|  |  |
| --- | --- |
| Use Case Description | Slurm Create Batch  |
| Actors | OFMF Fabric Manager, Administrator, Compute nodes, Slurm Spank Plug-In |
| Description | Build out Nodes and topology for an HPC Batch Run  |
| Normal Flow | * Batch run submitted to Slurm
* Init Spank plugin
* Spank extension sets up Allocator context
* Set up logging----use the logging capabilities from the fabric provider.
* Get node list values assigned by Slurm to run the batch job—node\_list array
* Correlate compute node\_list (physical or virtual) with fabric endpoints---physical, fully virtual and para-virtual, container
* Get service key from list of path choices between the endpoints
* Provide restrictions and features of batch limitations
* Get back necessary security keys, etc. from the fabric
	+ Fabric Manager is passing the keys, not interacting with the keys
	+ The Fabric Manager should not know key information.
	+ At most, the Fabric Manager is a ‘conduit’.
	+ Necessary addressing details
	+ Get the fabric up to where the connections can be established
* Create a Zone of nodes (worker, IO)
	+ Basic capabilities
	+ Define additional capabilities
	+ QoS, default and then additionally performant
* Default connections using physical connections between workers and resources

*Slurm Requirements in/Fabric Manager Responses out---information transferred back and forth** Run batch job
	+ You are running on this physical mapping, identifier and port on the fabric.
* Spank-Fini
* Exit job
 |
| Alternate Flow 1 | * Batch run submitted to Slurm
* Init Spank plugin
* Spank extension sets up Allocator context
* Set up logging for connection
* Get node list values—node\_list array
* Correlate node\_list with endpoints
* Provide routes to nodes
* Spank detects error
* Log error
* Report Errot
* Exit job
 |