Design document of

Container move (relocation) between nodes

towards the improvement of long-running service support in Hadoop YARN

version 0.1

Abstract

Support for relocating containers has become a must-have requirement for most multi-service applications, since the inevitable concept-drifts make SLAs hard to be satisfied. The relocation and co-location of services (long running containers) can help to reduce bottlenecks in a multi-service cluster, especially where data-intensive, streaming applications interfere.

Currently, such a container relocation mechanism is managed completely by the ApplicationMaster, which gives the implementation, maintenance burden to the services across the cluster. Additionally, in a moderately utilized cluster environment, simulating a container relocation with a new resource allocation will not react to the violation of SLAs fast enough. Workarounds usually lead to under-utilization.

The following section will describe the high level changes and improvements that should be made to YARN to allow containers to be moved across the cluster.

Proposed solution

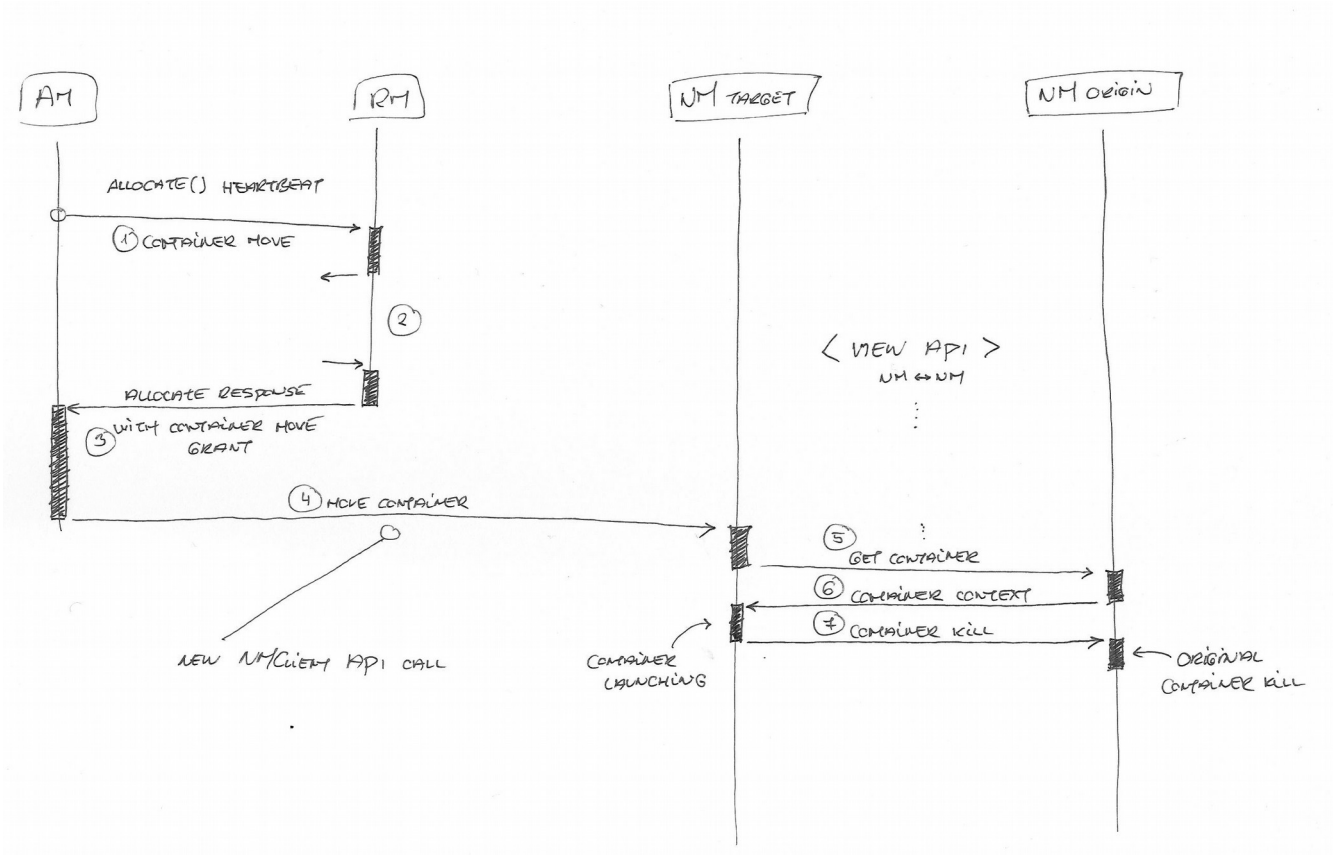

Figure 1: Container relocation mechanism

ApplicationMasters should be able to move containers between nodes in form of relocation requests sent to the ResourceManager (in step 1 of *Figure 1*). The relocation of a container should be requested upon the allocate heartbeat, attached with a

- priority,
- deadline in which the relocation should occur,
- set of target nodes or racks.

The scheduler should handle relocation requests differently, but within the same queue (in step 2).
Upon relocation grant (in step 3), the allocate response should include the assigned node, which is the target node where the container should be moved to.

It should be the ApplicationMaster's responsibility, to seek out the target node and request the container relocation (in step 4). For that (in step 5 and 6), the target node needs to retrieve the container's context from the origin, and relaunch it under the same conditions (environment, command, ...). Optionally (in step 7), the target NodeManager should notify the origin upon a successful container launch, to kill the original container – to avoid unnecessary service downtime.

Additional notes

Preserving the state of the application should remain the burden of the application. If necessary, the moving container should save and load its state from a distributed storage (for example HDFS).