NAME

page – inspect a cached page for objects

SYNOPSIS

```
VASResult shore_vas::num_cached_oids(
            int         *count
        );

VASResult shore_vas::cached_oids(
            int         *count,
                  lrid_t            *list
        );
```

DESCRIPTION

The purpose of these functions is to allow the caller to discover what objects were cached by the SVAS as a side effect of reading an anonymous object, or as a side effect of getting the system properties of an anonymous object. The caller can subsequently read these objects before they are thrown out of the SVAS's cache. The functions **readObj(svas)** and **sysprops(svas)** have forms that indicate *whether* a page of small anonymous objects is cached as a side effect. The functions described here allow the caller to learn the identities of the objects on the page.

Num_cached_oids returns the number of slots on the most-recently-cached page. This number may be larger than the number of small objects on the page. It is meant to be used to allocate space in which to hold the OIDs.

Cached_oids returns a list of the OIDs of the small objects on the most-recently-cached page. It also returns the number of OIDs returned.

ARGUMENTS

The argument *list* is a buffer in which to write the OIDs of the objects on the page. The buffer must be allocated and freed by the caller. The buffer must be large enough to accommodate the OIDs of all the small objects on the page.

The argument *count* is both an input argument and an output argument. When **cached_oids** is called, *count* must indicate the number of entries in the buffer.

ENVIRONMENT

These functions are available on both the server and clients, but since the server does not cache small-object pages the way the client does, these functions always return with `count == 0` on the server. Sometime in the future, these functions will be made more useful on the server.

SEE ALSO

sysprops(svas), and **readObj(svas).**