

CLAIMS

1. A data preservation subsystem in a computing system, the data preservation subsystem comprising:
 - a battery-backed memory having an aperture, the aperture having a flush order which is an order of copying data from the aperture to an aperture-associated non-volatile storage in response to a data integrity threat, the flush order defining a first-to-flush end of the aperture and a last-to-flush end of the aperture;
 - a data preservation circuitry in operable communication with the battery-backed memory, the data preservation circuitry configured to perform data preservation steps that include (a) receiving a request to store a dataset A-data in the aperture, A-data including data which is designated as critical data, (b) identifying a portion A-memory of unallocated memory of the aperture, A-memory being large enough to hold A-data, A-memory having an address which is closer to the aperture first-to-flush end than any other address of any other unallocated memory of the aperture, (c) marking A-memory as allocated and placing a copy of A-data in A-memory, (d) receiving a request to store a dataset Z-data in the aperture, Z-data not including any data which is designated as critical data, (e) identifying a portion Z-memory of unallocated memory of the aperture, Z-memory being large enough to hold Z-data, Z-memory having an address which is closer to the aperture last-to-flush end than any other address of any other unallocated memory of the aperture, (f) marking Z-memory as allocated and placing a copy of Z-data in Z-memory;wherein the data preservation subsystem provides a higher likelihood of successfully flushing critical data and thereby preserving it than by storing data in the battery-backed memory without regard to data criticality.
2. The data preservation subsystem of claim 1, wherein the battery-backed memory aperture resides in or is controlled by a network node X, and the request to store A-data in the aperture was sent from a different network node Y.
3. The data preservation subsystem of claim 2, wherein the network node X includes a storage node in a cloud, and the network node Y includes a compute node in the cloud.
4. The data preservation subsystem of claim 2, wherein the network node X

and the network node Y are related by a storage relationship R, where R is defined such that two nodes M and N are related by R when M stores data on behalf of N, and wherein R is not a one-to-one relationship of network node X and network node Y.

5. The data preservation subsystem of claim 1, wherein the battery-backed memory comprises NVDIMM memory.

6. The data preservation subsystem of claim 1, wherein the computing system comprises virtual machines, the dataset A-data is sent to the data preservation subsystem from a virtual machine VM-A, and the dataset Z-data is sent to the data preservation subsystem from a virtual machine VM-Z.

7. A method for data preservation in a computing system, the method comprising:

receiving a request to store a dataset A-data in an aperture of a battery-backed memory, the aperture having a flush order which is an order of copying data from the aperture to an aperture-associated non-volatile storage in response to a data integrity threat, the flush order defining a first-to-flush end of the aperture and a last-to-flush end of the aperture, A-data including data which is designated as critical data;

identifying a portion A-memory of unallocated memory of the aperture, A-memory being large enough to hold A-data, A-memory having an address which is closer to the aperture first-to-flush end than any other address of any other unallocated memory of the aperture;

marking A-memory as allocated and placing a copy of A-data in A-memory;

receiving a request to store a dataset Z-data in the aperture, Z-data not including any data which is designated as critical data;

identifying a portion Z-memory of unallocated memory of the aperture, Z-memory being large enough to hold Z-data, Z-memory having an address which is closer to the aperture last-to-flush end than any other address of any other unallocated memory of the aperture; and

marking Z-memory as allocated and placing a copy of Z-data in Z-memory;

whereby a likelihood of successfully flushing critical data and thereby preserving it is increased by placing the critical data ahead of non-critical data in the flush-order in the battery-backed memory.

8. The method of claim 7, further comprising at least one of the following:
determining a size of the aperture based at least in part on a battery characteristic

of the battery-backed memory;
determining a location of the aperture based at least in part on a battery characteristic of the battery-backed memory;
resizing the aperture based at least in part on a change in a battery characteristic of the battery-backed memory; or
moving the aperture based at least in part on a change in a battery characteristic of the battery-backed memory.

9. The method of claim 7, further comprising flush defragging at least a defrag portion of the aperture, the defrag portion including critical data.

10. The method of claim 7, further comprising designating at least a portion of A-data as critical data based on at least one of the following criteria: a workload criticality, a virtual machine priority, or a customer status.

11. The method of claim 7, further comprising detecting the data integrity threat, flushing all of the critical data from the aperture to the non-volatile storage, and failing to flush at least some non-critical data from the aperture to the non-volatile storage.

12. The method of claim 7, further comprising detecting the data integrity threat, flushing at least all of the critical data from the aperture to the non-volatile storage, and then restoring all of the flushed data from the non-volatile storage into at least one of: a volatile memory, the battery-backed memory from which the data was flushed, or a different battery-backed memory than the battery-backed memory from which the data was flushed.

13. The method of claim 7, further comprising at least one of the following:
keeping at least one copy of Z-data outside the aperture and outside the aperture-associated non-volatile storage;
specifying at least one of an aperture size or an aperture address, and wherein said specifying is based on at least one of the following: a battery capacity, a battery age, or a battery reliability value;
flushing critical data from the aperture to the non-volatile storage, wherein said flushing comprises executing firmware which has a unified extensible firmware interface;
reserving at least ten percent of the aperture to hold data which is not designated as critical data;
moving data which is not designated as critical data to a storage location outside the aperture in order to make space for data which is designated as critical

data; or
defragging at least a portion of the aperture, thereby reducing or eliminating an unallocated portion that was previously located between two allocated portions of the aperture.

14. The method of claim 7, wherein the battery-backed memory aperture resides in a physical machine denoted here as M and the request to store A-data in the aperture was sent from a different physical machine denoted here as N.

15. The method of claim 7, comprising storing the A-data in only one storage node and storing copies of the Z-data in multiple storage nodes.