

- **Partition and Cache Management**

VMs can be quite large and a lot of their disk content can be divided into partitions that are either (1) empty, (2) read-only and shareable by multiple running VMs, and/or (3) have been seen before and are thus cacheable.

The Workspace Service can currently handle multiple partitions, some decompression, and also blankspace creation. But future work entails a full partition management solution including more intelligent caching, security, and transfer functionality.

- **Networking**

The Workspace Service currently handles bridging VM NICs to various networks, can manage IP pools, and can deliver networking information to VMs via the DHCP protocol. But more is needed for a full networking solution, including allowing clients to express connectivity requirements (such as "this workspace should only be allowed to access to group X of workspaces" or "this workspace only needs outgoing WAN access", etc).

- **Fine-grained enforcement**

Simple resource enforcement is possible with the tools that VMMs inherently do well at, such as memory enforcement. When starting to manage I/O together with CPU % enforcement, there is an interdependence between them and "leakage" between VM resource consumption that must be accounted for in a fine-grained manner in order to be able to guarantee the client the exact resources it has negotiated.

- **Local Resource Management**

The Workspace Service currently contains a simple local resource management solution for managing a pool of VMM nodes. We are working on expanding and maturing the local resource situation, experimenting with both (a) the proper way to schedule VMs with overhead awareness and (b) ways to integrate with current local resource management systems running regular jobs.

- **Virtual Clusters**

Deploying multiple workspaces at once that depend on each other has both dependency and configuration management issues. We are working on *scalable* techniques for deploying, managing, and migrating virtual clusters.

- **Configuration Management**

Deploying one or many workspaces often means that configuration information only available at deploy-time must be made available. Doing this in a thoughtful manner involves both configuration management systems and having a set of standard virtual appliance components to piece together in predictable ways.

- **Security/Trust Management**

Resource provider policies may only accept certain VMs on their sites and developing an assertion/attestation framework for VMs is necessary for this to not be ad hoc and error prone (this includes not only assertions, but the signing and validation mechanisms for both particular VM files and workspace metadata).

- **VMware backend**

VMware is freely available now and is good additional backend candidate (currently the service supports only Xen).

- **Ease of use, performance**

An ongoing effort to make the service and clients easier and more effective for both programmatic integration and human use. A C based client will help with integration and performance. Recent examples of this priority are the new User Quickstart Guide, binary installation options, and pre-packaged sample VMs.