may be implemented as a microprocessor, microcontroller, application specific integrated circuit (ASIC), discrete logic, or a combination of other type of circuits or logic. Similarly, memories may be DRAM, SRAM, Flash or any other type of memory. Flags, data, databases, tables, entities, and other data structures may be separately stored and managed, may be incorporated into a single memory or database, may be distributed, or may be logically and physically organized in many different ways. The components may operate independently or be part of a same program. The components may be resident on separate hardware, such as separate removable circuit boards, or share common hardware, such as a same memory and processor for implementing instructions from the memory. Programs may be parts of a single program, separate programs, or distributed across several memories and processors.

The respective logic, software or instructions for implementing the processes, methods and/or techniques discussed throughout this disclosure may be provided on computer-readable media or memories or other tangible media, such as a cache, buffer, RAM, removable media, hard drive, other computer readable storage media, or any other tangible media or any combination thereof. The tangible media include various types of volatile and nonvolatile storage media. The functions, acts or tasks illustrated in the figures or described herein may be executed in response to one or more sets of logic or instructions stored in or on computer readable media. The functions, acts or tasks are independent of the particular type of instructions set, storage media, processor or processing strategy and may be performed by software, hardware, integrated circuits, firmware, micro code, or any type of other processor, operating alone or in combination. Likewise, processing strategies may include multiprocessing, multitasking, parallel processing and/or any other processing strategy known now or later discovered. In one embodiment, the instructions are stored on a removable media device for reading by local or remote systems. In other embodiments, the logic or instructions are stored in a remote location for transfer through a computer network or over telephone lines. In yet other embodiments, the logic or instructions are stored within a given computer, CPU, GPU, or system.

A second action may be said to be "in response to" a first action independent of whether the second action results directly or indirectly from the first action. The second action may occur at a substantially later time than the first action and still be in response to the first action. Similarly, the second action may be said to be in response to the first action even if intervening actions take place between the first action and the second action, and even if one or more of the intervening actions directly cause the second action to be performed. For example, a second action may be in response to a first action if the first action sets a flag and a third action later initiates the second action whenever the flag is set.

To clarify the use of and to hereby provide notice to the public, the phrases "at least one of <A>, <B>, . . . and <N>" or "at least one of <A>, <B>, . . . <N>, or combinations thereof" or "<A>, <B>, . . . and/or <N>" are defined by the Applicant in the broadest sense, superseding any other implied definitions hereinbefore or hereinafter unless expressly asserted by the Applicant to the contrary, to mean one or more elements selected from the group comprising A, B, . . . and N. In other words, the phrases mean any combination of one or more of the elements A, B, . . . or N including any one element alone or the one element in combination with one or more of the other elements which may also include, in combination, additional elements not listed.

While various embodiments of the innovation have been described, it will be apparent to those of ordinary skill in the art that many more embodiments and implementations are possible within the scope of the innovation. Accordingly, the innovation is not to be restricted except in light of the attached claims and their equivalents.

What is claimed is:

1. A memory appliance comprising:

a processor;

a communication interface;

a memory configured in an address space addressable by the processor; and

a region access unit configured to allocate a region of the memory for use as an external primary memory of a client on a network, wherein the external primary memory of the client is primary memory of the client that is external to the client and is accessible to the client over the network via the communication interface of the memory appliance,

wherein the communication interface is configured to provide the client access to the region of the memory via client-side memory access before initialization of all of the region, wherein the processor of the memory appliance is bypassed if a client-side memory access request is for an initialized portion of the region of the memory but the processor is further configured to initialize, if the client-side memory access request is for an uninitialized portion of the region, the uninitialized portion in response to the client-side memory access request, wherein the region access unit is configured to allocate the region of the memory in response to a first memory allocation request received from the client over the network, wherein a subset of the allocated region of the memory is allocatable at the client, as a slab of external memory in response to a second memory allocation request received at the client, and wherein the communication interface is configured to provide the client access to data in the slab of external memory in the region of the memory via client-side memory access before initialization of all of the region.

2. The memory appliance of claim **1**, wherein initialization of the uninitialized portion includes a setting of the uninitialized portion to all zeros or all ones.

3. The memory appliance of claim **1**, wherein initialization of the uninitialized portion includes a copying of data from a backing store to the uninitialized portion.

4. The memory appliance of claim **1**, wherein initialization of the uninitialized portion includes a copying of data from a second memory appliance to the uninitialized portion.

5. The memory appliance of claim **1**, wherein initialization of the region includes a restore of the region of the memory after a reboot of the memory appliance, and wherein a portion of the region of the memory that has never been written to after allocation is set to all zeros or all ones instead of being restored.

6. The memory appliance of claim **1**, wherein the region access unit is configured to configure the communication interface to treat portions of the region of the memory as uninitialized based on a setting and/or a clearing of one or more indicators indicative of presence and/or access permission of the portions of the region.

7. The memory appliance of claim 1, wherein the region access unit is configured to allocate the region of the memory for the client after client-side memory access to the region is provided.

8. The memory appliance of claim 1, wherein the processor is configured to:

memory map at least a portion of a file to the region of the memory, wherein a virtual address addressable by the processor is generated, and the at least a portion of the file is accessible through the region of the memory at the virtual address; and

register the virtual address with the communication interface, wherein registration of the virtual address provides client-side memory access to the region of memory, wherein the client-side memory access provides the client of the system access to the region of the memory over the network.

9. The memory appliance of claim 8, wherein the region of the memory is primary external memory to the client.

10. The memory appliance of claim 8, wherein the file is stored on a device external to the memory appliance.

11. The memory appliance of claim 1, wherein the processor is further configured to change data in a portion of the region of the memory from a first memory tier to a second memory tier.

12. The memory appliance of claim 8, wherein the processor is further configured to track and mark as dirty any portion of the region of the memory that is written to via an observable write, and write dirty portions of the region of the memory to the file but portions of the region of the memory that are not dirty are not written to the file.

13. The memory appliance of claim 1, wherein the processor is further configured to perform a batch portion invalidation and/or reclaim of portions of the region of the memory and indicate to the communication interface that the portions of the region of the memory are reclaimed and/or invalidated.

14. The memory appliance of claim 8, wherein the region of the memory is not large enough to store all of the at least a portion of the file at once.

* * * * *