CLAIMS

1. A computer-implemented method of managing memory in a computer comprising:

monitoring usage of a primary memory associated with the computer, the primary memory storing uncompressed memory blocks in a ready state;

in response to primary memory usage by the uncompressed memory blocks in the ready state exceeding a ready state threshold,

compressing at least some of the uncompressed memory blocks in the ready state to form compressed memory blocks,

storing the compressed blocks as memory blocks in a compressed state in the primary memory, and

storing the at least some of the uncompressed memory blocks as memory blocks in a ready and compressed state in the primary memory, wherein each of the memory blocks in the ready and compressed state in the primary memory is one of the uncompressed memory blocks that has a corresponding compressed block also stored in the primary memory;

in response to primary memory usage by the memory blocks in the ready and compressed state exceeding a release threshold, releasing at least some of the memory blocks in the ready and compressed state; and

in response to primary memory usage by the memory blocks in the compressed state exceeding a compressed threshold, transferring at least some memory blocks in the compressed state to a secondary memory associated with the computer to form memory blocks in an on disk state.

2. The method of claim 1, further comprising:

receiving a request from a module executing on the computer to access a memory block;

determining whether the requested memory block is in the ready state; and

responsive to a determination that the requested memory block is not in the ready state, changing the requested memory block into the ready state; and

serving the requested memory block in the ready state in response to the access request.

3. The method of claim 1, further comprising:

in response to primary memory usage falling below a usage threshold, transferring at least some memory blocks in the on disk state from the secondary memory to the primary memory.

4. The method of claim 3, wherein the memory blocks transferred from the secondary memory to the primary memory are stored in the primary memory in the compressed state, further comprising:

retaining the transferred memory blocks in the on disk state on the secondary memory; and

in response to primary memory usage exceeding the compressed threshold, releasing the memory blocks transferred from the secondary memory to the primary memory stored in the compressed state.

5. The method of claim 1, further comprising:

receiving a request from a module executing on the computer to access a memory block in the ready and compressed state;

determining whether the access request is a read request or a write request;

responsive to determining that the access request is a write request:

changing the requested memory block into the ready state; and

releasing a memory block in the compressed state corresponding to the requested memory block;

responsive to determining that the access request is a read request:

serving the read request using the memory block in the ready and compressed state; and

maintaining the requested memory block in the ready and compressed state.

6. The method of claim 1, wherein compressing at least some of the uncompressed memory blocks in the ready state to form compressed memory blocks comprises:

compressing the uncompressed memory blocks in the ready state in an ascending order of elapsed time since allocation of the uncompressed memory blocks.

7. The method of claim 1, wherein the primary memory provides faster access to memory blocks relative to the secondary memory.

8. A computer system for managing memory in a computer, the system comprising:

a non-transitory computer-readable storage medium storing executable computer program instructions, the computer program instructions comprising instructions that when executed cause a computer processor to perform steps, comprising:

monitoring usage of a primary memory associated with the computer, the primary memory storing uncompressed memory blocks in a ready state;

in response to primary memory usage by the uncompressed memory blocks in the ready state exceeding a ready state threshold,

compressing at least some of the uncompressed memory blocks in the ready state to form compressed memory blocks,

storing the compressed blocks as memory blocks in a compressed state in the primary memory, and

storing the at least some of the uncompressed memory blocks as memory blocks in a ready and compressed state in the primary memory, wherein each of the memory blocks in the ready and compressed state in the primary memory is one of the uncompressed memory blocks that has a corresponding compressed block also stored in the primary memory;

in response to primary memory usage by the memory blocks in the ready and compressed state exceeding a release threshold, releasing at least some of the memory blocks in the ready and compressed state; and

in response to primary memory usage by the memory blocks in the compressed state exceeding a compressed threshold, transferring at least some memory blocks in the compressed state to a secondary memory associated with the computer to form memory blocks in an on disk state.

9. The system of claim 8, wherein the steps performed by the computer processor further comprise:

receiving a request from a module executing on the computer to access a memory block;

determining whether the requested memory block is in the ready state; and

responsive to a determination that the requested memory block is not in the ready state, changing the requested memory block into the ready state; and serving the requested memory block in the ready state in response to the access request.

10. The system of claim 8, wherein the steps performed by the computer processor further comprise:

in response to primary memory usage falling below a usage threshold, transferring at least some memory blocks in the on disk state from the secondary memory to the primary memory.

11. The system of claim 10, wherein the memory blocks transferred from the secondary memory to the primary memory are stored in the primary memory in the compressed state, wherein the steps performed by the computer processor further comprise:

retaining the transferred memory blocks in the on disk state on the secondary memory; and

in response to primary memory usage exceeding the compressed threshold, releasing the memory blocks transferred from the secondary memory to the primary memory stored in the compressed state.

12. The system of claim 8, wherein the steps performed by the computer processor further comprise:

receiving a request from a module executing on the computer to access a memory block in the ready and compressed state;

determining whether the access request is a read request or a write request;

responsive to determining that the access request is a write request:

changing the requested memory block into the ready state; and

releasing a memory block in the compressed state corresponding to the requested memory block;

responsive to determining that the access request is a read request:

serving the read request using the memory block in the ready and compressed state; and

maintaining the requested memory block in the ready and compressed state.

13. The system of claim 8, wherein compressing at least some of the uncompressed memory blocks in the ready state to form compressed memory blocks comprises:

compressing the uncompressed memory blocks in the ready state in an ascending order of elapsed time since allocation of the uncompressed memory blocks.

14. The system of claim 8, wherein the primary memory provides faster access to memory blocks relative to the secondary memory.

15. A non-transitory computer readable storage medium storing executable computer program instructions for managing memory in a computer, the computer

program instructions comprising instructions that when executed cause a computer processor to:

monitor usage of a primary memory associated with the computer, the primary memory storing uncompressed memory blocks in a ready state;

in response to primary memory usage by the uncompressed memory blocks in the ready state exceeding a ready state threshold,

compressing at least some of the uncompressed memory blocks in the ready state to form compressed memory blocks,

storing the compressed blocks as memory blocks in a compressed state in the primary memory, and

storing the at least some of the uncompressed memory blocks as memory blocks in a ready and compressed state in the primary memory, wherein each of the memory blocks in the ready and compressed state in the primary memory is one of the uncompressed memory blocks that has a corresponding compressed block also stored in the primary memory;

in response to primary memory usage by the memory blocks in the ready and compressed state exceeding a release threshold, release at least some of the memory blocks in the ready and compressed state; and

in response to primary memory usage by the memory blocks in the compressed state exceeding a compressed threshold, transfer at least some memory blocks in the compressed state to a secondary memory associated with the computer to form memory blocks in an on disk state.

16. The computer readable medium of claim 15, wherein the computer program instructions for managing memory in the computer further comprise instructions that when executed cause the computer processor to:

receive a request from a module executing on the computer to access a memory block;

determine whether the requested memory block is in the ready state; and

responsive to a determination that the requested memory block is not in the ready state, change the requested memory block into the ready state; and

serve the requested memory block in the ready state in response to the access request.

17. The computer readable medium of claim 15, wherein the computer program instructions for managing memory in the computer further comprise instructions that when executed cause the computer processor to:

in response to primary memory usage falling below a usage threshold, transfer at least some memory blocks in the on disk state from the secondary memory to the primary memory.

18. The computer readable medium of claim 17, wherein the memory blocks transferred from the secondary memory to the primary memory are stored in the primary memory in the compressed state, wherein the computer program instructions for managing memory in the computer further comprise instructions that when executed cause the computer processor to:

retain the transferred memory blocks in the on disk state on the secondary memory; and

in response to primary memory usage exceeding the compressed threshold, release the memory blocks transferred from the secondary memory to the primary memory stored in the compressed state.

19. The computer readable medium of claim 15, wherein the computer program instructions for managing memory in the computer further comprise instructions that when executed cause the computer processor to:

receive a request from a module executing on the computer to access a memory block in the ready and compressed state;

determine whether the access request is a read request or a write request;

responsive to determining that the access request is a write request:

change the requested memory block into the ready state; and

release a memory block in the compressed state corresponding to the requested memory block;

responsive to determining that the access request is a read request:

serve the read request using the memory block in the ready and compressed state; and

maintain the requested memory block in the ready and compressed state.

20. The computer readable medium of claim 15, wherein the computer program instructions for compressing at least some of the uncompressed memory blocks in the ready state to form compressed memory blocks comprise instructions that when executed cause the computer processor to:

compress the uncompressed memory blocks in the ready state in an ascending order of elapsed time since allocation of the uncompressed memory blocks.