ABSTRACT
As computation outstrips I/O performance, consolidating as much of an end to end HPC workflow onto individual nodes is a promising technique to mitigate data movement bottlenecks. However, current HPC systems are ill-equipped to perform this consolidation due to the performance impacts and software challenges of integrating HPC simulations and other workflow components that have been designed to run in isolation. To address this problem we continue our work in the design of the Hobbes OS/R [2, 4] with mechanisms to support composed applications. We discuss Leviathan and Remora, system software components that support the composition of workflows across multiple stacks of the Hobbes infrastructure. These components ease developer effort and allow users to deploy existing applications without requiring invasive application modification.

1. INTRODUCTION
Consolidating as much of an end to end HPC workflow onto individual nodes is a promising technique to mitigate data movement bottlenecks. However, current HPC systems are ill-equipped to perform this consolidation, primarily due to the challenges of effectively integrating HPC simulations and other workflow components that have been designed to run in isolation.

The Hobbes project [2] makes composition possible by provisioning multiple enclaves on a node, each of which consists of a subset of the node’s hardware and an internalized OS and runtime stack. Heretofore, Hobbes has primarily focused on provisioning isolated environments that prevent cross-enclave interference from harming application performance. However, as a result support for cross-enclave communication must be explicitly supported by the underlying enclave environments, and until now this has been limited to shared memory mappings via the XEMEM system [3].

In this poster, we present novel components of the Hobbes runtime that provide additional support for application composition. We first present Leviathan, the Hobbes node manager which provides mechanisms to administer enclaves and resources for higher-level workflows. We also present Remora, which uses the Leviathan infrastructure to make application composition possible at the MPI middleware level.

2. THE HOBBES LEVIATHAN NODE MANAGER
Leviathan is an intranode information and control service which enables the management and configuration of multiple enclaves running on the same local compute node. In this section, we discuss its key features that support composed applications.

Node Information Service Leviathan provides a node information service whereby each enclave on the node can track the state of system resources. In addition to publishing the availability of hardware resources (such as CPU cores, memory blocks, and network devices), the service also provides higher-level information, including a naming service for XEMEM shared memory keys and process IDs for remote enclave processes. Leviathan currently publishes this information in an in-memory database, and exports it to enclaves via XEMEM.

User-Level Resource Management Leviathan internalizes the management of physical hardware resources by leveraging the resource offlining capabilities of modern Linux kernels. Once resources have been offlineed from the host Linux environment, users can allocate them to support the creation of enclaves and virtual machines. Further-
more, Leviathan processes can directly allocate additional resources by querying the information service, requesting resources, and assigning them to the host OS environment managing its enclave.

**Enclave Lifecycle Management** The resources managed by Leviathan can be space partitioned into multiple enclaves and virtual machines. Leviathan provides tools which allows users to create enclaves / virtual machines by specifying the target hardware (CPU cores, memory blocks, etc.) and software (operating system) environment. Leviathan also allows users to destroy enclaves or discover unexpected crashes via heartbeats. Furthermore, Leviathan allows users to launch applications into existing enclaves or virtual machines based on their desired OS environment.

**Inter-Enclave Communication** Finally, Leviathan provides inter-enclave communication to support higher-level application composition. Leviathan supports the construction of XEMEM shared memory segments by publishing XEMEM identifiers in the node information service. However, it also supports additional mechanisms, such as cross-enclave notifications/signals and remote procedure calls (RPC) for dispatching tasks for remote execution.

### 3. REMORA

As part of our work on the Node Virtualization Layer (NVL) of the Hobbes project we have provided a mechanism to compose MPI applications without requiring invasive source code modifications. Remora provides MPI components and support libraries which make it possible for MPI to seamlessly integrate into the Hobbes NVL environment. These components are as follows:

**The Remora Cross OS MPI Runtime** is a novel runtime that is designed to replace ORTE in OpenMPI. This runtime provides a set of services that minimize POSIX dependencies to run across a wide variety of operating systems, especially lightweight kernels like Kitten [5]. Most of the work of the runtime, including process startup, resource allocation and teardown are done by per-enclave control processes that execute in each enclave environment and query the Leviathan information service and communicate via XEMEM [3], which removes the kernel from the control plane. Remora utilizes the Vader [6] BTL, which is built on the XPMEM API and thus is supported via XEMEM without modification.

**Rank spaces** are a mechanism that Remora uses to make composition of MPI applications possible via custom intra-communicators. Individual applications maintain their original MPI_COMM_WORLD communicator. However, the entire composed set of applications now have a new intra-communicator MPI_COMM_UNIVERSE, which is the union of all application communicators.

**The Remora Process Management Infrastructure (RPMI)** is an implementation of the PMI [1] client/server infrastructure, built over Leviathan. The RPMI maintains a global database of mappings of processes to enclaves in an in-memory database. Individual MPI processes query the Leviathan information service in order to discover the global state of MPI applications.

**The Remora Process Launcher** launches composed applications using an XML file which specifies a multilevel topology mapping applications to communicators and enclaves to applications. The process launcher extends the support provided by the Leviathan lifecycle management tools to allow global rank assignment and spawning jobs across enclaves.

### References


