

NAME

stats – application statistics gathering

SYNOPSIS

```
#include <ShoreApp.h>
#include <ShoreStats.h>
class w_statistics_t; // forward;
// defined in <w_statistics.h>, included by <ShoreApp.h>

static shrc Shore::gather_stats(w_statistics_t &, bool remote=false);
void Shore::clear_stats();
```

DESCRIPTION

These methods are the interface to a statistics-gathering facility that application programmers may find useful for debugging or performance-tuning purposes. The statistics are gathered with **gather_stats**. Each statistic kept has a name and a manifest constant of that name for use with the generic statistics-gathering programming interface described in **statistics(fc)**. The names and descriptions of the statistics are listed in sections below.

Gather_stats fills in the *w_statistics_t* structure with the current local or remote statistics. Local and remote statistics are maintained at all the levels of the Shore software. There is no single data structure that contains all the statistics at all the layers. The *w_statistics_t* class is used to hold the statistics in a generic form, which, for each statistic, includes a description of the statistics (a character string), a value for the statistic (integer, unsigned integer, or floating point number), and a manifest constant for identifying the statistic. Details about programming with the *w_statistics_t* class are in **statistics(fc)**.

Clear_stats resets the statistics to zero. If the Object cache option **oc_pstats** has the value **true** or **yes**, the statistics are gathered, printed, and cleared each time a transaction is committed or aborted.

Shore::init (see **init(oc)**) initializes the local statistics to zero.

OBJECT CACHE STATISTICS

The Object Cache maintains the following statistics, identified by their manifest constants (which are defined in <ShoreStats.h>).

OCSTAT_anoncreates
the number of anonymous objects created

OCSTAT_regcreates
the number of registered objects created

OCSTAT_otentries
The number of objects about which the the object cache has any information.

OCSTAT_ncached
The number of objects fetched into the object cache. If pre-fetching is in effect, this can be more than the number of objects actually used by the application.

OCSTAT_nbytes
the number of bytes fetched into the object cache.

OCSTAT_nupdates
The number of times a reference to an object was made writable so that a non-const method could be called.

OCSTAT_ndestroys
The number of objects destroyed.

OCSTAT_total_pins
The number of times objects were pinned (a reference to an object was followed).

OCSTAT_total_unpins

The number of times objects were unpinned (a reference to an object went out of scope).

OCSTAT_mem_limit

The memory limit imposed on the object cache by the user through options.

OCSTAT_mem_used

The amount of memory used for caching objects (does not include the memory used for managing the object cache) at the time the statistics are gathered.

OCSTAT_obj_used

The number of objects cached at the time the statistics are gathered.

OCSTAT_extern_mem

The amount of memory used for caching objects that wouldn't fit in the cache (whose size is determined by OCSTAT_mem_limit) after replacing objects. This number does not include the memory used for managing the object cache.

OCSTAT_extern_obj

The number of objects that would not fit in the cache (whose size is determined by OCSTAT_mem_limit) after replacing objects.

The Shore Value-Added Server statistics are also gathered. They are described in **statistics(svas)**. Unix resource-usage statistics are also gathered. They are describe in **getrusage(2)**.

If the flag *remote* is false on a call to **gather_stats**, only the statistics for the client process are gathered. If the value of *remote* is true, statistics are gathered from the server instead. Remote statistics compose a collection of statistics gathered from 13 different software modules within the Shore server. They are not described in detail anywhere, but many of the statistics are self-describing when printed.

VERSION

This manual page applies to Version 1.1 of the Shore software.

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.

COPYRIGHT

Copyright © 1994, 1995, 1996, 1997, Computer Sciences Department, University of WisconsinMadison. All Rights Reserved.

SEE ALSO

statistics(svas), **statistics(fc)**, and **options(oc)**.