

High Performance Computing (HPC)
Cluster Management
with CIM
-
a Job Scheduling Environment

--- draft ---

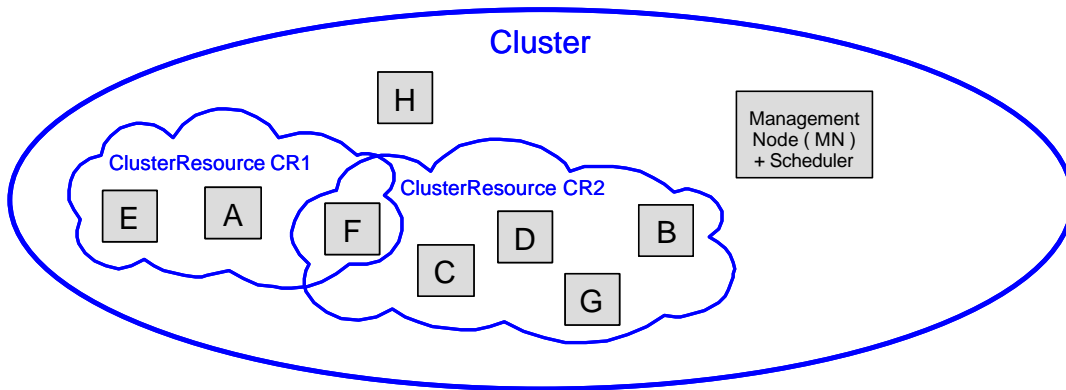
In Cooperation with University of Illinois (NCSA)
May 2002

author :
Linux Technology Center (LTC)
System Management
Heidi Neumann
heidneu@de.ibm.com

Version 1.3
Last Update October 2002

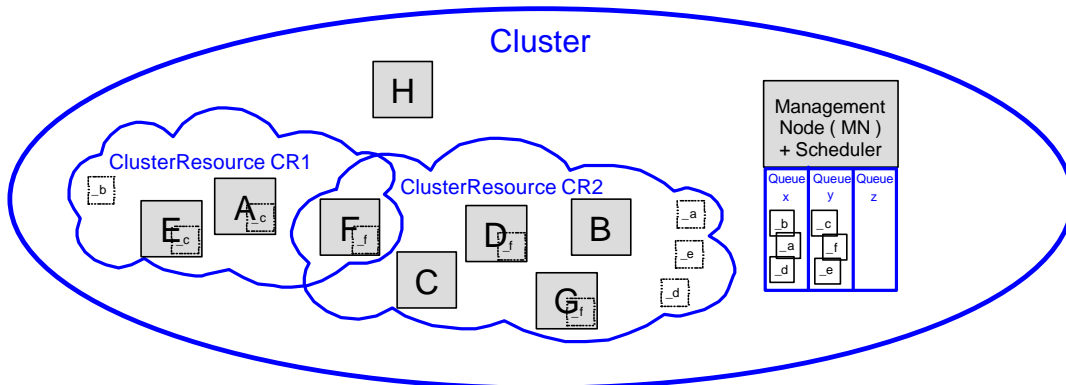
1 Introduction

This chapter describes the NCSA Cluster environment. NCSA hosts a High Performance Computing Cluster with OpenPBS (<http://www.openpbs.org/>) as Job Scheduler. The intention of our cooperation is the definition and implementation of a CIM Schema for this environment. Currently there are several activities to define a common Cluster Schema in CIM. To later synchronize with this common Cluster Schema, the NCSA-SBLIM Cluster Schema is as abstract and general from the CIM Schema as possible.



1.1 Cluster Topology (logical objects are marked blue , physical objects are marked grey)

Picture 1.1 gives an overview of the main Cluster parts. The Cluster itself is a logical entity and responsible for computations consuming an high amount of resources. A Cluster contains a defined set of Hosts (A, B, ... H, MN). These Hosts are logical grouped in so called Resources (CR1, CR2). A single Host can participate in more than one Resource. The MN does not participate in any Resource. (But can be used for computations, if all Resources are in use.). Theoretically one Host can execute more than one Jobs at a certain time. In the case of the University of Illinois Cluster this is not supported. That means one Host can execute zero or one Job at one point in time.



1.2 Job Management

A Job ($_{a, b, \dots, f}$) is a unit of work submitted to the Cluster for computation. The Job is submitted to a specific Queue (x, y, z) and certain Resource (CR1, CR2). The Job contains information under which specific conditions the computation has to run. The Job stays as long in its Queue as the computation is not finished or the Job killed. Once a Job is started, the Job runs on a list of Hosts. Each of these Hosts is member of the Resource specified by submission of the Job. A Resource can compute more than one Jobs at one point in time.

The Scheduler has knowledge about Jobs, Queues, Resources and Hosts and is responsible for the execution of the Jobs within their specified conditions. There is only one Scheduler responsible for the Cluster.

Definition of the Schema

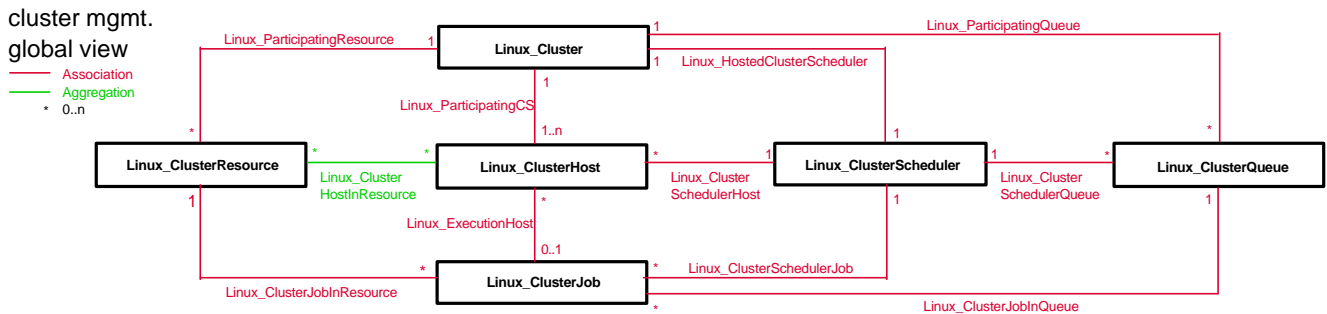
While the definition of the cluster schema in general is independent of a specific naming schema, we use CIM (Common Information Model) as underlying technology. CIM is an implementation-neutral schema to offer overall management information in a network / enterprise environment and an accepted industry standard defined by the Distributed Management Task Force (DMTF).

The schema definition is divided in two important parts. The first one represents the global view a Management Node / Management Server has onto the cluster (chapter 2). An implementation of this schema is only located on the Management Node (MN). The second part describes the Base Instrumentation of each single node participating in the cluster (chapter 3).

To offer each user of the Cluster special views with all necessary and important information, the objects (classes) in the schema need a well defined set of attributes. The different views itself are realised by using different subsets of information. An example can be : An administrator needs a extensive view onto the whole Cluster, while a manager only wants to know how many Jobs are currently scheduled for computation. In the case of the view for managers an application enumerates all Jobs and uses only this subset of information. If an administrator needs a complete overview, the enumeration of Jobs is only one step besides many others. The enumeration of all Resources, Hosts and Queues can be another step. Associations define the relations between the instances of this classes, e.g. which Hosts are defined for Resource CR1. But the application works with exactly the same classes for each view.

2 Cluster Management / Global View

2.1 Class and Association Description



2.1 Cluster Schema (global view) class & association overview - necessary CIM schema parts are Core and System

This schema is only available on the Management Node (MN) and offers a global view onto the Cluster. The MN needs a higher level Instrumentation compared to the single nodes (Hosts). This higher level functionality is described in this chapter. The dependency to the CIM schema is attached at the end of this document - 'Class view' and 'Association view'.

Linux_Cluster is a logical object to represent a certain topology - the HPC Cluster Domain. An enumeration of this class returns one single instance .

Linux_ClusterHost represents the Hosts located in this Cluster. An enumeration of this class returns the same amount of instances as Hosts are physical available. The MN is not part of this list.

Linux_ClusterResource represents the Resources. An enumeration of this class returns the same amount of instances as Resources are defined for this Cluster. A Resource is a static definition for a node group.

Linux_ClusterJob represents the Jobs. An enumeration of this class returns the same amount of instances as Jobs are submitted for computation to the Cluster.

Linux_ClusterQueue represents the Queues. An enumeration of this class returns the same amount of instances as Queues are defined for this Cluster.

Linux_ClusterScheduler represents the Scheduler. The Scheduler is a service. An enumeration of this class returns one single instance as there is only one Scheduler responsible for the Cluster.

Associations reflect the relations between classes and their instances. First a short description of associations. An association has a left and a right 'end', see picture 2.1 'Cluster Schema'. These ends can be compared with pointers to classes. An association is called with one end specified (one instance of the referred class), to figure out which instances are on the other side. For a better understanding I will give examples based on the pictures 1.2 and 2.1 .

Linux_ParticipatingCS reflects the relation between the Cluster Domain and the Hosts defined for it. If you call this association from the Cluster end (instance of class Linux_Cluster specified) all instances of Linux_ClusterHost are returned. If you call this association from the Host end (instance of Linux_ClusterHost specified) the one instance of Linux_Cluster is returned.

Linux_ParticipatingResource reflects the relation between the Cluster Domain and the Resources defined for it. If you call this association from the Cluster end (instance of class Linux_Cluster specified) all instances of Linux_ClusterResource are returned. If you call this association from the Resource end (instance of Linux_ClusterResource specified) the one instance of Linux_Cluster is returned.

Linux_ParticipatingQueue reflects the relation between the Cluster Domain and the Queues defined for it. If you call this association from the Cluster end (instance of class Linux_Cluster specified) all instances of Linux_ClusterQueue are returned. If you call this association from the Queue end (instance of Linux_ClusterQueue specified) the on instance of Linux_Cluster is returned.

Linux_HostedClusterScheduler reflects the relation between the Cluster Domain and the Scheduler defined for it. If you call this association from the Cluster end (instance of class Linux_Cluster specified) the one instance of Linux_ClusterScheduler is returned. If you call this association from the Scheduler end (instance of Linux_ClusterScheduler specified) the one instance of Linux_Cluster is returned.

These associations seem not to have a high importance, as they reflect relations that can be easily found out with the enumeration of the classes itself. Based on the fact that there exist only one instance of Linux_Cluster.

Linux_ClusterHostInResource reflects the relation between the Hosts and the Resources. If you call this association from the Resource end (instance of class Linux_ClusterResource specified) these instances of Linux_ClusterHost are returned, which have been defined to be part of this Resource. If you call this association from the Host end (instance of Linux_ClusterHost specified) these instances of Linux_ClusterResource are returned, where the Host is defined as member.

specified end		returned instance
Linux_ClusterResource	CR1	Linux_ClusterHost A , E , F
Linux_ClusterResource	CR2	Linux_ClusterHost C , D , G , B , F
Linux_ClusterHost	A	Linux_ClusterResource CR1
Linux_ClusterHost	F	Linux_ClusterResource CR1 , CR2

Linux_ClusterJobInQueue reflects the relation between the Jobs and the Queues. If you call this association from the Job end (instance of class Linux_ClusterJob specified) this instance of Linux_ClusterQueue is returned, where the Job was submitted to. If you call this association from the Queue end (instance of Linux_ClusterQueue specified) these instances of Linux_ClusterJob are returned, which have been submitted to this Queue for execution. Independent of the current execution state of the Jobs.

specified end		returned instance
Linux_ClusterJob	_d	Linux_ClusterQueue x
Linux_ClusterJob	_e	Linux_ClusterQueue y
Linux_ClusterQueue	x	Linux_ClusterJob _b , _a , _d
Linux_ClusterQueue	z	Linux_ClusterJob -

Linux_ClusterJobInResource reflects the relation between a Job and the Resource, used for the execution of this Job. The information in which Resource a Job has to be executed is submitted with the Job definition to the Scheduler. If you call this association from the Job end (instance of class Linux_ClusterJob specified) this instance of Linux_ClusterResource is returned, where the Job was submitted to. If you call this association from the Resource end (instance of Linux_ClusterResource specified) these instances of Linux_ClusterJob are returned, which have been submitted to this Resource for execution. Independent of the current execution state of the Job(s).

specified end		returned instance
Linux_ClusterJob	_f	Linux_ClusterResource CR2
Linux_ClusterJob	_e	Linux_ClusterResource CR1
Linux_ClusterResource	CR1	Linux_ClusterJob _b , _c
Linux_ClusterResource	CR2	Linux_ClusterJob _f

Linux_ExecutionHost reflects the relation between a running Job and the Hosts used for computation. If you call this association from the Job end (instance of class Linux_ClusterJob specified) and the Job is currently executed, these instances of Linux_ClusterHost are returned, which are listed by the 'Host Execution List'. If the Job is not running, no instances will be returned. If you call this association from the Host end (instance of Linux_ClusterHost specified) the instance of Linux_ClusterJob is returned, which is currently computed on this Host. If the Host is not in use, no instance will be returned.

specified end		returned instance
Linux_ClusterJob	_f	Linux_ClusterHost D , F , G
Linux_ClusterJob	_c	Linux_ClusterHost E , A
Linux_ClusterHost	A	Linux_ClusterJob _c
Linux_ClusterHost	D	Linux_ClusterJob _f
Linux_ClusterHost	C	Linux_ClusterJob -

Linux_ClusterSchedulerQueue reflects the relation between the Scheduler and the Queues to manage. If you call this association from the Scheduler end (instance of class Linux_ClusterScheduler specified) all instances of Linux_ClusterQueue are returned. If you call this association from the Queue end (instance of Linux_ClusterQueue specified) the one instance of Linux_ClusterScheduler is returned.

Linux_ClusterSchedulerJob reflects the relation between the Scheduler and the Jobs to manage. If you call this association from the Scheduler end (instance of class Linux_ClusterScheduler specified) all instances of Linux_ClusterJob are returned, independent of the Queue they have been submitted to. If you call this association from the Job end (instance of Linux_ClusterJob specified) the one instance of Linux_ClusterScheduler is returned.

Linux_ClusterSchedulerHost reflects the relation between the Scheduler and the Host(s) available for computation. If you call this association from the Scheduler end (instance of class Linux_ClusterScheduler specified) all instances of Linux_ClusterHost are returned, independent of the Resource they have been defined for. If you call this association from the Host end (instance of Linux_ClusterHost specified) the one instance of Linux_ClusterScheduler is returned.

2.2 Property Description

The blue marked properties are key properties of the classes. These properties are required and serve as unique identifier of the instances.

Linux_Cluster

Name	Type	Description
CreationClassName	string	Property derived from CIM_System; contains the name of the class -> 'Linux_Cluster'
Name	string	contains the unique name of the Cluster Domain, e.g. 'NCSA-SBLIM-Cluster'
NumberOfHosts	uint32	Represents the total number of Hosts (independent of the current status) of the Cluster.
NumberOfAvailableHosts	uint32	Represents the number of Hosts, currently not in use.
NumberOfAssignedHosts	uint32	Represents the number of Hosts, currently in use for job execution.
MaxNumberOfAssignableHosts	uint32	Represents the number of Hosts, currently can be used.
MaxNumberOfCapableHosts	uint32	Represents the number of Hosts, currently can be scheduled for use.
NumberOfCPUs	uint32	Represents the total number of CPUs (independent of the current status) of the Cluster.
NumberOfAvailableCPUs	uint32	Represents the number of CPUs, currently not in use.
NumberOfAssignedCPUs	uint32	Represents the number of CPUs, currently in use for job execution.
MaxNumberOfAssignableCPUs	uint32	Represents the number of CPUs, currently can be used.
MaxNumberOfCapableCPUs	uint32	Represents the number of CPUs, currently can be scheduled for use.
NumberOfJobs	uint32	Represents the total number of Jobs (status : running / pending) submitted to the Cluster.
NumberOfJobsQueues	uint32	Represents the number of Jobs, currently in status 'queued'.
NumberOfJobsHeld	uint32	Represents the number of Jobs, currently in status 'held'.
NumberOfJobsRunning	uint32	Represents the number of Jobs, currently in status 'running'.
NumberOfJobsExiting	uint32	Represents the number of Jobs, currently in status 'exiting'.
SchedulerAddress	string	The address of the machine running the scheduler.
PhysicalMemory	uint32	The sum of the physical memory of the Hosts.
TotalMemory	uint32	The sum of memory available on the Hosts for job execution.
Load	uint32	The aggregate load on the Cluster.
PercentLoad	uint64	The percentage load over all Hosts.

Linux_ClusterScheduler

Name	Type	Description
SystemCreationClassName	string	contains 'Linux_UnitaryComputerSystem'
SystemName	string	contains the 'host.domain' value of the system acting as Management Node (MN)
CreationClassName	string	Property derived from CIM_Service; contains name of the class -> 'Linux_ClusterScheduler'
Name	string	name of the service, e.g. 'Open PBS'
Address	string	...
Type	string	...

Linux_ClusterResource

Name	Type	Description
CollectionID	string	Property derived from CIM_CollectionOfMSEs, contains the unique name of the Resource
NumberOfAvailableHosts	uint32	Represents the number of Hosts, currently not in use.
NumberOfAssignedHosts	uint32	Represents the number of Hosts, currently in use for job execution.
MaxNumberOfAssignableHosts	uint32	Represents the number of Hosts, currently can be used.
MaxNumberOfCapableHosts	uint32	Represents the number of Hosts, currently can be scheduled for use.
NumberOfAvailableCPUs	uint32	Represents the number of CPUs, currently not in use.
NumberOfAssignedCPUs	uint32	Represents the number of CPUs, currently in use for job execution.
MaxNumberOfAssignableCPUs	uint32	Represents the number of CPUs, currently can be used.
MaxNumberOfCapableCPUs	uint32	Represents the number of CPUs, currently can be scheduled for use.
NumberOfJobs	uint32	Represents the total number of Jobs (status : running / pending) of the Resource instance.
NumberOfJobsQueues	uint32	Represents the number of Jobs, currently in status 'queued'.
NumberOfJobsHeld	uint32	Represents the number of Jobs, currently in status 'held'.
NumberOfJobsRunning	uint32	Represents the number of Jobs, currently in status 'running'.
NumberOfJobsExiting	uint32	Represents the number of Jobs, currently in status 'exiting'.

Linux_ClusterQueue

Name	Type	Description
SystemCreationClassName	string	contains 'Linux_UnitaryComputerSystem'
SystemName	string	contains the 'host.domain' value of the system acting as Management Node (MN)
CreationClassName	string	Property derived from CIM_JobDestination; contains the name of the class -> 'Linux_ClusterQueue'
Name	string	contains the unique name of the Queue, e.g. 'all'
NumberOfHosts	uint32	The number of Hosts assigned to Jobs submitted to this Queue (default if none is specified)
NumberOfAvailableHosts	uint32	Represents the number of Hosts, currently available to this Queue.
NumberOfAssignedHosts	uint32	Represents the number of Hosts, currently assigned to Jobs in this Queue.
MaxNumberOfHosts	uint32	The max number of Hosts available to Jobs submitted to this Queue.
MinNumberOfHosts	uint32	The min number of Hosts available to Jobs submitted to this Queue.
NumberOfCPUs	uint32	The number of CPUs assigned to Jobs submitted to this Queue (default if none is specified)
NumberOfAvailableCPUs	uint32	Represents the number of CPUs, currently available to this Queue.
NumberOfAssignedCPUs	uint32	Represents the number of CPUs, currently assigned to Jobs in this Queue.
MaxNumberOfCPUs	uint32	The max number of CPUs available to Jobs submitted to this Queue.
MinNumberOfCPUs	uint32	The min number of CPUs available to Jobs submitted to this Queue.
NumberOfJobs	uint32	Represents the total number of Jobs (status : running / pending) submitted to the Queue.
NumberOfJobsQueues	uint32	Represents the number of Jobs, currently in status 'queued'.
NumberOfJobsHeld	uint32	Represents the number of Jobs, currently in status 'held'.
NumberOfJobsRunning	uint32	Represents the number of Jobs, currently in status 'running'.
NumberOfJobsExiting	uint32	Represents the number of Jobs, currently in status 'exiting'.
Type	string	values : 'Execution' ...
Enabled	boolean	The Queue is able to accept Jobs.
Started	boolean	The Queue is active and can schedule Jobs.
MaxOfWallTime	uint64	The max time available to Jobs submitted to this Queue.
MinOfWallTime	uint64	The min time available to Jobs submitted to this Queue
WallTime	uint64	The default max time assigned to Jobs submitted to this Queue.

Linux_ClusterJob

Name	Type	Description
QueueCreationClassName	string	contains 'Linux_ClusterQueue'
QueueName	string	contains the unique name of the Queue, the Job was submitted to
CreationClassName	string	Property derived from CIM_Job; contains the name of the class -> 'Linux_ClusterJob'
Name	string	contains the unique name of the Job, e.g. 'Job1'
Owner	string	The user who submitted the Job.
User	string	...
Group	string	...
Scheduler	string	...
TimeSubmitted	datetime	The time when the Job was submitted to the Queue.
StartTime	datetime	The time when the Job was started.
ElapsedTime	datetime	Execution time of the Job.

UntilTime	datetime	Time after which the Job is invalid or should be stopped.
CPUTimeUsed	uint64	The CPU time used in seconds.
MemoryUsed	uint64	...
TotalMemoryUsed	uint64	...
WallTimeUsed	uint64	...
MaxOfWallTime	uint64	...
MaxOfCPUTime	uint64	...
NumberOfJobsExiting	uint32	Represents the number of Jobs, currently in status 'exiting'.
JobStatus	string	Free form string to represent the status of the Job.
Checkpoint	string	...
ErrorPath	string	...
OutputPath	string	...
Priority	uint32	Indicates the importance of the execution of the Job.
Rerunable	boolean	...
SessionID	string	...
VariableList	string	...
NumberOfNodes	uint16	...
ProcsPerNode	uint32	...

Linux_ClusterHost

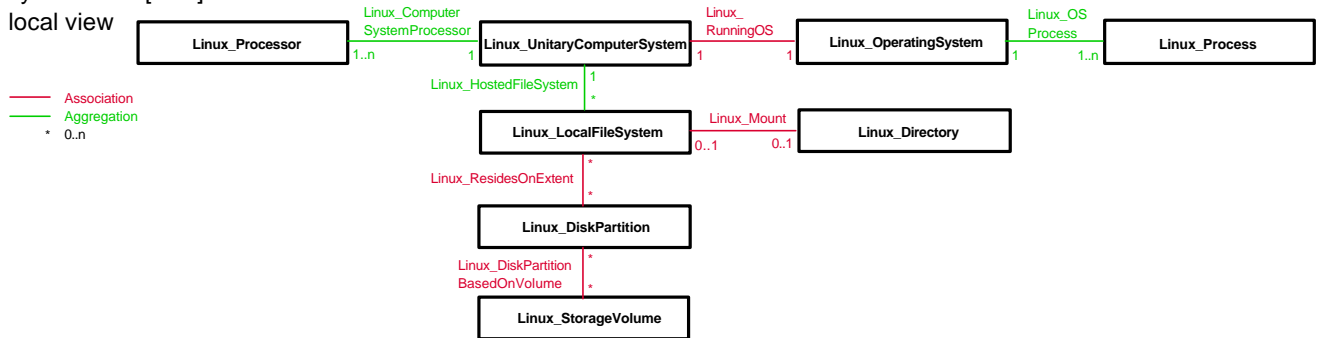
Name	Type	Description
CreationClassName	string	Property derived from CIM_System; contains the name of the class -> 'Linux_Cluster'Host
Name	string	contains the unique name of the machine -> value 'host.domain'
NumberOfCPUs	uint16	Represents the total number of CPUs of the local system.
MaxCPUSpeed	uint32	Represents the max CPU speed of the local system.

3 Cluster Management / Local Host View

3.1 Class and Association Description

system data [host]

local view



3.1 Resource Schema (local host) class & association overview - necessary CIM schema parts are Core and System

This schema reflects the Base Instrumentation of each single node participating in the cluster and will be described in this chapter at a later point in time.

tbd

Cluster Schema - Association view

