HyperCache

COLLABORATORS			
	TITLE : HyperCache		
ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY		January 19, 2023	

REVISION HISTORY			
NUMBER	DATE	DESCRIPTION	NAME

Contents

Chapter 1

HyperCache

1.1 HyperCache.guide

```
  This document describes the functions of HyperCache V2.0


Late Breaking News
 New Information Not In You Manual

Parameters
         Command Line and ARexx parameters

ARexx
              How to use the ARexx port

HyperCacheStats
    A utility to monitor HyperCache performance

Index
              Index
```

1.2 late breaking news

```
Late Breaking News
******************


  The following information is new, and may not be found
  in your manual:

  - If HyperCache cannot be started at boot time, a requester
    will be displayed indicating the cause of the failure.
    For example, if you set your HyperCachePrefs such that a
    non-existant device is specified, you will see the message:

    "HyperCache could not find the device specified"
```

Any such requesters indicate that you must investigate the cause of the failure, or as an alternative, re-install HyperCache.

- If you run the installation program twice, and specify the 68000 version during one session, and the 68020 version during the other, it is possible that you will have two copies of HyperCache in your WBStartup: drawer. If this occurs, simply use Workbench to remove the copy that you do not wish to start from your WBStartup: drawer.

1.3 HyperCache.guide/Parameters

Parameters
****

The available parameters and their descriptions are:

DEVICE
Specifies which device to use

UNIT
Specifies which unit to use

NO_UNIT_CHECK
Don't check which unit access is to

NUM_LINES
Specifies the number of lines to use

LINE_SIZE
Specifies the size of each line

WRITE_RETENTION_TIME
Specifies the maximum time to keep data

QUIET_WRITE_TIME
Specifies time to wait for a drive to go quiet

DONOTWAIT
A Workbench parameter (See AmigaDOS documentation)

TOOLPRI
A Workbench parameter (See AmigaDOS documentation)

?
Ask for help

Note that any parameters altered *after* HyperCache has been started will not take effect until the next time HyperCache is started. In general, this will

require restarting your Amiga.

1.4 HyperCache.guide/DEVICE

```
                DEVICE
------

Format:
     DEVICE=<devicename>

Default:
     gvpscsi.device

Example:
     DEVICE=gvpscsi.device

Description:
     This specifies the name of the device that is to be cached.  In order to
     determine the device name for a particular storage volume, you may wish
     to run HyperCacheStats and select the volume name in the list of
     available volumes.  The device will be automatically displayed in the
     "Device" field of that program for your reference.

See also:

                 UNIT
```

1.5 HyperCache.guide/UNIT

```
                UNIT
----

Format:
     UNIT=<unit number>

Default:
     0

Example:
     UNIT=0

Description:
     This parameter specifies which unit number to cache. Each physical
     device (eg: hard drive, floppy drive) connected to a controller
     (or "Device") has a unit number.  For example, if you have two
     floppy drives, they are generally units 0 and 1, respectively.

     A single hard drive has only one unit number, even if it has
     multiple partitions.
```

```
     You can determine which UNIT number corresponds to a volume by
     running the HyperCacheStats program, selecting a volume name,
     and examining the value of the UNIT display in that program.

See Also:

                 NO_UNIT_CHECK

                 DEVICE
```

1.6 HyperCache.guide/NO_UNIT_CHECK

```
                NO_UNIT_CHECK
-------------

Format:
     NO_UNIT_CHECK

Default:
     Off (units are checked)

Example:
     NO_UNIT_CHECK

Description:
     This parameter disables the check which HyperCache makes to
     determine which UNIT a read or write operation was intended for
     when it is received at the controller (or "Device").  This can
     provide a slight increase in performance.

     ** NOTE **

     Using this parameter when there is more than one UNIT connected
     to a DEVICE can cause data loss.

     IF YOU ARE UNSURE, DO NOT USE THIS PARAMTER.


See Also:

                 UNIT
```

1.7 HyperCache.guide/NUM_LINES

```
                NUM_LINES
---------

Format:
     NUM_LINES=<number of lines to use>
```

```
Default:
     256

Example:
     NUM_LINES=256

Description:
     The number of lines is your primary control over the size of
     the cache.  Adjusting this parameter up or down increases or
     decreases your cache size.  For example, doubling its value
     doubles the size of the cache.

     The total cache size can be calculated by multiplying the
     number of lines by line size, and multiplying that value by
     the block size of your device (typically, 512 bytes).

     For example, a cache with 400 lines and line size of 4 would
     be:

     400 * 4 * 512 = 819,200 bytes

     Due to internal overhead, your total memory consumption will be
     slightly higher than this figure.

See Also:

                 LINE_SIZE
```

1.8 HyperCache.guide/LINE_SIZE

```
                LINE_SIZE
---------

Format:
     LINE_SIZE=<size of each line in blocks>

Default:
     4

Example:
     LINE_SIZE=4

Description:
     The LINE_SIZE parameter controls the amount of data which
     the cache prefetches from the device on any particulay read
     operation.

     The default value of 4 is typically optimal.  If your drive
     is very defragmented, you may wish to experiment with a value
     of 8.

     Doubling the LINE_SIZE will also double the size of the cache.
     Therefore, you may have to make a corresponding change in the
     number of lines if you wish to maintain the same cache size
```

```
     after adjusting this paramter.

     Your primary control over the cache size should be NUM_LINES,
     and not the LINE_SIZE.

     NOTE: The LINE_SIZE must be a power of 2.  ie: 4, 8, 16, etc.

See Also:

                 NUM_LINES
```

1.9 HyperCache.guide/WRITE_RETENTION_TIME

```
                WRITE_RETENTION_TIME
--------------------

Format:
     WRITE_RETENTION_TIME=<maximum time (in seconds) to keep changed
     data>

Default:
     300

Example:
     WRITE_RETENTION_TIME=60

Description:
     This controls the MAXIMUM amount of time (in seconds) that data
     which has been changed can be held in the cache.  To disable write
     retention set this parameter to 0.

     *Warning:* If you want to reset your computer you must wait the
     smaller of WRITE_RETENTION_TIME or QUIET_WRITE_TIME (in the case
     that no IO is occurring, plus a few more seconds) *BEFORE* you can
     safely reset your computer.  Note that if you try to suspend
     HyperCache it may take WRITE_RETENTION_TIME before HyperCache
     responds.

See Also:

                 QUIET_WRITE_TIME
```

1.10 HyperCache.guide/QUIET_WRITE_TIME

```
QUIET_WRITE_TIME
----------------

Format:
     QUIET_WRITE_TIME=<time to flush cache in if drive quiet>
```

```
Default:
     1

Example:
     QUIET_WRITE_TIME=1

Description:
     This parameter only takes effect when write retention is enabled
     and there is data in the cache that has been changed.  If this is
     the case, then HyperCache must at some time write the changed data
     back to the disk.  During disk activity there are normally periods
     of inactivity while the disk is free for use.  The QUIET_WRITE_TIME
     determines how long HyperCache waits to declare that the disk
     is idle, and to begin writing.

     NOTE: Filing systems, like the FastFileSystem, often continue to
     present IO requests after you finish an activity for a further 1-2
     seconds.  Hence, a QUIET_WRITE_TIME of 1, will appear to wait
     three seconds before writing out a request during a quiet spell.
     This is normal and correct behavior.

     *Warning:* If you want to reset your computer you must wait the
     smaller of WRITE_RETENTION_TIME or QUIET_WRITE_TIME (in the case
     that no IO is occurring, plus a few more seconds) BEFORE you can
     safely reset your computer.
```

1.11 HyperCache.guide/DONOTWAIT

```
DONOTWAIT
---------

Format:
     DONOTAIT

Default:
     -

Example:
     DONOTWAIT

Description:
     This is actually a WorkBench parameter, and tells WorkBench not to
     wait for the program to finish.  This is recommended for WorkBench
     starts.

     Refer to your Amiga System Software documentation for full details
     on this parameter.
```

1.12 HyperCache.guide/TOOLPRI

```
TOOLPRI
-------

Format:
     TOOLPRI=<tool startup priority>

Default:
     0

Example:
     TOOLPRI=1

Description:
     This is actually a WorkBench parameter, and tells WorkBench the
     startup priority of this tool, when it is placed in the WBStartup
     drawer.

     Refer to your Amiga System Software documentation for full details
     on this parameter.
```

1.13 HyperCache.guide/?

```
?
-

Format:
     ?

Default:
     -

Example:
     ?

Description
     This is a CLI only parameter, and displays help on using
     HyperCache.
```

1.14 HyperCache.guide/ARexx

```
                ARexx
*****

    HyperCache V2.0 has an ARexx port that enables you to control the
cache, as well as get statiscal information about how it is performing.
The name of the ARexx port that is created consists of "HyperCache.",
followed by the device name, followed by a ".", followed by the unit
number.
```

```
    For example, for the device 'csi.device', unit 0, the ARexx
port name would be 'HyperCache.scsi.device.0'.  For the 'trackdisk.device',
unit 2, the ARexx port name would be 'HyperCache.trackdisk.device.2'.


                ACTIVATE
                              Tell HyperCache to cache data

                SUSPEND
                               Tell HyperCache not to cache data

                ALTERNATE
                             Alternate between suspended and activated

                FLUSH
                                 Write out any data changed in cache

                STATS
                                 Get statistical information about cache
```

1.15 HyperCache.guide/ACTIVATE

```
                ACTIVATE
--------

Format:
     ACTIVATE

Example:
     Address 'HyperCache.trackdisk.device.0'

     ACTIVATE

Description:
     This tells HyperCache to start caching data.  This command is most
     useful after doing a SUSPEND or ALTERNATE.  When HyperCache first
     starts up, it is in this mode.  If the cache was previously
     suspended when this command is received, then memory will be
     allocated for the cache again.

See also:

                SUSPEND

                ALTERNATE
```

1.16 HyperCache.guide/SUSPEND

```
                SUSPEND
-------

Format:
     SUSPEND

Example:
     Address 'HyperCache.trackdisk.device.0'

     SUSPEND

Description:
     This tells HyperCache to stop caching data.  This command is most
     useful after doing an ACTIVATE or ALTERNATE, or when you are about
     to do something that may cause memory corruption.  This command
     will also flush the cache, and free up most of the cache memory.

See also:

                 ACTIVATE

                 ALTERNATE

                 FLUSH
```

1.17 HyperCache.guide/ALTERNATE

```
                ALTERNATE
---------

Format:
     ALTERNATE

Example:
     Address 'HyperCache.trackdisk.device.0'

     ALTERNATE

Description:
     If HyperCache was suspended, if will become active.  If HyperCache
     was active it will become suspended.  If the cache is suspended
     then all data will be flushed first.

See also:

                 ACTIVATE

                 SUSPEND

                 FLUSH
```

1.18 HyperCache.guide/FLUSH

```
                FLUSH
-----

Format:
     FLUSH

Example:
     Address 'HyperCache.trackdisk.device.0'

     FLUSH

Description:
     Any data that has been written to the cache, but is still waiting
     to be written out to the device will be forced out with this
     command.  Note that the command may return before the data has
     actually been flushed (it only registers a request to flush the
     data).  To make certain that the data has been flushed, execute
     this command twice (the second call will be delayed until the
     first has been processed).

See also:

                 SUSPEND
```

1.19 HyperCache.guide/STATS

```
                STATS
-----

Format:
     STATS

Example:
     Address 'HyperCache.trackdisk.device.0'

     STATS statistics=result

     parse var statistics ReadHits ReadMisses WriteHits WriteMisses

     TotalReads=ReadHits+ReadMisses

     TotalWrites=WriteHits+WriteMisses

Description:
     This command asks HyperCache how many read hits, read misses,
     write hits and write misses have occurred since the last time the
     STATS command was received.

See also:

                 SUSPEND
```

1.20 HyperCache.guide/HyperCacheStats

HyperCacheStats
****

HyperCacheStats is a utility that uses the HyperCache ARexx port to display cache statistics.

When started from Workbench, it takes one parameter in the ToolTypes field of its icon, 'PORT'. This parameter should be set to the name of the ARexx port that HyperCache is using (refer to the ARexx section). For example, for gvpscsi.device, unit 0:

```
PORT=HyperCache.gvpscsi.device.0
```

When run from CLI, a parameter of '-PORT' should be specified, and set equal to the ARexx port that HyperCache is using. For example, for gvpscsi.device, unit 0:

```
HyperCacheStats -PORT=HyperCache.gvpscsi.device.0
```

When started, 18 numeric panels will be displayed and all are reset back to 0. Please refer to the ARexx command STATS for more of an explanation of how some of the numbers are generated.

1. "Total Read Hits" gives the number of bytes that have been supplied from the cache during read requests.

2. "Total Read Misses" gives the number of bytes that were not supplied from the cache (had to be obtained from the drive) during read requests.

3. "Total Read Samples" says how many times a sample by HyperCacheStats resulted in information. HyperCacheStats queries HyperCache every second to see if there has been any activity. If a read request has been presented, then it is counted as a sample. This is done because some of the statistics are not defined when no activity has occurred.

4. "Total Write Hits" gives the number of bytes that were written to the cache, and did not have to be written to disk immediately.

5. "Total Write Misses" gives the number of bytes that had to be written back to disk immediately.

6. "Total Write Samples" indicates how many times a sample by HyperCacheStats resulted in information. HyperCacheStats queries HyperCache every second to see if there has been any activity. If a write request has been presented, then it is counted as a sample. This is done because some of the statistics are not defined when no activity has occurred.

7. "Instantaneous Read Hits" gives the number of bytes that were

supplied to a read request during the last sample taken by HyperCacheStats. "Instantaneous Read Misses" gives the number of bytes that had to be read from disk during the last sample.

8. "Instantaneous Read Misses" is the number of bytes that had to be fetched from disk for a read request during the last sample.

9. "Instantaneous Read Hit Ratio" is the ratio of instantaneous read hits to the total instantaneous read operations. The ideal value is 1.000, and the worst is 0.000. Note that if no operations occur, then this value will be held until one does occur, and a new sample takes place.

10. "Average Read Hits" is the average number of bytes supplied from the cache to meet a read request per sample.

11. "Average Read Misses" is the average number of bytes that had to be fetched from disk to meet a read request per sample.

12. "Average Read Hit Ratio" is the ratio of total read hits to total reads. The ideal value is 1.000, and the worst is 0.000.

13. "Instantaneous Write Hits" is the number of bytes that were supplied from cache for a read request during the last sample.

14. "Instantaneous Write Misses" is the number of bytes were unable to be stored directly in cache during a write request during the last sample.

15. "Instantaneous Write Hit Ratio" is the ratio of instantaneous write hits to the total instantaneous write operations. The ideal value is 1.000, and the worst is 0.000. Note that if no operations occur, then this value will be held until one does occur, and a new sample takes place.

16. "Average Write Hits" is the average number of bytes supplied from the cache to meet a write request per sample.

17. "Average Write Misses" is the average number of bytes that were not able to be stored directly in cache during a write request per sample.

18. "Average Write Hit Ratio" is the ratio of total write hits to total writes. The ideal value is 1.000, and the worst is 0.000.

1.21 HyperCache.guide/Index

```
                Index
*****


                 ?
                  ?
```

```
ACTIVATE
                          ACTIVATE

ALTERNATE
                        ALTERNATE

ARexx
                            ARexx

ARexx, ACTIVATE
                   ACTIVATE

ARexx, ALTERNATE
                 ALTERNATE

ARexx, FLUSH
                     FLUSH

ARexx, STATS
                     STATS

ARexx, SUSPEND
                   SUSPEND

Cache Size
                        LINE_SIZE

Cache Size
                        NUM_LINES

DEVICE
                            DEVICE

DONOTWAIT
                         DONOTWAIT

FLUSH
                            FLUSH

HyperCacheStats
                  HyperCacheStats

LINE_SIZE
                         LINE_SIZE

NO_UNIT_CHECK
                    NO_UNIT_CHECK

NUM_LINES
                         NUM_LINES

Parameters
                        Parameters

Parameters, ?
                     ?
```

```
Parameters, DEVICE
               DEVICE

Parameters, DONOTWAIT
            DONOTWAIT

Parameters, LINE_SIZE
            LINE_SIZE

Parameters, NO_UNIT_CHECK
        NO_UNIT_CHECK

Parameters, NUM_LINES
            NUM_LINES

Parameters, QUIET_WRITE_TIME
     QUIET_WRITE_TIME

Parameters, TOOLPRI
              TOOLPRI

Parameters, UNIT
                 UNIT

Parameters, WRITE_RETENTION_TIME
 WRITE_RETENTION_TIME

QUIET_WRITE_TIME
                 QUIET_WRITE_TIME

STATS
                             STATS

SUSPEND
                           SUSPEND

TOOLPRI
                           TOOLPRI

UNIT
                              UNIT

WRITE_RETENTION_TIME
             WRITE_RETENTION_TIME
```