Definition and Overview

DIRECTORY POOL: A Heap-Like Data Structure Handled Directory by the Operating System for the Combined Random and Nonrandom Individual Selection of Unique Constituent Nodes.

“A Heap-Like Data Structure...”: A directory pool works like a heap in the following respects:
1.) The Pool is initialized prior to runtime, wherein it is populated with Nodes.
2.) Nodes can be selected from the Pool at runtime.
3.) Selecting a Node removes it from the Pool.

“...for the Combined Random and Nonrandom...”: Once a directory pool is initialized, nodes may be selected from it randomly. Additionally, specific nodes may also be selected from the pool. Finally, a random selection may be made from a subset of nodes within the directory pool.
Individual Selection: Nodes are selected individually at runtime.

“..of Unique Constituent Nodes”: Because nodes are removed upon selection, it is impossible to re-select a particular node until the directory pool has been reinitialized.

Advantages
The Directory Pool Data Structure has the following advantages:
1.) May be accessed by multiple programs simultaneously without the possibility of deadlocking due to multiple scripts accessing and modifying the file together.
2.) Uses simple OS functions (create/delete/remove files) which are fast and safe.
3.) Supports the combined random and non-random selection of member nodes depending on the users needs at runtime.
4.) Time and memory intensive operations (re-initialization of the pool) can be done outside of regular runtime, improving performance during.

Procedure
1.) A directory known as the pool is created
2.) Files are created in the pool, each with a unique filename
3.) Data may be added to the files. This may be skipped as unique filenames are sufficient
4.) A file is selected from the pool based on some criteria (random, specific, random-within-subset).
5.) Data from the file is passed to a program from the selected file.
6.) The selected file is deleted from the directory.