# API summary

|  |  |  |
| --- | --- | --- |
| # | **API name and description** | **Where to look for it (initial ideas)** |
| 1 | Get physical topology – Find out all existing switches in the domain, connectivity, connection to racks, connection to hosts | * **Vitrage APIs**: Includes getting resources and topology, currently no auto discovery, switch data is loaded from configuration files, racks support not included in Mitaka, it is on the road map |
| 2 | Get virtual topology | **Vitrage API**: Includes getting resources and topology.  **Neutron** API: Currently defines networks, subnetworks and ports so we need those and I believe that neutron provides it. In my mind virtual topology is made of VMs, virtual switches and tunnels. There is a need of course build topology per tenant and per subnet |
| 3 | Get mappings: VMs to hosts, Hosts to racks, racks to switch ports, apps to VMs | **Vitrage API**: Racks will not be supported in Mitaka **Nova API**: VM to hosts  **IPMI**:??? |
| 4 | SNMP Manager – Receive SNMP traps | The most obvious example is port up/down indication which is supports on all physical switches, there may be other ways to understand it but this is defiantly the fastest way.  No current support in the virtual layer |
| 5 | Get switch status/ event | 1. **Vitrage API**: Get the state of a resource. This might either be the original state (like Nova state), or a state deduced by Vitrage (in case there is an alarm on the resource). 2. Targeted for Mitaka, but is at risk due to time limitations. 3. **SNMP traps**: Get switch port down from different switches |
| 6 | Get switch port status /event | 1. **SNMP traps**: Get switch port down/ up trap |
| 7 | Get NIC status /event | 1. **Vitrage API**: NICs support is scheduled for Mitaka. Regarding the get state API, see use case #5. 2. **SNMP traps**: Get neighbor switch port down/ up trap 3. **IPMI**: ??? |
| 8 | Get VMs status /Event | 1. **Vitrage API**: VMs are already supported in Vitrage. 2. **Nova**: ??? |
| 9 | Activate link OAM tool | **Yaedstick**: ???  **Linux tools**: Ping, sFlow, etc. use remote shell to activate  **Other:** ??? |
| 10 | Get host’s NIC statistics | **IPMI**: ???  **Other**: ??? |
| 11 | Get host process list and status | **Linux tools**: Use remote shell |
| 12 | Get OVS switch status | 1. **Vitrage API**: OVS switch support is scheduled for Mitaka. |
| 13 | Get L2 agent status | **Neutron:** ??? |
| 14 | Get OVS/ SDN switch port status | **Linux tools**: Use remote shell  **Other:** ??? |
| 15 | Get OVS port down event | **Onos:** ???  **Linux tools**: Use remote shell |
| 16 | Get Hypervisor status/ event | **Linux tools:** Use remote shell  **Nova:** ???  **Neutron:** ???  **Onos:** ???  **IPMI:** ??? |
| 17 | Check process up time | **IPMI:** ??? |
| 18 | Get host restarted event | **IPMI:** Get sys up time will show the status |
| 19 | VNC ??? | What is the use case? vNIC down, can we access the VM in this case what kind of information can we get form the VM in this case? Can it be helpful for us in anyway? I don’t have an answer for it yet |
| 20 | Get expected configuration | All related projects |
| 21 | Get actual configuration | All related projects |
|  |  |  |
|  |  |  |