Image Load Flow

- vcld daemon is running
    - polls database every 12 seconds to check if the management node has any reservations which need to be processed
- vcld finds a reservation that needs to be processed
- vcld gathers all of the information for the reservation from the database by calling utils.pm::get_request_info() and then a DataStructure object is created
- vcld creates a new state object
    - a new state object is created based on the request.stateid column for the reservation
    - the main state modules are new.pm, image.pm, reserved.pm, reclaim.pm
- the constructor for all of the state modules is Module.pm::new()
- this constructor automatically calls an initialize() subroutine if initialize() has been implemented for the class being constructed
- State.pm contains an initialize() subroutine, and all of the state modules (new.pm, image.pm...) inherit from State.pm so the initialize() subroutine in State.pm is automatically called when vcld creates the state object
- State.pm::initialize() creates an OS object based on the OS of the image assigned to the reservation
- State.pm::initialize() creates a provisioning engin object based on the provisioning engine of the computer assigned to the reservation
- vcld forks a new process for the newly created state object
- the new process calls the state module's process() subroutine
- process() performs all of the tasks needed to process the reservation, returns true if successful, false if failed