NAME

updateObj – combined write and append or trunc

SYNOPSIS

```
        // truncate followed by write:
        VASResult shore_vas::updateObj(
                    const  lrid_t     &obj,
                    ObjectOffset      offset,
                    const  vec_t      &wdata,
                    ObjectSize        newlen,
                    ObjectOffset      newtstart
                );

        // append followed by write:
        VASResult shore_vas::updateObj(
                    const  lrid_t     &obj,
                    ObjectOffset      offset,
                    const  vec_t      &wdata,
                    ObjectOffset      aoffset,
                    const  vec_t      &adata,
                    ObjectOffset      newtstart
                );
```

DESCRIPTION

UpdateObj combines two functions: **truncObj** and **writeObj** (in its first form), or **appendObj** and **writeObj** (in its second form). The purpose of combining these pairs of functions is to reduce the number of remote operations performed when these combinations of operations are apropos.

ARGUMENTS

The argument *obj* is the full logical object identifier of the object to be read or updated.

In the first form, the argument *newlen* indicates the desired length of the object after the truncation occurs. If this is larger than the size of the object before truncation, the SVAS pads the object with zero-valued bytes up to the desired length.

In the second form, the size of the object is increased by appending the data given in *adata.* This form of requires the value *aoffset* to indicate the size of the object when the function is called. The reason for this is that the combined vectors *adata* and *wdata* may exceed internal limits on the size of a remote request. In that case, the append portion of this request is accomplished by a single append request with uninitialized data, along with a set of smaller write requests.

The vectors may be empty.

UpdateObj can change the location and size of the TEXT portion of the object. The argument *newtstart* indicates the new location of the beginning of the TEXT attribute.

An exclusive lock is acquired by **updateObj.**

ENVIRONMENT

All these methods are available to both the server and to clients. The argument *aoffset* is ignored on the server.

Both forms of **updateObj** must be called when a transaction is active.

ERRORS

Deadlocks can occur while locks are being acquired. See **transaction(svas)** for information about deadlocks.

A complete list of errors is in **errors(svas).**

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

SEE ALSO

appendObj(svas), truncObj(svas), writeObj(svas), transaction(svas), errors(svas), and **text(svas).**