

**StorageTek Linear Tape File System, Library  
Edition**

Web Services Application Programming Interface Guide

Release 1.0

**E55756-02**

May 2015

StorageTek Linear Tape File System, Library Edition Web Services Application Programming Interface Guide, Release 1.0

E55756-02

Copyright © 2014, 2015, Oracle and/or its affiliates. All rights reserved.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

---

---

# Contents

<b>Preface</b> .....	ix
Audience .....	ix
Documentation Accessibility .....	ix
Related Documents .....	ix
Conventions .....	ix
<b>1 Overview</b>	
Limitations .....	1-1
<b>2 Using the API</b>	
Accessing the Web Services Description Language File .....	2-1
API Functions .....	2-1
Invoking API Functions .....	2-2
Signatures .....	2-2
General Notes .....	2-2
<b>3 Status Services</b>	
General Status Services .....	3-1
getJobResult .....	3-1
Format .....	3-1
Parameters .....	3-1
Roles .....	3-1
Exceptions Thrown .....	3-1
Recommended Timeout .....	3-2
SOAP Request .....	3-2
SOAP Response .....	3-2
SOAP Fault .....	3-2
getJobStatus .....	3-3
Format .....	3-3
Parameters .....	3-3
Roles .....	3-3
Exceptions Thrown .....	3-3
Recommended Timeout .....	3-3
SOAP Request .....	3-3
SOAP Response .....	3-4

SOAP Fault.....	3-4
<b>Volume and Drive Status Services .....</b>	<b>3-4</b>
getAllDriveInfo.....	3-4
Format.....	3-4
Input Type.....	3-4
Response.....	3-4
Roles.....	3-4
Exceptions Thrown .....	3-4
Recommended Timeout.....	3-5
SOAP Request.....	3-5
SOAP Response.....	3-5
SOAP Fault.....	3-5
getAllVolumeInfo.....	3-5
Format.....	3-6
Response.....	3-6
Roles.....	3-6
Exceptions Thrown .....	3-6
Recommended Timeout.....	3-6
SOAP Request.....	3-6
SOAP Response.....	3-6
SOAP Fault.....	3-7
getVolumeInfoByStatus .....	3-7
Format.....	3-7
Input Type.....	3-7
Response.....	3-7
Roles.....	3-7
Exceptions Thrown .....	3-7
Recommended Timeout.....	3-8
SOAP Request.....	3-8
SOAP Response.....	3-8
SOAP Fault.....	3-9
getVolumeInfoByVolser .....	3-9
Format.....	3-9
Parameter .....	3-9
Roles.....	3-9
Exceptions Thrown .....	3-9
Recommended Timeout.....	3-9
SOAP Request.....	3-9
SOAP Response.....	3-10
SOAP Fault.....	3-10

## 4 Management Services

<b>Overview .....</b>	<b>4-1</b>
assignVolume .....	4-1
Format.....	4-2
Parameters.....	4-2
Response.....	4-2

Roles .....	4-2
Exception Thrown.....	4-2
Recommended Timeout.....	4-3
Recommended Job Poll Interval .....	4-3
Job Error .....	4-3
SOAP Request.....	4-3
SOAP Response .....	4-4
SOAP Fault.....	4-4
ejectVolume.....	4-4
Format.....	4-4
Parameters.....	4-5
Response.....	4-5
Roles .....	4-5
Exception Thrown.....	4-5
Recommended Timeout.....	4-6
Recommended Job Poll Interval .....	4-6
Job Error .....	4-6
SOAP Request.....	4-6
SOAP Response .....	4-6
SOAP Fault.....	4-7
enterVolume.....	4-7
Format.....	4-7
Parameters.....	4-7
Response.....	4-8
Roles .....	4-8
Exception Thrown.....	4-8
Recommended Timeout.....	4-8
Recommended Job Poll Interval .....	4-8
Job Error .....	4-8
SOAP Request.....	4-8
SOAP Response .....	4-9
SOAP Fault.....	4-9
finalizeVolume.....	4-9
Format.....	4-9
Response.....	4-10
Parameters.....	4-10
Roles .....	4-10
Exception Thrown.....	4-10
Recommended Timeout.....	4-11
Recommended Job Poll Interval .....	4-11
Job Errors.....	4-11
SOAP Request.....	4-11
SOAP Response .....	4-11
SOAP Fault.....	4-12
formatVolume.....	4-12
Format.....	4-12
Parameters.....	4-13

Response.....	4-13
Roles.....	4-13
Exception Thrown.....	4-13
Recommended Timeout.....	4-13
Recommended Job Poll Interval .....	4-14
Job Errors.....	4-14
SOAP Request.....	4-14
SOAP Response.....	4-14
SOAP Fault.....	4-14
formatVolumeWithDrive.....	4-15
Format.....	4-15
Parameters.....	4-15
Response.....	4-15
Roles.....	4-16
Exception Thrown.....	4-16
Recommended Timeout.....	4-17
Recommended Job Poll Interval .....	4-17
Job Errors.....	4-17
SOAP Request.....	4-17
SOAP Response.....	4-17
SOAP Fault.....	4-18
getVolumePath .....	4-18
Format.....	4-18
Parameters.....	4-18
Response.....	4-18
Roles.....	4-18
Exception Thrown.....	4-18
Recommended Timeout.....	4-19
Recommended Job Poll Interval .....	4-19
Job Errors.....	4-19
SOAP Request.....	4-19
SOAP Response.....	4-19
SOAP Fault.....	4-20
offlineVolume .....	4-20
Format.....	4-20
Parameters.....	4-21
Response.....	4-21
Roles.....	4-21
Exception Thrown.....	4-21
Recommended Timeout.....	4-22
Recommended Job Poll Interval .....	4-22
Job Errors.....	4-22
SOAP Request.....	4-22
SOAP Response.....	4-22
SOAP Fault.....	4-23
onlineVolume .....	4-23
Format.....	4-23

Parameters.....	4-23
Response.....	4-24
Roles.....	4-24
Exception Thrown.....	4-24
Recommended Timeout.....	4-24
Recommended Job Poll Interval.....	4-24
Job Error.....	4-24
SOAP Request.....	4-25
SOAP Response.....	4-25
SOAP Fault.....	4-25
prepareVolume.....	4-26
Format.....	4-26
Parameters.....	4-26
Response.....	4-26
Roles.....	4-26
Exception Thrown.....	4-26
Recommended Timeout.....	4-27
Recommended Job Poll Interval.....	4-27
Job Error.....	4-27
SOAP Request.....	4-27
SOAP Response.....	4-28
SOAP Fault.....	4-28
unassignVolume.....	4-28
Format.....	4-28
Parameters.....	4-29
Response.....	4-29
Roles.....	4-29
Exception Thrown.....	4-29
Recommended Timeout.....	4-30
Recommended Job Poll Interval.....	4-30
Job Error.....	4-30
SOAP Request.....	4-30
SOAP Response.....	4-30
SOAP Fault.....	4-31
unformatVolume.....	4-31
Format.....	4-32
Parameters.....	4-32
Response.....	4-32
Roles.....	4-32
Exception Thrown.....	4-32
Recommended Timeout.....	4-33
Recommended Job Poll Interval.....	4-33
Job Error.....	4-33
SOAP Request.....	4-33
SOAP Response.....	4-33
SOAP Fault.....	4-33
unformatVolumeWithDrive.....	4-34

Format.....	4-34
Parameters.....	4-34
Response.....	4-35
Exception Thrown.....	4-35
Roles.....	4-36
Recommended Timeout.....	4-36
Recommended Job Poll Interval .....	4-36
Job Errors.....	4-36
SOAP Request.....	4-36
SOAP Response.....	4-36
SOAP Fault.....	4-37
vaultVolume .....	4-37
Format.....	4-37
Parameters.....	4-38
Response.....	4-38
Roles.....	4-38
Exception Thrown.....	4-38
Recommended Timeout.....	4-39
Recommended Job Poll Interval .....	4-39
Job Errors.....	4-39
SOAP Request.....	4-39
SOAP Response.....	4-40
SOAP Fault.....	4-40

## Glossary

## Index



---

---

# Preface

This programmer's guide describes the programmatic approach to integrating the Web Services Application Programming Interface (API) with third-party products. Developers are encouraged to refer to the LTFS-LE 1.0.4 Web Services API signatures and return codes for each function as discussed in "Status Services" on page 3-1 and "Management Services" on page 4-1.

## Audience

This manual is intended to be used by developers responsible for implementing the LTFS-LE 1.0.4 API.

## Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>.

### Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> or visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> if you are hearing impaired.

## Related Documents

For more information, see the following documents in the LTFS-LE documentation set:

- *LTFS-LE 1.0 Planning and Installation Guide*
- *LTFS-LE 1.0 Administration Guide*
- *LTFS-LE Security Guide*
- *LTFS-LE Third-Party Licenses and Notices*

## Conventions

The following text conventions are used in this document:

<b>Convention</b>	<b>Meaning</b>
<b>boldface</b>	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
<i>italic</i>	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
monospace	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

---

---

## Overview

The LTFS LE Web Services are the key part of the integration strategy with third-party applications, and create more opportunities for an integrated solution with both internal and external products.

The availability of the LTFS-LE 1.0.4 Web Services API provides critical BUI functions to allow developers of third-party products to create a seamless integrated user experience. In this approach, the developers embed client LTFS-LE Web Services into their BUI, allowing the customers to interact with a single BUI for the integrated solution.

This programmer's guide describes the programmatic approach to integrating the Web Services API with third-party products. Developers are encouraged to refer to the LTFS-LE Web Services API signatures and return codes for each function as discussed in "[Status Services](#)" on page 3-1 and "[Management Services](#)" on page 4-1.

### Limitations

Using Web Services and the LTFS-LE BUI at the same time is not supported. Doing so leads to undefined behavior. The recommended method of operations is to either use the BUI to perform the required operations or to use the Web Services to perform the operation.

Only one Web services client is supported. Web Services and the integration must target volume management capabilities using a single-threaded client.



---

---

## Using the API

The web services discussed in this document are only available in the LTFS-LE 1.0.4 release of the software.

### Accessing the Web Services Description Language File

Programmers can obtain the supported signature of the Web Service available from the installed version of LTFS-LE software from the Web Services Description Language 1.1 (WSDL) file. This file can be accessed from the following location:

```
https://server.mydomain.com:7002/LTFS/LTFSLEWS?WSDL
```

In the above example, *server.mydomain.com* is the hostname and domain name.

For some programming instances, you may need the following port name:

```
port name="LTFSLEWSSoap12HttpPort"
```

### API Functions

The back-end execution times of the LTFS LE Web Services vary. Web Services that pertain to the status of the volume or operation can be returned quickly. However, the execution time of some Web Services depends on several factors out of control of the software. Mechanical factors such as user actions cannot be predicted by software. For example, certain Web Services such as the Assign Volume service may take longer if the volume contains a lot files. Using a different example, the completion of the eject operation depends on the operator walking up to the library and removing the volume from the CAP.

In order to address these discrepancies, and yet provide a consistent programmatic interface across all the Web Services calls, the API service call returns immediately. The returned results will depend on the type of service. The LTFS LE has two categories of Web Services:

- Status services

Status services are the services that enables users to get the volume and drive status. For more information, see "[Status Services](#)" on page 3-1.

- Management services

Management services are the services that manage the hardware resources. For example, use a management service to enter a volume, eject a volume, and bring a volume online or offline. For more information, see "[Management Services](#)" on page 4-1.

## Invoking API Functions

The management services use a two-step process to invoke an API function and return results:

1. Invoke the function. A unique request job ID is immediately returned.
2. Invoke the `getJobStatus` function, specifying the job ID returned in step 1.

This returns a one of the following job statuses:

- 0 (Web service call associated with job ID is complete.)
- 1 (Web service call associated with job ID is still in progress.)
- -1 (No such web service call with job ID is found in the system.)

## Signatures

The LTFS-LE Web Services signatures include each SOAP request message format, SOAP response message format and SOAP fault message format. All LTFS LE Web Services only support SOAP messages. The SOAP message format implements SOAP 1.2 specification. Refer to "[Status Services](#)" on page 3-1 and "[Management Services](#)" on page 4-1 for the valid SOAP signature for each function.

## General Notes

- The Web Services outlined in this document are only available in the LTFS-LE 1.0.4 release of the software.
- All Web Service APIs are only accessible through HTTPS.
- Using WS-Security, all Web Service API SOAP request messages include a username token in the SOAP header in which the username and password are all in plain text
- The `username` and `password` is used in username token as an example only. It must be replaced by a valid username and password
- All the data used in the SOAP request and response message are example data only to show the SOAP request and response message data structure.

---

---

## Status Services

Status services enable users to get drive and volume statuses. For example, use a status service to get all volume information, get volume status, or get drive online or offline statuses. These services can obtain the service result almost instantaneously by querying the database, so the status services are implemented as a synchronous call. The completion of the operation returns the result of the operation.

### General Status Services

The following two Web Services are the general status Web Services for retrieving the job status and job result of each asynchronous management service execution.

- `getJobResult(int jobId)`
- `getJobStatus(int jobId)`

The `getjobstatus` and `getJobResult` Web Services are an example of pairing up asynchronous (management) services with synchronous (status) services to obtain the results of an execution.

### getJobResult

This synchronous call retrieves the job result for the given job ID entered.

#### Format

```
getJobResult(int jobId)
```

*LTFS-LE Web Service to call:*

```
getJobResult(jobId)
```

#### Parameters

*jobId*

The job ID returned by management and file Webservice call to indicate the running web service job.

#### Roles

Admin, Monitor, or Service

#### Exceptions Thrown

`ValidationFailedException` and `ResultNotFoundException`

- If the jobID is 0 or an invalid value, the exception thrown is: Invalid jobID.

ResultNotFoundException

- If the job result cannot be found, the exception thrown is: No job result is found.

Example: The JobResult cannot be found with the jobId1234.

## Recommended Timeout

600 seconds

## SOAP Request

```
<?xml version = '1.0' encoding = 'UTF-8'?>
<env:Envelope xmlns:env="http://www.w3.org/2003/05/soap-envelope" >
xmlns:ns1="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd"
xmlns:ns2="http://sb.ejb.ws.mds.ltfsle.oracle/">
  <env:Header>
    <ns1:Security>
      <ns1:UsernameToken>
        <ns1:Username>username</ns1:Username>
        <ns1:Password>password</ns1:Password>
      </ns1:UsernameToken>
    </ns1:Security>
  </env:Header>
  <env:Body>
    <ns1:getJobResult>
      <arg0>1212</arg0>
    </ns2:getJobResult>
  </env:Body>
</env:Envelope>
```

## SOAP Response

```
<?xml version = '1.0' encoding = 'UTF-8'?>
<S:Envelope xmlns:S="http://www.w3.org/2003/05/soap-envelope">
  <S:Body>
    <ns2:getJobResultResponse xmlns:ns2="http://sb.ejb.ws.mds.ltfsle.oracle/">
      <return>
        <callerIp>127.0.0.1</callerIp>
        <jobCompletedOn>1399190120</jobCompletedOn>
        <jobId>607</jobId>
        <jobStatus>Success</jobStatus>
        <jobUser>Admin</jobUser>
        <serviceCalled>enterVolume</serviceCalled>
      </return>
    </ns2:getJobResultResponse>
  </S:Body>
</S:Envelope>
```

## SOAP Fault

```
<?xml version = '1.0' encoding = 'UTF-8'?>
<S:Envelope xmlns:S="http://www.w3.org/2003/05/soap-envelope">
  <S:Body>
    <S:Fault xmlns:ns4="http://schemas.xmlsoap.org/soap/envelope/"
      <S:Code>
        <S:Value>S:Receiver</S:Value>
      <S:Code>
      <S:Reason>
```



```

    <S:Text xml:lang="en">The JobResult cannot be found with the jobId 607</S:Text>
  <S:Reason>
  <S:Detail>
    <ns2:ResultNotFoundException xmlns:ns2="http://wsb.ejb.ws.mds.ltfsle.oracle/">
      <message>The JobResult cannot be found with the jobId 607</message>
    </ns2:ResultNotFoundException>
  <S:Detail>
  <S:Fault>
</S:Body>
</S:Envelope>

```

## getJobStatus

This synchronous call retrieves the job status of the given job ID. The return codes are:

- 0 (Web service call associated with job ID is complete.)
- 1 (Web service call associated with job ID is still in progress.)
- -1 (No such web service call with job ID is found in the system.)

### Format

```
getJobStatus(int jobId)
```

*LTFS-LE Web Service chain to call:*

```
getJobStatus(jobId)
```

### Parameters

*jobId*

The job id returned by a WebService call to indicate the running web service job.

### Roles

Admin, Monitor, or Service

### Exceptions Thrown

ValidationFailedException

If the jobId is 0 or an invalid value, the exception thrown is: Invalid jobId.

### Recommended Timeout

600 seconds

### SOAP Request

```

<?xml version = '1.0' encoding = 'UTF-8'?
<env:Envelope xmlns:env="http://www.w3.org/2003/05/soap-envelope"
xmlns:ns1="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" xmlns:ns2="http://sb.ejb.ws.mds.ltfsle.oracle/">
  <env:Header>
    <ns1:Security>
      <ns1:UsernameToken>
        <ns1:Username>username</ns1:Username>
        <ns1:Password>password</ns1:Password>
      </ns1:UsernameToken>
    </ns1:Security>
  </env:Header>

```

```
<env:Body>
  <ns2:getJobStatus>
    <arg0>1212</arg0>
  </ns2:getJobStatus>
</env:Body>
</env:Envelope>
```

### SOAP Response

```
<?xml version = '1.0' encoding = 'UTF-8'?>
<S:Envelope xmlns:S="http://www.w3.org/2003/05/soap-envelope">
  <S:Body>
    <ns2:getJobStatusResponse xmlns:ns2="http://sb.ejb.ws.mds.ltfsle.oracle/">
      <return>0</return>
    </ns2:getJobStatusResponse>
  </S:Body>
</S:Envelope>
```

### SOAP Fault

None

## Volume and Drive Status Services

These service signatures retrieve volume and drive statuses.

- `getAllDriveInfo()`
- `getAllVolumeInfo()`
- `getVolumeInfoByStatus (String status)`
- `getVolumeInfoByVolser (String[] volsers)`

### getAllDriveInfo

A synchronous call that retrieves all the library tape drive information.

#### Format

`getAllDriveInfo()`

#### Input Type

None

#### Response

Valid drive responses are shown in the SOAP Response example. Refer to the Glossary for definitions.

#### Roles

Admin, Monitor, or Service

#### Exceptions Thrown

None. If there are some errors in getting all drive info, the return value is an empty drive array.

## Recommended Timeout

600 seconds

## SOAP Request

```
<?xml version = '1.0' encoding = 'UTF-8'?>
<env:Envelope xmlns:env="http://www.w3.org/2003/05/soap-envelope" xmlns:ns1=
"http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd"
xmlns:ns2="http://sb.ejb.ws.mds.ltfsle.oracle/">
  <env:Header>
    <ns1:Security>
      <ns1:UsernameToken>
        <ns1:Username>username</ns1:Username>
        <ns1:Password>password</ns1:Password>
      </ns1:UsernameToken>
    </ns1:Security>
  </env:Header>
  <env:Body>
    <ns1:getAllDriveInfo/>
  </env:Body>
</env:Envelope>
```

## SOAP Response

```
<?xml version = '1.0' encoding = 'UTF-8'?>
<S:Envelope xmlns:S="http://www.w3.org/2003/05/soap-envelope">
  <S:Body>
    <ns2:getAllDriveInfoResponse xmlns:ns2="http://ws.ltfsle.oracle/">
      <return>
        <acslsPath>0 ,0 ,12 ,5</acslsPath>
        <availableStatus>Available</availableStatus>
        <driveCompress>Compression Enabled</driveCompress>
        <driveEncry>Encryption NO</driveEncry>
        <driveFwRev>RF53</driveFwRev>
        <driveGenDev>/dev/sg10</driveGenDev>
        <driveLtfsCap>LTFS Capable</driveLtfsCap>
        <drivePoolName>Default Drive Pool</drivePoolName>
        <driveProdId>T10000C</driveProdId>
        <driveSn>576001000421</driveSn>
        <driveStatus>Assigned</driveStatus>
        <driveStdDev>/dev/st3</driveStdDev>
        <driveType>T1C</driveType>
        <driveVendor>STK</driveVendor>
        <onlineStatus>Online</onlineStatus>
      </return>
    </ns2:getAllDriveInfoResponse>
  </S:Body>
</S:Envelope>
```

## SOAP Fault

None. The response will return empty drives if there are any errors during this WebService call.

## getAllVolumeInfo

This request function is a synchronous call that retrieves all the volume information in LTFS-LE.

**Format**`getAllVolumeInfo()`**Response**

Valid responses are shown in the SOAP Response example. Refer to the Glossary for definitions.

**Roles**

Admin, Monitor, or Service

**Exceptions Thrown**

None. If there are errors in getting all volume information, the return value is an empty volume array.

**Recommended Timeout**

600 seconds

**SOAP Request**

```
<?xml version = '1.0' encoding = 'UTF-8'?>
<env:Envelope xmlns:env="http://www.w3.org/2003/05/soap-envelope"
xmlns:ns1="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd"
xmlns:ns2="http://sb.ejb.ws.mds.ltfsle.oracle/">
  <env:Header>
    <ns1:Security>
      <ns1:UsernameToken>
        <ns1:Username>username</ns1:Username>
        <ns1:Password>password</ns1:Password>
      </ns1:UsernameToken>
    </ns1:Security>
  </env:Header>
  <env:Body>
    <ns2:getAllVolumeInfo/>
  </env:Body>
</env:Envelope>
```

**SOAP Response**

```
<?xml version = '1.0' encoding = 'UTF-8'?>
<S:Envelope xmlns:S="http://www.w3.org/2003/05/soap-envelope">
  <S:Body>
    <ns2:getAllVolumeInfoResponse xmlns:ns2="http://sb.ejb.ws.mds.ltfsle.oracle/">
      <return>
        <acslsPath>0, ,0 ,12 ,24 ,4</acslsPath>
        <assignedStatus>Assigned</assignedStatus>
        <availableStatus>Available</availableStatus>
        <capacityAvailable>1453410222080</capacityAvailable>
        <capacityUsed>24117248</capacityUsed>
        <currentLocation>home</currentLocation>
        <inuseStatus>In use</inuseStatus>
        <ltfsUUID>e255c63b-54c9-4fc7-a60b-e63d58ed0005</ltfsUUID>
        <onlineStatus>Online</onlineStatus>
        <volser>F50191</volser>
        <volumeCreatedOn>1403014692</volumeCreatedOn>
        <volumePoolName>Default Volume Pool</volumePoolName>
        <volumeType>LTO-1.5T</volumeType>
      </return>
    </ns2:getAllVolumeInfoResponse>
  </S:Body>
</S:Envelope>
```

```

<return>
  <acslsPath>0, ,0 ,12 ,23 ,5</acslsPath>
  <assignedStatus>Assigned</assignedStatus>
  <availableStatus>Available</availableStatus>
  <capacityAvailable>1453410222080</capacityAvailable>
  <capacityUsed>24117248</capacityUsed>
  <currentLocation>home</currentLocation>
  <inuseStatus>In use</inuseStatus>
  <ltfsUUID>e255c63b-54c9-4fc7-a60b-e63d58ed0005</ltfsUUID>
  <onlineStatus>Online</onlineStatus>
  <volser>F50192</volser>
  <volumeCreatedOn>1403014692</volumeCreatedOn>
  <volumePoolName>Default Volume Pool</volumePoolName>
  <volumeType>LTO-1.5T</volumeType>
</return>
<return>
</ns2:getAllVolumeInfoResponse>
</S:Body>
</S:Envelope>

```

### SOAP Fault

None. The response returns empty volumes if there are any errors during this Web Service call.

## getVolumeInfoByStatus

A synchronous call that retrieves volume information by volume status.

### Format

```
getVolumeInfoByStatus(String status)
```

*LTFS-LE Web Service chain to call:*

```
getVolumeInfoByStatus(status)
```

### Input Type

Valid volume statuses are: Online, Offline, Assigned, Unassigned, In Use, Idle, Available, or Unavailable.

It does not support the entering of multiple statuses, nor does it support the pending status.

### Response

Valid responses are shown in the SOAP Response example. Refer to the Glossary for definitions.

### Roles

Admin, Monitor, or Service

### Exceptions Thrown

StatusNotFoundException and VolumeNotFoundException

If a volume is not found by the given status, the message is *No volume found with the [status] status.*

## Recommended Timeout

600 seconds

## SOAP Request

```
<?xml version = '1.0' encoding = 'UTF-8'?>
<env:Envelope xmlns:env="http://www.w3.org/2003/05/soap-envelope" xmlns:ns1
="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" xmlns:ns2
="http://sb.ejb.ws.mds.ltfsle.oracle/">
  <env:Header>
    <ns1:Security>
      <ns1:UsernameToken>
        <ns1:Username>username</ns1:Username>
        <ns1:Password>password</ns1:Password>
      </ns1:UsernameToken>
    </ns1:Security>
  </env:Header>
  <env:Body>
    <ns2:getVolumeInfoByStatus>
      <arg0>online</arg0>
    </ns2:getVolumeInfoByStatus>
  </env:Body>
</env:Envelope>
```

## SOAP Response

```
<?xml version = '1.0' encoding = 'UTF-8'?>
<S:Envelope xmlns:S="http://www.w3.org/2003/05/soap-envelope">
  <S:Body>
    <ns2:getVolumeInfoByStatusResponse xmlns:ns2="http://ws.ltfsle.oracle/">
      <return>
        <acslsPath>3 ,0 ,12 ,38 ,1</acslsPath>
        <assignedStatus>Assigned</assignedStatus>
        <availableStatus>Available</availableStatus>
        <capacityAvailable>1453410222080</capacityAvailable>
        <capacityUsed>24117248</capacityUsed>
        <currentLocation>home</currentLocation>
        <inuseStatus>In use</inuseStatus>
        <ltfsUUID>e255c63b-54c9-4fc7-a60b-e63d58ed0005</ltfsUUID>
        <onlineStatus>Online</onlineStatus>
        <volser>050191</volser>
        <volumeCreatedOn>1402933089</volumeCreatedOn>
        <volumePoolName>Default Volume Pool</volumePoolName>
        <volumeType>LTO-1.5T</volumeType>
      </return>
    </return>
    <return>
      <acslsPath>3 ,0 ,12 ,41 ,2</acslsPath>
      <assignedStatus>Assigned</assignedStatus>
      <availableStatus>Available</availableStatus>
      <capacityAvailable>1453410222080</capacityAvailable>
      <capacityUsed>24117248</capacityUsed>
      <currentLocation>home</currentLocation>
      <inuseStatus>In use</inuseStatus>
      <ltfsUUID>e255c63b-54c9-4fc7-a60b-e63d58ed0005</ltfsUUID>
      <onlineStatus>Online</onlineStatus>
      <volser>050191</volser>
      <volumeCreatedOn>1402933089</volumeCreatedOn>
      <volumePoolName>Default Volume Pool</volumePoolName>
      <volumeType>LTO-1.5T</volumeType>
    </return>
  </S:Body>
</S:Envelope>
```

```

    </ns2:getVolumeInfoByStatusResponse>
  </S:Body>
</S:Envelope>

```

### SOAP Fault

```

<?xml version = '1.0' encoding = 'UTF-8'?>
<S:Envelope xmlns:S="http://www.w3.org/2003/05/soap-envelope">
  <S:Body>
    <ns3:Fault xmlns:ns2="http://schemas.xmlsoap.org/soap/envelope/" xmlns:ns3
="http://www.w3.org/2003/05/soap-envelope">
      <ns3:Code>
        <ns3:Value>ns3:Receiver</ns3:Value>
      </ns3:Code>
      <ns3:Reason>
        <ns3:Text xml:lang="en">Status:[status] is an invalid status.</ns3:Text>
      </ns3:Reason>
      <ns3:Detail>
        <ns2:StatusNotFoundException xmlns:ns2="http://ws.ltfslsle.oracle/">
          <message>Status:[status] is an invalid status.</message>
        </ns2:StatusNotFoundException>
      </ns3:Detail>
    </ns3:Fault>
  </S:Body>
</S:Envelope>

```

### getVolumeInfoByVolser

A synchronous call that retrieves the volume information of one or more volumes by volume serial number.

#### Format

```
getVolumeInfoByVolser (String[] volsers)
```

*LTFS-LE Web Service to call:*

```
getVolumeInfoByVolser([volser1, volser2, ...])
```

#### Parameter

*volser* is the volume serial number

#### Roles

Admin, Monitor, or Service

#### Exceptions Thrown

If any one of volume serial numbers is not found, a `VolumeNotFoundException` exception is thrown.

#### Recommended Timeout

600 seconds

#### SOAP Request

```

<?xml version = '1.0' encoding = 'UTF-8'?>
<env:Envelope xmlns:env="http://www.w3.org/2003/05/soap-envelope" xmlns:ns1

```

```
= "http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" xmlns:ns2
="http://1.0.xsd" xmlns:ns2="http://sb.ejb.ws.mds.ltfsle.oracle/">
  <env:Header>
    <ns1:Security>
      <ns1:UsernameToken>
        <ns1:Username>username</ns1:Username>
        <ns1:Password>password</ns1:Password>
      </ns1:UsernameToken>
    </ns1:Security>
  </env:Header>
  <env:Body>
    <ns2:getVolumeInfoByVolser>
      <arg0>F52270</arg0>
      <arg0>EL0010</arg0>
      <arg0>F50331</arg0>
    </ns2:getVolumeInfoByVolser>
  </env:Body>
</env:Envelope>
```

### SOAP Response

```
<?xml version = '1.0' encoding = 'UTF-8'?>
<S:Envelope xmlns:S="http://www.w3.org/2003/05/soap-envelope">
  <S:Body>
    <ns2:getVolumeInfoByVolserResponse xmlns:ns2="http://ws.ltfsle.oracle/">
      <return>
        <acslsPath>3 ,0 ,12 ,38 ,1</acslsPath>
        <assignedStatus>Assigned</assignedStatus>
        <availableStatus>Available</availableStatus>
        <capacityAvailable>1453410222080</capacityAvailable>
        <capacityUsed>24117248</capacityUsed>
        <currentLocation>home</currentLocation>
        <inuseStatus>In use</inuseStatus>
        <ltfsUUID>e255c63b-54c9-4fc7-a60b-e63d58ed0005</ltfsUUID>
        <onlineStatus>Online</onlineStatus>
        <volser>050191</volser>
        <volumeCreatedOn>1403015625</volumeCreatedOn>
        <volumePoolName>Default Volume Pool</volumePoolName>
        <volumeType>LTO-1.5T</volumeType>
      </return>
      <return>
        <acslsPath>3 ,0 ,12 ,38 ,2</acslsPath>
        <assignedStatus>Assigned</assignedStatus>
        <availableStatus>Available</availableStatus>
        <capacityAvailable>1453410222080</capacityAvailable>
        <capacityUsed>24117248</capacityUsed>
        <currentLocation>home</currentLocation>
        <inuseStatus>In use</inuseStatus>
        <ltfsUUID>e255c63b-54c9-4fc7-a60b-e63d58ed0005</ltfsUUID>
        <onlineStatus>Online</onlineStatus>
        <volser>050191</volser>
        <volumeCreatedOn>1403015625</volumeCreatedOn>
        <volumePoolName>Default Volume Pool</volumePoolName>
        <volumeType>LTO-1.5T</volumeType>
      </ns2:getVolumeInfoByVolserResponse>
    </S:Body>
  </S:Envelope>
```

### SOAP Fault

```
<?xml version = '1.0' encoding = 'UTF-8'?>
```



```
<S:Envelope xmlns:S="http://www.w3.org/2003/05/soap-envelope">
  <S:Body>
    <ns3:Fault xmlns:ns2="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:ns3="http://www.w3.org/2003/05/soap-envelope">
      </ns3:Code>
      <ns3:Value>ns3:Receiver</ns3:Value>
      </ns3:Code>
      <ns3:Reason>
        <ns3:Text xml:lang="en"> The volume(s) [vol1],[vol2],...,[vols] is(are) not found in the
LTFS-LE library</ns3:Text>
      </ns3:Reason>
      <ns3:Detail>
        <ns2:VolumeNotFoundException xmlns:ns2="http://ws.ltfslsle.oracle/">
          <message> The volume(s) [vol1],[vol2],...,[vols] is(are) not found in the LTFS-LE
library</message>
        </ns2:VolumeNotFoundException>
      </ns3:Detail>
    </ns3:Fault>
  </S:Body>
</S:Envelope>
```



---

---

## Management Services

Management services are the services that manage the storage resources. For example, use a management service to enter a volume, eject a volume, and bring a volume online or offline.

### Overview

These services need to communicate with the mechanical component of the solution, and the response time depends on the type of operation and the mechanical latencies. The management services are implemented as an asynchronous call. A management services is asynchronous because it returns before the job has been executed.

- The completion of the operation returns a program or job ID which the calling service needs to call to obtain the results when they are available.
- When the asynchronous service returns, it provides a job ID as part of the response. The return codes are.
  - 0 (Web service call associated with this job ID is complete.)
  - 1 (Web service call associated with this job ID is still in progress.)
  - -1 (No such web service call with the job ID is found in the system.)
- The programmer implementing the Web Services client needs to periodically call the `getJobStatus` Web Service by passing it the job ID to obtain the status of the job.
- Once `getJobStatus` returns a "0", call the `getJobResult` Web Service to obtain the service result.

Each asynchronous Web Services can take a varying amount of time to complete. The back-end times out if there is a failure on the mechanical side. This guide provides guidelines on the maximum recommended time for the third-party Web Services client to wait before timing out.

Refer to "[Status Services](#)" on page 3-1 for information on `getJobStatus` and `getJobResult`.

### assignVolume

To manage volumes using LTFS-LE, volumes must be assigned to the LTFS-LE application.

When a volume is assigned to LTFS-LE, it is automatically placed online and added to the default volume pool. When a non-formatted volume is assigned, the volume is automatically formatted for use with LTFS-LE.

## Format

```
assignVolume(String[] volsers)
```

### *Pre-conditions:*

Verify that a compatible drive is available and that the volume is offline, unassigned, idle, not in a pool, and available before assigning the volume. The volume or volumes must exist in the library.

### To do this:

- Obtain the volume status:

```
getVolumeInfoByVolser([volser1, volser2, ...])
```

- Ensure that the volume is in one of the required states and take corrective action, if necessary, before proceeding.

### *LTFS-LE Web Service to call:*

```
assignVolume([volser1, volser2, ...])  
getJobStatus(jobId) < poll each 10s until status is complete >  
getJobResult(jobId)
```

### *Post-condition:*

Once the series of Web Services calls have been executed, the results are an XML document. You must convert them into appropriate return codes. If the API calls succeed, the volume is now assigned to the LTFS-LE software. If the volume is not LTFS-formatted, the volume is automatically LTFS-formatted.

## Parameters

- *volsers* are the volume serial numbers of the volume or volumes to be assigned. The input parameters are an array of strings and enable multiple volumes to be assigned.
- *jobId* is the job ID used to retrieve the job status and result.

## Response

Valid responses are shown in the SOAP Response example. Refer to the Glossary for definitions.

## Roles

Admin or Service

## Exception Thrown

```
ValidationFailedException
```

### Exceptions can be thrown:

- If one or more volser serial numbers are not entered, the exception thrown is: The parameter *volsers* is null.

The *volser* is the first volume that does not meet any constraint (not all the volumes specified by *volsers*).

- If the volume does not exist in LTFS-LE, the exception thrown is: The volume *<volser>* is not in the LTFS-LE Database.

Example: The volume EL0010 is not in the LTFS-LE Database.

- If the volume status is not Available, Unassigned, Offline, and Idle, the exceptions thrown are:
  - The volume `<volser>`'s Availability Status is `<volser's real availableStatus>`. The volume's Availability Status must be Available to be assigned.  
Example: The volume EL0010's Availability Status is Unavailable.
  - The volume `<volser>`'s LTFS-LE Assignment Status is `<volser's real assignedStatus>`. The volume's LTFS-LE Assignment Status must be Unassigned to be assigned.  
Example: The volume EL0010's LTFS-LE Assignment Status is Assigned.
  - The volume `<volser>`'s Status is `<volser's real onlineStatus>`. The volume's Status must be Offline to be assigned.  
Example: The volume EL0010's Status is Online.
  - The volume `<volser>`'s Inuse Status is `<volser's real inuseStatus>`. The volume's Inuse Status must be Idle to be assigned.  
Example: The volume SL0010's Inuse Status is In Use.
- If the volume is already in the default volume pool, the exception thrown is: The volume `<volser>` is already assigned in the Default Volume Pool.  
Example: The volume EL0010 is already assigned in the Default Volume Pool.

### Recommended Timeout

7200 seconds

### Recommended Job Poll Interval

10 seconds

### Job Error

Failed to assign volume `<volser>`.

### SOAP Request

```
<?xml version = '1.0' encoding = 'UTF-8'?>
<env:Envelope xmlns:env="http://schemas.xmlsoap.org/soap/envelope/" xmlns:ns1
="http://sb.ejb.ws.mds.ltfsle.oracle/">
  <env:Header>
    <ns1:Security>
      <ns1:UsernameToken>
        <ns1:Username>username</ns1:Username>
        <ns1:Password>password</ns1:Password>
      </ns1:UsernameToken>
    </ns1:Security>
  </env:Header>
  <env:Body>
    <ns1:assignVolume>
      <arg0>F50233</arg0>
      <arg0>F50234</arg0>
    </ns1:assignVolume>
  </env:Body>
</env:Envelope>
```

## SOAP Response

```
<?xml version = '1.0' encoding = 'UTF-8'?>
<S:Envelope xmlns:S="http://www.w3.org/2003/05/soap-envelope">
  <S:Body>
    <ns2:assignVolumeResponse xmlns:ns2="http://sb.ejb.ws.mds.ltfsle.oracle/">
      <return>1220</return>
    </ns2:assignVolumeResponse>
  </S:Body>
</S:Envelope>
```

## SOAP Fault

```
<?xml version = '1.0' encoding = 'UTF-8'?>
<S:Envelope xmlns:S="http://www.w3.org/2003/05/soap-envelope">
  <S:Body>
    <ns3:Fault xmlns:ns2="http://schemas.xmlsoap.org/soap/envelope/" xmlns:ns3
="http://www.w3.org/2003/05/soap-envelope">
      <ns3:Code>
        <ns3:Value>ns3:Receiver</ns3:Value>
      </ns3:Code>
      <ns3:Reason>
        <ns3:Text xml:lang="en">The volume SL0010's Assignment Status is Assigned</ns3:Text>
      </ns3:Reason>
      <ns3:Detail>
        <ns2:ValidationFailedException xmlns:ns2="http://sb.ejb.ws.mds.ltfsle.oracle/">
          <message>The volume SL0010's Assignment Status is Assigned</message>
        </ns2:ValidationFailedException>
      </ns3:Detail>
    </ns3:Fault>
  </S:Body>
</S:Envelope>
```

## ejectVolume

This API function ejects one or more LTFS-LE volumes from the library.

Considering the message size, LTFS LE only supports ejecting up to 30 volumes using the mailslot (CAP). Ejects are done in fours until all volumes are completely ejected. There is no timeout. This is an asynchronous call.

### Format

```
ejectVolume(String[] volsers, String acs, String lsm, String cap)
```

#### Pre-condition:

Verify that: the volume(s) is finalized and offline before ejecting; a CAP is available; and the ACS is the same as the library configured. To do this:

1. Obtain the volume status:

```
getVolumeInfoByVolser([volser1, volser2, ...])
```

2. If the volume is online and not in use, finalize the volume:

```
finalizeVolume(volser)
getJobStatus(jobId) < poll each 10s until status is complete >
getJobResult(jobId)
```

3. If the volume is online, take the volume offline before executing the eject API call:

```
offlineVolume(volser)
```

```

    getJobStatus(jobId) < poll each 10s until status is complete >
    getJobResult(jobId)

```

*LTFS-LE WebService chain to call:*

```

ejectVolume([volser1,volser2,...],acs,lsm,cap)
getJobStatus(jobId) < poll each 10s until status is complete >
getJobResult(jobId)

```

*Post-condition:*

Once the series of Web Services calls have been executed, the results are an XML document. You must convert them into appropriate return codes. If the API calls succeed, the volume is ejected from the library. The user must walk up to the library and retrieve the volume after opening the CAP door.

### Parameters

- *volser* is the serial numbers of the volume or volumes to be ejected.
- *acs* is the library ACS identifier.
- *lsm* is the library LSM identifier.
- *cap* is the library CAP (mailslot) ID used to eject the volume.

### Response

Valid responses are shown in the SOAP Response example. Refer to the Glossary for definitions.

### Roles

Admin or Service

### Exception Thrown

ValidationFailedException

Exceptions can be thrown:

- If one or more *volser* serial numbers are not entered, the exception thrown is: The parameter *volsers* is null.  
The *volser* is the first volume that does not meet any constraint (not all the volumes specified by *volsers*).
- If the number of volumes specified is greater than 30, the exception thrown is: The number of volumes to eject must be 30 or less.
- If the *acs* parameter is null, the exception thrown is: The parameter *acs* is null.
- If the *lsm* parameter is null, the exception thrown is: The parameter *lsm* is null.
- If the *cap* parameter is null, the exception thrown is: The parameter *cap* is null.
- If the volume does not exist in LTFS-LE, the exception thrown is: The volume *<volser>* is not in the LTFS-LE Database.  
Example: The volume EL0010 is not in the LTFS-LE Database.
- If the volume status is not Offline and Idle, the exceptions thrown are:

- The volume <volser>'s Status is <volser's real onlineStatus>. The volume's Status must be Offline to eject the volume.

Example: The volume EL0010's Status is Online.

- The volume <volser>'s Inuse Status is <volser's real inuseStatus>. The volume's Inuse Status must be Idle to eject the volume.

Example: The volume SL0010's Inuse Status is In Use.

### Recommended Timeout

This depends on the time taken to remove the volume from the CAP (mailslot).

### Recommended Job Poll Interval

10 seconds

### Job Error

Failed to eject volume <volser>.

### SOAP Request

```
<?xml version = '1.0' encoding = 'UTF-8'?>
<env:Envelope xmlns:env="http://schemas.xmlsoap.org/soap/envelope/" xmlns:ns1
="http://sb.ejb.ws.mds.ltfsle.oracle/">
  <env:Header>
    <ns1:Security>
      <ns1:UsernameToken>
        <ns1:Username>username</ns1:Username>
        <ns1:Password>password</ns1:Password>
      </ns1:UsernameToken>
    </ns1:Security>
  </env:Header>
  <env:Body>
    <ns1:ejectVolume>
      <arg0>F50234</arg0>
      <arg0>F50233</arg0>
      <arg1>3</arg1>
      <arg2>0</arg2>
      <arg3>1</arg3>
    </ns1:ejectVolume>
  </env:Body>
</env:Envelope>
```

### SOAP Response

```
<?xml version = '1.0' encoding = 'UTF-8'?>
<S:Envelope xmlns:S="http://www.w3.org/2003/05/soap-envelope">
  <S:Header>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401
-wss-wssecurity-secext-1.0.xsd" S:mustUnderstand="true">
      <wsu:Timestamp xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd">
        <wsu:Created>2014-06-17T14:36:36Z</wsu:Created>
        <wsu:Expires>2014-06-17T14:37:36Z</wsu:Expires>
      </wsu:Timestamp>
    </wsse:Security>
  </S:Header>
  <S:Body>
    <ns2:ejectVolumeResponse xmlns:ns2="http://sb.ejb.ws.mds.ltfsle.oracle/">
```



```

    <return>1213</return>
  </ns2:ejectVolumeResponse>
</S:Body>
</S:Envelope>

```

## SOAP Fault

```

<?xml version = '1.0' encoding = 'UTF-8'?>
<S:Envelope xmlns:S="http://www.w3.org/2003/05/soap-envelope">
  <S:Body>
    <ns3:Fault xmlns:ns2="http://schemas.xmlsoap.org/soap/envelope/" xmlns:ns3
="http://www.w3.org/2003/05/soap-envelope">
      <ns3:Code>
        <ns3:Value>ns3:Receiver</ns3:Value>
      </ns3:Code>
      <ns3:Reason>
        <ns3:Text xml:lang="en">The volume SL0010's Status is Online</ns3:Text>
      </ns3:Reason>
      <ns3:Detail>
        <ns2:ValidationFailedExceptionxmlns:ns2="http://sb.ejb.ws.mds.ltfsle.oracle/">
          <message>The volume SL0010's Status is Online</message>
        </ns2:ValidationFailedException>
      </ns3:Detail>
    </ns3:Fault>
  </S:Body>
</S:Envelope>

```

## enterVolume

This is an asynchronous call that enters one or more volumes into the library using the CAP (mailslot).

### Format

```
enterVolume(String acs, String lsm, String cap)
```

#### Pre-condition:

There is no specific pre-condition for entering a volume into the CAP. However, the administrator or service must gather the CAP identifier (*acs*, *lsm*, and *cap\_id*), and ensure that a CAP is available and is not in the middle of an operation (pending, eject, enter, or audit) before executing this command.

#### LTFS-LE WebService chain to call:

```

enterVolume(acs, lsm, cap)
getJobStatus(jobId) <poll each 10s until status is complete>
getJobResult(jobId)

```

#### Post-condition:

Once the series of Web Services calls have been executed, the results are an XML document. You must convert them into appropriate return codes. If the API calls succeed, the volume is entered into the library.

### Parameters

- *acs* is the library ACS identifier.
- *lsm* is the library LSM identifier.
- *cap* is the library CAP (mailslot) ID used to enter the volume.

- *jobId* is the job ID used to retrieve the job status and result.

### Response

Valid responses are shown in the SOAP Response example. Refer to the Glossary for definitions.

### Roles

Admin or Service

### Exception Thrown

ValidationFailedException

Exceptions can be thrown:

- If the *acs* parameter is empty, the exception thrown is: The parameter *acs* is null.
- If the *lsm* parameter is empty, the exception thrown is: The parameter *lsm* is null.
- If the *cap* parameter is empty, the exception thrown is: The parameter *cap* is null.

Examples:

- ACS 4 is inconsistent with the library *acs 7*.
- Invalid ACS number.

### Recommended Timeout

This depends on the time it takes the administrator to enter the volume into the CAP.

### Recommended Job Poll Interval

10 seconds

### Job Error

Failed to enter volume *<volser>*.

### SOAP Request

```
<?xml version = '1.0' encoding = 'UTF-8'?>
<env:Envelope xmlns:env="http://schemas.xmlsoap.org/soap/envelope/" xmlns:ns1
="http://sb.ejb.ws.mds.ltfsle.oracle/">
  <env:Header>
    <ns1:Security>
      <ns1:UsernameToken>
        <ns1:Username>username</ns1:Username>
        <ns1:Password>password</ns1:Password>
      </ns1:UsernameToken>
    </ns1:Security>
  </env:Header>
  <env:Body>
    <ns1:enterVolume>
      <arg0>3</arg0>
      <arg1>0</arg1>
      <arg2>1</arg2>
    </ns1:enterVolume>
  </env:Body>
</env:Envelope>
```

```

</env:Body>
</env:Envelope>

```

### SOAP Response

```

<?xml version = '1.0' encoding = 'UTF-8'?>
<S:Envelope xmlns:S="http://www.w3.org/2003/05/soap-envelope">
  <S:Header>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/
oasis-200401-wss-wssecurity-secext-1.0.xsd" S:mustUnderstand="true">
      <wsu:Timestamp xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd">
        <wsu:Created>2014-06-17T14:36:36Z</wsu:Created>
        <wsu:Expires>2014-06-17T14:37:36Z</wsu:Expires>
      </wsu:Timestamp>
    </wsse:Security>
  </S:Header>
  <S:Body>
    <ns2:enterVolumeResponse xmlns:ns2="http://sb.ejb.ws.mds.ltfsle.oracle/">
      <return>1212</return>
    </ns2:enterVolumeResponse>
  </S:Body>
</S:Envelope>

```

### SOAP Fault

```

<?xml version = '1.0' encoding = 'UTF-8'?>
<S:Envelope xmlns:S="http://www.w3.org/2003/05/soap-envelope">
  <S:Body>
    <ns3:Fault xmlns:ns2="http://schemas.xmlsoap.org/soap/envelope/" xmlns:ns3
="http://www.w3.org/2003/05/soap-envelope">
      <ns3:Code>
        <ns3:Value>ns3:Receiver</ns3:Value>
      </ns3:Code>
      <ns3:Reason>
        <ns3:Text xml:lang="en"> Invalid ACS number. </ns3:Text>
      </ns3:Reason>
      <ns3:Detail>
        <ns2:ValidationFailedExceptionxmlns:ns2="http://sb.ejb.ws.mds.ltfsle.oracle/">
          <message>Invalid ACS number.</message>
        </ns2:ValidationFailedException>
      </ns3:Detail>
    </ns3:Fault>
  </S:Body>
</S:Envelope>

```

## finalizeVolume

Use the `finalizeVolume` command to commit any changes to the volume. Typically, this action is performed before the volume is taken offline to be ejected out of the library. Finalizing a volume applies any unwritten metadata or unapplied file deletes to the volume. This action ensures that the actions on the global namespace are reflected on the physical volume. Failing to finalize a volume before taking it offline can likely delay taking the volume offline.

### Format

```
finalizeVolume(String volser)
```

*Pre-conditions:*

Verify that a compatible drive is available and that the volume is online, in a pool, idle, and available before finalizing the volume. To do this:

- Obtain the volume status:

```
getVolumeInfoByVolser([volser1, volser2, ...])
```

- Ensure that the volume is in all the required states and take corrective action, if necessary, before proceeding.

*LTFS-LE Web Service to call:*

```
finalizeVolume(volser)
getJobStatus(jobId) < poll each 10s until status is complete >
getJobResult(jobId)
```

*Post-condition:*

Once the series of Web Services calls have been executed, the results are an XML document. You must convert them into appropriate return codes. If the API calls succeed, any pending operations that need to be applied to the volume have been applied to the volume.

## Response

Valid responses are shown in the SOAP Response example. Refer to the Glossary for definitions.

## Parameters

- *volser* is the volume serial number of the volume to be finalized.
- *jobId* is the job ID used to retrieve the job status and result.

## Roles

Admin or Service

## Exception Thrown

ValidationFailedException

Exceptions can be thrown:

- If the *volser* is not entered, the exception thrown is: The parameter *volser* is null.
- If the volume does not exist in LTFS-LE, the exception thrown is: The volume *<volser>* is not in the LTFS-LE Database.  
Example: The volume EL0010 is not in the LTFS-LE Database.
- If the volume status is not Available, Assigned, Offline, and Idle, the exceptions thrown are:
  - The volume *<volser>*'s Availability Status is *<volser's real availableStatus>*. The volume's Availability Status must be Available to be finalized.  
Example: The volume EL0010's Availability Status is Unavailable.
  - The volume *<volser>*'s LTFS-LE Assignment Status is *<volser's real assignedStatus>*. The volume's LTFS-LE Assignment Status must be Assigned to be brought offline.

Example: The volume EL0010's LTFS-LE Assignment Status is Unassigned.

- The volume `<volser>`'s Status is `<volser's real onlineStatus>`. The volume's Status must be Online to be brought offline.

Example: The volume EL0010's Status is Offline.

- The volume `<volser>`'s Inuse Status is `<volser's real inuseStatus>`. The volume's Inuse Status must be Idle to be brought offline.

Example: The volume SL0010's Inuse Status is In Use.

- If the volume is not in the default volume pool, the exception thrown is: The volume `<volser>` is not in the Default Volume Pool.

Example: The volume EL0010 is not in the Default Volume Pool.

### Recommended Timeout

7200 seconds

### Recommended Job Poll Interval

10 seconds

### Job Errors

Failed to finalize volume `<volser>`.

### SOAP Request

```
<?xml version = '1.0' encoding = 'UTF-8'?>
<env:Envelope xmlns:env="http://schemas.xmlsoap.org/soap/envelope/" xmlns:ns1
="http://sb.ejb.ws.mds.ltfsle.oracle/">
  <env:Header>
    <ns1:Security>
      <ns1:UsernameToken>
        <ns1:Username>username</ns1:Username>
        <ns1:Password>password</ns1:Password>
      </ns1:UsernameToken>
    </ns1:Security>
  </env:Header>
  <env:Body>
    <ns1:finalizeVolume>
      <arg0>F50233</arg0>
    </ns1:finalizeVolume>
  </env:Body>
</env:Envelope>
```

### SOAP Response

```
<?xml version = '1.0' encoding = 'UTF-8'?>
<S:Envelope xmlns:S="http://www.w3.org/2003/05/soap-envelope">
  <S:Header>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
secext-1.0.xsd" S:mustUnderstand="true">
      <wsu:Timestamp xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd">
        <wsu:Created>2014-06-17T14:36:36Z</wsu:Created>
        <wsu:Expires>2014-06-17T14:37:36Z</wsu:Expires>
      </wsu:Timestamp>
    </wsse:Security>
```

```
</S:Header>
<S:Body>
  <ns2:finalizeVolumeResponse xmlns:ns2="http://sb.ejb.ws.mds.ltfsle.oracle/">
    <return>1219</return>
  </ns2:finalizeVolumeResponse>
</S:Body>
</S:Envelope>
```

### SOAP Fault

```
<?xml version = '1.0' encoding = 'UTF-8'?>
<S:Envelope xmlns:S="http://www.w3.org/2003/05/soap-envelope">
  <S:Body>
    <ns3:Fault xmlns:ns2="http://schemas.xmlsoap.org/soap/envelope/" xmlns:ns3
="http://www.w3.org/2003/05/soap-envelope">
      <ns3:Code>
        <ns3:Value>ns3:Receiver</ns3:Value>
      </ns3:Code>
      <ns3:Reason>
        <ns3:Text xml:lang="en">The volume SL0010's Status is Offline</ns3:Text>
      </ns3:Reason>
      <ns3:Detail>
        <ns2:ValidationFailedException xmlns:ns2="http://sb.ejb.ws.mds.ltfsle.oracle/">
          <message>The volume SL0010's Status is Offline</message>
        </ns2:ValidationFailedException>
      </ns3:Detail>
    </ns3:Fault>
  </S:Body>
</S:Envelope>
```

## formatVolume

An asynchronous call that formats the volume into an LTFS-formatted volume. The LTFS-LE WebService does not require confirmation nor does it perform any validation of the volume. You must be aware of the volume that is being formatted and ensure that the correct volume is formatted.

When a volume is formatted, all previous data on the volume is deleted.

### Format

`formatVolume(String volser)`

#### *Pre-condition:*

Verify the volume is offline, assigned, and available before formatting the volume. Check that there is a compatible drive available.

To do this:

- Obtain the volume status:  
`getVolumeInfoByVolser([volser1, volser2, ...])`
- Ensure that the volume is in the state required by the precondition.
- Take corrective actions, if necessary, before proceeding.

#### *LTFS-LE Web Service chain to call:*

```
formatVolume(volser)
getJobStatus(jobId) < poll each 10s until status is complete >
getJobResult(jobId)
```

*Post-condition:*

Once the series of Web Services calls have been executed, the results are an XML document. You must convert them into appropriate return codes. If the API calls succeed, the volume is now LTFS-LE formatted.

**Parameters**

- *volser* is the serial number of the volume to be LTFS-LE formatted.
- *jobId* is the job ID used to retrieve the job status and result.

**Response**

Valid responses are shown in the SOAP Response example. Refer to the Glossary for definitions.

**Roles**

Admin or Service

**Exception Thrown**

ValidationFailedException

Exceptions can be thrown:

- If the *volser* is not entered, the exception thrown is: The parameter *volser* is null.
  - If the volume does not exist in LTFS-LE, the exception thrown is: The volume *<volser>* is not in the LTFS-LE Database.
- Example: The volume EL0010 is not in the LTFS-LE Database.
- If the volume status is not Available, Assigned, Offline, and Idle, the exceptions thrown are:

- The volume *<volser>*'s Availability Status is *<volser's real availableStatus>*. The volume's Availability Status must be Available to be formatted.

Example: The volume EL0010's Availability Status is Unavailable.

- The volume *<volser>*'s LTFS-LE Assignment Status is *<volser's real assignedStatus>*. The volume's LTFS-LE Assignment Status must be Assigned to be formatted.

Example: The volume EL0010's LTFS-LE Assignment Status is Unassigned.

- The volume *<volser>*'s Status is *<volser's real onlineStatus>*. The volume's Status must be Offline to be finalized.

Example: The volume EL0010's Status is Online.

- The volume *<volser>*'s Inuse Status is *<volser's real inuseStatus>*. The volume's Inuse Status must be Idle to be formatted.

Example: The volume SL0010's Inuse Status is In Use.

**Recommended Timeout**

7,200 seconds

## Recommended Job Poll Interval

10 seconds

## Job Errors

Failed to format volume <volser>.

## SOAP Request

```
<?xml version = '1.0' encoding = 'UTF-8'?>
<env:Envelope xmlns:env="http://schemas.xmlsoap.org/soap/envelope/" xmlns:ns1
="http://sb.ejb.ws.mds.ltfsle.oracle/">
  <env:Header>
    <ns1:Security>
      <ns1:UsernameToken>
        <ns1:Username>username</ns1:Username>
        <ns1:Password>password</ns1:Password>
      </ns1:UsernameToken>
    </ns1:Security>
  </env:Header>
  <env:Body>
    <ns1:formatVolume>
      <arg0>F50234</arg0>
    </ns1:formatVolume>
  </env:Body>
</env:Envelope>
```

## SOAP Response

```
<?xml version = '1.0' encoding = 'UTF-8'?>
<S:Envelope xmlns:S="http://www.w3.org/2003/05/soap-envelope">
  <S:Header>
    <wsse:Security
xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
secext-1.0.xsd" S:mustUnderstand="true">
      <wsu:Timestamp xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd">
        <wsu:Created>2014-06-17T14:36:36Z</wsu:Created>
        <wsu:Expires>2014-06-17T14:37:36Z</wsu:Expires>
      </wsu:Timestamp>
    </wsse:Security>
  </S:Header>
  <S:Body>
    <ns2:formatVolumeResponse xmlns:ns2="http://sb.ejb.ws.mds.ltfsle.oracle/">
      <return>1215</return>
    </ns2:formatVolumeResponse>
  </S:Body>
</S:Envelope>
```

## SOAP Fault

```
<?xml version = '1.0' encoding = 'UTF-8'?>
<S:Envelope xmlns:S="http://www.w3.org/2003/05/soap-envelope">
  <S:Body>
    <ns3:Fault xmlns:ns2="http://schemas.xmlsoap.org/soap/envelope/" xmlns:ns3
="http://www.w3.org/2003/05/soap-envelope">
      <ns3:Code>
        <ns3:Value>ns3:Receiver</ns3:Value>
      </ns3:Code>
      <ns3:Reason>
        <ns3:Text xml:lang="en">The volume EL0010's Available Status is Unavailable</ns3:Text>
      </ns3:Reason>
    </ns3:Fault>
  </S:Body>
</S:Envelope>
```



```

</ns3:Reason>
<ns3:Detail>
  <ns2:ValidationFailedExceptionxmlns:ns2="http://sb.ejb.ws.mds.ltfsle.oracle/">
    <message>The volume EL0010's Aailable Status is Unavailable</message>
  </ns2:ValidationFailedException>
</ns3:Detail>
</ns3:Fault>
</S:Body>
</S:Envelope>

```

## formatVolumeWithDrive

This asynchronous call formats the volume into an LTFS format and is used when a volume must be formatted in a specific drive. The volume and drive must be compatible.

### Format

`formatVolumeWithDrive (String volser, String driveSN)`

#### Pre-condition:

Verify that the volume is offline, assigned, and available before formatting the volume. The volume and drive must exist in library and the volume and drive must be compatible. The drive must be in an idle or available state.

To do this:

- Obtain the volume status:
 

```
getVolumeInfoByVolser([volser1, volser2, ...])
```
- Ensure that the volume is in the state required by the precondition.
- Take corrective actions, if necessary, before proceeding.

#### LTFS-LE Web Service chain to call:

```

formatVolumeWithDrive(volser, driveSN)
getJobStatus(jobId) < poll each 10s until status is complete >
getJobResult(jobId)

```

#### Post-condition:

Once the series of Web Services calls have been executed, the results are an XML document. You must convert them into appropriate return codes. If the API calls succeed, the volume is now LTFS-LE formatted.

### Parameters

- `volser` is the volume serial number.
- `driveSN` is the drive serial number of the drive formatting the volume.
- `jobId` is the job ID used to retrieve the job status and result.

### Response

Valid responses are shown in the SOAP Response example. Refer to the Glossary for definitions.

## Roles

Admin or Service

## Exception Thrown

ValidationFailedException

Exceptions that can be thrown:

### Volumes:

- If the volser serial number is not entered, the exception thrown is: The parameter `volser` is null.
- If the volume does not exist in LTFS-LE, the exception thrown is: The volume `<volser>` is not in the LTFS-LE Database.

Example: The volume EL0010 is not in the LTFS-LE Database.

- If the volume status is not Available, Assigned, Offline, and Idle, the exceptions thrown are:
  - The volume `<volser>`'s Availability Status is `<volser's real availableStatus>`. The volume's Availability Status must be Available to be formatted.  
Example: The volume EL0010's Availability Status is Unavailable.
  - The volume `<volser>`'s LTFS-LE Assignment Status is `<volser's real assignedStatus>`. The volume's LTFS-LE Assignment Status must be Assigned to be formatted.  
Example: The volume EL0010's LTFS-LE Assignment Status is Unassigned.
  - The volume `<volser>`'s Status is `<volser's real onlineStatus>`. The volume's Status must be Offline to be formatted.  
Example: The volume EL0010's Status is Online.
  - The volume `<volser>`'s Inuse Status is `<volser's real inuseStatus>`. The volume's Inuse Status must be Idle to be formatted.  
Example: The volume SL0010's Inuse Status is In Use.

### Drives:

- If the drive serial number is not entered, the exception thrown is: The parameter `driveSN` is null.
- If the drive does not exist in LTFS-LE, the exception thrown is: The drive `<driveSN>` is not in the LTFS-LE Database.  
Example: The drive 123456 is not in the LTFS-LE Database.
- If the drive status is not Available, Assigned, and Online the exceptions thrown are:
  - The drive `<driveSN>`'s Availability Status is `<drive's real availableStatus>`. The drive must be Available to be used.  
Example: The drive 123456's Availability Status is Unavailable.
  - The drive `<driveSN>`'s LTFS-LE Assignment Status is `<drive's real assignedStatus>`. The drive must be Assigned to be used.  
Example: The drive 123457's LTFS-LE Assignment Status is Unassigned.

- The drive `<driveSN>`'s Status is `<drive's real onlineStatus>`. The drive must be Online to be used.

Example: The drive 123456's Status is Offline.

### Volume Type and Drive Type Incompatibility:

- If the volume type and drive type are not compatible, the exception thrown is: The volume `<volser>`'s type `<vol_type>` is not compatible with the drive `<driveSN>`'s type `<drive_type>`.

Example: The volume EL0010's type LTO-5 is not compatible with drive 123456's type T1C.

### Recommended Timeout

7200 seconds

### Recommended Job Poll Interval

10 seconds

### Job Errors

Failed to format volume `<volser>`.

### SOAP Request

```
<?xml version = '1.0' encoding = 'UTF-8'?>
<env:Envelope xmlns:env="http://schemas.xmlsoap.org/soap/envelope/" xmlns:ns1
="http://sb.ejb.ws.mds.ltfsle.oracle/">
  <env:Header>
    <ns1:Security>
      <ns1:UsernameToken>
        <ns1:Username>username</ns1:Username>
        <ns1:Password>password</ns1:Password>
      </ns1:UsernameToken>
    </ns1:Security>
  </env:Header>
  <env:Body>
    <ns1:formatVolumeWithDrive>
      <arg0>F50234</arg0>
      <arg1>576001000421</arg1>
    </ns1:formatVolumeWithDrive>
  </env:Body>
</env:Envelope>
```

### SOAP Response

```
<?xml version = '1.0' encoding = 'UTF-8'?>
<S:Envelope xmlns:S="http://www.w3.org/2003/05/soap-envelope">
  <S:Header>
    <wsse:Security
xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
secext-1.0.xsd" S:mustUnderstand="true">
      <wsu:Timestamp xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd">
        <wsu:Created>2014-06-17T14:36:36Z</wsu:Created>
        <wsu:Expires>2014-06-17T14:37:36Z</wsu:Expires>
      </wsu:Timestamp>
    </wsse:Security>
  </S:Header>
```

```
<S:Body>
  <ns2:formatVolumeWithDriveResponse xmlns:ns2="http://sb.ejb.ws.mds.ltfsle.oracle/">
    <return>1215</return>
  </ns2:formatVolumeWithDriveResponse>
</S:Body>
</S:Envelope>
```

### SOAP Fault

```
<?xml version = '1.0' encoding = 'UTF-8'?>
<S:Envelope xmlns:S="http://www.w3.org/2003/05/soap-envelope">
  <S:Body>
    <ns3:Fault xmlns:ns2="http://schemas.xmlsoap.org/soap/envelope/" xmlns:ns3
="http://www.w3.org/2003/05/soap-envelope">
      <ns3:Code>
        <ns3:Value>ns3:Receiver</ns3:Value>
      </ns3:Code>
      <ns3:Reason>
        <ns3:Text xml:lang="en">The volume EL0010's Availability Status is
Unavailable</ns3:Text>
      </ns3:Reason>
      <ns3:Detail>
        <ns2:ValidationFailedExceptionxmlns:ns2="http://sb.ejb.ws.mds.ltfsle.oracle/">
          <message>The volume EL0010's Availability Status is Unavailable</message>
        </ns2:ValidationFailedException>
      </ns3:Detail>
    </ns3:Fault>
  </S:Body>
</S:Envelope>
```

## getVolumePath

This asynchronous call finds the volume path in the global namespace.

### Format

```
getVolumePath(volser)
```

#### *Pre-Condition:*

The volume status should be available and assigned. The unassigned volume has no *volser* folder.

### Parameters

- *volser* is the volume serial number for the volume path to be found.
- *jobId* is the job ID used to retrieve the job status and result.

### Response

Valid responses are shown in the SOAP Response example. Refer to the Glossary for definitions.

### Roles

Admin or Service

### Exception Thrown

```
ValidationFailedException
```

Exceptions can be thrown:

- If the `volser` is not entered, the exception thrown is: The parameter `volser` is null.
- If the volume does not exist in LTFS-LE, the exception thrown is: The volume `<volser>` is not in the LTFS-LE Database.

Example: The volume EL0010 is not in the LTFS-LE Database.

- If the volume status is not Available and Assigned the exceptions thrown are:
  - The volume `<volser>`'s Availability Status is `<volser's real availableStatus>`. The volume's Availability Status must be Available to get the volume path.

Example: The volume EL0010's Availability Status is Unavailable.

- The volume `<volser>`'s LTFS-LE Assignment Status is `<volser's real assignedStatus>`. The volume's LTFS-LE Assignment Status must be Assigned to get the volume path.

Example: The volume EL0010's LTFS-LE Assignment Status is Unassigned.

### Recommended Timeout

3600 seconds

### Recommended Job Poll Interval

10 seconds

### Job Errors

Volume path not found.

### SOAP Request

```
<?xml version = '1.0' encoding = 'UTF-8'?>
<env:Envelope xmlns:env="http://schemas.xmlsoap.org/soap/envelope/" xmlns:ns1
="http://sb.ejb.ws.mds.ltfsle.oracle/">
  <env:Header>
    <ns1:Security>
      <ns1:UsernameToken>
        <ns1:Username>username</ns1:Username>
        <ns1:Password>password</ns1:Password>
      </ns1:UsernameToken>
    </ns1:Security>
  </env:Header>
  <env:Body>
    <ns1:getVolumePath>
      <arg0>F50233</arg0>
    </ns1:getVolumePath>
  </env:Body>
</env:Envelope>
```

### SOAP Response

```
<?xml version = '1.0' encoding = 'UTF-8'?>
<S:Envelope xmlns:S="http://www.w3.org/2003/05/soap-envelope">
  <S:Header>
```

```

    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
secext-1.0.xsd" S:mustUnderstand="true">
      <wsu:Timestamp xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd">
        <wsu:Created>2014-06-17T14:36:36Z</wsu:Created>
        <wsu:Expires>2014-06-17T14:37:36Z</wsu:Expires>
      </wsu:Timestamp>
    </wsse:Security>
  </S:Header>
  <S:Body>
    <ns2: getVolumePath xmlns:ns2="http://sb.ejb.ws.mds.ltfsle.oracle/">
      <return>1225</return>
    </ns2: getVolumePath >
  </S:Body>
</S:Envelope>

```

## SOAP Fault

```

<?xml version = '1.0' encoding = 'UTF-8'?>
<S:Envelope xmlns:S="http://www.w3.org/2003/05/soap-envelope">
  <S:Body>
    <ns3:Fault xmlns:ns2="http://schemas.xmlsoap.org/soap/envelope/" xmlns:ns3
="http://www.w3.org/2003/05/soap-envelope">
      <ns3:Code>
        <ns3:Value>ns3:Receiver</ns3:Value>
      </ns3:Code>
      <ns3:Reason>
        <ns3:Text xml:lang="en">The volume EL0010's Availability Status is
Unavailable</ns3:Text>
      </ns3:Reason>
      <ns3:Detail>
        <ns2:ValidationFailedExceptionxmlns:ns2="http://sb.ejb.ws.mds.ltfsle.oracle/">
          <message>The volume EL0010's Availability Status is Unavailable</message>
        </ns2:ValidationFailedException>
      </ns3:Detail>
    </ns3:Fault>
  </S:Body>
</S:Envelope>

```

## offlineVolume

This is an asynchronous call that places the volume in an offline state and enables the ejecting, vaulting, formatting, assigning, and unassigning of one or more volumes.

### Format

```
offlineVolume(String[] volser)
```

#### Pre-conditions:

Verify that the volume exists in the library, is online, assigned, available, and in a pool before setting it offline. Check that there is a compatible drive available. The offline operation finalizes the volume automatically.

- Obtain the volume status:

```
getVolumeInfoByVolser([volser1, volser2, ...])
```

- Finalize the volume if the volume is in an online assigned state.

```
finalizeVolume(volser)
getJobStatus(jobId) < poll each 10s until status is complete >
```

```
getJobResult(jobId)
```

*LTFS-LE Web Service chain to call:*

```
offlineVolume([volser1, volser2, ...])
getJobStatus(jobId) < poll each 10s until status is complete >
getJobResult(jobId)
```

*Post-condition:*

Once the series of Web Services calls have been executed, the results are an XML document. You must convert them into appropriate return codes. If the API calls succeed, the volume is now offline.

### Parameters

- *volser*s are the volume serial numbers of the volume or volumes to be taken offline. The input parameters are an array of strings; the call allows multiple volumes to be taken offline.
- *jobId* is the job ID used to retrieve the job status and result.

### Response

Valid responses are shown in the SOAP Response example. Refer to the Glossary for definitions.

### Roles

Admin or Service

### Exception Thrown

ValidationFailedException

Exceptions can be thrown:

- If one or more of the *volser* serial numbers are not entered, the exception thrown is: The parameter *volser*s is null.  
The *volser* is the first volume that does not meet any constraint (not all the volumes specified by *volser*s).
- If the volume does not exist in LTFS-LE, the exception thrown is: The volume *<volser>* is not in the LTFS-LE Database.  
Example: The volume EL0010 is not in the LTFS-LE Database.
- If the volume status is not Available, Assigned, Online, and Idle, the exceptions thrown are:
  - The volume *<volser>*'s Availability Status is *<volser's real availableStatus>*. The volume's Availability Status must be Available to be brought offline.  
Example: The volume EL0010's Availability Status is Unavailable.
  - The volume *<volser>*'s LTFS-LE Assignment Status is *<volser's real assignedStatus>*. The volume's LTFS-LE Assignment Status must be Assigned to be brought offline.  
Example: The volume EL0010's LTFS-LE Assignment Status is Unassigned.

- The volume `<volser>`'s Status is `<volser's real onlineStatus>`. The volume's Status must be Online to be brought offline.

Example: The volume EL0010's Status is Offline.

- If the volume is not in the default volume pool, the exception thrown is: The volume `<volser>` is not in the Default Volume Pool.

Example: The volume EL0010 is not in the Default Volume Pool.

### Recommended Timeout

7200 seconds

### Recommended Job Poll Interval

10 seconds

### Job Errors

Failed to bring volume `<volser>` offline.

### SOAP Request

```
<?xml version = '1.0' encoding = 'UTF-8'?>
<env:Envelope xmlns:env="http://schemas.xmlsoap.org/soap/envelope/" xmlns:ns1
="http://sb.ejb.ws.mds.ltfsle.oracle/">
  <env:Header>
    <ns1:Security>
      <ns1:UsernameToken>
        <ns1:Username>username</ns1:Username>
        <ns1:Password>password</ns1:Password>
      </ns1:UsernameToken>
    </ns1:Security>
  </env:Header>
  <env:Body>
    <ns1:offlineVolume>
      <arg0>F50234</arg0>
      <arg0>F50233</arg0>
    </ns1:offlineVolume>
  </env:Body>
</env:Envelope>
```

### SOAP Response

```
<?xml version = '1.0' encoding = 'UTF-8'?>
<S:Envelope xmlns:S="http://www.w3.org/2003/05/soap-envelope">
  <S:Header>
    <wsse:Security
xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
secext-1.0.xsd" S:mustUnderstand="true">
      <wsu:Timestamp xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd">
        <wsu:Created>2014-06-17T14:36:36Z</wsu:Created>
        <wsu:Expires>2014-06-17T14:37:36Z</wsu:Expires>
      </wsu:Timestamp>
    </wsse:Security>
  </S:Header>
  <S:Body>
    <ns2:offlineVolumeResponse xmlns:ns2="http://sb.ejb.ws.mds.ltfsle.oracle/">
      <return>1218</return>
    </ns2:offlineVolumeResponse>
```



```
</S:Body>
</S:Envelope>
```

### SOAP Fault

```
<?xml version = '1.0' encoding = 'UTF-8'?>
<S:Envelope xmlns:S="http://www.w3.org/2003/05/soap-envelope">
  <S:Body>
    <ns3:Fault xmlns:ns2="http://schemas.xmlsoap.org/soap/envelope/" xmlns:ns3
="http://www.w3.org/2003/05/soap-envelope">
      <ns3:Code>
        <ns3:Value>ns3:Receiver</ns3:Value>
      </ns3:Code>
      <ns3:Reason>
        <ns3:Text xml:lang="en">The volume EL0010's Availability Status is
Unavailable</ns3:Text>
      </ns3:Reason>
      <ns3:Detail>
        <ns2:ValidationFailedExceptionxmlns:ns2="http://sb.ejb.ws.mds.ltfsle.oracle/">
          <message>The volume EL0010's Availablility Status is Unavailable</message>
        </ns2:ValidationFailedException>
      </ns3:Detail>
    </ns3:Fault>
  </S:Body>
</S:Envelope>
```

## onlineVolume

An asynchronous call that places the volume in an online state which enables the reading, writing or finalizing of volume.s.

### Format

```
onlineVolume(String[] volsers)
```

#### *Pre-conditions:*

Verify that the volume exists in the library and is offline, assigned, in pool, idle, and available before setting it online. Check that there is a compatible drive available.

```
getVolumeInfoByVolser([volser1, volser2, ...])
```

#### *LTFS-LE Web Service to call:*

```
onlineVolume([volser1, volser2, ...])
getJobStatus(jobId) < poll each 10s until status is complete >
getJobResult(jobId)
```

#### *Post-condition:*

Once the series of Web Services calls have been executed, the results are an XML document. You must convert them into appropriate return codes. If the API calls succeed, the volume is now online.

### Parameters

- *volsers* are the volume serial numbers of the volume or volumes to be brought online. The input parameters are an array of strings; the call allows multiple volumes to be brought online.
- *jobId* is the job ID used to retrieve the job status and result.

## Response

Valid responses are shown in the SOAP Response example. Refer to the Glossary for definitions.

## Roles

Admin or Service

## Exception Thrown

`ValidationFailedException`

Exceptions can be thrown:

- If one or more `volser` serial numbers are not entered, the exception thrown is: The parameter `volser`s is null.

The *volser* is the first volume that does not meet any constraint (not all the volumes specified by *volser*s).

- If the volume does not exist in LTFS-LE, the exception thrown is: The volume `<volser>` is not in the LTFS-LE Database.

Example: The volume EL0010 is not in the LTFS-LE Database.

- If the volume status is not Available, Assigned, Offline, and Idle, the exceptions thrown are:

- The volume `<volser>`'s Availability Status is `<volser's real availableStatus>`. The volume's Availability Status must be Available to be brought online.

Example: The volume EL0010's Availability Status is Unavailable.

- The volume `<volser>`'s LTFS-LE Assignment Status is `<volser's real assignedStatus>`. The volume's LTFS-LE Assignment Status must be Assigned to be brought online.

Example: The volume EL0010's LTFS-LE Assignment Status is Unassigned.

- The volume `<volser>`'s Status is `<volser's real onlineStatus>`. The volume's Status must be Offline to be brought online.

Example: The volume EL0010's Status is Online.

- The volume `<volser>`'s Inuse Status is `<volser's real inuseStatus>`. The volume's Inuse Status must be Idle to be brought online.

Example: The volume SL0010's Inuse Status is In Use.

## Recommended Timeout

7200 seconds

## Recommended Job Poll Interval

10 seconds

## Job Error

Failed to bring volume `<volser>` online.

## SOAP Request

```
<?xml version = '1.0' encoding = 'UTF-8'?>
<env:Envelope xmlns:env="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:ns1="http://sb.ejb.ws.mds.ltfsle.oracle/">
  <env:Header>
    <ns1:Security>
      <ns1:UsernameToken>
        <ns1:Username>username</ns1:Username>
        <ns1:Password>password</ns1:Password>
      </ns1:UsernameToken>
    </ns1:Security>
  </env:Header>
  <env:Body>
    <ns1:onlineVolume>
      <arg0>F40244</arg0>
      <arg0>F43022</arg0>
    </ns1:onlineVolume>
  </env:Body>
</env:Envelope>
```

## SOAP Response

```
<?xml version = '1.0' encoding = 'UTF-8'?>
<S:Envelope xmlns:S="http://www.w3.org/2003/05/soap-envelope">
  <S:Header>
    <wsse:Security
xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
secext-1.0.xsd" S:mustUnderstand="true">
      <wsu:Timestamp xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd">
        <wsu:Created>2014-06-17T14:36:36Z</wsu:Created>
        <wsu:Expires>2014-06-17T14:37:36Z</wsu:Expires>
      </wsu:Timestamp>
    </wsse:Security>
  </S:Header>
  <S:Body>
    <ns2:onlineVolumeResponse xmlns:ns2="http://sb.ejb.ws.mds.ltfsle.oracle/">
      <return>1217</return>
    </ns2:onlineVolumeResponse>
  </S:Body>
</S:Envelope>
```

## SOAP Fault

```
<?xml version = '1.0' encoding = 'UTF-8'?>
<S:Envelope xmlns:S="http://www.w3.org/2003/05/soap-envelope">
  <S:Body>
    <ns3:Fault xmlns:ns2="http://schemas.xmlsoap.org/soap/envelope/" xmlns:ns3
="http://www.w3.org/2003/05/soap-envelope">
      <ns3:Code>
        <ns3:Value>ns3:Receiver</ns3:Value>
      </ns3:Code>
      <ns3:Reason>
        <ns3:Text xml:lang="en">The volume SL0010's Aailblity Status is Unavailable</ns3:Text>
      </ns3:Reason>
      <ns3:Detail>
        <ns2:ValidationFailedExceptionxmlns:ns2="http://sb.ejb.ws.mds.ltfsle.oracle/">
          <message>The volume SL0010's Availability Status is Unavailable</message>
        </ns2:ValidationFailedException>
      </ns3:Detail>
    </ns3:Fault>
  </S:Body>
</S:Envelope>
```

```
</ns3:Fault>
</S:Body>
</S:Envelope>
```

## prepareVolume

This asynchronous call moves the volume folder from the `/LTFSLE/LOST+FOUND+VOLUMES/` to the `/LTFSLE/` folder.

### Format

```
prepareVolume(volser)
```

#### *Pre-conditions:*

Verify that the volume folder is located in `/LTFSLE/LOST+FOUND+VOLUMES` and that the volume exists in the library and is offline, assigned, and available before preparing the volume. To do this:

- Obtain the volume status:

```
getVolumeInfoByVolser([volser1, volser2, ...])
```

- Obtain the location of the volume.

```
getVolumePath(volser)
```

- Ensure that the volume is in all required states and take corrective action, if necessary, before proceeding.

#### *LTFS-LE Web Service to call:*

```
prepareVolume(volser)
getJobStatus(jobId) < poll each 10s until status is complete >
getJobResult(jobId)
```

#### *Post-condition:*

Once the series of Web Services calls have been executed, the results are an XML document. You must convert them into appropriate return codes. If the API calls succeed, the volume folder is now accessible from the `/LTFSLE` folder.

### Parameters

- *volser* is the volume serial number to prepare to be accessible through the `/LTFSLE/` location.
- *jobId* is the job ID used to retrieve the job status and result.

### Response

Valid responses are shown in the SOAP Response example. Refer to the Glossary for definitions.

### Roles

Admin or Service

### Exception Thrown

```
ValidationFailedException
```

Exceptions can be thrown:

- If the `volser` is not entered, the exception thrown is: The parameter `volser` is null.
- If the volume does not exist in LTFS-LE, the exception thrown is: The volume `<volser>` is not in the LTFS-LE Database.  
Example: The volume EL0010 is not in the LTFS-LE Database.
- If the volume status is not Available, Assigned, and Offline the exceptions thrown are:
  - The volume `<volser>`'s Availability Status is `<volser's real availableStatus>`. The volume's Availability Status must be Available to initiate the `prepareVolume` command.  
Example: The volume EL0010's Availability Status is Unavailable.
  - The volume `<volser>`'s LTFS-LE Assignment Status is `<volser's real assignedStatus>`. The volume's LTFS-LE Assignment Status must be Assigned to initiate the `prepareVolume` command.  
Example: The volume EL0010's LTFS-LE Assignment Status is Unassigned.
  - The volume `<volser>`'s Status is `<volser's real onlineStatus>`. The volume's Status must be Offline to initiate the `prepareVolume` command.  
Example: The volume EL0010's Status is Online.
  - The volume `<volser>`'s Inuse Status is `<volser's real inuseStatus>`. The volume's Inuse Status must be Idle to initiate the `prepareVolume` command.  
Example: The volume SL0010's Inuse Status is In Use.

**Recommended Timeout**

3600 seconds

**Recommended Job Poll Interval**

10 seconds

**Job Error**Failed to prepare volume `<volser>`.**SOAP Request**

```
<?xml version = '1.0' encoding = 'UTF-8'?>
<env:Envelope xmlns:env="http://schemas.xmlsoap.org/soap/envelope/" xmlns:ns1
="http://sb.ejb.ws.mds.ltfsle.oracle/">
  <env:Header>
    <ns1:Security>
      <ns1:UsernameToken>
        <ns1:Username>username</ns1:Username>
        <ns1:Password>password</ns1:Password>
      </ns1:UsernameToken>
    </ns1:Security>
  </env:Header>
  <env:Body>
    <ns1:prepareVolume>
      <arg0>F50233</arg0>
    </ns1:prepareVolume>
  </env:Body>
</env:Envelope>
```

```
</env:Body>
</env:Envelope>
```

### SOAP Response

```
<?xml version = '1.0' encoding = 'UTF-8'?>
<S:Envelope xmlns:S="http://www.w3.org/2003/05/soap-envelope">
  <S:Header>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd" S:mustUnderstand="true">
      <wsu:Timestamp xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd">
        <wsu:Created>2014-06-17T14:36:36Z</wsu:Created>
        <wsu:Expires>2014-06-17T14:37:36Z</wsu:Expires>
      </wsu:Timestamp>
    </wsse:Security>
  </S:Header>
  <S:Body>
    <ns2:prepareVolumeResponse xmlns:ns2="http://sb.ejb.ws.mds.ltfsle.oracle/">
      <return>1221</return>
    </ns2:prepareVolumeResponse>
  </S:Body>
</S:Envelope>
```

### SOAP Fault

```
<?xml version = '1.0' encoding = 'UTF-8'?>
<S:Envelope xmlns:S="http://www.w3.org/2003/05/soap-envelope">
  <S:Body>
    <ns3:Fault xmlns:ns2="http://schemas.xmlsoap.org/soap/envelope/" xmlns:ns3="http://www.w3.org/2003/05/soap-envelope">
      <ns3:Code>
        <ns3:Value>ns3:Receiver</ns3:Value>
      </ns3:Code>
      <ns3:Reason>
        <ns3:Text xml:lang="en">The volume EL0010's Availability Status is Unavailable</ns3:Text>
      </ns3:Reason>
      <ns3:Detail>
        <ns2:ValidationFailedException xmlns:ns2="http://sb.ejb.ws.mds.ltfsle.oracle/">
          <message>The volume EL0010's Availability Status is Unavailable</message>
        </ns2:ValidationFailedException>
      </ns3:Detail>
    </ns3:Fault>
  </S:Body>
</S:Envelope>
```

## unassignVolume

This command unassigns a volume from LTFS-LE.

### Format

```
unassignVolume(String[] volsers)
```

#### *Pre-conditions:*

Verify that the volume exists in the library, is offline, assigned, in a pool, and available before unassigning the volume. To do this:

- Obtain the volume status:

```
getVolumeInfoByVolser([volser1, volser2, ...])
```

- Ensure that the volume is in all required states and take corrective action, if necessary, before proceeding.

*LTFS-LE Web Service to call:*

```
unassignVolume([volser1, volser2, ...])
getJobStatus(jobId) < poll each 10s until status is complete >
getJobResult(jobId)
```

*Post-condition:*

Once the series of Web Services calls have been executed, the results are an XML document. You must convert them into appropriate return codes. If the API calls succeed, the volume is unassigned from LTFS-LE.

### Parameters

- *volser*s are the volume serial numbers of the volume or volumes to be unassigned. The input parameters are an array of strings and enable multiple volumes to be unassigned.
- *jobId* is the job ID used to retrieve the job status and result.

### Response

Valid responses are shown in the SOAP Response example. Refer to the Glossary for definitions.

### Roles

Admin or Service

### Exception Thrown

ValidationFailedException

Exceptions can be thrown:

- If one of or more volser serial numbers are not entered, the exception thrown is: The parameter *volser*s is null.  
The *volser* is the first volume that does not meet any constraint (not all the volumes specified by *volser*s).
- If the volume does not exist in LTFS-LE, the exception thrown is: The volume *<volser>* is not in the LTFS-LE Database.  
Example: The volume EL0010 is not in the LTFS-LE Database.
- If the volume status is not Available, Assigned, Offline, and Idle, the exceptions thrown are:
  - The volume *<volser>*'s Availability Status is *<volser's real availableStatus>*. The volume's Availability Status must be Available to be unassigned.  
Example: The volume EL0010's Availability Status is Unavailable.
  - The volume *<volser>*'s LTFS-LE Assignment Status is *<volser's real assignedStatus>*. The volume's LTFS-LE Assignment Status must be Assigned to be unassigned.

**Example:** The volume EL0010's LTFS-LE Assignment Status is Unassigned.

- The volume `<volser>`'s Status is `<volser's real onlineStatus>`. The volume's Status must be Offline to be unassigned.

**Example:** The volume EL0010's Status is Online.

- The volume `<volser>`'s Inuse Status is `<volser's real inuseStatus>`. The volume's Inuse Status must be Idle to be unassigned.

**Example:** The volume SL0010's Inuse Status is In Use.

## Recommended Timeout

7200 seconds

## Recommended Job Poll Interval

10 seconds

## Job Error

Failed to unassign volume `<volser>`.

## SOAP Request

```
<env:Envelope xmlns:env="http://schemas.xmlsoap.org/soap/envelope/" xmlns:ns1
="http://sb.ejb.ws.mds.ltfsle.oracle/">
  <env:Header>
    <ns1:Security>
      <ns1:UsernameToken>
        <ns1:Username>username</ns1:Username>
        <ns1:Password>password</ns1:Password>
      </ns1:UsernameToken>
    </ns1:Security>
  </env:Header>
  <env:Body>
    <ns1:unassignVolume>
      <arg0>F50233</arg0>
      <arg0>F50234</arg0>
    </ns1:unassignVolume>
  </env:Body>
</env:Envelope>
```

## SOAP Response

```
<?xml version = '1.0' encoding = 'UTF-8'?>
<S:Envelope xmlns:S="http://www.w3.org/2003/05/soap-envelope">
  <S:Header>
    <wsse:Security xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
secext-1.0.xsd" S:mustUnderstand="true">
      <wsu:Timestamp xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd">
        <wsu:Created>2014-06-17T14:36:36Z</wsu:Created>
        <wsu:Expires>2014-06-17T14:37:36Z</wsu:Expires>
      </wsu:Timestamp>
    </wsse:Security>
  </S:Header>
  <S:Body>
    <ns2:unassignVolumeResponse xmlns:ns2="http://sb.ejb.ws.mds.ltfsle.oracle/">
      <return>1221</return>
    </ns2:unassignVolumeResponse>
  </S:Body>
</S:Envelope>
```



```

    </ns2:unassignVolumeResponse>
  </S:Body>
</S:Envelope>

```

## SOAP Fault

```

<?xml version = '1.0' encoding = 'UTF-8'?>
<S:Envelope xmlns:S="http://www.w3.org/2003/05/soap-envelope">
  <S:Body>
    <ns3:Fault xmlns:ns2="http://schemas.xmlsoap.org/soap/envelope/" xmlns:ns3
="http://www.w3.org/2003/05/soap-envelope">
      <ns3:Code>
        <ns3:Value>ns3:Receiver</ns3:Value>
      </ns3:Code>
      <ns3:Reason>
        <ns3:Text xml:lang="en">The volume EL0010's Inuse Status is In Use</ns3:Text>
      </ns3:Reason>
      <ns3:Detail>
        <ns2:ValidationFailedException xmlns:ns2="http://sb.ejb.ws.mds.ltfsle.oracle/">
          <message>The volume EL0010's Inuse Status is In Use</message>
        </ns2:ValidationFailedException>
      </ns3:Detail>
    </ns3:Fault>
  </S:Body>
</S:Envelope>

```

## unformatVolume

This asynchronous `unformatVolume` call removes the LTFS format from the volume. When an LTFS volume is unformatted, the LTFS format headers (metadata and data partitions) are removed from the volume and the volume becomes a standard, non-LTFS volume.

The LTFS-LE Web Service does not require confirmation nor does it perform any validation of the volume. However, when a volume is unformatted, all previous data on the volume is deleted. You must be aware of the volume that is being unformatted and ensure that the correct volume is unformatted.

### *Pre-condition:*

Verify that the volume is offline, unassigned, available, and idle before unformatting the volume. Check that there is a compatible drive available.

To do this:

- Obtain the volume status:
 

```
getVolumeInfoByVolser([volser1, volser2, ...])
```
- Ensure that the volume is in the state required by the pre-condition.
- Take corrective actions, if necessary, before proceeding.

### *LTFS-LE Web Service chain to call:*

```

unformatVolume(volser)
getJobStatus(jobId) < poll each 10s until status is complete >
getJobResult(jobId)

```

### *Post-condition:*

Once the series of Web Services calls have been executed, the results are an XML document. You must convert them into appropriate return codes. If the API calls

succeed, the volume is LTFS-LE no longer formatted and the LTFS-LE format has been removed from the volume.

## Format

`unformatVolume(String volser)`

## Parameters

- `volser` is the serial numbers of the volume to be unformatted.
- `jobId` is the job ID used to retrieve the job status and result.

## Response

Valid responses are shown in the SOAP Response example. Refer to the Glossary for definitions.

## Roles

Admin or Service

## Exception Thrown

`ValidationFailedException`

Exceptions can be thrown:

- If the `volser` is not entered, the exception thrown is: The parameter `volser` is null.
- If the volume does not exist in LTFS-LE, the exception thrown is: The volume `<volser>` is not in the LTFS-LE Database.  
Example: The volume EL0010 is not in the LTFS-LE Database.
- If the volume status is not Available, Unassigned, Offline, and Idle, the exceptions thrown are:
  - The volume `<volser>`'s Availability Status is `<volser's real availableStatus>`. The volume's Availability Status must be Available to be unformatted.  
Example: The volume EL0010's Availability Status is Unavailable.
  - The volume `<volser>`'s LTFS-LE Assignment Status is `<volser's real assignedStatus>`. The volume's LTFS-LE Assignment Status must be Unassigned to be unformatted.  
Example: The volume EL0010's LTFS-LE Assignment Status is Assigned.
  - The volume `<volser>`'s Status is `<volser's real onlineStatus>`. The volume's Status must be Offline to be unformatted.  
Example: The volume EL0010's Status is Online.
  - The volume `<volser>`'s Inuse Status is `<volser's real inuseStatus>`. The volume's Inuse Status must be Idle to be unformatted.  
Example: The volume SL0010's Inuse Status is In Use.
- If the volume is in the default volume pool, the exception thrown is: The volume `<volser>` is in the Default Volume Pool. The volume must be removed from the Default Volume Pool before it can be unformatted.

Example: The volume EL0010 is in the Default Volume Pool.

### Recommended Timeout

7200 seconds

### Recommended Job Poll Interval

10 seconds

### Job Error

Failed to unformat volume <volser>.

### SOAP Request

```
<?xml version = '1.0' encoding = 'UTF-8'?>
<env:Envelope xmlns:env="http://schemas.xmlsoap.org/soap/envelope/" xmlns:ns1
="http://sb.ejb.ws.mds.ltfsle.oracle/">
  <env:Header>
    <ns1:Security>
      <ns1:UsernameToken>
        <ns1:Username>username</ns1:Username>
        <ns1:Password>password</ns1:Password>
      </ns1:Security>
    </env:Header>
    <env:Body>
      <ns1:unformatVolume>
        <arg0>F50344</arg0>
      </ns1:unformatVolume>
    </env:Body>
  </env:Envelope>
```

### SOAP Response

```
<?xml version = '1.0' encoding = 'UTF-8'?>
<S:Envelope xmlns:S="http://www.w3.org/2003/05/soap-envelope">
  <S:Header>
    <wsse:Security
xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
secext-1.0.xsd" S:mustUnderstand="true">
      <wsu:Timestamp xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd">
        <wsu:Created>2014-06-17T14:36:36Z</wsu:Created>
        <wsu:Expires>2014-06-17T14:37:36Z</wsu:Expires>
      </wsu:Timestamp>
    </wsse:Security>
  </S:Header>
  <S:Body>
    <ns2:unformatVolumeResponse xmlns:ns2="http://sb.ejb.ws.mds.ltfsle.oracle/">
      <return>1216</return>
    </ns2:unformatVolumeResponse>
  </S:Body>
</S:Envelope>
```

### SOAP Fault

```
<?xml version = '1.0' encoding = 'UTF-8'?>
<S:Envelope xmlns:S="http://www.w3.org/2003/05/soap-envelope">
  <S:Body>
    <ns3:Fault xmlns:ns2="http://schemas.xmlsoap.org/soap/envelope/" xmlns:ns3
```

```
= "http://www.w3.org/2003/05/soap-envelope">
  <ns3:Code>
    <ns3:Value>ns3:Receiver</ns3:Value>
  </ns3:Code>
  <ns3:Reason>
    <ns3:Text xml:lang="en">The volume EL0010's inuseStatus is In Use</ns3:Text>
  </ns3:Reason>
  <ns3:Detail>
    <ns2:ValidationFailedException xmlns:ns2="http://sb.ejb.ws.mds.ltfsle.oracle/">
      <message>The volume EL0010's inuseStatus is In Use</message>
    </ns2:ValidationFailedException>
  </ns3:Detail>
</ns3:Fault>
</S:Body>
</S:Envelope>
```

## unformatVolumeWithDrive

This asynchronous call removes the LTFS format from the volume. When an LTFS volume is unformatted, the LTFS format headers (metadata and data partitions) are removed from the volume and the volume becomes a standard, non-LTFS volume.

### Format

`unformatVolumeWithDrive (String volser, String drivesn)`

#### Pre-condition:

The volume and drive must exist in library. The volume and drive must be compatible. The volume must be unassigned, offline, not in a pool, and available. The drive must be in an idle, online, or available state.

#### To do this:

- Obtain the volume status:  
`getVolumeInfoByVolser ([volser1, volser2, ...])`
- Ensure that the volume is in the state required by the precondition.
- Take corrective actions, if necessary, before proceeding.

#### LTFS-LE Web Service chain to call:

```
unformatVolumeWithDrive(volser, drivesn)
getJobStatus(jobId) < poll each 10s until status is complete >
getJobResult(jobId)
```

#### Post-condition:

Once the series of Web Services calls have been executed, the results are an XML document. You must convert them into appropriate return codes. If the API calls succeed, the volume is now LTFS-LE formatted.

### Parameters

- `volser` is the volume serial number to be unformatted.
- `drivesn` is the drive serial number that is a string of characters and numbers,
- `jobId` is the job ID used to retrieve the job status and result.

## Response

Valid responses are shown in the SOAP Response example. Refer to the Glossary for definitions.

## Exception Thrown

ValidationFailedException

Exceptions that can be thrown:

### Volumes:

- If the volser serial number is not entered, the exception thrown is: The parameter `volser` is null.
- If the volume does not exist in LTFS-LE, the exception thrown is: The volume `<volser>` is not in the LTFS-LE Database.

Example: The volume EL0010 is not in the LTFS-LE Database.

- If the volume status is not Available, Unassigned, Offline, and Idle, the exceptions thrown are:

- The volume `<volser>`'s Availability Status is `<volser's real availableStatus>`. The volume's Availability Status must be Available to be unformatted.

Example: The volume EL0010's Availability Status is Unavailable.

- The volume `<volser>`'s LTFS-LE Assignment Status is `<volser's real assignedStatus>`. The volume's LTFS-LE Assignment Status must be Unassigned to be unformatted.

Example: The volume EL0010's LTFS-LE Assignment Status is Assigned.

- The volume `<volser>`'s Status is `<volser's real onlineStatus>`. The volume's Status must be Offline to be unformatted.

Example: The volume EL0010's Status is Online.

- The volume `<volser>`'s Inuse Status is `<volser's real inuseStatus>`. The volume's Inuse Status must be Idle to be unformatted.

Example: The volume SL0010's Inuse Status is In Use.

- If the volume is in the default volume pool, the exception thrown is: The volume `<volser>` is in the Default Volume Pool. The volume must be removed from the Default Volume Pool before it can be unformatted.

Example: The volume EL0010 is in the Default Volume Pool.

### Drives:

- If the drive serial number is not entered, the exception thrown is: The parameter `driveSN` is null.
- If the drive does not exist in LTFS-LE, the exception thrown is: The drive `<driveSN>` is not in the LTFS-LE Database.

Example: The drive 123456 is not in the LTFS-LE Database.

- If the drive status is not Available, Assigned, and Online the exceptions thrown are:

- The drive `<driveSN>`'s Availability Status is `<drive's real availableStatus>`. The drive must be Available to be used.

Example: The drive 123456's Availability Status is Unavailable.

- The drive <driveSN>'s LTFS-LE Assignment Status is <drive's real assignedStatus>. The drive must be Assigned to be used.

Example: The drive 123457's LTFS-LE Assignment Status is Unassigned.

- The drive <driveSN>'s Status is <drive's real onlineStatus>. The drive must be Online to be used.

Example: The drive 123456's Status is Offline.

### Volume Type and Drive Type Incompatibility:

- If the volume type and drive type are not compatible, the exception thrown is: The volume <volser>'s type <vol\_type> is not compatible with the drive <driveSN>'s type <drive\_type>.

Example: The volume EL0010's type LTO-5 is not compatible with drive 123456's type T1C.

### Roles

Admin or Service

### Recommended Timeout

7200 seconds

### Recommended Job Poll Interval

10 seconds

### Job Errors

Failed to unformat volume <volser>.

### SOAP Request

```
<?xml version = '1.0' encoding = 'UTF-8'?>
<env:Envelope xmlns:env="http://schemas.xmlsoap.org/soap/envelope/" xmlns:ns1
="http://sb.ejb.ws.mds.ltfsle.oracle/">
  <env:Header>
    <ns1:Security>
      <ns1:UsernameToken>
        <ns1:Username>username</ns1:Username>
        <ns1:Password>password</ns1:Password>
      </ns1:UsernameToken>
    </ns1:Security>
  </env:Header>
  <env:Body>
    <ns1:unformatVolumeWithDrive>
      <arg0>F50344</arg0>
      <arg1>576001000421</arg1>
    </ns1:unformatVolumeWithDrive>
  </env:Body>
</env:Envelope>
```

### SOAP Response

```
<?xml version = '1.0' encoding = 'UTF-8'?>
<S:Envelope xmlns:S="http://www.w3.org/2003/05/soap-envelope">
  <S:Header>
```

```

    <wsse:Security
xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
secext-1.0.xsd" S:mustUnderstand="true">
    <wsu:Timestamp      xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd">
        <wsu:Created>2014-06-17T14:36:36Z</wsu:Created>
        <wsu:Expires>2014-06-17T14:37:36Z</wsu:Expires>
    </wsu:Timestamp>
    </wsse:Security>
</S:Header>
<S:Body>
    <ns2:unformatVolumeWithDriveResponse xmlns:ns2="http://sb.ejb.ws.mds.ltfsle.oracle/">
        <return>1216</return>
    </ns2:unformatVolumeWithDriveResponse>
</S:Body>
</S:Envelope>

```

### SOAP Fault

```

<?xml version = '1.0' encoding = 'UTF-8'?>
<S:Envelope xmlns:S="http://www.w3.org/2003/05/soap-envelope">
    <S:Body>
        <ns3:Fault xmlns:ns2="http://schemas.xmlsoap.org/soap/envelope/"  xmlns:ns3
="http://www.w3.org/2003/05/soap-envelope">
            <ns3:Code>
                <ns3:Value>ns3:Receiver</ns3:Value>
            </ns3:Code>
            <ns3:Reason>
                <ns3:Text xml:lang="en">The volume EL0010's type LTO-5 is not compatible with drive
123456's type T1C</ns3:Text>
            </ns3:Reason>
            <ns3:Detail>
                <ns2:ValidationFailedExceptionxmlns:ns2="http://sb.ejb.ws.mds.ltfsle.oracle/">
                    <message>The volume EL0010's type LTO-5 is not compatible with drive 123456's type
T1C</message>
                </ns2:ValidationFailedException>
            </ns3:Detail>
        </ns3:Fault>
    </S:Body>
</S:Envelope>

```

## vaultVolume

This API function vaults one or more LTFS-LE volumes that are present in the LTFS-LE default volume pool. You can only vault one volume at a time.

This asynchronous call vaults up to 30 volumes using the CAP (mailslot). There is room for four volumes on the mailslot. If the user vaults 30 volumes, then volumes are sent to the mailslot four at a time until all ejects are complete.

### Format

```
vaultVolume(String[] volsers, String acs, String lsm, String cap)
```

#### *Pre-condition:*

Verify that a CAP is available and that the volume is available, assigned, offline, idle, and in a pool before vaulting. The total number of volumes should be less than or equal to 30. Volumes must be in the same library configured. To do this:

- Obtain the volume status:

```
getVolumeInfoByVolser([volser1,volser2,...])
```

- If the volume is in an online assigned state, finalize the volume:

```
finalizeVolume(volser)
getJobStatus(jobId) < poll each 10s until status is complete >
getJobResult(jobId)
```

- If the volume is online, take the volume offline before executing the vaultVolume API call:

```
offlineVolume(volser)
getJobStatus(jobId) < poll each 10s until status is complete >
getJobResult(jobId)
```

*LTFS-LE WebService chain to call:*

```
vaultVolume([volser1,volser2,...],acs,lsm,cap)
getJobStatus(jobId) < poll each 10s until status is complete >
getJobResult(jobId)
```

*Post-condition:*

Once the series of Web Services calls have been executed, the results are an XML document. You must convert them into appropriate return codes. If the API calls succeed, the volume is entered into the library. The user must walk up to the library and retrieve the volume after opening the CAP door.

### Parameters

- *volser*s are the serial number of the volume or volumes to be vaulted.
- *acs* is the library ACS identifier.
- *lsm* is the library LSM identifier.
- *cap* is the library CAP (mailslot) ID used to eject the volume.
- *jobId* is the job ID used to retrieve the job status and result.

### Response

Valid responses are shown in the SOAP Response example. Refer to the Glossary for definitions.

### Roles

Admin or Service

### Exception Thrown

ValidationFailedException

Exceptions can be thrown:

- If one or more volser serial numbers are not entered, the exception thrown is: The parameter *volser*s is null.  
The *volser* is the first volume that does not meet any constraint (not all the volumes specified by *volser*s).
- If the number of volumes specified is greater than 30, the exception thrown is: The number of volumes to vault must be 30 or less.



- If the `acs` parameter is null, the exception thrown is: The parameter `acs` is null.
- If the `lsm` parameter is null, the exception thrown is: The parameter `lsm` is null.
- If the `cap` parameter is null, the exception thrown is: The parameter `cap` is null.
- If the volume does not exist in LTFS-LE, the exception thrown is: The volume `<volser>` is not in the LTFS-LE Database.

Example: The volume EL0010 is not in the LTFS-LE Database.

- If the volume status is not Available, Assigned, Offline, and Idle, the exceptions thrown are:
  - The volume `<volser>`'s Availability Status is `<volser's real availableStatus>`. The volume's Availability Status must be Available to vault the volume.
 

Example: The volume EL0010's Availability Status is Unavailable.
  - The volume `<volser>`'s LTFS-LE Assignment Status is `<volser's real assignedStatus>`. The volume's LTFS-LE Assignment Status must be Assigned to vault the volume.
 

Example: The volume EL0010's LTFS-LE Assignment Status is Unassigned.
  - The volume `<volser>`'s Status is `<volser's real onlineStatus>`. The volume's Status must be Offline to vault the volume.
 

Example: The volume EL0010's Status is Online.
  - The volume `<volser>`'s Inuse Status is `<volser's real inuseStatus>`. The volume's Inuse Status must be Idle to vault the volume.
 

Example: The volume SL0010's Inuse Status is In Use.

### Recommended Timeout

This depends on the time taken for the admin to remove the volume from the CAP (mailslot).

### Recommended Job Poll Interval

10 seconds

### Job Errors

Failed to vault volume `<volser>`.

### SOAP Request

```
<?xml version = '1.0' encoding = 'UTF-8'?>
<env:Envelope xmlns:env="http://schemas.xmlsoap.org/soap/envelope/" xmlns:ns1
="http://sb.ejb.ws.mds.ltfsle.oracle/">
  <env:Header>
    <ns1:Security>
      <ns1:UsernameToken>
        <ns1:Username>username</ns1:Username>
        <ns1:Password>password</ns1:Password>
      </ns1:UsernameToken>
    </ns1:Security>
  </env:Header>
</env:Envelope>
```

```

</env:Header>
<env:Body>
  <ns1:vaultVolume>
    <arg0>F50234</arg0>
    <arg0>F50233</arg0>
    <arg1>3</arg1>
    <arg2>0</arg2>
    <arg3>1</arg3>
  </ns1:vaultVolume>
</env:Body>
</env:Envelope>

```

## SOAP Response

```

<?xml version = '1.0' encoding = 'UTF-8'?>
<S:Envelope xmlns:S="http://www.w3.org/2003/05/soap-envelope">
  <S:Header>
    <wsse:Security
xmlns:wsse="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
-secext-1.0.xsd" S:mustUnderstand="true">
      <wsu:Timestamp xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-
wssecurity-utility-1.0.xsd">
        <wsu:Created>2014-06-17T14:36:36Z</wsu:Created>
        <wsu:Expires>2014-06-17T14:37:36Z</wsu:Expires>
      </wsu:Timestamp>
    </wsse:Security>
  </S:Header>
  <S:Body>
    <ns2:vaultVolumeResponse xmlns:ns2="http://sb.ejb.ws.mds.ltfsle.oracle/">
      <return>1214</return>
    </ns2:vaultVolumeResponse>
  </S:Body>
</S:Envelope>

```

## SOAP Fault

```

<?xml version = '1.0' encoding = 'UTF-8'?>
<S:Envelope xmlns:S="http://www.w3.org/2003/05/soap-envelope">
  <S:Body>
    <ns3:Fault xmlns:ns2="http://schemas.xmlsoap.org/soap/envelope/" xmlns:ns3
="http://www.w3.org/2003/05/soap-envelope">
      <ns3:Code>
        <ns3:Value>ns3:Receiver</ns3:Value>
      </ns3:Code>
      <ns3:Reason>
        <ns3:Text xml:lang="en">The volume EL0010's Inuse Status is In Use</ns3:Text>
      </ns3:Reason>
      <ns3:Detail>
        <ns2:ValidationFailedExceptionxmlns:ns2="http://sb.ejb.ws.mds.ltfsle.oracle/">
          <message>The volume EL0010's Inuse Status is In Use</message>
        </ns2:ValidationFailedException>
      </ns3:Detail>
    </ns3:Fault>
  </S:Body>
</S:Envelope>

```

---

---

# Glossary

## **acs**

Automated Cartridge System This is the library acs identifier.

## **ACSLs**

Oracle's StorageTek Automated Cartridge System Library Software. This software manages contents of multiple StorageTek tape libraries and controls library hardware to mount and dismount cartridges on StorageTek tape drives.

ACSLs assigns resources to LTFS-LE as if it were a logical library. LTFS-LE then issues commands to ACSLS to read and write files, much in the same way that backup software does today.

## **acslsPath**

A string represents the ACSLS path of the volume (acs,lsm,panel,row,column) or drive (acs,lsm,panel,drive).

## **API**

Application Programming Interface. A source code interface that an operating system or library provides to support requests for services to be made of it by computer programs. It facilitates the exchange of data between two or more software applications and extends the capabilities of the software.

## **assignedStatus**

Volume or drive assigned status to LTFS LE. Valid assigned status values are: "Assigned" and "Unassigned".

## **availableStatus**

Volume or drive availability status. Valid availability statuses are:

- Volume - "Available" and Unavailable"
- Drive - "Available", "Unavaiable", and "In Use"

## **CAP (mailslot)**

Cartridge Access Port (mailslot in the SL150). A bidirectional port built into the door panel of an LSM, which provides for the manual entry or automatic ejection of data or cleaning cartridges.

## **capacityAvailable**

The capacity available on this volume in Bytes

**capacityUsed**

The capacity used on this volume in Bytes.

**currentLocation**

The volume current location. Valid values of currentLocation, such as "home", "enter", "mounting", "dismounting".

**driveCompress**

Whether or not the drive is in compression mode. Valid values for are: Compression Enabled or Compression Not Enabled.

**driveEncry**

Whether or not the drive is in encrypted mode. Valid values are: Encryption Enabled or Encryption Not Enabled.

**driveFwRev**

Drive firmware revision.

**driveGenDev**

This gives the location of the device in the operating system, /dev/sg6.

**driveLtfsCap**

Whether or not the drive meets LTFS compliance. Valid values are: "LTFS Capable" or "Not Capable"

**drivePoolName**

The drive pool name to which this drive belongs. Valid values are: empty value "" or "Default Drive Pool".

**driveSn**

The drive serial number. It is a string of characters and numbers.

**driveStatus**

Notifies if the drive is assigned to LTFS LE. Valid drive statuses are "Assigned" and "Unassigned".

**driveStdDev**

This gives the location of the device in the operating system, /dev/st0.

**driveType**

The type of the drive, such as T10000C, T10000D, HP-LTO5, IBM-LTO5, HP-LTO6, IBM-LTO6.

**driveVendor**

The name of drive vendor, such as "STK", "IBM", "HP".

**inuseStatus**

Valid statuses are "In use" and "Idle".

**jobCompleOn**

The job execution complete time.

**jobDetail**

Array of strings that contains the return information when the Web service call successfully completes. The valid values are an empty value or strings returned by the Web service call.

**jobErrors**

Array of strings that contain the return information when the Web service call has failed. It is an empty value if the Web service call is successful.

**jobId**

The job ID returned by a management and file WebService call to indicate the running web service job.

**job poll interval**

The recommended polling interval and timeout values for each individual Web Service call. Default is 10 seconds.

**jobStatus**

The `jobStatus` from `getJobResult` is either "Success" or "Failed"

**jobUser**

The user name, such as `ltfsledmin`.

**LSM**

Library Storage Module. An ACS structure that provides the storage area for cartridges, cartridge drives, CAPs, and the robot necessary for moving them.

**ltfsUUID**

The UUID (Universally Unique Identifier) of a volume is a string of characters and numbers that is created when the volume is formatted using the `mkltfs` command.

**onlineStatus**

Valid volume or drive status in LTFS-LE is "Online" or "Offline".

**pool**

A collection of tape cartridges having one or more similar features or attributes, such as a pool of data tapes.

**roles**

- Admin

This role is reserved for the system administrator responsible for configuring and managing the LTFS-LE system. It provides access to all LTFS-LE system facilities.

- Monitor

This role applies to a basic user who wants to monitor drives, volumes, and system resources. It provides limited access to the LTFS-LE system, with read-only permissions

- Service

This role applies to the Service Representative. It provides Administrator level access, but uses login credentials that are specific to Services. This role may include additional command-line interface (CLI) capabilities for servicing procedures.

**serviceCalled**

The status or management service called, such as assignVolume.

**volume**

A data or cleaning cartridge.

**volumeCreatedOn**

The time when this volume is discovered in the library. The time format is epoch time.

**volumePoolName**

The volume pool name to which this volume belongs. Valid values for volumePoolName are empty value " " or "Default Volume Pool".

**volser**

The volume serial number. A synonym for external label identifier.

**volumeType**

The type of the volume, such as LTO-1.5T, LTO-2.5T, T10000T2.

## A

---

### API

- accessing the WSDL, 2-1
- functions, 2-1
- signatures, 2-2

## D

---

### drive status services

- getAllDriveInfo, 3-3, 3-4

## G

---

### general status services

- getJobResult, 3-1
- getJobStatus, 3-3

## M

---

### management services

- assignVolume, 4-1, 4-7
- ejectVolume, 4-4
- enterVolume, 4-7
- finalizeVolume, 4-9
- formatVolume, 4-12
- formatVolumeWithDrive, 4-15
- getVolumePath, 4-18
- offlineVolume, 4-20
- onlineVolume, 4-23
- prepareVolume, 4-26
- unassignVolume, 4-28
- unformatVolume, 4-31
- unformatVolumeWithDrive, 4-34
- vaultVolume, 4-37

## V

---

### volume status services

- getAllVolumeInfo, 3-5
- getVolumeInfoByStatus, 3-7
- getVolumeInfoByVolser, 3-9

