License Resources

License resources work exactly like global resources with the added advantage that they can be dynamically updated in the Supervisor by an external application.

Normally, as jobs are dispatched and executed, the Supervisor continually updates the global resource counts to match. However, many sites subscribe to a single floating license server to control all licensed applications, not just the ones running on a compute farm. If those licenses are allocated to other processes not monitored by Qube!, the Supervisor runs the risk of dispatching a job where there are no licenses available for the application it will try to execute.

By using an external application to monitor the state of the license server, the Supervisor can be updated on a regular basis to ensure the Supervisor is fully aware of the actual number of licenses available.

Configuring the system requires two separate components:

1. The Supervisor must be configured with a license resource equal to the maximum license allocation to the compute farm.
2. An external application must be developed that can monitor the license server and call `qbupdateresource` to update the Supervisor on a regular basis.

In this case, only the Supervisor needs to be configured with a license resource specification:
`supervisor_global_resources=` *resspec* `[,` *resspec*…`]`

The *resspec* uses the keyword license:
`license.`*name*`=`*quantity*

It is beyond the scope of this document to describe how to write a script that can poll a license server. At some point, however qbupdateresource should be called:
`qbupdateresource -name license.`*name*` -total` *license_total* `-used` *license_used*

where *name* is the name of the license resource in the Supervisor, *license_total* is the total number of application licenses managed by the license server, and *license_used* is the number in use at the time.

> ⚠️ When a supervisor first boots up, and **before `qbupdateresource` is run** to inform the supervisor of the actual licenses in use on the external license server, **all `license.*` resources are assumed to be fully in use**: every one of the `license.*` resources will be set to **N/N**, indicating that no resources are available. This "pessimistic" view is to avoid dispatching jobs until the supervisor receives an updated count from the external license server via **`qbupdateresource`**.

Example

Here is an example of the license resource configuration of five mental ray licenses:
`supervisor_global_resources = license.mentalray=5`

and an example call to `qbupdateresource` to register that a total 5 licenses are installed, and 2 are currently in use:
`% qbupdateresource -name license.mentalray -total 5 -used 2`