

NAME

ssm_options – Run-time options used by the Shore Storage Manager

DESCRIPTION

The Shore Storage Manager (SSM) recognizes following run-time options. Values for these options are generally stored in a file, but can also be read from command lines of servers (it all depends on what the server is written to do). A sample configuration option file is `ssh/.shoreconfig.example`, which is in the source release.

The following options must be supplied (they do not have default values):

sm_bufpoolsize

Determines the size of the buffer pool in kilobytes. **Values:** integer greater than or equal to 80 (10 pages' worth).

sm_logdir

Determines the location of the log. **Values:** path name for a raw device or a directory. May be relative or absolute.

The following options have default values:

sm_reformat_log

Determines whether the logs will be reformatted, bypassing any recovery. **Values:** yes, no. **Default value:** no.

sm_prefetch

Determines whether, during a file scan, pages will be prefetched using some experimental prefetch code. **Values:** yes, no. **Default value:** no.

sm_locktablesize

Determines size of lock table in buckets. some experimental prefetch code. **Values:** integer greater than or equal to 64. Whatever value you give is rounded up to the nearest power of 2, then converted to the largest prime number less than that power of 2, up to a maximum of 8388593 (prime less than $2^{23} = 8388608$). **Default value:** 64000 (results in 65521 buckets).

sm_logsize

Determines the upper limit on the log in kilobytes. If 0 is given, and the path given for names a raw device, the real device size is used. When the log is a directory, it comprises 8 partitions. Each partition can reach at most 1/8 the total logsize given here. When half a max-sized partition is written or 1 MB is written, whichever comes first, a checkpoint is forced. **Values:** Values: 0 or integer > 1000. **Default value:** 10000 (10 MB).

sm_logbufsize

Determines the size of the buffer used for buffering log records in shared memory (in Unix only) in kilobytes. **Values:** Values: integer representing between 4 and 128 pages, inclusive. **Default value:** 128 (128 KB or 16 pages when built with 8KB pages).

sm_backgroundflush

Determines whether the background buffer-cleaner thread is enabled. (This thread implements the 'steal' buffer-management policy.) **Values:** Values: yes, no **Default value:** yes

sm_diskrw

Path name to the disk I/O program forked by the SM. Not applicable on NT. **Values:** Values: path name **Default value:** diskrw

sm_errlog

Determines where certain error logging will be written. **Values:** Values: - (indicates stderr) or a file name. **Default value:** -

sm_errlog_level

Determines what level of seriousness of messages will be logged. These values correspond to the Unix syslog levels. All the logging messages in the SM have seriousness `error` or `info`. **Values:** Values: none, emerg, fatal, alert, internal, error, warning, info, debug **Default value:** error

The next three options determine how many fine-grained locks on a given resource cause the lock manager to escalate to a resource-level lock. With the default values, the lock manager will never escalate a request for a file lock to a request for a volume lock; after 25 requests for pages in the same store, it will escalate a request for a page lock to a request for a store lock, and after 5 requests for records on the same page, it will escalate a request for a record lock to a request for a page lock. If you do not want any lock escalation, set all three options to 0.

sm_lock_escalate_to_page_threshold

Values: Values: 0 (no escalation) or positive integer **Default value:** 5

sm_lock_escalate_to_store_threshold

Values: Values: 0 (no escalation) or positive integer **Default value:** 25

sm_lock_escalate_to_volume_threshold

Values: Values: 0 (no escalation) or positive integer **Default value:** 0

sm_num_lid_cache_entries

Determines the size of the logical-id cache. **Values:** Values are integers greater than 0. **Default value:** 10000.

sm_cc_alg

Default locking protocol for files. **Values:** record, page, file. **Default value:** record. (A transaction's locking protocol can be set explicitly with `ss_m::set_xct_lock_level(concurrency_t`

sm_log_warn

Threshold (in percent) of log used by active transactions, at which callback is made. **Values:** 0 - 100 **Default value:** 40.

The following options are for internal use by the developers of the Storage Manager and should not be changed:

sm_bfm_strategy

For testing different buffer-manager strategies. No alternatives are implemented in this release.

sm_processor_set

Ignored unless the SM is configured and built on a Sparc/Solaris with SOLARIS2_PSETS #defined in the shore.def file, and processor sets are supported on the architecture.

sm_logging

Turn on/off logging. SM is not guaranteed to work if turned off.

sm_scriptlog

Ignored.

sm_scriptlog_level

Ignored.

sm_dcommit_timeout

Used for distributed transaction commit coordinator (not in this release).

VERSION

This manual page applies to Version 2.0 of the Shore Storage Manager.

SPONSORSHIP

The Shore project is sponsored by the Advanced Research Project Agency, ARPA order number 018 (formerly 8230), monitored by the U.S. Army Research Laboratory under contract DAAB07-91-C-Q518. Further funding for this work was provided by DARPA through Rome Research Laboratory Contract No. F30602-97-2-0247.

COPYRIGHT

Copyright (c) 1994-1999, Computer Sciences Department, University of Wisconsin -- Madison. All Rights Reserved.

SEE ALSO

options(common),