Lyris ListManager API

User Manual

Table of Contents

Introduction to the Application Programming Interface (API)

ListManager allows administrators to access and manipulate the database tables using SOAP (Simple Object Access Protocol). SOAP provides a way to communicate between applications running on different operating systems, with different technologies and programming languages. SOAP is a key element of Microsoft's .NET architecture for future Internet application development. SOAP provides a higher-level interface to database information so that no knowledge of database query protocols is required.

ListManager's API can be invoked from any language with SOAP support, including:

.NET
Java
Php
Tcl
Ruby

Examples of ListManager's functions in different languages are available in your ListManager's soap directory:

http://yourlistmanagerurl/soap

Or, you may navigate to Utilities: Other: Programming API.

To use the ListManager API effectively, it is important to understand the information stored in each table column. Many functions require modification of record fields in order to work properly. It is also important to understand the possible side effects of modifying these fields. For example, if you set a member's 'MemberType' attribute to 'unsub', then that member's record may be deleted if the list is set up to purge unsubscribed members.

Matching API Versions to ListManager Versions

The ListManager API is updated separately from ListManager; as a result, API updates don't necessary correspond directly to ListManager updates. Here is a list showing API versions and which version(s) of ListManager they work with:

ListManager Version	API Version(s) Available
9.0a and earlier	1.4.2 (default status: enabled)
9.0b	1.4.2 (default status: enabled) 1.5 beta
9.1	1.4.2 (default status: disabled) 1.5.1
9.2 (1Q- 2007)	1.4.2 (default status: disabled) 1.5.2
9.3	1.4.2 (default status: disabled) 1.5.2a
10.0	10.0 (**NOTE**: as of ListManager 10.0, the API version matches the program version.)
10.1	10.0
10.2	10.0
11.0	11.0
11.1	11.1

Access Rights

Starting with ListManager 10.0, a user has to be a list admin of at least one list in order to use API functions (i.e., all API functions require authentication). Some administration functions may require additional information or administrative access beyond the list administrator level. An http 401 (authentication) error will be returned if the user does not have privilege to access a particular function.

Name of the function	Version	List	Site	Server	Listname Required	Return Value	Data Type
ApiVersion	LM 8.1	●	●	●		Current API version	string
CheckMember Password	LM 8.1	●	●	●	✓	True if password match member	bool
CopyMember	LM 8.1	●	●	●	✓	Member ID of the new copied member created	int
CreateContent	LM 10.0	●	●	●		Returns an ID number of a content created or raises error with a verbose message	int
CreateList	LM 8.1		●	●	✓	New List ID	int
CreateListAdmin	LM 8.1	●	●	●	✓	New Member ID	int
CreateMany Members	LM 8.1	●	●	●	✓	Number of members created	int
CreateMemberBan	LM 8.1	●	●	●	✓	New member ID	int
CreateMember Column	LM 9.1 / 1.5.1			●		Success flag – true if update successful	bool
CreateSegment	LM 9.1 / 1.5.1	●	●	●	✓	New segment ID	int
CreateServer Admin	LM 10.0			●		Returns an ID number of a server admin created or raises error with a vergose message	int
CreateSingleMember	LM 8.1	●	●	●	✓	New Member ID	int
CreateSite	LM 10.0			●		ID of the site created	int

Name of the function	Version	List	Site	Server	Listname Required	Return Value	Data Type
CreateSiteAdmin	LM 10.0		●	●		Returns an Id number of a server admin created or raises error with a verbose message	int
CreateTopic	LM 10.0		●	●		Returns true if succeeds, raises error with a vergose message otherwise	bool
CurrentUserEmailAddress	LM 8.1	●	●	●		User's e-mail address	string
DeleteContent	LM 10.0	●	●	●		Returns true if succeeds, raises error with a verbose message otherwise	bool
DeleteList	LM 8.1		●	●	✓	Success flag--true if update successful	bool
DeleteMemberColumn	LM 9.1 / 1.5.1			●		Success flag – true if update successful	bool
DeleteMembers	LM 8.1	●	●	●	✓	Number of members successfully deleted	int
DeleteSegment	LM 9.1 / 1.5.1	●	●	●	✓	Success flag – true if update successful	bool
DeleteServerAdmin	LM 10.0			●		Returns true if succeeds, raises error with a verbose message otherwise	bool
DeleteSite	LM 10.0		●	●		Returns true if succeeds, raises error with a verbose message otherwise	bool
DeleteSiteAdmin	LM 10.0		●	●		Returns true if succeeds, raises error with a verbose message otherwise	bool
DeleteTopic	LM 10.0		●	●		Returns true if succeeds, raises	bool

Name of the function	Version	List	Site	Server	Listname Required	Return Value	Data Type
						error with a verbose message otherwise	
EmailOnWhatLists	LM 8.1	●	●	●		A list of lists which contain given e-mail address	ArrayOfString
EmailPasswordOnWhat Lists	LM 8.1	●	●	●		A list of lists which contain given e-mail address	ArrayOfString
GetEmailFromMember ID	LM 8.1	●	●	●		E-mail corresponding to the ID	string
GetListID	LM 8.1	●	●	●	✓	List ID	int
GetListnameFrom MemberID	LM 8.1	●	●	●		ListName a member ID belongs to	string
GetMemberID	LM 8.1	●	●	●		Member ID	int
GetPreviewMailing	LM 10.0	●	●	●		Returns a merged text as string or raises error with verbose message	string
ImportContent	LM 8.1	●	●	●		Content corresponding to the ID given	SimpleMailingStruct
MailingStatus	LM 8.1	●	●	●		Transact log for the given mailing ID	string
ModerateMailing	LM 8.1	●	●	●	✓	Mailing ID of the scheduled mailing	int
ScheduleMailing	LM 8.1	●	●	●	✓	Mailing ID of the scheduled mailing	int
SelectContent	LM 8.1	●	●	●		An array of content matching criteria	ArrayOf ContentStruct
SelectLists	LM 8.1	●	●	●		An array of list struct matching criteria	ArrayOfListStruct
SelectListsEx	LM 10.0	●	●	●		Result set from the query selecting lists or NULL if no	ArrayOfArray OfString

Name of the function	Version	List	Site	Server	Listname Required	Return Value	Data Type
						records returned. First row contains column names which should be mapped to a more readable form.	
SelectMembers	LM 8.1	●	●	●	✓	An array of members matching criteria	ArrayOf MemberStruct
SelectSegments	LM 8.1	●	●	●	✓	Array of segment struct matching criteria	ArrayOf SegmentStruct
SelectSimpleMembers	LM 8.1	●	●	●	✓	An array of members matching criteria	ArrayOfSimple MemberStruct
SendMailing	LM 8.1	●	●	●	✓	Mailing ID of the sent mailing	int
SendMailingDirect	LM 8.1	●	●	●	✓	Mailing ID of the scheduled mailing	int
SendMemberDoc	LM 8.1	●	●	●		Mailing ID of the sent document to member	int
SendMessage	LM 9.1 / 1.5.1	●	●	●	✓	ID of the Message sent	int
SqlDelete	LM 8.1			●		Success flag--true if update successful	bool
SqlInsert	LM 8.1			●		Identity of the row created	int
SqlSelect	LM 8.1			●		Result set from the query or NULL if no records returned. First row contains column names.	ArrayOfArrray OfString
SqlUpdate	LM 8.1			●		Success flag--true if update successful	bool
TrackingSummary	LM 8.1	●	●	●		Tracking summary requested for given	Tracking SummaryStruct

Name of the function	Version	List	Site	Server	Listname Required	Return Value	Data Type
						outmailID	
UnSubscribe	LM 8.1	●	●	●	✓	Number of members unsubscribed	int
UpdateContent	LM 10.0	●	●	●		Returns true if succeeds, raises error with a verbose message otherwise	bool
UpdateList	LM 8.1		●	●	✓	Success flag – true if update successful	bool
UpdateListAdmin	LM 8.1	●	●	●	✓	Success flag – true if update successful	bool
UpdateMember Demographics	LM 8.1	●	●	●	✓	Success flag – true if update successful	bool
UpdateMemberEmail	LM 8.1	●	●	●	✓	Success flag--true if update successful	bool
UpdateMemberKind	LM 8.1	●	●	●	✓	Success flag – true if update successful	bool
UpdateMember Password	LM 8.1	●	●	●	✓	Success flag--true if update successful	bool
UpdateMemberStatus	LM 8.1	●	●	●	✓	Success flag – true if update successful	bool
UpdateSegment	LM 9.1 / 1.5.1			●	✓	Success flag – true if update successful	bool
UpdateServerAdmin	LM 10.0			●		Returns true if succeeds, raises error with a verbose message otherwise	bool
UpdateSite	LM 10.0		●	●		Returns true if succeeds, raises error with a verbose message otherwise	bool
UpdateSiteAdmin	LM 10.0		●	●		Returns true if succeeds, raises error with a verbose message otherwise	bool
UpdateTopic	LM 10.0		●	●		Returns true if	bool

Name of the function	Version	List	Site	Server	Listname Required	Return Value	Data Type
						succeeds, raises error with a verbose message otherwise	

SOAP Resources
Overview of SOAP and Web Services (Microsoft)
Microsoft Web Services Developer Center
Java Web Services - Axis
Pocket SOAP
W3C SOAP v 1.2 Specification
SOAP Tutorial (W3Schools)

Troubleshooting

Problem: ListManager Will Not Run

While starting ListManager, it is possible that the API cannot run because Port 82 is taken. If this occurs, the following message displays during startup:

Unable to bind to LMAPI port '82'. LMAPI will be unavailable until this issue is resolved and LM is restarted.

Possible resolutions:

1. Configure your LMAPI 'network settings/port' to use a different port.
2. Free the service currently running on the API port.
3. Run as root OR configure on a port available for non-root users (usually >1024).

After the port issue is resolved, you must restart the server.

Running a Select statement on a large table results in error message: "Hit max iterations setting of 500000. Possible infinite loop."

If you run a SELECT statement on a table that has more than 1,000,000 records, you may see an error message that states: "Hit max iterations of 5000000. Possible infinite loop."

To prevent this error message from appearing, you must change two lines of code in one of your ListManager configuration files:

1. Open the following file in a text editor:

    c:\Program Files\Listmanager\tclweb\lib\lm\lmlib.tcl

2. Find the following text:

```
# this variable is the absolute maximum numbers of iterations through the
for(;;) loop in the TclExecuteByteCode loop
set absoluteMaxIterations 1000000
set defaultMaxIterations 500000
```

3. Change the **set absoluteMaxIterations** amount from 1000000 to 5000000.

4. Change the **set defaultMaxIterations** amount from 500000 to 200000.

5. Save and exit.

Examples of ListManager's functions are available in your ListManager's soap directory:

http://yourlistmanagerurl/soap

Programming Quick Start

To use ListManager's API, follow these steps:

1. Download and Install the SOAP Client for Your Language

2. Point SOAP Toolkit to WSDL URL
To use the API, you must point your SOAP toolkit at your ListManager installation's WSDL URL, located at:

http://yourlistmanagerurl:82/?wsdl

3. Call the API Version
Always call ApiVersion before running any other functions to ensure your client and the ListManager server you are attempting to communicate with are using the same version.

4. Set the Login
The security for the ListManager API is the same as that for the ListManager web interface; the same user name/password must be set in the script.

More Information
The following pages give more detailed and specific instructions for various languages:

PHP
Java
C#.NET
C#.NET 2.0

Examples of ListManager's functions are available in your ListManager's soap directory:

http://yourlistmanagerurl/soap

PHP SOAP Quickstart Guide

You can find example code written in PHP showing the usage of ListManager's API functions in the soap directory available at :
http://yourlistmanagerurl/soap

Your ListManager installation includes the example code as well as the required NuSoap library.

To use the client example code
1. Download your packed example code from http://yourlistmanagerurl/soap/php.zip.
2. Unpack the ZIP file to a web-accessible directory on a PHP-enabled webserver (e.g., Apache, IIS). For example, create a lmapiTest directory in your /var/www/html/ path, and then unpack files there.

NOTES

- The built-in ListManager webservice is not suitable for PHP integration. For assistance with choosing or installing a PHP webserver implementation, please refer to http://www.php.net/
- Packed NuSoap library can also be found at http://yourlistmanagerurl/soap/php/nusoap-0.6.7.zip

Your directory structure after unpacking should look like the one below:
lmapiTest/php/
|
|_ lmapiTest.php
|_ readme.txt
|_ lib/
|_ samples/

3. To run your script, point your browser to the location:

http://yourlistmanagerurl/lmapiTest/php/lmapiTest.php

If you are using php as a standalone, run

php lmapiTest.php

4. The following code snippet is a minimal client script for PHP. To test the code below, change the value "youradminname" to your admin name and the value "yourlyrispassword" to your

password . This code connects you to your local ListManager server and retrieves the current API version.

---------CUT HERE----------

```php
<?php
$wsdl_location = 'http://127.0.0.1:82/?wsdl';
$userName = 'youradminname';
$password = 'yourlyrispassword';

// Pull in the NuSOAP code
if ( PHP_VERSION >= 5 )
    require_once('lib/nusoap_php5.php');
else
    require_once('lib/nusoap_php4.php');

// create client
if ( PHP_VERSION >= 5 )
    $lmapiClient = new nusoapclient( $wsdl_location, true );
else
    $lmapiClient = new soapclient( $wsdl_location, true );

//set basic authentication
$lmapiClient->setCredentials($userName,$password, 'basic');

//make sure there was no error.
$err= $lmapiClient->getError();
if ($err) {
    echo "<h2>Error</h2><pre> $err <hr> $lmapiClient->debug_str;\n\n";
    return false;
    }

$lmapi = $lmapiClient->getProxy();

//set basic authentication
$lmapi->setCredentials($userName,$password, 'basic');

echo "<h3> Current version of API at " . $wsdl_location . " is: " .
$api_version . "</h3>\n";

?>
```

Java SOAP Quickstart Guide

To run the sample client code
1. Download sample client from:

http://yourlistmanagerserver/soap/

2. Unzip client files to a convenient location (c:\lm_java or ~/lm_java)
3. Go to:

http://www.lyris.com/products/listmanager/api/java/

4. Download java dependencies (Supporting Libraries) and unzip in the same location.
5. If you are running on a WINDOWS machine:
    a. Edit set_vars.bat: change the location of JAVA_HOME to the location of the JDK.
    b. Set TEST_HOME to the directory you created in step 1.
    c. Run set_vars.bat.
6. Edit build.xml, change the hostname to your listmanger server name.
7. Make sure ListManager IS running before proceeding.
8. Run **ant gen-stubs**.
9. Run **ant run-client**.
10.There is also a "clean" target that removes all generated code.

* To avoid error messages remove directories jdk/sample/ and jdk/demo/ before executing

Below is a skeleton api class:

```java
----JAVA-----
package lmapi;
import java.util.*;
import org.apache.axis.client.Stub;

public class Client {

    // named after objects dyna-generated by axis
    public static Lmapi service;
    public static LmapiSoap lm;

    // static means class global variables, it does not change between
instances.
    public static String userName ="admin";
    public static String password = "lyris";

    public static void main (String [] args) throws Exception
    {
    // Make a service
    service = new LmapiLocator();
```

```java
        try
        {
            // Get a stub that implements the service operations
            lm = service.getlmapiSoap();
            // Lyris Api uses authorization
            ((Stub)lm).setUsername(userName);
            ((Stub)lm).setPassword(password);

            System.out.println("Current API version: " + lm.apiVersion()
);

            // run code here

        } catch (Exception ex) {
            ex.printStackTrace();
        }
    }
}
----END JAVA-----
```

C#.NET SOAP Quickstart Guide

1. Open Visual studio & create new project.
2. Add Web reference:

 http://YOURSERVER:82/?wsdl

3. Name it lmapiSoap

------C#---------

```csharp
// Initialize ListManager soap object
lm = new lmapiSoap.lmapi();

// set up basic authorization
String username = "admin";
String password = "lyris";
lm.Credentials = new System.Net.NetworkCredential(username, password);

// Display current API version
MessageBox.Show("Current API Version: "+ lm.ApiVersion());

// verify API functions work
MessageBox.Show(lm.CurrentUserEmailAddress());
```

Running Lyris ListManager Client Tests Using C#.NET

1. Unzip project file
2. Open **Lyris Listmanager SOAP API Tests.csproj**.
3. Update your web reference to your server (if your reference is different from http://localhost:82/?wsdl)

http://YOURSERVER:82/?wsdl

4. Name the web reference **ImapiSoap**.
5. You can now run the test suite by choosing **Debug > Start**.

C#.NET 2.0 SOAP Quickstart Guide

1. Open Visual studio 2005 & create new a project.
2. Add Web reference:

http://YOURSERVER:82/?wsdl

3. Name it ImapiSoap.

------C#---------

```csharp
// Initialize listmanager soap object
lm = new lmapiSoap.lmapi();

// setup basic authorization
String username = "admin";
String password = "lyris";
lm.Credentials = new System.Net.NetworkCredential(username, password);

// Display current API version
MessageBox.Show("Current API Version: "+ lm.ApiVersion());

// verify API functions work...
MessageBox.Show(lm.CurrentUserEmailAddress());
```

Running Lyris Listmanager Client Tests Using C#.NET

1. Unzip the project file.
2. Open **Lyris Listmanager SOAP API Tests.csproj**.
3. Update your web reference to your server (if your reference is different from http://localhost:82/?wsdl).

http://YOURSERVER:82/?wsdl

4. Name the web reference **ImapiSoap**.
5. You can now run the test suite by choosing **Debug > Start**.

Function Reference

Member Functions

SelectMembers

Description
Selects an array of MemberStructs using FilterCriteriaArray. Example:

```
ar[0] = (ListName = list1); ar[1] = (domain = example.com)
```

Arguments
FilterCriteriaArray - an array of member field/value pairs (values from MemberStruct)

Returns
An array of members matching criteria

Return Data Type
ArrayOfMemberStruct

Access Rights (learn more)

List	Site	Server	Listname Required
x	x	x	x

Version
ListManager 8.1

SelectMembersEx

Description
This function is similar to the SelectMembers function; however, its extended functionality lets you select only specified fields. The result is significant performance improvement.

Arguments

FieldsToFetch (ArrayOfString)
List of fields from members_ table to be fetched.

FilterCriteriaArray (ArrayOfString)
List of criteria to filter members returned.

Return
Result set from the query selecting lists, or NULL if no records returned. The first row contains column names which should be mapped to a more readable form.

Notes
The purpose of this function is to provide a more flexible and portable solution than the SelectMembers function. The more fields you request the longer the request takes to execute, so you can specify the exact fields you're interested in while retrieving information about existing lists. Note that allowed selectable fields are pre-defined and can be flexibly modified by the server administrator.

FilterCriteriaArray lets you filter out members not matching your criteria.

A list of allowed fields to be selected from members_ table is pre-defined for security reasons but can be extended, limited or changed by manually editing lyrTableFieldMappings.

By default following fields are allowed for FieldsToFetch / FilterCriteria arguments.

DateHeld
DateJoined
Domain
EmailAddress
FullName
IsListAdmin
ListName
MailFormat
MemberID
MemberType
ReceiveAdminMail
UserNameLC

Rights
Only members that a user is allowed access to will be returned. If there are members that match the criteria, but a user is not allowed to administrate any of them, an empty list will be returned.

Return Data Type
ArrayOfArrayOfString

Access Rights (learn more)

List	Site	Server	Listname Required
x	x	x	

 Version
ListManager 10.2a

CreateSingleMember

Description
Creates a single member of a list. CreateSingleMember is easier to use programmatically than CreateManyMembers, as it requires three strings instead of the MemberStructArray that CreateManyMembers requires.

Arguments
EmailAddress - The email address of the member (string)
FullName - The full name of the member (string)
ListName - The name of the list the member will be added to (string)

Returns
New Member ID

Return Data Type
int

Access Rights (learn more)

List	Site	Server	Listname Required
x	x	x	x

 Version
ListManager 8.1

CreateManyMembers

Description
Creates Members from an array of TinyMemberStructs. Example:

```
TinyMemberStructArray = new TinyMemberStruct[300];

TinyMemberStructArray[1].EmailAddress = "email1@domain.com"
TinyMemberStructArray[2].EmailAddress = "email2@domain.com"
...
TinyMemberStructArray[3].EmailAddress = "email300@domain.com"

CreateManyMembers(TinyMemberStructArray,"list1",true)
```

Arguments
TinyMemberStructs[] - an array of member TinyMemberStructs.
Listname - Name of list to add users to.
SkipBadRecords - Stop processing if error occurs during add.

Returns
Number of members created

Return Data Type
int

Access Rights (learn more)

List	Site	Server	Listname Required
x	x	x	x

Version
ListManager 8.1

CreateMemberColumn
Description
Creates a new column in the members_ table.

Arguments
FieldName - name of the column to create (char)
FieldType - datatype of the column (FieldTypeEnum)

Returns
Success flag – true if update successful

Return Data Typebool

Access Rights ([learn more](#))

List	Site	Server	Listname Required
		x	

 Version
ListManager 9.1/API 1.5.1

DeleteMemberColumn
Description
Deletes a user column from the members_ table. You may delete member columns that you previously added.

Arguments
FieldName - name of the column to be deleted (char)

Returns
Success flag – true if update successful

Return Data Type
bool

Access Rights ([learn more](#))

List	Site	Server	Listname Required
		x	

 Version
ListManager 9.1/API 1.5.1

UpdateMemberDemographics

Update demographics for a member. Use field names as defined in your members_ table paired with the correct values in the demographics array.

Arguments
SimpleMemberStruct - A [SimpleMemberStruct](#)
KeyValueType - An array of KeyValueType(s) as defined by your database.

Note: In the previous version of the ListManager API, the second argument was **DemographicsArray**. DemographicsArray and KeyValueType are equivalent, so code should convert by altering the type for any variables of this type.

Returns
Success flag – true if update successful

Return Data Type
bool

Access Rights (learn more)

List	Site	Server	Listname Required
x	x	x	x

Version
ListManager 8.1

UpdateMemberPassword

Updates the password of a member.

Arguments
SimpleMemberStruct - A SimpleMemberStruct
NewPassword - The new member password (string)

Returns
Success flag – true if update successful

Return Data Type
bool

Access Rights (learn more)

List	Site	Server	Listname Required
x	x	x	x

Version
ListManager 8.1

UpdateMemberKind

Updates the "kind" of a member. The member "kind" defines what kind of mail the member receives (e.g., mail or digest; see MemberKindEnum for member kind options). The member kind correlates to the member_.SubType_ field.

Arguments
SimpleMemberStruct - A SimpleMemberStruct
NewMemberKind - The member kind, from MemberKindEnum

Returns
Success flag – true if update successful

Return Data Type
bool

Access Rights (learn more)

List	Site	Server	Listname Required
x	x	x	x

Version
ListManager 8.1

UpdateMemberEmail

Updates the email address of a member.

Arguments
SimpleMemberStruct - A SimpleMemberStruct
NewEmailAddress - The new email address of the member (string)

Returns
Success flag – true if update successful

Return Data Type
bool

Access Rights (learn more)

List	Site	Server	Listname Required
x	x	x	x

Version
ListManager 8.1

UpdateMemberStatus

Updates the "status" of a member. The member "status" defines the membership status (e.g., normal, unsub, confirm; see MemberStatusEnum for member status options). The member status correlates to the member_.MemberType_ field.

Arguments
SimpleMemberStruct - A SimpleMemberStruct
NewStatus - The member's new status, from MemberStatusEnum

Returns
Success flag – true if update successful

Return Data Type
bool

Access Rights (learn more)

List	Site	Server	Listname Required
x	x	x	x

Version
ListManager 8.1

DeleteMembers

Completely removes members based on FilterCriteraArray. Returns number of users deleted.

Arguments
FilterCriteriaArray - an array of member field/value pairs (values from MemberStruct)

Returns
Number of members successfully deleted

CAUTION Passing empty parameters to this function will delete all members from the list.

Return Data Type
int

Access Rights (learn more)

List	Site	Server	Listname Required
x	x	x	x

Version
ListManager 8.1

CopyMember

Copies a member.

Arguments
SimpleMemberStruct - A SimpleMemberStruct
NewEmailAddress - The email address of the new member (string)
NewFullName - The full name of the new member (string)
NewListName - The name of the list this member should be copied into (string)

Returns
Member ID of the new copied member created

Return Data Type
int

Access Rights (learn more)

List	Site	Server	Listname Required
x	x	x	x

Version
ListManager 8.1

SendMemberDoc

Sends a "confirm", "hello", "goodbye", "held", "private" or "delivery" document to any email address or member. Returns Outmail ID.

Arguments
SimpleMemberStruct - A SimpleMemberStruct
DocTypeIn - The message to be sent, from MessageTypeEnum

Returns
Mailing ID of the sent document to member

Return Data Type
int

Access Rights ([learn more](#))

List	Site	Server	Listname Required
x	x	x	

Version
ListManager 8.1

Unsubscribe

Unsubscribes members based on [FilterCriteraArray](#). Members who are unsubscribed remain in the database but their MemberType_ = 'unsub'. Returns number of users unsubscribed.

Arguments
SimpleMemberStructArray - An array of [SimpleMemberStructs](#)

Returns
Number of members usubscribed

NOTE The Unsubscribe call always sends a goodbye message to unsubscribed members.

Return Data Type
int

Access Rights ([learn more](#))

List	Site	Server	Listname Required
x	x	x	x

Version
ListManager 8.1

CheckMemberPassword

Returns true if the user in the SimpleMemberStruct exists with password "Password"

Arguments
SimpleMemberStruct - A [SimpleMemberStruct](#)
Password - The password to be checked for this member (string)

Returns
True if password match member

Return Data Typebool

Access Rights (learn more)

List	Site	Server	Listname Required
x	x	x	x

Version
ListManager 8.1

GetMemberID

Returns the MemberID for a given EmailAddress and ListName as defined in a SimpleMemberStruct.

Arguments
SimpleMemberStruct - A SimpleMemberStruct

Returns
Member ID

Return Data Type
int

Access Rights (learn more)

List	Site	Server	Listname Required
x	x	x	

Version
ListManager 8.1

GetListnameFromMemberID
Returns the list name for a particular member id.

Arguments
MemberID - The ID of the member (int)

Returns
ListName a member ID belongs to

Return Data Type
string

Access Rights (learn more)

List	Site	Server	Listname Required
x	x	x	

Version
ListManager 8.1

GetEmailFromMemberID

Returns the email address for a particular member id.

Arguments
MemberID - The ID of the member (int)

Returns
E-mail corresponding to the ID

Return Data Type
string

Access Rights (learn more)

List	Site	Server	Listname Required
x	x	x	

Version
ListManager 8.1

EmailOnWhatLists
Returns an array of list names that the specified EmailAddress is subscribed to.

Arguments
EmailAddress - The email address (string)

Returns
A list of lists which contain given E-mail address

Return Data Type
ArrayOfString

Access Rights (learn more)

List	Site	Server	Listname Required
x	x	x	

 Version
ListManager 8.1

EmailPasswordOnWhatLists
Returns an array of list names that the specified EmailAddress and Password is subscribed to.

Arguments
EmailAddress - The email address (string)
Password - The password (string)

Returns
A list of lists which contain given E-mail address

Return Data Type
ArrayOfString

Access Rights (learn more)

List	Site	Server	Listname Required
x	x	x	

 Version
ListManager 8.1

CreateMemberBan

Creates a member ban. Returns BanID. BanLogic is: A = "accepted" C = "conditionally accepted" R = "banned". Most member bans will be R (banned); other types may prevent others from joining the list.

Arguments
MemberBan - A member to be banned, from MemberBanStruct

Returns
New Member ID

Return Data Type
int

Access Rights (learn more)

List	Site	Server	Listname Required
x	x	x	x

Version
ListManager 8.1

List Functions

CreateListAdmin

Creates a new list administrator. Returns MemberID of successfully created list admin. Note that this function will not be successful if the EmailAddress specified already exists on the list.

Arguments
EmailAddress - The email address of the administrator (string)
Password - The password of the administrator (string)
ListName - The name of the list (string)
FullName - The full name of the administrator (string)
RecieveListAdminMail - Whether the administrator should receive email messages for list admins (boolean)
RecieveModerationNotification - Whether the administrator should receive notification of messages pending moderation (boolean)
BypassListModeration - Whether the administrator should be able to post messages directly to the list, bypassing list moderation (boolean)

Returns
New Member ID

Return Data Type
int

Access Rights (learn more)

List	Site	Server	Listname Required
x	x	x	x

Version
ListManager 8.1

UpdateListAdmin
Updates the list admin settings of any member.

Arguments
SimpleMemberStruct - A SimpleMemberStruct
IsListAdmin - Whether this member is a list admin (boolean)
RecieveListAdminMail - Whether the administrator should receive email messages for list admins (boolean)
RecieveModerationNotification - Whether the administrator should receive notification of messages pending moderation (boolean)
BypassListModeration - Whether the administrator should be able to post messages directly to the list, bypassing list moderation (boolean)

Returns
Success flag – true if update successful

Return Data Type
bool

Access Rights (learn more)

List	Site	Server	Listname Required
x	x	x	x

Version
ListManager 8.1

DeleteList
Deletes a list, and everything associated with it (members, content, etc.).

Arguments
ListName - The name of the list to be deleted (string)

Returns
Success flag – true if update successful

Return Data Type
bool

Access Rights (learn more)

List	Site	Server	Listname Required
	x	x	x

Version
ListManager 8.1

CreateList

Creates a new list and returns the newly created listID.

Arguments
ListType - The type of list to be created, from ListTypeEnum
ListName - The name of the list (string)
ShortDescription - A short description of the list (string)
AdminName - The name of the list administrator (string)
AdminEmail - The email address of the list administrator (string)
AdminPassword - The password of the list administrator (string)
Topic - the category the mailing list belongs to (char)**

** If no topic is specified, revert to previous (v1.5.2) behavior.

Name	Description	Notes
ListType	The type of list to be created, from ListTypeEnum	
ListName	The name of the list (string)	
ShortDescription	A short description of the list (string)	
AdminName	The name of the list administrator (string)	
AdminEmail	The email address of the list administrator (string)	
AdminPassword	The password of the list administrator (string)	

Returns
New List ID

Return Data Type
int

Access Rights (learn more)

List	Site	Server	Listname Required
	x	x	x

Version
ListManager 8.1

UpdateList

Updates the settings of a list. Returns boolean success.

The following fields cannot be updated:
- ListID
- ListName
- DateCreated

To update MailStreamID, the user must be a server admin, or the $mailstreams_restricted flag must not be set to 'yes'.

Arguments
ListStruct - The list to be updated, from ListStruct

Returns
Success flag – true if update successful

Return Data Type
bool

Access Rights (learn more)

List	Site	Server	Listname Required
	x	x	x

Version
ListManager 8.1

SelectLists

Select an array of ListStructs.

Arguments
ListName - The name of the list. If blank or null, all lists are selected (string)
SiteName - The name of the site. If entered, selects all lists on that site (string)

Returns
An array of list struct matching criteria. NULL if empty set

Return Data Type
ArrayOfListStruct

Access Rights (learn more)

List	Site	Server	Listname Required
x	x	x	

Version
ListManager 8.1

Note
SelectListEx function is an extended version of SelectList. It has additional functionality and improved performance. You can specify fields which will be retrieved from the database thereby shorten time spent on encoding and sending redundant fields.

SelectListsEx

Function
SelectListsEx

Description
SelectListEx function is an extended version of SelectList. It has additional functionality and improved performance. You can specify fields which will be retrieved from the database thereby shorten time spent on encoding and sending redundant fields.

Arguments
ListName – (string) list name

SiteName – (string) site name

FieldsToFetch – (ArrayOfString) list of fields from lists_ table to be fetched

FilterCriteriaArray – (ArrayOfString) list of criteria to filter the lists returned.

Returns
Result set from the query selecting lists or NULL if no records returned. First row contains column names which should be mapped to a more readable form.

Notes
The purpose of this function is to provide a more flexible and portable solution than the SelectLists function. The more fields you request the longer the request takes to execute, so you can specify the exact fields you're interested in while retrieving information about existing lists.

FilterCriteriaArray lets you filter out lists not matching your criteria.

A list of allowed fields to be selected from lists_ table is pre-defined for security reasons but can be extended, limited or changed by manually editing lyrTableFieldMappings.

By default, the following fields are allowed for FieldsToFetch / FilterCriteria arguments.

Admin
ListID
Description
CreationTimestamp
ConferenceVisibility
ConferenceDuration
MessageFooterHTML
MessageHeaderHTML
ListName
Keywords

MriVisibility
Topic
TrackAllUrls
To
From
ReplyTo
DefaultTo
DefaultFrom
DefaultSubject

Rights
Only lists that a user is allowed access to will be returned. If there are lists that match the criteria, but a user is not allowed to administrate any of them, an empty list will be returned.

Return Data Type
ArrayOfArrayOfString

Access Rights (learn more)

List	Site	Server	Listname Required
x	x	x	

Version
ListManager 10.0

GetListID

Gets the ListID from the ListName for a list.

Arguments
ListName - The name of the list (string)

Returns
List ID

Return Data Type
int

Access Rights (learn more)

List	Site	Server	Listname Required
x	x	x	x

Content Functions

CreateContent

Function
CreateContent

Description
Creates a content

Arguments
ContentStruct containing information for a new content

Return Value
Returns an ID number of a content created or raises error with a verbose message

Notes
Title, Description, HeaderTo, HeaderFrom, DocType, NativeTitle (subject) and DocParts must be provided.

ContentID member of the structure is ignored.

Content title must be unique.

* MimePartName member of every DocPart record can be 'html', 'text' for html and text parts respectively OR a filename for attaching a file to the content (e.g. "document.txt")

Rights
If neither SiteName or ListName are provided a user must be a server admin and the content will be created for all lists.

If ListName is provided – user must be allowed to admin that list; content will be created for this list only.

If SiteName is provided – user must be allowd to admin that site; content will be created for this site only.

Return Data Type
int

Access Rights ([learn more](#))

List	Site	Server	Listname Required
x	x	x	

Version
ListManager 10.0

DeleteContent

Function
DeleteContent

Description
Deletes a content

Arguments
SiteAdminStruct – containing ContentID to delete

Return Value
Returns true if succeeds, raises error with a verbose message otherwise

Notes
ContentID must be provided

Rights
A user must be allowed to administrate that content, so either a server admin, or a site admin (if the content is created for that site or a list which belongs to that site) or list admin (if the content has been created for his list only).

Return Data Type
bool

Access Rights ([learn more](#))

List	Site	Server	Listname Required
x	x	x	

Version

ListManager 10.0

SelectContent

Select an array of content structs based on FilterCriteriaArray.

Example:

```
ar[0] = (ListName = list1); ar[1] = (Title like %newsletter%);
```

Arguments
FilterCriteriaArray- an array of content field/value pairs (values from ContentStruct)

Returns
An array of content matching criteria

Return Data Type
ArrayOfContentStruct

Access Rights (learn more)

List	Site	Server	Listname Required
x	x	x	

 Version
ListManager 8.1

UpdateContent

Function
UpdateContent

Description
Updates a content

Arguments
ContentStruct containing information to be changed/overwritten in the content

Return Value
Returns true if succeeds, raises error with a verbose message otherwise

Notes
Title, Description, HeaderTo, HeaderFrom, DocType, NativeTitle (subject) and DocParts must be provided.

ContentID must be provided.

Content title must be unique.

* MimePartName member of every DocPart record can be 'html', 'text' for html and text parts respectively OR a filename for attaching a file to the content (e.g. "document.txt")

Rights
If neither SiteName or ListName are provided a user must be a server admin and the content will be available for all lists after updating.

If ListName is provided – user must be allowed to admin that list; content will be available for this list only after updating.

If SiteName is provided – user must be allowd to admin that site; content will be available for this site only after updating.

Return Data Type
bool

Access Rights (learn more)

List	Site	Server	Listname Required
x	x	x	

Version
ListManager 10.0

Mailings Functions

GetPreviewMailing

Function
GetPreviewMailing

Description
Returns merged text containing merge tags against given member and the subset.

Arguments
PreviewStruct - structure filled with information needed for merging
struct

Return Value
Returns a merged text as string or raises error with verbose message

Notes
This function is used to preview how a message is going to look after merging for a particular member and subset. Merge tags will be replaced with the data corresponding to a user. TextToMerge and MemberID are mandatory members of the PreviewStruct structure

Rights
User must be a list admin of a list which a member of MemberID given belongs to. So server, site or list admin are allowed with above limitations.

Return Data Type
string

Access Rights (learn more)

List	Site	Server	Listname Required
x	x	x	

Version
ListManager 10.0

ImportContent

Takes a Content ID and returns a SimpleMailingStruct. The SimpleMailingStruct can be used to create a new mailing by copying all of its elements to a new MailingStructand sending that.

Arguments
ContentID - The ID of the content to be made into a mailing (int)

Returns
Content corresponding to the ID given

Return Data Type
SimpleMailingStruct

Access Rights (learn more)

List	Site	Server	Listname Required
x	x	x	

Version
ListManager 8.1

MailingStatus

Get the status of a mailing sent with list processing (e.g., sent with SendMailing). MailingStatus will not work with a mailing created with SendMailingDirect.

Arguments
InMailID - The ID of the inmail to be checked (int)

Returns
Transact log for the given mailing ID

Return Data Type
string

Access Rights (learn more)

List	Site	Server	Listname Required
x	x	x	

Version

ListManager 8.1

ScheduleMailing

Schedules a mailing for a future date based on MailingStruct. The message is then subject to list processing and validation as if it were a mailing created by an administrator through the administrator interface. If SegmentID is blank or 0, the mailing is sent to the entire list as defined in MailngStruct. If successful, returns the InMail ID of the new mailing.

Arguments
SegmentID - The ID of the segment to be sent to. Sends to the entire list if blank or null (int)
SendDate - The date and time the mailing should be sent /soap/object_reference.tml" /l "dateTime(datetime)
MailingStructIn - A MailingStruct

Returns
Mailing ID of the scheduled mailing

Return Data Type
int

Access Rights (learn more)

List	Site	Server	Listname Required
x	x	x	x

Version
ListManager 8.1

SendMailing

Sends a mailing based on MailingStruct by creating a row in the inmail_ table. The message is then subject to list processing and validation as if it were a mailing created by an administrator through the administrator GUI. If SegmentID is blank or 0, the mailing is sent to the entire list as defined in MailngStruct. If successful, returns the InMailID of new mailing.

Arguments
SegmentID - The ID of the segment to be sent to. Sends to the entire list if blank or null (int)
MailingStruct - The contents of the mailing as a MailingStruct

Returns
Mailing ID of the sent mailing

Return Data Type
int

Access Rights (learn more)

List	Site	Server	Listname Required
x	x	x	x

Version
ListManager 8.1

SendMailingDirect

Sends a mailing based on MailingStruct to any email address or member IDs, or an array of email addresses and member IDs. Creates mailing directly in the OutMail_ table, so does not perform any list processing. The email address(es) defined in the EmailAddressArray do not need to be subscribed to any list. Returns true if message was successfully queued.

Arguments
EmailAddressArray - An array of email addresses to be sent the message (ArrayOfString)
MemberIDArray - An array of member IDs to be sent the message (ArrayOfInt)
MailingStruct - The contents of the mailing, as defined by a MailingStruct

Returns
Mailing ID of the scheduled mailing

Return Data Type
int

Access Rights (learn more)

List	Site	Server	Listname Required
x	x	x	x

Version
ListManager 8.1

SendMessage

Description
A simplified way to send mail through ListManager acting as a Mail Transfer Agent. Creates an entry in log_ table of sending of those mails.

Arguments
MessageStructIn - Message to be sent as structure (MessageStruct)

Returns
ID of the Message sent

Return Data Type
int

Access Rights (learn more)

List	Site	Server	Listname Required
x	x	x	x

Version
ListManager 9.1/API 1.5.1

TMSendMessage
This API function sends a transactional message to a specific member ID or email address.

TMSendMessage uses an activated Transactional Message which must be pre-defined in UI. The MailingID for this Transactional Message must be provided in the TMMessageStruct structure. You can find the Message ID on the Home page or the Outgoing Mailing page.

The TMMessageStruct structure contains an array of key/value pairs called Payload. You can use the payload data to pass recipient-specific information. This data will be merged dynamically when the message is sent.

An example of use of these messages would be a point of sale transaction where the message can contain the amount due from the time of purchase. This amount and, perhaps, a description of the item sold would be merged into the Transaction Message created in the UI.

A MemberID or RecipientEmail can be used to specify the recipient. If a MemberID is used, the API user must have access to that user (e.g. be a list admin of a list to which this member belongs). RecipientEmail, if provided, must be a valid email address. If both MemberID and RecipientEmail are provided, an error will be returned if the email of the member identified by MemberID differs from RecipientEmail provided. Payload provided in the TMMessageStruct structure will be used in %%payload name%% merge tags within the message.

Note: Unlike the SendMailing and SendMessage functions, TMSendMessage does not create a new outmail_ record.

Arguments
TMMessageStruct - a struct including email or member id, payload, and message id

Returns
The recipient id or an error if problems occur at the time of processing the transactional message

Return Data Type
unsigned long

Access Rights (learn more)

List	Site	Server
x	x	x

Version
ListManager 11.1

Site Functions

CreateSite

Function
CreateSite Description
Creates a new site

Arguments
SiteStruct – site to be created as a structure.
Struct

Return Value
ID of the site created

Notes
Leave the SiteID of SiteStruct empty.
SiteName and HostName are required.
SiteName has to be unique for the server.
HostName should have the URL form starting from 'http://' or 'https://'.

Rights
You have to be a server admin

Return Data Type
int

Access Rights (learn more)

List	Site	Server	Listname Required
		x	

Version
ListManager 10.0

CreateSiteAdmin

Function
CreateSiteAdmin

Description
Creates a site admin

Arguments
SiteAdminStruct – containing information needed to create an admin

Return Value
Returns an ID number of a server admin created or raises error with a verbose message

Notes
Name, Password, EmailAddress and WhatSites members if the SiteAdminStructure must be provided.

AdminID member of the structure is ignored.

Rights
User must be allowed to administrate all the sites a site admin being created is supposed to be admin of. In other words it should be either a server admin or a site admin of all sites he wants to grant access to the newly created admin.

Return Data Type
int

Access Rights (learn more)

List	Site	Server	Listname Required
	x	x	

Version
ListManager 10.0

DeleteSite

Function
DeleteSite

Description
Deletes a site

Arguments
SiteStruct – containing SiteID of a site to be deleted
struct

Return Value
Returns true if succeeds, raises error with a verbose message otherwise

Notes
You have to provide the SiteID member of SiteStruct structure you want to delete.
All lists, topics, members, log data, completed recipients, mailings and other information related to that site will be deleted.

Rights
Server or Site admin of the updated site

Return Data Type
bool

Access Rights ([learn more](#))

List	Site	Server	Listname Required
	x	x	

Version
ListManager 10.0

DeleteSiteAdmin

Function
DeleteSiteAdmin

Description
Deletes a site admin

Arguments
SiteAdminStruct – containing AdminID to delete

Return Value
Returns true if succeeds, raises error with a verbose message otherwise

Notes
AdminID must be provided.

Rights
User must be allowed to administrate all the sites a site admin being created is supposed to be admin of. In other words it should be either a server admin or a site admin of all sites he wants to grant access to the updated admin.

Return Data Type
bool

Access Rights ([learn more](#))

List	Site	Server	Listname Required
	x	x	

Version
ListManager 10.0

UpdateSite

Function
UpdateSite

Description
Updates a site

Arguments
SiteStruct – structure containing data to be overwritten (updated)
struct

Return Value
Returns true if succeeds, raises error with a verbose message otherwise

Notes
You have to provide the SiteID member of SiteStruct structure you want to update.

Rights
Server or Site admin of the updated site

Return Data Type
bool

Access Rights (learn more)

List	Site	Server	Listname Required
	x	x	

Version
ListManager 10.0

UpdateSiteAdmin

Function
UpdateSiteAdmin

Description
Updates a site admin

Arguments
SiteAdminStruct – containing information to update

Return Value
Returns true if succeeds, raises error with a verbose message otherwise

Notes
AdminID must be provided.

Rights
User must be allowed to administrate all the sites a site admin being created is supposed to be admin of. In other words it should be either a server admin or a site admin of all sites he wants to grant access to the updated admin.

Return Data Type
bool

Access Rights ([learn more](#))

List	Site	Server	Listname Required
		x	

Version
ListManager 10.0

SQL and Misc. Functions

SqlSelect

Runs an SQL SELECT Statement, and returns an array of array of strings. The first element of the array contains field names. Only server administrators may access this function.

Arguments
SqlStatement - The SQL SELECT statement to run (string)

Returns
Result set from the query or NULL if no records returned. First row contains column names.

NOTE You can use SqlSelect when doing a mail-merge using SOAP with non-ListManager tables; it can use execute any query, i.e., INSERT INTO < non-ListManager table >.

Return Data Type
ArrayOfArrayOfString

Access Rights (learn more)

List	Site	Server	Listname Required
		x	

Version
ListManager 8.1

SqlDelete

Deletes row(s) from a table using a "where" clause. Only server administrators may access this function.

Arguments
Table - The table the rows should be deleted from. See the Programmer's Guide in the Reference Manual for more information about ListManager's tables (string)
SqlWhere - The "where" clause (string)

Returns
Success flag – true if update successful

Return Data Type
bool

Access Rights (learn more)

List	Site	Server	Listname Required
		x	

Version
ListManager 8.1

SqlInsert

Inserts rows into a table. DataArray should contain field names and values. If returnid is true, the id of the row is returned. otherwise functions returns 1. Only server administrators may access this function.

Arguments
Table - The table the rows should be inserted. See the Programmer's Guide in the Reference Manual for more information about ListManager's tables (string)
DataArray - ArrayOfKeyValueType
ReturnID - Whether the ID of the new row should be returned (Boolean)

Returns
Identity of the row created

NOTE You cannot use SqlInsert when doing a mail-merge using SOAP with non-ListManager tables. Instead, use SqlSelect; it can use execute any query, i.e., INSERT INTO < non-ListManager table >.

Return Data Type
int

Access Rights ([learn more](#))

List	Site	Server	Listname Required
		x	

Version
ListManager 8.1

SqlUpdate

Updates rows into a Table using an SQL "where" clause. DataArray should contain field names and values. Only server administrators may access this function.

Arguments
Table - The table whose the rows should be updated. See the Programmer's Guide in the [Reference Manual](#) for more information about ListManager's tables (string)
DataArray - The data array to update the table (ArrayOfKeyValueType)
SqlWhere - The SQL "where" clause that should be run (string)

Returns
Success flag – true if update successful

Return Data Type
bool

Access Rights ([learn more](#))

List	Site	Server	Listname Required
		x	

Version
ListManager 8.1

ApiVersion

Verifies the version of the API the target ListManager server is running. Different api versions have incompatible clients, so this function ensures the version you expect is the version hosted on the listmanager server.

Arguments
(none)

Returns
Current API version

Return Data Type
string

Access Rights (learn more)

List	Site	Server	Listname Required
x	x	x	

Version
ListManager 8.1

CurrentUserEmailAddress

This function returns the email address associated with the user logged in. For example, this command will return the email address associated with the server administrator "admin".

Arguments
(none)

Returns
User's e-mail address

Return Data Type
string

Access Rights (learn more)

List	Site	Server	Listname Required
x	x	x	

Version
ListManager 8.1

Segment Functions

CreateSegment

Description
Creates a segment.

Arguments
SegmentStructIn - segment specification as struct - (SegmentStruct)

NOTE SegmentID of SegmentStructIn will be ignored.

Returns
New segment ID

Return Data Type
int

Access Rights (learn more)

List	Site	Server	Listname Required
x	x	x	x

Version
ListManager 9.1/API 1.5.1

SelectSegments

Selects an array of SegmentStructs using FilterCriteriaArray.

Example:
```
ar[0] = (SegmentName = aol); ar[1] = (ListName like %lis%);
```

Arguments
FilterCriteriaArray- an array of member field/value pairs (values from SegmentStruct)

Returns
Array of segment struct matching criteria

Return Data Type
ArrayOfSegmentStruct

Access Rights (learn more)

List	Site	Server	Listname Required
x	x	x	x

Version
ListManager 8.1

UpdateSegment

Executes update query to update all corresponding fields of the structure. Overrides the existing structure of a segment uniquely identified by the SegmentID member of SegmentStructIn.

Arguments
SegmentStructIn - new segment description (to be updated) (SegmentStruct)

Returns
Success flag – true if update successful

Return Data Type
Bool

Access Rights ([learn more](#))

List	Site	Server	Listname Required
		x	x

Version
ListManager 9.1/API 1.5.1

DeleteSegment

Description
Deletes a segment.

Arguments
Segment ID - segment ID to be deleted (int)

Returns
Success flag – true if update successful

Return Data Type
bool

Access Rights ([learn more](#))

List	Site	Server	Listname Required
x	x	x	x

Version
ListManager 9.1/API 1.5.1

Server Functions

CreateServerAdmin

Function
CreateServerAdmin

Description
Creates a server admin

Arguments
ServerAdminStruct – containing information needed to create an admin

Return Value
Returns an ID number of a server admin created or raises error with a verbose message

Notes
Name, Password and EmailAddress members of the ServerAdminStructure must be provided.

AdminID member structure is ignored.

Rights
You have to be a server admin.

Return Data Type
int

Access Rights (learn more)

List	Site	Server	Listname Required
		x	

Version
ListManager 10.0

DeleteServerAdmin

Function
DeleteServerAdmin

Description
Deletes a server admin

Arguments
ServerAdminStruct – containing AdminID to delete

Return Value
Returns true if succeeds, raises error with a verbose message otherwise

Notes
AdminID must be provided

Rights
You have to be a server admin.

Return Data Type
bool

Access Rights ([learn more](#))

List	Site	Server	Listname Required
		x	

Version
ListManager 10.0

UpdateServerAdmin

Function
UpdateServerAdmin

Description
Updates a server admin

Arguments
ServerAdminStruct – containing information to update

Return Value
Returns true if succeeds, raises error with a verbose message otherwise

Notes
AdminID must be provided.

Rights
You have to be a server admin.

Return Data Type
bool

Access Rights (learn more)

List	Site	Server	Listname Required
	x	x	

Version
ListManager 10.0

Report Functions

TrackingSummary

Exports a report of delivery success and failures, unique and total opens, and unique clicks per URL.

Arguments
OutmailID - The MessageID_ of an outmail.

Returns
Tracking summary requested for given outmailID_

Return Data Type
TrackingSummaryStruct

Access Rights (learn more)

List	Site	Server	Listname Required
x	x	x	

 Version
ListManager 8.1

Topic Functions

CreateTopic

Function
CreateTopic

Description
Creates a topic

Arguments
TopicStruct – structure filled with topic information
struct

Return Value
Returns true if succeeds, raises error with a verbose message otherwise**l**

Notes
TopicName, TopicDescription, SiteName are mandatory fields which need to be given.
TopicName should be unique.

Rights
User must be allowed to admin a site topic is about to be created for. So server or site admin is required.

Return Data Type
bool

Access Rights (learn more)

List	Site	Server	Listname Required
	x	x	

Version
ListManager 10.0

Function
DeleteTopic

Description
Deletes a topic

Arguments
TopicStrict – containing TopicName of a site to be deleted
struct

Return Value
Returns true if succeeds, raises error with a verbose message otherwise

Notes
You have to provide the TopicName member of TopicStruct structure you want to delete. All information corresponding to the topic will be deleted like active or completed recipients or clicktrtracking data.

Rights
User must be allowed to admin a site topic is about to be updated for. So server or site admin is required.

Return Data Type
bool

Access Rights (learn more)

List	Site	Server	Listname Required
	x	x	

 Version
ListManager 10.0

Function
UpdateTopic

Description
Updates a site

Arguments
TopicStruct – structure containing data to be overwritten (updated)
struct

Return Value
Returns true if succeeds, raises error with a verbose message otherwise

Notes
You have to provide the TopicName member of TopicStruct structure you want to update.

Rights
User must be allowed to admin a site topic is about to be updated for. So server or site admin is required.

Return Data Type
bool

Access Rights ([learn more](#))

List	Site	Server	Listname Required
	x	x	

Version
ListManager 10.0

Deprecated Functions

SelectSimpleMembers

This function is **deprecated** and will be removed in a future release. Please use the SelectMembers()function

Description
Selects an array of MemberStructs using FilterCriteriaArray. Example:

```
ar[0] = (ListName = list1); ar[1] = (domain = example.com)
```

Arguments
FilterCriteriaArray - an array of member field/value pairs (values from MemberStruct)

Returns
An array of members matching criteria

Return Data Type
ArrayOfSimpleMemberStruct

Access Rights (learn more)

List	Site	Server	Listname Required
x	x	x	x

Version
ListManager 8.1

Type Reference

Classes and Structs

ContentStruct

Used to create content.

Name	Type	Description
ContentID	int	Unique ID in Docs_ table
DateCreated	date	Date content was created
Description	string	The description of this content
DocParts	ArrayOfDocPart	An array of doc parts to be created along with the content
DocType	DocTypeEnum	The document type
HeaderFrom	string	The From: header
HeaderTo	string	The To: header
IsReadOnly	boolean	If true, web interface will not let user change this document in the web interface
IsTemplate	boolean	If true, this content is a template
ListName	string	a list name
NativeTitle	string	Title of the document in its native language. A subject of the content.
SiteName	string	a site name
Title	string	Title of the document, in English

Note: Since Microsoft .NET does not support NULL date/times, the date/time "2001-01-01 00:00:00" is considered null.

CharSetStruct

The CharSetStruct defines the character set used by a mailing. The CharSetStruct is used with the MailingStruct.

Name   Type Description
CharSetID int   An auto-incrementing ID field that uniquely identifies a record in this table
Description string A text description. Informational only
Name   string The MIME compatible name of the character set

DocPart

Structure Name
DocPart

Members
Body – (string) a body
MimePartName* – (string) mime name
CharSetID – (int) a charset id for this doc part
Encoding – (MailSectionEncodingEnum) encoding for this doc part * MimePartName can be 'html', 'text' for html and text parts respectively OR a filename for attaching a file to the content.

* MimePartName can be 'html', 'text' for html and text parts respectively OR a filename for attaching a file to the content

KeyValueType

 This is a Key-Value pair.

Name	Type	Description
Value	string	Value to insert
Name	string	Name of the key

ListStruct

The ListStruct describes all of the parameters for a particular list.

Name	Type	Description	Notes
ListID	int	The unique identifier for this list	
Additional	string	Additional information about this list (not used by ListManager)	
AddListNameToSubject	boolean	If true, adds the list's name to the "Subject:" header	
Admin	string	Administrator responsible for this list	
AllowCrossPosting	boolean	If true, this list should accept cross-posted messages (identical messages sent to multiple lists)	
AllowDuplicatePosts	boolean	If true, this list should accept duplicate messages	
AllowInfo	boolean	If true, this list should allow non-members to read information about it	
AnyoneCanPost	boolean	If true, this list should allow anyone to post; if false, it requires those who post to be members	
ApproveNum	int	The number of approvals required of new members before they may post to the list without moderation	
ArchiveDays	int	The number of days archived messages are kept	
ArchiveNum	int	The number of archived messages kept	
AutoReleaseHour	int	The time of day messages should be automatically approved and released	
BlankSubjectOk	boolean	If true, the list should permit messages with blank "Subject:" headers	
Child	boolean	If true, this list should accept postings from other lists	

Name	Type	Description	Notes
CleanAuto	boolean	If true, this list should automatically put members who bounce too much mail on hold	
CleanNotif	int	The length of time (days) a "held" member will be notified	
Comment	string	Additional information about this list (not used by ListManager)	
CommentsID	string	Comment about this list (links to a document)	
ConfirmSubscribes	boolean	Whether this list should confirm new subscriptions	
ConfirmUnsubscribes	ConfirmUnsubEnum	Whether this list should confirm unsubscribes	
DateCreated	dateTime	The date and time this list was created	
DefaultFrom	string	The default "From:" header	
DefaultSubject	string	The default "Subject:" header	
DefaultTo	string	The default "To:" header	
DeliveryReports	int	What delivery reports should be sent to administrators	
DescLongDocID	string	Long description of this list (links to a document)	
DetectHtmlByDefault	boolean	If true, this list should detect HTML capability by default	
DetectOpenByDefault	boolean	If true, this list should detect HTML opens by default	
DigestFooter	string	The digest footer	
DigestHeader	string	The digest header	
Disabled	boolean	If true, this list is disabled	
DontRewriteMessageIDHeader	boolean	If true, ListManager will not rewrite the Message-ID, but instead leave it alone, and use X-Lyris-Message-ID	
EnableScripting	EnableScriptingEnum	What level of scripting the list allows	
ErrHold	int	How many bounces before a member is put on hold	
From	string	If set, specifies the "From:" header of every message	

Name	Type	Description	Notes
		send on this list	
HeaderRemove	string	Which headers should be removed	
KeepOutmailPostings	int	How long mailings sent to the list should be kept	
Keywords	string	Keywords which describe this list	
ListName	string	The name of the list	
MakePostsAnonymous	boolean	If true, posts to this list will be anonymous	
MaxMembers	int	The maximum number of members this list is allowed	
MaxMessageSize	int	The maximum size, in bytes, non-administrators may post to the list	
MaxMessNum	int	The maximum number of messages that may be sent through the list in a 24-hour period	
MaxPostsPerUser	int	The maximum number of posts a single person may post to the list	
MaxQuoting	int	The maximum number of contiguous lines that may be quoted at once in a single message	
MemberListSecurity	MemberListSecurityEnum	Who may access the list of members	
MergeCapOverride	MergeCapOverrideEnum	Whether this list should override the server scripting levels	
MessageFooter	string	The default message footer	
MessageHeader	string	The default message header	
Moderated	ModeratedEnum	What level of moderation this list requires	
ModifyHeaderDate	boolean	Whether the "Date:" header should automatically be rewritten to be the date and time a message is approved	
MRIVisibility	MRIVisibilityEnum	Whether the Discussion Forum Interface for this list should be (v)isible, (h)idden or (i)nvisible	

Name	Type	Description	Notes
NameRequired	NameRequiredEnum	Whether the subscriber must provide a name when joining	
NewSubscriberSecurity	NewSubscriberPolicyEnum	What security controls who may join the list	
NoArchive	boolean	If true, this list maintains a message archive	
NoBodyOk	boolean	If true, messages posted to this list may be blank	
NoEmail	boolean	If true, this list rejects messages posted via email, and requires postings through the ListManager web interface	
NoEmailSubscriptions	boolean	If true, this list prohibits subscriptions via email, and requires them through the ListManager web interface	
NoListHeader	boolean	If true, the list-help headers should not be included	
NoNNTP	boolean	If true, the list will not be accessible via NNTP	
NoSearch	boolean	If true, the list's archives, if available, will not be searchable	
NotifyHeldInterval	int	How frequently held members should be notified	
OnlyAllowAdminSend	boolean	If true the list will only allow administrators to send	
PasswordRequired	int	Whether a password is required when joining	
PostPassword	PostPasswordEnum	Whether a member's password must be included in the message in order to be posted	
PrivApprov	string	Sets rules for automatically approving applications to a private list	
PurgeExpiredInterval	int	How long expired members should be kept in the database before being purged, in days	
PurgeHeldInterval	int	How long held members should be kept in the	

Name	Type	Description	Notes
		database before being purged, in days	
PurgeReferredInterval	int	How long referred members should be kept in the database before being purged, in days	
PurgeUnapprovedInterval	int	How long unapproved members should be kept in the database before being purged, in days	
PurgeUnconfirmedInterval	int	How long unconfirmed members should be kept in the database before being purged, in days	
PurgeUnsubInterval	int	How long unsubscribed members should be kept in the database before being purged, in days	
RecencyDayCount	int	How many days to check for recent mailings	
RecencyEmailEnabled	boolean	If ture, recency checking/purging is enabled	
RecencyMailCount	int	The number of mailings that should trigger recency limits	
RecencyOperator	RecencyOperatorEnum	When checking for recency, should the number of mailings be (m)ore than, (e)qual to or (l)ess than the number in RecencyMailCount	
RecencySequentialEnabled	boolean	If true, recency limits should be applied to sequential mailings	
RecencyTriggeredEnabled	boolean	If true, recency limits should be applied to triggered mailings	
RecencyWebEnabled	boolean	If true, recency limits should be applied to web mailings	
RecipientLoggingLevel	LoggingLevelEnum	What level of delivery logging should there be? (N)o logging, non-SMTP errors (m), all (e)rrors, errors and successes (f)	
ReferralsPerDay	int	The number of referrals	

Name	Type	Description	Notes
		allowed per member per day	
ReleasePending	int	The length of time unapproved messages should be kept before being sent automatically	
RemoveDuplicateCrossPostings	boolean	If true, if a list allows cross-postings, it should clean out duplicate recipients so that members of multiple lists only receive one copy of a message	
ReplyTo	string	The "Reply-To:" header	
RunProgAfterPosting	string	Program to execute after posting	
RunProgAfterSub	string	Program to execute after subscribing	
RunProgAfterUnsub	string	Program to execute after unsubscribing	
RunProgBeforePosting	string	Program to execute before posting a message	
RunProgBeforeSub	string	Program to execute before processing a subscribe	
RunProgBeforeUnsub	string	Program to execute before processing an unsubscribe	
ScriptingLevel	[ScriptingLevelEnum](#)	What level of scripting should this list allow	
ShortDescription	string	A short description of this list	
SimpleSub	boolean	If true, this list should only allow name, email address, or allow members to set other options (such as noack, norepro)	
SMTPFrom	string	The SMTP "MAIL FROM:" header	
SMTPHeaders	string	Text that should be included in the SMTP header of every message	
SponsOrgID	string	Organization that is sponsoring this list (link to document, type is organization)	
SubscribePassword	string	If this list is 'password'	

Name	Type	Description	Notes
		protected for new subscribers, what is the password?	
SubscriptionReports	SubscriptionReportEnum	What subscription reports should be emailed to list administrators?	
To	string	The default "To:" header	
Topic	char	The category the mailing list belongs to.	** New in v.1.5.1
Visitors	boolean	If true, the list should allow non-members to see messages in the Discussion Forum Interface	

Note: Since Microsoft .NET does not support NULL date/times, the date/time "2001-01-01 00:00:00" is considered null.

MailingStruct

The MailingStruct defines a mailing.

Name	Type	Description
AdditionalHeaders	string	Any addtional headers (e.g., X-type headers)
Attachments	string	Any attachments. The syntax for using this element is: 'Attachments' => '{foo 2003 times.txt}' where **times.txt** is the name of the attached file, and **2003** is the value for 'uniqueid_' in the 'docparts_' table. "Foo" is simply a placeholder and can be replaced with any string you choose. Note that both the quotes and the curly brackets are significant. Using 'foo 2003 times.txt' instead of '{foo 2003

Name	Type	Description
		times.txt}' will cause the SOAP server to reply with an error.
BypassModeration	boolean	If true, the mailing may bypass moderation
Campaign	string	What campaign the mailing should belong to
CharSetID	int	The ID of the charset for this mailing (from CharSetStruct)
DetectHtml	boolean	If true, the mailing should detect if the recipient has the ability to read HTML
DontAttemptAfterDate	dateTime	The "kill date" after which this mailing should no longer be delivered
EnableRecency	boolean	If true, recency checking should be done for this mailing
From	string	The header "From:"
HtmlMessage	string	The HTML portion of the mailing
HtmlSectionEncoding	int	The HTML message encoding
IsHtmlSectionEncoded	boolean	If true, the HTML portion of the mailing has been encoded
IsTextSectionEncoded	boolean	If true, the text portion of the mailing has been encoded
ListName	string	The list this mailing is posted to
RecencyNumberOfMailings	int	If enabled, the number of mailings that should trigger recency limits.
RecencyWhich	string	When checking for recency, should the number of mailings be more than, equal to or less than the number in RecencyNumberof Mailings (from RecencyWhichEnum)
ReplyTo	string	The "Reply To:" header
ResendAfterDays	int	If set, means that when the message is approved, it should be resubmitted for automatic approval after so many days
SampleSize	int	The maximum number of recipients to receive this mailing. If not set, there is no

Name	Type	Description
		maximum
ScheduledMailingDate	dateTime	The date and time the mailing is scheduled to be sent
Subject	string	The "Subject:" header
TextMessage	string	The text portion of the mailing
TextSectionEncoding	int	The encoding of the text section
Title	string	The title of the mailing
To	string	The "To:" header
TrackOpens	boolean	If true, this mailing should detect if recipients open it
RewriteDateWhenSent	boolean	If true, the mailing should have its "Date:" header rewritten to be the time it was actually sent, removing the original "Date:" header

NOTES

- Since Microsoft .NET does not support NULL date/times, the date/time "2001-01-01 00:00:00" is considered null.
- "MailingName" has been removed from MailingStruct in ListManager 10.0 . The "Title" member holds the mailing name.

MemberStruct

Most member functions use the SimpleMemberStruct. The MemberStruct provides more information about members. It is returned by SelectMembers.

Name	Type	Description
MemberID	integer	Unique member ID
Additional	string	Additional information about this member (not used by ListManager)
ApprovalNeeded	boolean	Can this member bypass approval to send messages?
CanApprovePending	boolean	If true, and if this member is an admin, can this member approve pending (moderated) messages?
CleanAuto	boolean	If true, do not mark as 'held' this member if they bounce too much email (i.e.: let them bounce all they want)
Comment	string	Additional information about this member (not used by ListManager)

Name	Type	Description
DateBounce	dateTime	The date of the most recent bounce
DateConfirm	dateTime	Date the member was last sent a 'confirm' message
DateHeld	dateTime	Date the member was held
DateJoined	dateTime	Date when member joined this list
DateUnsubscribed	dateTime	Date when member unsubscribed from this list
Demographics	ArrayOfKeyValueType	An array of KeyValueType(s) as defined by your database
EmailAddress	string	The email address of the member
ExpireDate	dateTime	The date the membership expired
FullName	string	The full name of the member
IsListAdmin	boolean	If true, the member is a list administrator
ListName	string	The name of the list this member belongs to
MailFormat	MailFormatEnum	What format does the user want to receive mail, (T)ext, (M)ultipart, or (H)TML.
MembershipKind	MemberKindEnum	The kind of subscription this member has ('mail', 'nomail', 'digest', 'mimedigest', 'index')
MemberStatus	MemberStatusEnum	What is the status of this member? ('normal', 'confirm', 'private', 'expired', 'held', 'unsub', 'needs-goodbye', 'needs-hello', 'needs-confirm')
NoRepro	boolean	If true, the member should receive a copy of his own postings
NotifyError	boolean	If true, the member (if a list administrator) should receive list error mail
NotifySubmission	boolean	If true, the member (if a list administrator) should receive notification of messages pending moderation
NumApprovalsNeeded	integer	The number of approvals this member needs before posting to the list unapproved
NumberOfBounces	integer	The number of bounces this member has had
Password	string	The member's password
ReadsHtml	boolean	If true, this member has been detected to read HTML
ReceiveAcknowlegment	boolean	If true, this member receives a message acknowledging a posting to the list
ReceiveAdminEmail	boolean	If true, this member (if a list administrator) should receive messages for list administrators
UserID	string	Additional information about this member (not used by ListManager)

Note: Since Microsoft .NET does not support NULL date/times, the date/time "2001-01-01 00:00:00" is considered null.

MemberBanStruct

The MemberBanStruct defines a member ban, which controls who may or may not join lists.

Name	Type	Description
BanLogic	BanLogicEnum	The type of ban this is
Domain	string	The domain this ban applies to
ListName	string	The list this ban applies to; NULL for a site or server ban
SiteName	string	The site this ban applies to; NULL for a list or server ban
UserName	string	The user name this ban applies to, if any

MessageStruct

Name	Type	Description
RecipientEmailsIn	ArrayOfString	Array of email addresses you send messages to (optional)
RecipientsMemberIDsIn	ArrayOfInt	Array of IDs of members you want to send messages to (optional)
HeadersIn	ArrayOfKeyValueTypes	Headers of the message, in the form of key/value pairs
Body	char	Raw ready to send body (excluding headers)
SegmentID	int	An integer determining associated segment ID (if 0 or NULL, omitted). Does not send to this segment; only associates the message with it
ListName	char	List you want to associate mailing with (if 0 or NULL, omitted). Does not send to this segment; only associates the message with it

PreviewStruct

Structure name

PreviewStruct

Members
TextToMerge – (string) Text including merge tags to be merged
MemberID – (int) MemberID of a user whose data will be used for merging
SubsetID – (int) SubsetID to be merged against while using member data

SegmentStruct

The SegmentStruct defines a segment (SQL subset) of a list.

Name	Type	Description
SegmentID	int	Unique identifier of this segment
Description	string	A description of this segment
ListName	string	The name of the list this segment is associated with
NumTestRecords	int	The number of records to retrieve when testing this segment
SegmentName	string	The name of this segment
SegmentType	SegmentTypeEnum	The type of segment, normal, triggered or sequential
ClauseWhere	char	Essentials/Query
ClauseAfterSelect	char	Advanced/After SELECT
ClauseFrom	char	Advanced/Join other tables
ClauseOrderBy	char	Advanced/Sorting
ClauseSelect	char	Advanced/Columns to display
AddWhereList	bool	Boundaries/Pull only this list's members
AddWhereMemberType	bool	Boundaries/Pull only normal members
AddWhereSubType	bool	Boundaries/Pull only members receiving regular email

ServerAdminStruct

Structure name
ServerAdminStruct

Members
AdminID – (int) id of a server admin (needed for update and delete – leave empty for creating an admin)
Name – (string) name of the administrator
EmailAddress – (string) e-mail address of an administrator
Password – (string) password of the administrator

SimpleMailingStruct

The SimpleMailingStruct is created from the ImportContent function. It may then be used in SendMailingor SendMailingDirectfunction.

Name	Type	Description
AdditionalHeaders	string	Any additional SMTP headers for the mailing
Attachments	string	Any attachments to the mailing (Note: this field should only be set from the results of ImportContent; it should not be set by the user)
CharSetID	int	The charset ID
From	string	The From header of the mailing
HtmlMessage	string	The HTML message
HtmlSectionEncoding	int	The encoding of the HTML section
IsHtmlSectionEncoded	boolean	If true, the HTML section is encoded
IsTextSectionEncoded	boolean	If true, the text section is encoded
Subject	string	The Subject header of the mailing
TextMessage	string	The text message
TextSectionEncoding	int	The encoding of the text section
To	string	The To header
Title	string	The title of the mailing

SimpleMemberStruct

The SimpleMemberStruct requires minimal information about a member, and is used in many member functions.

Name	Type	Description
MemberID	int	The unique identifier for this member
EmailAddress	string	The email address for this member
ListName	string	The list this member belongs to

SiteAdminStruct

Structure name
SiteAdminStruct

Members
AdminID – (int) id of a server admin (needed for update and delete – leave empty for creating an admin)
Name – (string) name of the administrator

EmailAddress – (string) e-mail address of an administrator
Password – (string) password of the administrator
WhatSites – (ArrayOfString) array of sites the site admin should be able to administrate

SiteStruct

Structure name
SiteStruct

Members
SiteID – (int) id of the site (needed for update and delete – leave empty for creating site)
SiteName –(string) name of the site
SiteDescription – (string) description of the site
HostName – (string) hostname to associate with site
WebInterfaceURL – URL address of the web interface

TinyMemberStruct

The TinyMemberStruct requires minimal information about a member, and is used in the CreateManyMembers function.

Name	Type	Description
EmailAddress	string	The email address for this member.
FullName	string	The full name for this member.

TMMessageStruct

Provides information about a transactional message.

Name	Type	Description
RecipientEmail	string	The recipient's email address (must be a valid address; can be used insted of MemberID)
MemberID	int	The recipient's membmerID (can be used instead of RecipientEmail)
MailingID	int	The MailingID of the transactional message
Payload	ArrayOfKeyValueType	Array of key/value pairs carrying the payload for the message (for example {Name='day',Value='Tuesday'} will be used in a transactional message containing the %%payload day%% merge tag)

TopicStruct

Structure name
TopicStruct

Members
TopicName – (string) name of the topic
TopicDescription – (string) name of the topic
SiteName – (string) site name this topic should belong to
HiddenTopic – (bool) determines if the topic should be hidden

TrackingSummaryStruct

The TrackingSummaryStruct is returned by the TrackingSummary function, and gives basic delivery and tracking information for a mailing.

Name	Type	Description
TransientFailure	string	Temporary failures due to non-permanent SMTP or DNS failures
Success	int	The number of successful recipients
Expired	int	The expiration date to send the mailing was reached before all attempts were completed
Paused	int	The recipient was not sent to because the mailing was paused by an administrator
MailMergeSkipped	int	The recipient was not processed due to the skip command in mailmerge
Active	int	The number of recipients currently receiving mailing
Opens	int	The total number of times a message is opened
Created	date	The date that the mailing was created as an outmail record for sending
NotAttempted	int	The number of recipients not yet attempted
Clickthroughs	int	The total number of tracked clickthroughs for a message
Title	string	The title of the mailing
TotalRecipients	int	The total number of recipients
PermanentFailure	int	Permanent failures due to permanent SMTP or DNS failures, receipt of a bounce message, invalid email address or failure to retrieve the message body from the database
TotalUndelivered	int	The total number of recipients who have been attempted but failed
MailMergeAbort	int	The recipient was not sent to because the mailing was aborted during mail merging by the abort command
UniqueOpens	int	The total number of recipients tracked opening a message
Clickstreams	int	The total number of tracked clickstreams for a message
Pending	int	The pending number of recipients
URLs	Array of	All tracked URLs

	UrlTrackin gStruct	
MailingID	int	The outmail_.MessageID_ value
Retry	int	The number of recipients waiting for mailings who have already been tried but who did not receive due to a transient error

URLTrackingStruct

The URLTrackingStruct is returned as part of the TrackingSummaryStruct.

Name	Type	Description
UniqueOpens	string	Number of unique clicks for this URL.
Opens	string	Total number of clicks for this URL
URL	string	The URL

Enumerations

BanLogicEnum

Specifies the kind of ban in the MemberBanStruct.

Key	Description
A	Accepted. Bans addresses other than accepted addresses from joining
C	Conditionally accepted. Bans other addresses from joining, except those explicitly rejected with a "Reject" ban.
R	Banned. This member or domain may not join the list.

ConfirmUnsubEnum

Specifies whether unsubscribe requests should be confirmed.

Key	Description
0	Require unsubscribe confirmation when the request is questionable (e.g., the request is from a different address than the member's email address)
1	Always require unsubscribe confirmation
2	Never require unsubscribe confirmation

DeliveryReportEnum

Specifies at what interval(s) delivery reports should be emailed to administrators.

Key	Value
0	NoReports
1	ReportAfter1Hour
2	ReportAfter4Hours
4	ReportAfter12Hours
8	ReportAfter24Hours
16	ReportAfter48Hours

DocTypeEnum

Specifies the content document type.

Key	Description
BILLING	Billing invoice data
CONTENT	Content created to be sent to a list
MESSAGE	An internal message sometimes sent to a user or an administrator

EnableScriptingEnum

Specifies who may use scripting for a list.

Key	Description
0	Only administrators may use scripting
1	Any list member may use scripting
2	Scripting disabled

FieldTypeEnum

Key
integer
date
char1
char2
char3
varchar10
varchar20
varchar30
varchar40
varchar50
varchar60
varchar70
varchar80
varchar90
varchar100
varchar150
varchar200
varchar250

ListTypeEnum

Specifies the kind of list being created.

Key	Description
marketing	All list settings will be appropriate for a list intended for email marketing. The list will be configured so only administrators may send email to the list. If members reply to the message, their responses will go to the message's author. Open and Clickthrough tracking is on by default for all new mailings sent from an email marketing list.
	Full personalization of messages (conditional content, mail merge) is possible because full scripting is enabled for administrators. The list will accept messages from other lists as a child list. Mailing and clickthrough data will be kept forever.
	Messages sent to the list must be approved before they will be sent. The list settings may be changed later in the Utilities: List Settings menu.
announcement-moderated	All list settings will be appropriate for a one-way announcement list. The list will be configured so only administrators may send email to the list. If members reply to the message, their responses will go to the message's author.
	Open and Clickthrough tracking is off by default for all new mailings sent from an announcement list.
	Messages can be personalized because safe scripting is enabled for administrators. The list will not accept messages from other lists as a child list. Mailing and clickthrough data will be kept for 35 days.
	Messages sent to the list must be approved before they will be sent. The list settings may be changed later in the Utilities: List Settings menu.
discussion-moderated	All list settings will be appropriate for a moderated discussion list. The list will be configured so that all list members may send email to the list. If members reply to the message, their responses will go to the list. Messages sent to the list must be moderated by an administrator.
discussion-unmoderated	All list settings will be appropriate for an unmoderated discussion list. The list will be configured so that all list members may send email to the list. If members reply to the message, their responses will go to the list.

LoggingLevelEnum

Specifies the level of delivery logging for a list.

Key	Description
N	No logs
M	Displays errors but not SMTP transaction info
E	All errors, including SMTP transaction info
F	Errors and successes

MailFormatEnum

Specifies the mail format a member prefers.

Key	Description
T	Text only
H	HTML
M	Multipart-alternative

MailSectionEncodingEnum

Specifies the encoding for a section of a message.

Key	Description
8bit	8 bit encoding
quoted-printable	quoted-printable encoding
base64	base 64 encoding
7bit	7 bit encoding

MemberKindEnum

Specifies what kind of messages a member should receive from the list.

Key	Description
digest	A digest
mimedigest	A MIME digest
index	Subject line index
nomail	No mail
mail	Normal list mail

MemberListSecurityEnum

Specifies who may access the list of members.

Key	Description
0	Only administrators may access the list of members
1	List members may access the list of members
2	Anyone may access the list of members

MemberStatusEnum

Specifies the status of the member. Only members with status = normal will typically receive mail from a list.

Key	Description
normal	A normal member who receives mailings from the list
confirm	The person has requested to join this mailing list, has received an email message asking them to confirm their membership, but has not replied to the confirmation message.
private	The person has requested to join this mailing list, but because this mailing list is set to private, their request to join is waiting for an administrator to approve it by changing their membership to normal.
expired	The member had their membership set to expire on a certain date, and that date has past.
held	The member's email address was invalid, or bounced mailing list messages for several days.
unsub	The member was a member of the list at one time, but has since unsubscribed
referred	The member was invited to join using a referral, but has not yet joined the list.
needs-confirm	The member will receive a message asking them to confirm their membership, and then their status will be set to Confirm.
needs-hello	The member will receive a message welcoming them to the list, and their status will be Normal.
needs-goodbye	The member will receive a message informing them they have left the list, and their status will be Unsubscribed.

MergeCapOverrideEnum

Specifies the level at which this list can override server scripting levels.

Key	Description
0	No scripting permitted
1	Safe scripting permitted
2	Full scripting permitted
3	Use server settings

MessageTypeEnum

Specifies the type of document that should be sent.

Key	Description
confirm	The document which will be sent via email to new subscribers when they join the mailing list, and the list is set to confirm new subscribers.
hello	The document which will be sent via email to new members of this mailing list when they join. If blank, a default Hello message will be sent.
goodbye	The document which will be sent via email to members of this mailing list when they leave the mailing list. If blank, a default Goodbye message will be sent.
held	The document which will be sent via email to subscribers whose subscriptions are being held due to delivery problems. If blank, a default Held message will be sent.
private	The document which will be sent via email to new members who join but cannot receive any messages until an administrator subscribes them to the list.

ModeratedEnum

Specifies the kind of moderation required by a list.

Key	Description
no	No moderation required
all	All messages must be moderated
number	The first *n* number of messages for a member must be moderated. This number is defined by the value for ApproveNum in the ListStruct

MriVisibilityEnum

Specifies whether the Discussion Forum Interface is visible, hidden, or unavailble. Used by the ListStruct.

Key   **Description**
V   List is visible
H   List is hidden; users may acces the list by going to the list URL directly
I   List is invisible and inaccessible

NameRequiredEnum

Specifies whether a prospective member must provide a name when joining a list.

Key   **Description**
0   Providing a name is optional
1   Do not ask for a name
2   Providing a name is required

NewSubscriberPolicyEnum

Specifies what requirements there are (if any) for new members.

Key   **Description**
open   Anyone may join
private   Administrator must approve new members
closed   New members may only be added by administrator
password Members may join if they supply the list password

PostPasswordEnum

Specifies whether the user's password must be included in a message to be posted.

Key   **Description**
0   Password required
1   Password not required

RecencyOperatorEnum

Specifies whether a list's recency limit should be for more, fewer or exactly the number specified by the RecencyMailCount in the ListStruct.

Key   **Description**
m   More than the number of messages specified
f   Fewer than the number of messages specified
e   Exactly the number of messages specified

RecencyWhichEnum

Specifies whether a mailing's recency limit should be for more, fewer or exactly the number specified by the RecencyNumberOfMailings in the MailingStruct.

Key   **Description**
m   More than the number of messages specified
f   Fewer than the number of messages specified
e   Exactly the number of messages specified

ScriptingLevelEnum

Specifies the scripting level for the list.

Key   **Description**
0   No scripting permitted
1   Safe scripting permitted
2   Full scripting permitted

SegmentTypeEnum

Specifies the type of segment.

Key   **Description**
normal   Normal segment
triggered Triggered segment
sequential Sequential segment

SubscriptionReportEnum

Specifies the type of subscription report that should be mailed to list administrators.

Key	Description
none	Do not email sub/unsub delivery reports to list admins
monthly	Monthly: send a standard monthly report on the 1st of the month
monthly-email	Monthly: send a detailed monthly report on the 1st of the month
monthly-nochart	Monthly: send a short monthly report on the 1st of the month
daily	Daily: send a standard report every day
daily-email	Daily: send a detailed report every day
daily-nochart	Daily: send a short report every day
monday	Weekly: every Monday send a standard weekly report
monday-email	Weekly: every Monday send a detailed weekly report
monday-nochart	Weekly: every Monday send a short weekly report
tuesday	Weekly: every Tuesday send a standard weekly report
tuesday-email	Weekly: every Tuesday send a detailed weekly report
tuesday-nochart	Weekly: every Tuesday send a short weekly report
wednesday	Weekly: every Wednesday send a standard weekly report
wednesday-email	Weekly: every Wednesday send a detailed weekly report
wednesday-nochart	Weekly: every Wednesday send a short weekly report
thursday	Weekly: every Thursday send a standard weekly report
thursday-email	Weekly: every Thursday send a detailed weekly report
thursday-nochart	Weekly: every Thursday send a short weekly report
friday	Weekly: every Friday send a standard weekly report
friday-email	Weekly: every Friday send a detailed weekly report
friday-nochart	Weekly: every Friday send a short weekly report
saturday	Weekly: every Saturday send a standard weekly report
saturday-email	Weekly: every Saturday send a detailed weekly report
saturday-nochart	Weekly: every Saturday send a short weekly report
sunday	Weekly: every Sunday send a standard weekly report
sunday-email	Weekly: every Sunday send a detailed weekly report
sunday-nochart	Weekly: every Sunday send a short weekly report