Storing and updating extra fpga resource attributes in hdfs v1

Ye, Yuqiang

Motivation

Currently YARN-3926 has supported various resource-types such as Disk, network, HDFS bandwidth, but YARN-3926 only consider the countable resources. For FPGA, GPU, network port and Disk Volume Mount , these cannot be treated as a unit of resource. It's not enough to schedule them only depend on a number value, all of them have their own specification. Furthermore, some resource attributes may be updated during an application running cycle, like FPGA AFU ID (Accelerated Function Unit, Hardware Accelerator implemented in FPGA logic that accelerates or intends to accelerate an application kernel).

This proposal is mainly to solve these changeable resource attributes by storing and updating them in hdfs. It can be regarded as a supplement for FPGA on YARN.

Usecases

First, we list a use case about FPGA which will be applied by the yarn application.

Resource resourceCapability = Resource.newInstance(*containerMemory*, *containerVirtualCores*);

//e.g. containerFpgaType can be MCP, count can be 1

resourceCapability.setResourceValue(*containerFpgaType*, *count*);

//e.g. nodeLabelExpression can be (AFU_ID == 0001)

ContainerRequest request = new ContainerRequest(resourceCapability, ..., *nodeLabelExpression*);

This case shows that AM request a container with 1 MCP and installed AFU package with AFU_ID 0001

YARN-3926 has considered all scalar resource type such as memory , cpu, disk , network , hdfs bandwidth and so on. The schedule policy will depend on the scalar of resource,

current it's long value. But as the listed use case, we think YARN-3926 is not enough for FPGA, GPU, network port which has particular attributes like FPGA AFU . Only a long value(MCP:1) cannot represent these resources. And the resource value is set in configuration that means it cannot be changed after cluster starting.

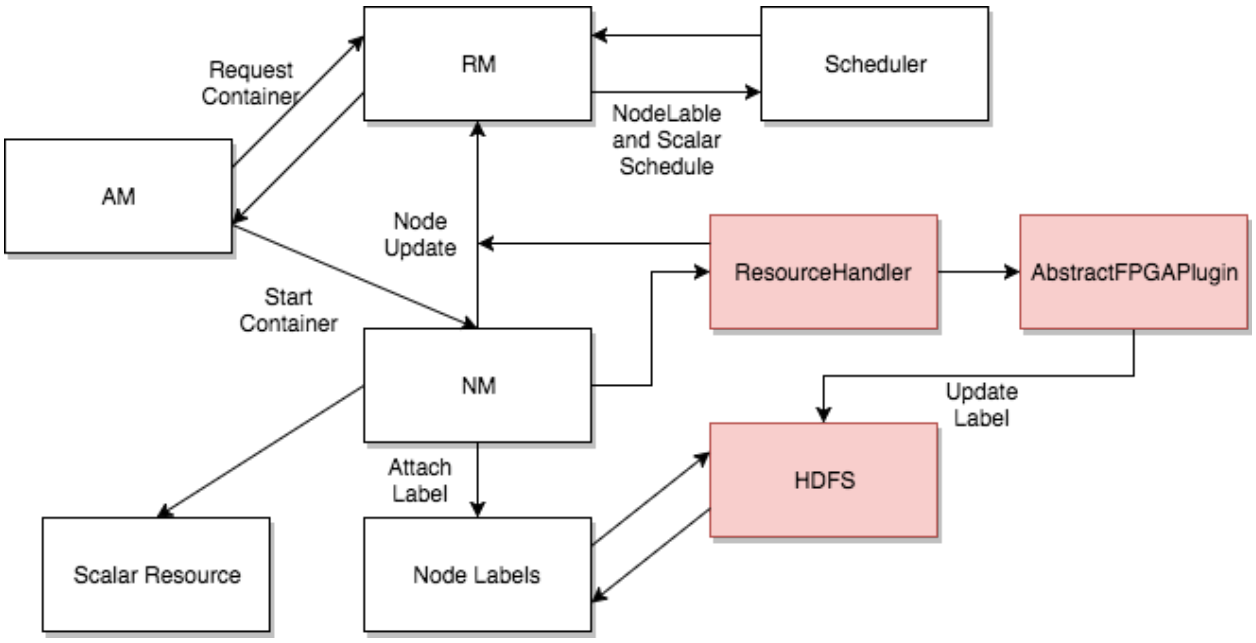
Solution

One solution is combining YARN-3926 and YARN-3409(Add constraint node labels). YARN-3409 will extend the existing node labels to characterize a node based on many of attributes and select a node based on expression of these attributes during scheduling.

For example , we have 4 FPGA MCP Nodes and each MCP already flashed afu package.

Here, MCP will be treated as a kind of resource and the value is 4 in YARN-3926. The extra resource attribute (afu id) will be a node label. And FPGA plugin(details in YARN-5983-Support-FPGA-resource-on-NM-side_v1) will responsible for updating node label in HDFS.

Below is a whole picture.



Some key steps here:

1. ResourceHandler will get the request afu id from container environment
2. Whether AbstractFPGAPlugin will flash and updating the afu id is determined by the current afu id node label in HDFS
3. After updating node label, RM,NM and hdfs should keep consistency.

Reference

YARN-3926 https://issues.apache.org/jira/browse/YARN-3926

YARN-3409 https://issues.apache.org/jira/browse/YARN-3409