NAME

mkVolRef, offVolRef, snapRef – Shore logical object identifiers

SYNOPSIS

```
VASResult shore_vas::mkVolRef(
            const  lvid_t     &onvol,
            lrid_t            *result,
                  int                          number = 1
          );

VASResult shore_vas::offVolRef(
            const  lvid_t     &onvol,
            const  lrid_t     &toobj,
            lrid_t            *result
          );

VASResult shore_vas::snapRef(
            const  lrid_t     &off,
            lrid_t            *result
          );
```

DESCRIPTION

MkVolRef allocates one or more logical object identifiers that are not associated with any object.

OffVolRef allocates a logical object identifier on one volume, and associates it with a logical object identifier on another volume, thereby creating an *indirect reference.*

SnapRef analyzes a logical object identifier, and if it is part of a chain of indirect references, the last logical object identifier in the chain is returned. (The true end of the chain is a physical object identifier. See **oid(shore)** for more details.)

ARGUMENTS

The argument *onvol* indicates the volume on which to allocate a new OID.

The argument *result* points to an area in the caller's address space into which the SVAS will write the resulting OID. *Result* may *not* be *null.* The optional argument *number* to **mkVolRef** indicates the number of serial numbers (hence, OIDs) to allocate. If it is larger than 1, the SSM allocates a series of serial numbers and the first one in the series is returned in **result.*

The argument *toobj* to **offVolRef** is an OID with which the new OID will be associated. There need not be an object associated with *toobj*, but *toobj* must a legitimate OID in form.

SnapRef's argument *off* is an object identifier that might be part of a chain of indirect references. If it is, the SSM follows the chain until it reaches an OID that is associated with a physical object identifier or is not associated with anything; that last OID is the one returned.

ERRORS

ENVIRONMENT

All these methods require that a transaction be active when they are called.

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

oid(shore), errors(svas), and **transaction(svas).**