodeAffinity


```yaml
apiVersion: v1
kind: Pod
metadata:
  name: my-pod
spec:
  containers:
  - image: k8s.gcr.io/pause
    name: pause
  affinity:
    nodeAffinity:
        requiredDuringSchedulingIgnoredDuringExecution:
          nodeSelectorTerms:
          - matchExpressions:
            - key: feature.node.kubernetes.io/kernel-version.major
              operator: Gt
              values: ['3']
            - key: feature.node.kubernetes.io/kernel-version.minor
              operator: Gt
              values: ['4']
        preferredDuringSchedulingIgnoredDuringExecution:
        - weight: 1
          preference:
            matchExpressions:
            - key: feature.node.kubernetes.io/cpu-hardware_multithreading
              operator: NotIn
              values: ['true']
```

Demo Time

- Deploy NFD v0.4.0 on a cluster
- Deploy Intel® GPU device plugin on GPU-capable node(s)

The (Near) Future

- NFD operator
- Multi-arch builds
- Usability and configurability improvements
- Support for Taints(?)
- Support Extended Resources(?)

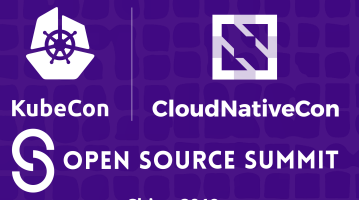
- Project logo ;)

- The usual boring
  - Usage examples
  - Documentation

Want To Help?

- Feature requests
- Patches
- Tell us your story

https://github.com/kubernetes-sigs/node-feature-discovery



THANK YOU