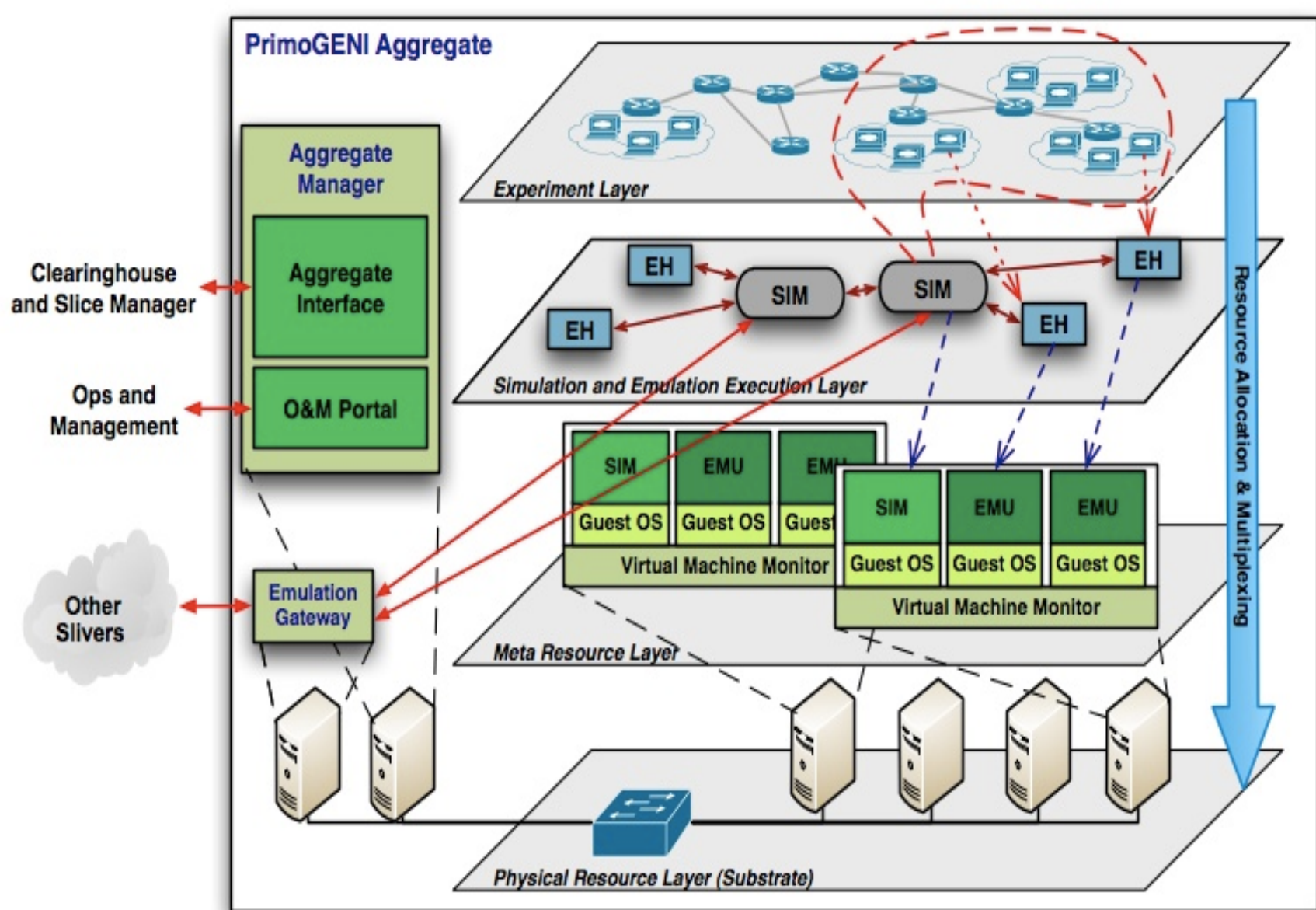


1 Overview

The **goal** of PrimoGENI is to incorporate **real-time network simulation** into the GENI "ecosystem". We are extending our existing real-time large-scale network simulator **PRIME** to become part of the GENI federation.

PrimoGENI will support large-scale GENI experiments with *millions* of **simulated** network entities (hosts, routers, and links) and *thousands* of **emulated** elements running unmodified network protocols and applications.

2 PrimoGENI Architecture



Important features of **PrimoGENI** include:

- **Resources**: abstraction of sharable features managed by a component manager and described in **Rspecs**, which define two types of resources:
 - **Meta resources**: physical or virtual hosts, and other resources managed by EmuLab.
 - **Virtual resources**: virtual network resources simulated and emulated by **PrimoGENI**.
- The **PrimoGENI** aggregate is **multi-layer**:
 - **Physical resources layer (substrate)**: cluster nodes, switches, and other physical resources, which can be queried during resource discovery.
 - **Meta resources layer**: virtual machines upon resource assignment during sliver creation.

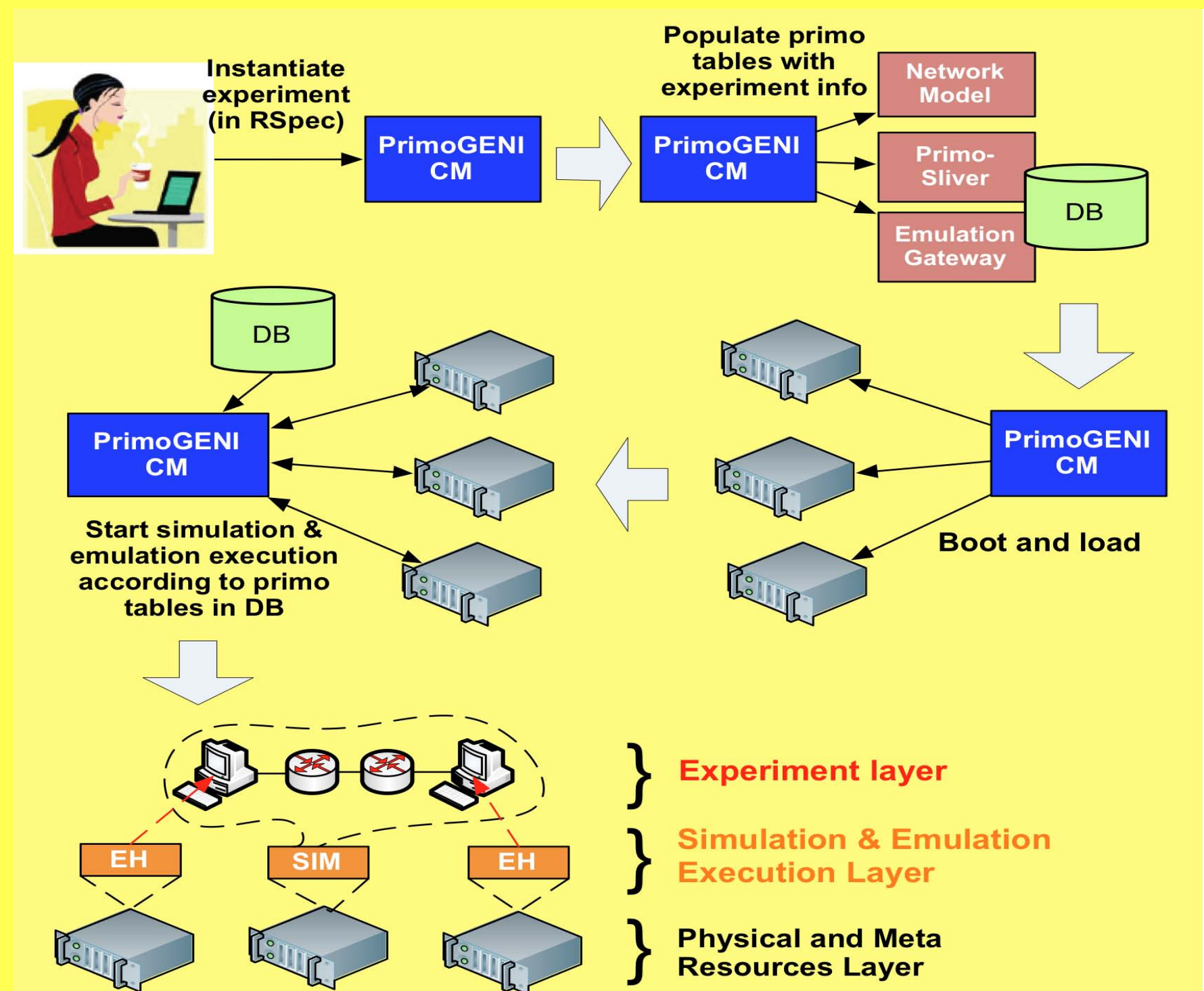
PrimoGENI uses the ProtoGENI/Emulab suite to manage physical and meta resources.

- **Simulation and emulation execution layer**: simulator instances and emulated hosts, created upon virtual network specification, and mapped to the meta resources at the layer below.
- **Experiment layer**: researchers can conduct live simulation and emulation experiments on the virtual network.

3 Integration to ProtoGENI

PrimoGENI currently has been integrated with the ProtoGENI environment at FIU:

- **Modified Rspecs** to include specification of virtual network resources
- **Added GENI-CM database tables** to store experiment information.
- **Added protocol** for initiating simulation and emulation during sliver creation.



Ongoing Efforts:

- IDE/GUI for large-scale network experiments (*network scripting* in python).
- Real-time experiment monitoring and control.

