

MBS Real Studio DotMacKit Plugin Documentation

Christian Schmitz

May 15, 2012

0.1 Introduction

This is the PDF version of the documentation for the Real Studio Plug-in from Monkeybread Software Germany. Plugin part: MBS Real Studio DotMacKit Plugin

0.2 Content

- 1 List of all topics 3
- 2 All items in this plugin 11
- 3 List of all classes 83
- 4 List of all global methods 85

Chapter 1

List of Topics

• 2 DotMacSDK	11
– 2.1 class DMMessageSubscriberMBS	11
* 2.1.1 Constructor(credentials as DMMemberAccountMBS)	11
* 2.1.1 credentials as DMMemberAccountMBS	12
* 2.1.1 messagePublisherWithCredentials(credentials as DMMemberAccountMBS) as DMMessageSubscriberMBS	12
* 2.1.1 subscribeToTopic(topic as DMTopicMBS) as DMTransactionMBS	12
* 2.1.1 topicNamed(name as string) as DMTopicMBS	12
* 2.1.1 topicNamed(name as string, memberName as string) as DMTopicMBS	13
* 2.1.1 topicNamed(name as string, memberName as string, creatorCode as string) as DMTopicMBS	13
* 2.1.1 unsubscribeFromTopic(topic as DMTopicMBS) as DMTransactionMBS	14
* 2.1.2 Handle as Integer	14
* 2.1.3 kDMLaunchAlways = 1	14
* 2.1.3 kDMLaunchDefault = 0	14
* 2.1.3 kDMLaunchInBackground = 2	15
– 2.3 class DMMessagePublisherMBS	18
* 2.3.1 Constructor(credentials as DMMemberAccountMBS)	18
* 2.3.1 createTopicNamed(topicName as string) as DMTransactionMBS	19
* 2.3.1 credentials as DMMemberAccountMBS	19
* 2.3.1 defaultTimeToLive as double	19
* 2.3.1 isSynchronous as boolean	19
* 2.3.1 kDMModify as string	20
* 2.3.1 kDMRead as string	20
* 2.3.1 kDMReadChildren as string	20

* 2.3.1 kDMWrite as string	21
* 2.3.1 messagePublisherWithCredentials(credentials as DMMemberAccountMBS) as DMMessagePublisherMBS	21
* 2.3.1 publishMessage(message as DMMessageMBS, topic as DMTopicMBS) as DMTransactionMBS	21
* 2.3.1 removeTopicNamed(topicName as string) as DMTransactionMBS	21
* 2.3.1 topicNamed(name as string) as DMTopicMBS	22
* 2.3.1 topicNamed(name as string, memberName as string) as DMTopicMBS	22
* 2.3.1 topicNamed(name as string, memberName as string, creatorCode as string) as DMTopicMBS	22
* 2.3.1 topics as DMTransactionMBS	23
* 2.3.1 transactionDelegate as DMTransactionDelegateMBS	23
* 2.3.2 Handle as Integer	23
* 2.3.3 kDMInsufficientStorage=102	24
* 2.3.3 kDMInvalidCredentials=101	24
* 2.3.3 kDMNetworkError=103	24
* 2.3.3 kDMResourceExists=105	24
* 2.3.3 kDMResourceNotFound=104	24
* 2.3.3 kDMServiceBusy=106	25
* 2.3.3 kDMSuccess=1	25
– 2.2 class DMSecondaryUserMBS	15
* 2.2.1 Constructor(name as string, password as string, owner as string, applicationID as string)	15
* 2.2.1 isSynchronous as boolean	16
* 2.2.1 name as string	16
* 2.2.1 owner as string	16
* 2.2.1 password as string	16
* 2.2.1 principalID as string	17
* 2.2.1 principalIDWithName(name as string, memberName as string) as string	17
* 2.2.1 secondaryUserWithName(name as string, password as string, owner as string, applicationID as string) as DMSecondaryUserMBS	17
* 2.2.1 transactionDelegate as DMTransactionDelegateMBS	17
* 2.2.2 Handle as Integer	18
– 2.11 class DMMessageMBS	65
* 2.11.1 Constructor(messageString as string)	65
* 2.11.1 expirationDate as date	65
* 2.11.1 isPublished as boolean	65
* 2.11.1 isSynchronous as boolean	66
* 2.11.1 message(messageString as string) as DMMessageMBS	66
* 2.11.1 messageID as string	66
* 2.11.1 messageString as string	66
* 2.11.1 payload as DMTransactionMBS	67

* 2.11.1 payloadType as string	67
* 2.11.1 properties as dictionary	67
* 2.11.1 publicationDate as date	67
* 2.11.1 publisher as string	68
* 2.11.1 setMessageString(messageString as string) as integer	68
* 2.11.1 setProperties(dic as dictionary) as integer	68
* 2.11.1 setTimeToLive(timeToLive as double) as integer	68
* 2.11.1 timeToLive as double	69
* 2.11.1 topic as DMTopicMBS	69
* 2.11.1 transactionDelegate as DMTransactionDelegateMBS	69
* 2.11.2 Handle as Integer	70
* 2.11.3 kDMInsufficientStorage=102	70
* 2.11.3 kDMInvalidCredentials=101	70
* 2.11.3 kDMNetworkError=103	70
* 2.11.3 kDMResourceExists=105	70
* 2.11.3 kDMResourceNotFound=104	71
* 2.11.3 kDMServiceBusy=106	71
* 2.11.3 kDMSuccess=1	71
– 2.6 class DMTransactionMBS	31
* 2.6.1 abort	31
* 2.6.1 bytesTransferred as Int64	31
* 2.6.1 contentLength as Int64	31
* 2.6.1 DataArrayResult as Memoryblock()	31
* 2.6.1 DataResult as Memoryblock	32
* 2.6.1 DictionaryResult as dictionary	32
* 2.6.1 errorType as integer	32
* 2.6.1 httpStatusCode as integer	32
* 2.6.1 IsDataArrayResult as boolean	33
* 2.6.1 IsDataResult as boolean	33
* 2.6.1 IsDictionaryResult as boolean	33
* 2.6.1 isFinished as boolean	33
* 2.6.1 IsNumberResult as boolean	34
* 2.6.1 IsStringArrayResult as boolean	34
* 2.6.1 IsStringResult as boolean	34
* 2.6.1 isSuccessful as boolean	34
* 2.6.1 NumberResult as double	34
* 2.6.1 Result as variant	35
* 2.6.1 StringArrayResult as string()	35
* 2.6.1 StringResult as string	35
* 2.6.1 transactionID as string	35
* 2.6.1 transactionState as integer	35

* 2.6.1 uri as string	36
* 2.6.2 context as variant	36
* 2.6.2 Handle as Integer	36
* 2.6.3 kDMTransactionAborted=4	37
* 2.6.3 kDMTransactionActive=2	37
* 2.6.3 kDMTransactionHadError=5	37
* 2.6.3 kDMTransactionNotStarted=1	37
* 2.6.3 kDMTransactionSuccessful=3	38
– 2.4 Globals	25
* 2.4 LoadDotMacKitFrameworkMBS(path as folderitem) as boolean	25
– 2.5 class DMTopicMBS	26
* 2.5.1 applicationID as string	26
* 2.5.1 credentials as DMMemberAccountMBS	26
* 2.5.1 exists as DMTransactionMBS	26
* 2.5.1 isAutoArrivalEnabledForMessagePayloads as boolean	27
* 2.5.1 isSynchronous as boolean	27
* 2.5.1 messages as DMMessageMBS()	28
* 2.5.1 messageWithID(messageID as string) as DMMessageMBS	28
* 2.5.1 name as string	28
* 2.5.1 newestMessage as DMMessageMBS	28
* 2.5.1 newMessages as DMMessageMBS()	29
* 2.5.1 oldestMessage as DMMessageMBS	29
* 2.5.1 owner as string	29
* 2.5.1 removeMessages as DMTransactionMBS	29
* 2.5.1 removeMessageWithID(messageID as string) as DMTransactionMBS	29
* 2.5.1 resetTimeToLive(ttl as double, messageID as string) as DMTransactionMBS	30
* 2.5.1 transactionDelegate as DMTransactionDelegateMBS	30
* 2.5.2 Handle as Integer	30
– 2.7 class DMTransactionDelegateMBS	38
* 2.7.1 Constructor	38
* 2.7.2 Handle as Integer	38
* 2.7.3 transactionAborted(Transaction as DMTransactionMBS)	39
* 2.7.3 transactionHadError(Transaction as DMTransactionMBS)	39
* 2.7.3 transactionSuccessful(Transaction as DMTransactionMBS)	39
– 2.8 class DMiDiskSessionMBS	39
* 2.8.1 account as DMAccountMBS	40
* 2.8.1 basicAttributesAtPath(path as string) as DMTransactionMBS	40
* 2.8.1 Constructor(c as DMAccountMBS) as DMiDiskSessionMBS	40
* 2.8.1 Constructor(c as DMAccountMBS, memberName as string) as DMiDiskSessionMBS	41
* 2.8.1 Constructor(c as DMSecondaryUserMBS) as DMiDiskSessionMBS	41

* 2.8.1 Constructor(c as DMSecondaryUserMBS, memberName as string) as DMiDiskSessionMBS	41
* 2.8.1 contentsAtPath(path as string) as memoryblock	42
* 2.8.1 copyPath(source as string, destination as string) as boolean	42
* 2.8.1 copyResourceAtPath(sourcePath as string, destinationPath as string) as DMTransactionMBS	42
* 2.8.1 createDirectoryAtPath(path as string) as boolean	42
* 2.8.1 createFileAtPath(path as string, contentData as memoryblock) as boolean	43
* 2.8.1 credentials as variant	43
* 2.8.1 defaultLockDuration as double	43
* 2.8.1 deleteResourceAtPath(path as string) as DMTransactionMBS	44
* 2.8.1 directoryContentsAtPath(Path as string) as string()	44
* 2.8.1 extendedAttributesAtPath(path as string) as DMTransactionMBS	44
* 2.8.1 fileExistsAtPath(path as string) as boolean	45
* 2.8.1 fileExistsAtPath(path as string, byref isDirectory as boolean) as boolean	45
* 2.8.1 getDataAtPath(path as string) as DMTransactionMBS	45
* 2.8.1 getDataAtPath(sourcePath as string, IfModifiedSince as date) as DMTransactionMBS	46
* 2.8.1 getDataAtPath(sourcePath as string, IfModifiedSince as date, ByteStart as double, ByteLength as double) as DMTransactionMBS	46
* 2.8.1 iDiskSessionWithCredentials(c as DMAccountMBS) as DMiDiskSessionMBS	47
* 2.8.1 iDiskSessionWithCredentials(c as DMAccountMBS, memberName as string) as DMiDiskSessionMBS	47
* 2.8.1 iDiskSessionWithCredentials(c as DMSecondaryUserMBS) as DMiDiskSessionMBS	47
* 2.8.1 iDiskSessionWithCredentials(c as DMSecondaryUserMBS, memberName as string) as DMiDiskSessionMBS	48
* 2.8.1 isSynchronous as boolean	48
* 2.8.1 kDMModify as string	49
* 2.8.1 kDMRead as string	49
* 2.8.1 kDMReadChildren as string	49
* 2.8.1 kDMWrite as string	49
* 2.8.1 listCollectionAtPath(path as string) as DMTransactionMBS	49
* 2.8.1 lockResourceAtPath(path as string) as DMTransactionMBS	50
* 2.8.1 lockResourceAtPath(Path as string, lockDuration as double) as DMTransactionMBS	50
* 2.8.1 makeCollectionAtPath(path as string) as DMTransactionMBS	51
* 2.8.1 movePath(source as string, destination as string) as boolean	51
* 2.8.1 moveResourceAtPath(sourcePath as string, destinationPath as string) as DMTransactionMBS	51
* 2.8.1 owner as string	52
* 2.8.1 putData(data as memoryblock, destinationPath as string) as DMTransactionMBS	52
* 2.8.1 putData(data as memoryblock, destinationPath as string, ByteStart as double, ByteLength as double) as DMTransactionMBS	52

* 2.8.1 putLocalFileAtPath(localPath as string, destinationPath as string) as DMTransactionMBS	53
* 2.8.1 quotaAttributes as DMTransactionMBS	53
* 2.8.1 relockResourceAtPath(path as string) as DMTransactionMBS	53
* 2.8.1 relockResourceAtPath(Path as string, lockDuration as double) as DMTransactionMBS	54
* 2.8.1 removeFileAtPath(path as string) as boolean	54
* 2.8.1 resourceExistsAtPath(path as string) as boolean	54
* 2.8.1 transactionDelegate as DMTransactionDelegateMBS	55
* 2.8.1 unlockResourceAtPath(path as string) as DMTransactionMBS	55
* 2.8.2 Handle as Integer	55
* 2.8.3 kDMContentLength = "DAV:getcontentlength"	56
* 2.8.3 kDMContentType = "DAV:getcontenttype"	56
* 2.8.3 kDMDateCreated = "DAV:creationdate"	56
* 2.8.3 kDMDebugLogging = "DMDebugLogging"	56
* 2.8.3 kDMDisplayName = "DAV:displayname"	56
* 2.8.3 kDMErrorLogging = "DMErrorLogging"	57
* 2.8.3 kDMGroup = "Group"	57
* 2.8.3 kDMiDiskQuotaInBytes = "DAV:quota"	57
* 2.8.3 kDMiDiskSpaceUsedInBytes = "DAV:quotaused"	57
* 2.8.3 kDMIsCollection = "IsCollection"	57
* 2.8.3 kDMIsLocked = "IsLocked"	58
* 2.8.3 kDMLastModified = "DAV:getlastmodified"	58
* 2.8.3 kDMLockOwner = "LockOwner"	58
* 2.8.3 kDMLockTimeout = "LockTimeout"	58
* 2.8.3 kDMLockToken = "LockToken"	58
* 2.8.3 kDMLoggingLevel = "com.apple.DotMacKit.LoggingLevel"	59
* 2.8.3 kDMMemberAccount = ".Mac Member"	59
* 2.8.3 kDMPrincipalName = "PrincipalName"	59
* 2.8.3 kDMPrincipalOwner = "PrincipalOwner"	59
* 2.8.3 kDMPrincipalType = "PrincipalType"	59
* 2.8.3 kDMSecondaryUser = "Secondary User"	60
* 2.8.3 kDMURI = "URI"	60
– 2.10 class DMGroupMBS	62
* 2.10.1 addMembers(members() as string) as DMTransactionMBS	62
* 2.10.1 isSynchronous as boolean	62
* 2.10.1 members as DMTransactionMBS	63
* 2.10.1 name as string	63
* 2.10.1 owner as string	63
* 2.10.1 principalID as string	63
* 2.10.1 principalIDWithName(name as string, memberName as string) as string	64
* 2.10.1 removeMembers(members() as string) as DMTransactionMBS	64

* 2.10.1 transactionDelegate as DMTransactionDelegateMBS	64
* 2.10.2 Handle as Integer	64
– 2.9 class DMAccountMBS	60
* 2.9.1 applicationID as string	60
* 2.9.1 applicationName as string	60
* 2.9.1 name as string	61
* 2.9.1 owner as string	61
* 2.9.1 password as string	61
* 2.9.1 principalID as string	61
* 2.9.1 principalIDWithName(name as string, memberName as string) as string	61
* 2.9.2 Handle as Integer	62
– 2.12 class DMMemberAccountMBS	71
* 2.12.1 accountFromPreferencesWithApplicationID(creatorCode as string) as DMMemberAccountMBS	72
* 2.12.1 accountUpgradeURL as string	72
* 2.12.1 accountWithName(name as string, password as string, creatorCode as string) as DMMemberAccountMBS	72
* 2.12.1 canUpgradeAccount as DMTransactionMBS	72
* 2.12.1 Constructor(creatorCode as string)	73
* 2.12.1 Constructor(name as string, password as string, creatorCode as string)	73
* 2.12.1 createGroupWithName(name as string, members() as string) as DMTransactionMBS	73
* 2.12.1 createSecondaryUserWithName(name as string, password as string) as DMTransactionMBS	74
* 2.12.1 daysLeftUntilExpiration as DMTransactionMBS	74
* 2.12.1 groupNamed(name as string) as DMGroupMBS	74
* 2.12.1 groupNames as DMTransactionMBS	74
* 2.12.1 isSynchronous as boolean	74
* 2.12.1 matchesSystemCredentials as boolean	75
* 2.12.1 removeGroupNamed(name as string) as DMTransactionMBS	75
* 2.12.1 removeSecondaryUserNamed(name as string) as DMTransactionMBS	75
* 2.12.1 resetPassword(password as string, SecondaryUserNamed as string) as DMTransactionMBS	76
* 2.12.1 secondaryUserNames as DMTransactionMBS	76
* 2.12.1 servicesAvailableForAccount as DMTransactionMBS	76
* 2.12.1 signUpNewMember	76
* 2.12.1 signUpNewMemberWithApplicationID(appID as string)	77
* 2.12.1 signUpURL as string	77
* 2.12.1 signUpURLWithApplicationID(appID as string) as string	77
* 2.12.1 transactionDelegate as DMTransactionDelegateMBS	78
* 2.12.1 upgradeAccount	78
* 2.12.1 validateCredentials as integer	78

* 2.12.2 kDMBackupService = "Backup"	78
* 2.12.2 kDMEmailService = "Email"	79
* 2.12.2 kDMiDiskService = "iDisk"	79
* 2.12.2 kDMInsufficientStorage=102	79
* 2.12.2 kDMInvalidCredentials=101	79
* 2.12.2 kDMInvalidParameter=107	79
* 2.12.2 kDMiSyncService = "iSync"	80
* 2.12.2 kDMNetworkError=103	80
* 2.12.2 kDMResourceExists=105	80
* 2.12.2 kDMResourceNotFound=104	80
* 2.12.2 kDMServiceBusy=106	80
* 2.12.2 kDMSuccess=1	81
* 2.12.2 kDMUndefined=-1	81
* 2.12.2 kDMUnknownError=100	81
* 2.12.2 kDMWebHostingService = "WebHosting"	81

Chapter 2

DotMacSDK

2.1 class DMMMessageSubscriberMBS

class DMMMessageSubscriberMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** The class for a message subscriber.

Notes: The thread-safe DMMMessageSubscriber session object is used to subscribe to specific topics and to access new messages in topics of interest. The .Mac credentials held by the DMMMessageSubscriber session represent the subscriber. The subscriber can subscribe to topics that it itself owns and to accessible topics owned by other .Mac members. [Note: Only .Mac members can publish or subscribe to messages. Currently, secondary users cannot.]

2.1.1 Methods

Constructor(credentials as DMMAccountMBS)

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Creates an initialized DMMMessageSubscriber session that will use the provided objects credentials when subscribing to DMTopics or accessing a DMTopics messages.

Notes: The provided .Mac member credentials can be used to subscribe to accessible topics owned by any .Mac member.

credentials as DMMemberAccountMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Returns the .Mac member object containing the credentials set for this DMMessagePublisher session.

Notes: These credentials are used to determine which topics can be subscribed to using this session.

messagePublisherWithCredentials(credentials as DMMemberAccountMBS) as DMMessageSubscriberMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Returns a new DMMessageSubscriber session that will use the provided objects credentials when subscribing to DMTopics or accessing a DMTopics messages.

Notes: The provided .Mac member credentials can be used to subscribe to accessible topics owned by any .Mac member.

subscribeToTopic(topic as DMTopicMBS) as DMTransactionMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Creates a persistent subscription to the given topic.

Notes: This subscription is specific to this session's .Mac member credentials, the machine and local OS X user that the calling application is running under and the calling application itself. As an argument, this method takes a DMTopic to which this session's .Mac member credentials have access. Calling this method creates an entry for the given subscription in the local user preferences, which will cause the subscription to be reactivated every time the calling application runs and creates a DMMessageSubscriber object. Therefore, subscribeToTopic only needs to be called once for a given topic by an application. Calling unsubscribeFromTopic removes the persistent subscription from the preferences. This method returns a DMTransaction object whose -isSuccessful method can be used to test for success.

topicNamed(name as string) as DMTopicMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Returns an autoreleased DMTopic instance with the provided name.

Notes: The returned topic's owner will be the .Mac member name specified by this DMMessageSubscribers credentials and the topics application ID will be set to the calling applications creator code. This method is used to obtain a DMTopic object that represents an existing topic on .Mac. The credentials set for this DMMessageSubscriber instance will be used to access the topic represented by the returned DMTopic object. To access messages in a topic, a DMTopic object is required. If you are trying to obtain a DMTopic object that represents a topic not owned by this DMMessageSubscriber's credentials, use the other topicNamed

methods, as described below.

See also:

- 2.1.1 topicNamed(name as string, memberName as string) as DMTopicMBS 13
- 2.1.1 topicNamed(name as string, memberName as string, creatorCode as string) as DMTopicMBS 13

topicNamed(name as string, memberName as string) as DMTopicMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Returns an DMTopic instance with the provided name and owner, where the owner is the name of the .Mac member who hosts the given topic.

Notes: The returned topic's application ID will be set to the calling applications creator code. This method is used to obtain a DMTopic object that represents an existing topic on .Mac. The credentials set for this DMMessageSubscriber instance will be used to access the topic represented by the returned DMTopic object. To access messages in a topic, a DMTopic object is required. If you are trying to obtain a DMTopic object that represents a topic that was created by an application different from the calling application, use the topicNamed method, as described below.

See also:

- 2.1.1 topicNamed(name as string) as DMTopicMBS 12
- 2.1.1 topicNamed(name as string, memberName as string, creatorCode as string) as DMTopicMBS 13

topicNamed(name as string, memberName as string, creatorCode as string) as DMTopicMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Returns a DMTopic instance with the provided name, owner and application ID, where the owner is the name of the .Mac member who hosts the given topic and the application ID is the creator code of the application that originally created the given topic.

Notes: This method is used to obtain a DMTopic object that represents an existing topic on .Mac. The credentials set for this DMMessageSubscriber instance will be used to access the topic represented by the returned DMTopic object. To access messages in a topic, a DMTopic object is required.

See also:

- 2.1.1 topicNamed(name as string) as DMTopicMBS 12
- 2.1.1 topicNamed(name as string, memberName as string) as DMTopicMBS 13

unsubscribeFromTopic(topic as DMTopicMBS) as DMTransactionMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Removes an existing subscription to the given topic.

Notes: Removes a subscription specific to the .Mac member for this session, the machine and local OS X user that the calling application is running under and the calling application itself. Calling this method deletes the entry for the given subscription from the local user preferences. As an argument, this method takes a DMTopic object to which this session's .Mac member credentials have access. Returns a DMTransaction object whose `isSuccessful` method can be used to test for success.

2.1.2 Properties**Handle as Integer**

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** The internal object reference.

Notes: (Read and Write property)

2.1.3 Constants**kDMLaunchAlways = 1**

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of the auto-launch constants.

Notes:

Auto-launch constants are used for passing launch options to `DMMessageSubscriber` methods that schedule applications and tools to launch when a given topic changes (these launch options can be ord together as appropriate).

Always launch a new instance of application (applies to bundled apps only).

kDMLaunchDefault = 0

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of the auto-launch constants.

Notes:

Auto-launch constants are used for passing launch options to DMMessageSubscriber methods that schedule applications and tools to launch when a given topic changes (these launch options can be ord together as appropriate).

Only one instance, launch in foreground.

kDMLaunchInBackground = 2

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of the auto-launch constants.

Notes:

Auto-launch constants are used for passing launch options to DMMessageSubscriber methods that schedule applications and tools to launch when a given topic changes (these launch options can be ord together as appropriate).

Dont bring launched app to foreground (applies to bundled apps only).

2.2 class DMSecondaryUserMBS

class DMSecondaryUserMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** The class for a secondary user.

2.2.1 Methods

Constructor(name as string, password as string, owner as string, applicationID as string)

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Creates a newly initialized DMSecondaryUser object keyed to a specific applications identification string.

Notes:

The secondary user is owned by the .Mac member with the given member name. Pass the unique 4-character application identifier obtained from

<http://developer.apple.com/datatype/creatorcode.html>

isSynchronous as boolean

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Whether transactions are performed synchronous (true) or asynchronous (false).

Notes:

The default mode is asynchronous.

This property is used to set the synchronicity state for generated transactions. Passing true will cause the object's transaction methods to block until they have either completed their transaction or failed with an error. Passing false will cause the methods to that return a DMTransaction object to return immediately, using any previously set delegate object to signal completion or error. It is safe to call this method to switch between asynchronous and synchronous modes as needed—since an in-flight DMTransaction object uses the settings that were in place at the time it was created, calling this method will not affect any transaction already in progress. [Note: The default mode is asynchronous.]

(Read and Write computed property)

name as string

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** A username string is returned.

owner as string

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** The name of the .Mac member who owns the credential is returned.

password as string

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** A password string is returned.

principalID as string

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** This method is implemented by DMMemberAccount, DMSecondaryUser or DMGroup and returns the appropriate principal ID string.

principalIDWithName(name as string, memberName as string) as string

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Returns a principal ID string identifying the principal with the given name and owner (the owner is the name of the .Mac member that owns the given principal).

Notes: This class method is useful when the caller needs to obtain a principle ID representing a DMMemberAccount, DMSecondaryUser or DMGroup without creating an instance of one of those classes.

secondaryUserWithName(name as string, password as string, owner as string, applicationID as string) as DMSecondaryUserMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Returns an autoreleased instance of DMSecondaryUser keyed to a specific applications identification string.

Notes:

The secondary user is owned by the .Mac member with the given member name. Pass the unique 4-character application identifier obtained from

<http://developer.apple.com/datatype/creatorcode.html>

transactionDelegate as DMTransactionDelegateMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** The transaction delegate in use for all the transactions made by this object.

Notes:

This method is used to set the delegate object which will receive status messages from asynchronous transactions generated by the object that conforms to this protocol. The object passed in this method's delegate parameter must conform to the informal DMTransactionDelegate protocol. Transaction delegates receive their messages on the run loop originally used to create the given DMTransaction. It is safe to call this

method to change the delegate as needed—since an in-flight DMTransaction object uses the settings that were in place at the time it was created, calling this method will not affect any transaction already in progress. A transaction delegate object does not receive callbacks for transactions executed synchronously. (Read and Write computed property)

2.2.2 Properties

Handle as Integer

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** The internal object reference.

Notes: (Read and Write property)

2.3 class DMMessagePublisherMBS

class DMMessagePublisherMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** The class for a message publisher.

Notes: The DMMessagePublisher session object is used to publish DMMessages to DMTopics and is thread-safe. The .Mac member credential held by the DMMessagePublisher session represents the member who will be publishing messages to topics. The publishing member can publish messages to topics that it itself owns or to topics owned by other .Mac members. The DMMessagePublisher session conforms to the DMSecurity protocol and DMTopics are considered securable entities that can be passed to DMSecurity methods for this session. [Note: Only .Mac members can publish or subscribe to messages. Currently, secondary users cannot.]

2.3.1 Methods

Constructor(credentials as DMMemberAccountMBS)

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Creates a newly initialized DMMessagePublisher session that will use the provided objects credentials when publishing DMMessages or otherwise accessing or modifying DMTopics.

Notes: The publishing member can publish messages to topics that it itself owns or to topics owned by

other .Mac members.

createTopicNamed(topicName as string) as DMTransactionMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Adds a new, empty DMTopic to .Mac. The new topic will be owned by the .Mac member specified by this DMMessagePublishers credentials.

Notes: Returns a DMTransaction object whose isSuccessfull method can be used to test for success.

credentials as DMMemberAccountMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Returns the .Mac member object containing the credentials set for this DMMessagePublisher session.

Notes: These credentials are used to determine which topics can have messages published to them or be otherwise modified using this session.

defaultTimeToLive as double

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** The default time-to-live that will be used by all DMMessages published by this session.

Notes:

The default time-to-live value is 7 days. Returns an NSTimeInterval (precision in seconds).

Sets the default time-to-live that will be used by all DMMessages published by this session. The default time-to-live value cannot exceed 30 days. Takes an NSTimeInterval (precision in seconds). If the provided ttl value not in the valid range, kDMInvalidParameter will be returned; otherwise, this method returns kDMSuccess.

(Read and Write computed property)

isSynchronous as boolean

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Whether transactions are performed synchronous (true) or asynchronous (false).

Notes:

The default mode is asynchronous.

This property is used to set the synchronicity state for generated transactions. Passing true will cause the object's transaction methods to block until they have either completed their transaction or failed with an error. Passing false will cause the methods to that return a DMTransaction object to return immediately, using any previously set delegate object to signal completion or error. It is safe to call this method to switch between asynchronous and synchronous modes as needed—since an in-flight DMTransaction object uses the settings that were in place at the time it was created, calling this method will not affect any transaction already in progress. [Note: The default mode is asynchronous.]
(Read and Write computed property)

kDMModify as string

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** One of the privileges constants.

Notes: Includes modify access to a given entity. Does not include create or delete access.

kDMRead as string

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** One of the privileges constants.

Notes: Includes read and list access to a given entity.

kDMReadChildren as string

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** One of the privileges constants.

Notes: Includes read and list access to children of a given entity only. Does not include read and list access to the entity itself.

kDMWrite as string

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** One of the privileges constants.

Notes: Includes create, modify and delete access to a given entity.

messagePublisherWithCredentials(credentials as DMMemberAccountMBS) as DMMessagePublisherMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Creates a DMMessagePublisher session that will use the provided objects credentials when publishing DMMessages or otherwise accessing or modifying DMTopics.

Notes: The publishing member can publish messages to topics that it itself owns or to topics owned by other .Mac members.

publishMessage(message as DMMessageMBS, topic as DMTopicMBS) as DMTransactionMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Publishes a new DMMessage to a DMTopic that already exists on .Mac (an existing topic will have been previously created on .Mac by the topics owner using DMMessagePublishers createTopicNamed method).

Notes: The default timetolive for a newly published DMMessage is 7 days. This default can be overridden using the sessions setDefaultTimeToLive: method. Timetolive can also be set on a messageby message basis using DMMessages setTimeToLive: method. This method returns a DMTransaction object whose isSuccessful method can be used to test for success. [Note: If .Mac is too busy to handle the message publication request, kDMServiceBusy will be returned by the transactions errorType method, and publication should be retried later. If the given DMMessage has already been published to .Mac, kDMResourceExists will be returned by the transactions errorType method.]

removeTopicNamed(topicName as string) as DMTransactionMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Removes the named DMTopic from .Mac, including all of the messages it contains.

Notes: A topic can be removed only by its owner, using the same application that originally created it. Therefore, the .Mac member name specified by this DMMessagePublishers credentials must be the owner of the topic and the calling application must be the same application used to originally create the topic. Returns a DMTransaction object whose isSuccessful method can be used to test for success.

topicNamed(name as string) as DMTopicMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Returns an autoreleased DMTopic instance with the provided name.

Notes: The returned topic's owner will be the .Mac member name specified by this DMMessagePublishers credentials and the topics application ID will be set to the calling applications creator code. This method is used to obtain a DMTopic object that represents an existing topic on .Mac. The credentials set for this DMMessagePublisher instance will be used to access or modify the topic represented by the returned DMTopic object. To publish messages to a topic or to list the messages in a topic, a DMTopic object is required. If you are trying to obtain a DMTopic object that represents a topic not owned by this DMMessagePublisher's credentials.

See also:

- 2.3.1 topicNamed(name as string, memberName as string) as DMTopicMBS 22
- 2.3.1 topicNamed(name as string, memberName as string, creatorCode as string) as DMTopicMBS 22

topicNamed(name as string, memberName as string) as DMTopicMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Returns an autoreleased DMTopic instance with the provided name and owner, where the owner is the name of the .Mac member who hosts the given topic.

Notes: The returned topic's application ID will be set to the calling applications creator code. This method is used to obtain a DMTopic object that represents an existing topic on .Mac. The credentials set for this DMMessagePublisher instance will be used to access or modify the topic represented by the returned DMTopic object. To publish messages to a topic or to list the messages in a topic, a DMTopic object is required. If you are trying to obtain a DMTopic object that represents a topic that was created by an application different from the calling application, use the topicNamed method, as described below.

See also:

- 2.3.1 topicNamed(name as string) as DMTopicMBS 22
- 2.3.1 topicNamed(name as string, memberName as string, creatorCode as string) as DMTopicMBS 22

topicNamed(name as string, memberName as string, creatorCode as string) as DMTopicMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Returns an autoreleased DMTopic instance with the provided name, owner and application ID, where the owner is the name of the .Mac member who hosts the given topic and the application ID is the creator code of the application that originally created the topic.

Notes: This method is used to obtain a DMTopic object that represents an existing topic on .Mac. The credentials set for this DMMessagePublisher instance will be used to access or modify the topic represented

by the returned DMTopic object. To publish messages to a topic or to list the messages in a topic, a DMTopic object is required.

See also:

- 2.3.1 topicNamed(name as string) as DMTopicMBS 22
- 2.3.1 topicNamed(name as string, memberName as string) as DMTopicMBS 22

topics as DMTransactionMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Lists the topics that this DMMessagePublisher can publish messages to or remove messages from.

Notes: Each topic listed may be owned by this session's .Mac member credentials or by another .Mac member. Returns a DMTransaction object whose isSuccessful method can be used to test for success and whose result method, upon success, returns an NSArray of all accessible DMTopic objects.

transactionDelegate as DMTransactionDelegateMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** The transaction delegate in use for all the transactions made by this object.

Notes:

This method is used to set the delegate object which will receive status messages from asynchronous transactions generated by the object that conforms to this protocol. The object passed in this method's delegate parameter must conform to the informal DMTransactionDelegate protocol. Transaction delegates receive their messages on the run loop originally used to create the given DMTransaction. It is safe to call this method to change the delegate as needed—since an in-flight DMTransaction object uses the settings that were in place at the time it was created, calling this method will not affect any transaction already in progress. A transaction delegate object does not receive callbacks for transactions executed synchronously. (Read and Write computed property)

2.3.2 Properties

Handle as Integer

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** The internal object reference.

Notes: (Read and Write property)

2.3.3 Constants

kDMInsufficientStorage=102

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of the DotMacKit status constants.

Notes: The .Mac account does not have sufficient disk space to accommodate the given operation.

kDMInvalidCredentials=101

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of the DotMacKit status constants.

Notes: The .Mac credentials set for this object do not have access to a .Mac account.

kDMNetworkError=103

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of the DotMacKit status constants.

Notes: Unable to interact with .Mac because of a networking error or lack of network availability.

kDMResourceExists=105

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of the DotMacKit status constants.

Notes: Indicates an error due to the pre-existence of a given resource.

kDMResourceNotFound=104

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of the DotMacKit status constants.

Notes: A required resource was not found.

kDMServiceBusy=106

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of the DotMacKit status constants.

Notes: The given .Mac service was too busy to perform the requested operation and the operation should be retried later.

kDMSuccess=1

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of the DotMacKit status constants.

Notes: The operation was successful.

2.4 Globals

LoadDotMacKitFrameworkMBS(path as folderitem) as boolean

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Loads the DotMacKit framework from the given folderitem.

Example:

```
dim f as FolderItem = getfolderitem("DotMacKit.framework")

if LoadDotMacKitFrameworkMBS(f) then
  MsgBox "Framework loaded."
else
  MsgBox "Framework failed to load."
end if
```

Notes: Returns true on success and false on failure.

2.5 class DMTopicMBS

class DMTopicMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** The class for a topic.
Notes:

DMTopics have DMMessages published into them using a DMMessagePublisher session. Users of a DMMessageSubscriber session can subscribe to a DMTopic object in order to be informed when new messages are published to it. DMTopic objects can only be created using the factory methods of a DMMessagePublisher or a DMMessageSubscriber session, and they inherit the credentials of their parent message session. A topics namespace is scoped to its owner (the hosting .Mac member) and the application that created it.

A DMTopic object can be passed to DMSecurity protocol methods that take an 'entity' argument, in order to set access privileges on the given topic. When a .Mac member is granted access to a DMTopic, an invitation is automatically generated and can be accessed using a DMMessageSubscriber session.

2.5.1 Methods

applicationID as string

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Returns the creator code of the application that created the topic.

credentials as DMMemberAccountMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Returns the .Mac member object containing the credentials that will be used to access or modify this DMTopic object.

Notes: The returned credentials are the same credentials held by the DMMessagePublisher or DMMessageSubscriber instance that was used to get a reference to this DMTopic.

exists as DMTransactionMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** The returned DMTransaction objects result method returns a Boolean value.

Notes: This value will be YES if the DMTopic exists and NO otherwise. [Note: If an error is encountered that prevents the existence check, the returned DMTransactions -errorType method can be used to identify

the error.]

isAutoArrivalEnabledForMessagePayloads as boolean

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** A boolean value indicating whether future messages received for the topic should have their payloads downloaded automatically to the local message cache

Notes:

This method sets whether future messages received for the topic should have their payloads downloaded automatically to the local message cache before any callbacks are made on a DMMessageSubscribers subscription delegate. When automatic downloading (or auto-arrival) of payloads is enabled, which is the default, a call to a DMMessages payload method simply returns the already-downloaded payload. When auto-arrival is not enabled for a given topic, a DMMessages payload method will do the extra work of downloading the payload on request. This method can be called regardless of whether a subscription to the given topic already exists. (A messages meta dataeverything but the payloadalways arrives automatically). [Note: Any auto-arrival default set by this method for a topic is automatically cleared from the local preferences cache if an unsubscribe call on the given topic is ever made in the future.] If this method is being called around the same time that a subscription to the given topic is being created, it is recommended that this method be called first, before the call to DMMessageSubscribers subscribeToTopic method. This ensures that the desired auto-arrival behavior is in place for the given topic before any topic-related callbacks are made into a DMMessageSubscribers subscription delegate.

This method returns a boolean value indicating whether future messages received for the topic should have their payloads downloaded automatically to the local message cache before any callbacks are made on a DMMessageSubscribers subscription delegate. When automatic downloading (or auto-arrival) of payloads is enabled, which is the default, a call to a DMMessages payload method simply returns the already-downloaded payload. When auto-arrival is not enabled for a given topic, a DMMessages -payload method will do the extra work of downloading the payload on request.

(Read and Write computed property)

isSynchronous as boolean

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Whether transactions are performed synchronous (true) or asynchronous (false).

Notes:

The default mode is asynchronous.

This property is used to set the synchronicity state for generated transactions. Passing true will cause the object's transaction methods to block until they have either completed their transaction or failed with an error. Passing false will cause the methods to that return a DMTransaction object to return immediately, using any previously set delegate object to signal completion or error. It is safe to call this method to switch between asynchronous and synchronous modes as needed—since an in-flight DMTransaction object uses the settings that were in place at the time it was created, calling this method will not affect any transaction already in progress. [Note: The default mode is asynchronous.]

(Read and Write computed property)

messages as DMMessageMBS()

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Returns all DMMessages found in this DMTopic on .Mac, as an NSArray of DMMessages.

Notes: Messages are ordered by publication time, starting with the earliest.

messageWithID(messageID as string) as DMMessageMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Retrieves a DMMessage using its globally unique identifier.

Notes: Returns a DMMessage if the provided message identifier is valid, and nil otherwise.

name as string

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Returns the name of the topic.

newestMessage as DMMessageMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Returns the most recent DMMessage in the topic, based on publication time.

newMessages as DMMessageMBS()

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Returns all DMMessages in this DMTopic that have not yet been accessed by the caller.

Notes: (The caller is defined as a combination of the .Mac credentials set for the session that created this DMTopic, the machine and local OS X user that the calling application is running under and the calling application itself.) This method will most often be called after a DMMessageSubscriber sessions subscription delegate is informed of a change in this DMTopic. Returns an NSArray of DMMessages ordered by publication time, starting with the earliest.

oldestMessage as DMMessageMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Returns the oldest DMMessage in the topic, based on publication time.

owner as string

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Returns the name of the .Mac member who owns and hosts the topic.

removeMessages as DMTransactionMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Removes all DMMessages from this DMTopic on .Mac.

Notes: Returns a DMTransaction object whose isSuccessful method can be used to test for success. This method can only be called if the .Mac credentials being used have write access to the given topic.

removeMessageWithID(messageID as string) as DMTransactionMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Removes the identified DMMessage from this DMTopic on .Mac.

Notes: Returns a DMTransaction object whose isSuccessful method can be used to test for success. This method can only be called if the .Mac credentials being used have write access to the given topic.

resetTimeToLive(ttl as double, messageID as string) as DMTransactionMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Resets the time-to-live value of the DMMessage specified by given identifier.

Notes: Returns a DMTransaction object whose `-isSuccessful` method can be used to test for success. The provided time-to-live value, an `NSTimeInterval` with precision in seconds, cannot be greater than 30 days. This method can only be called if the `.Mac` credentials being used have write access to the given topic.

transactionDelegate as DMTransactionDelegateMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** The transaction delegate in use for all the transactions made by this object.

Notes:

This method is used to set the delegate object which will receive status messages from asynchronous transactions generated by the object that conforms to this protocol. The object passed in this method's delegate parameter must conform to the informal `DMTransactionDelegate` protocol. Transaction delegates receive their messages on the run loop originally used to create the given `DMTransaction`. It is safe to call this method to change the delegate as needed—since an in-flight `DMTransaction` object uses the settings that were in place at the time it was created, calling this method will not affect any transaction already in progress. A transaction delegate object does not receive callbacks for transactions executed synchronously. (Read and Write computed property)

2.5.2 Properties

Handle as Integer

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** The internal object reference.

Notes: (Read and Write property)

2.6 class DMTransactionMBS

class DMTransactionMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** The class for a transaction.

2.6.1 Methods

abort

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Stops the current transaction if it is running asynchronously, halting its progress.

Notes: If the transaction is already complete (or resulted in an error) this method has no effect.

bytesTransferred as Int64

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Most useful for asynchronous operations, this method returns the number of bytes downloaded or uploaded thus far.

contentLength as Int64

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Returns the length in bytes of the content being downloaded or uploaded.

Notes: This returned length does not change over the life of the given transaction, except in the case where this method is called immediately after a GET request is started, before the content length has been received from the server this method returns kDMUndefined if called too early.

DataArrayResult as Memoryblock()

Plugin Version: 11.2 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** The result of this transaction as a memoryblock array.

Notes: Returns the result object of the given transaction as defined by the type of request the transaction

is handling, if currently available and applicable. Otherwise returns an empty array.

DataResult as Memoryblock

Plugin Version: 11.2 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** The result of the transaction as a memoryblock.

Notes: Returns the result object of the given transaction as defined by the type of request the transaction is handling, if currently available and applicable. Otherwise returns nil.

DictionaryResult as dictionary

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Returns true if the result contains a dictionary.

Notes: Returns the result object of the given transaction as defined by the type of request the transaction is handling, if currently available and applicable. Otherwise returns nil.

errorType as integer

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** The error type.

Notes: If an error occurred while generating the response associated with this object, this method will return a specific error type, as defined in the Constants section of this document. If an unknown error occurred, `kDMUnknownError` is returned. However, if no error occurred or the response has not yet completed when this method is called, `kDMUndefined` will be returned. [Note: If the `http` status code that was received is considered a failure code for the given transaction, `kDMUnknownError` is returned the `http` status code can be retrieved using the `httpStatusCode` method of this class.]

httpStatusCode as integer

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Returns the HTTP status code for the request if applicable.

Notes:

Check RFC 2616 (HTTP/1.1) for a complete list of all of the applicable status codes. Some of the most noteworthy ones are:

- 200 Success
- 304 Not Modified
- 401 Authorization required
- 403 Forbidden. Your authorization credentials will not allow the transaction

attempted.

404 Not Found

[Note: kDMUndefined will be returned if the transaction has not yet completed, if an http status code is not applicable for the given transaction type, or if the transaction failed before an HTTP status code could even be received, as might be the case if the client computer could not contact the .Mac environment for some reason.]

IsDataArrayResult as boolean

Plugin Version: 11.2 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Returns true if the result contains a data array.

IsDataResult as boolean

Plugin Version: 11.2 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Returns true if the result contains a data object.

IsDictionaryResult as boolean

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Returns true if the result contains a dictionary.

isFinished as boolean

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Whether the transaction has finished.

Notes:

This method will return true if the given transaction is no longer active, due to success, error or cancellation. Otherwise, false is returned.

IsNumberResult as boolean

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Returns true if the result contains a number.

IsStringArrayResult as boolean

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Returns true if the result contains a string array.

IsStringResult as boolean

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Returns true if the result contains a string.

isSuccessful as boolean

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Whether the transaction was successful.

Notes: Returns true if the transaction has finished successfully, no network errors occurred, and a valid http status code was returned. Otherwise, returns false.

NumberResult as double

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** The result of this transaction as a number.

Notes: Returns the result object of the given transaction as defined by the type of request the transaction is handling, if currently available and applicable. Otherwise returns 0.

Result as variant

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** The result as a variant.

Notes: Returns the result object of the given transaction as defined by the type of request the transaction is handling, if currently available and applicable. Otherwise returns nil.

StringArrayResult as string()

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** The result of this transaction as a string array.

Notes: Returns the result object of the given transaction as defined by the type of request the transaction is handling, if currently available and applicable. Otherwise returns an empty array.

StringResult as string

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** The result of the transaction as a string.

Notes: Returns the result object of the given transaction as defined by the type of request the transaction is handling, if currently available and applicable. Otherwise returns "".

transactionID as string

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Returns a globally unique identifier string for this transaction.

transactionState as integer

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** The state of this transaction.

Notes:

Returns one of the following integer constants:

Transaction State Constant	Meaning
kDMTransactionNotStarted	The transaction has not yet started.
kDMTransactionActive	The transaction is in progress.
kDMTransactionSuccessful	The transaction completed successfully and without error.
kDMTransactionAborted	DMTransactions abort method was called, halting the given transaction.
kDMTransactionHadError	A non-success http status code was returned, a network error occurred, or the expected data was not received. To get the exact error that occurred, call DMTransactions statusCode or errorType methods.

uri as string

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Returns the target uri sent in this transactions originating request.

2.6.2 Properties

context as variant

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** The context object set for this transaction.

Notes:

This value is nil by default.

Sets an object of the callers choosing on the given transaction that can be used to help identify a transaction. This method is especially useful in tracking transactions done asynchronously. After obtaining an asynchronously-fired DMTransaction object, the caller need not worry about the transaction completing and generating a delegate callback

8before the context can be set—since callbacks are made on the same client thread which launched the transaction, setting the context simply needs to be done immediately after generating the transaction and in the same thread. That way, even if the transaction completes immediately, the context will still be set before the completion callback is made.

(Read and Write property)

Handle as Integer

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** The internal object reference.

Notes: (Read and Write property)

2.6.3 Constants

kDMTransactionAborted=4

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of the status constants returned by the DMTransactionMBS.transactionState method.

Notes: DMTransactions abort method was called, halting the given transaction.

kDMTransactionActive=2

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of the status constants returned by the DMTransactionMBS.transactionState method.

Notes: The transaction is in progress.

kDMTransactionHadError=5

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of the status constants returned by the DMTransactionMBS.transactionState method.

Notes: A non-success http status code was returned, a network error occurred, or the expected data was not received. To get the exact error that occurred, call DMTransactions httpStatusCode or errorType methods.

kDMTransactionNotStarted=1

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of the status constants returned by the DMTransactionMBS.transactionState method.

Notes: The transaction has not yet started.

kDMTransactionSuccessful=3

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of the status constants returned by the DMTransactionMBS.transactionState method.

Notes: The transaction completed successfully and without error.

2.7 class DMTransactionDelegateMBS

class DMTransactionDelegateMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** The class for getting transaction events.

2.7.1 Methods

Constructor

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** The constructor to create a new object.

2.7.2 Properties

Handle as Integer

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** The internal object reference.

Notes: (Read and Write property)

2.7.3 Events

transactionAborted(Transaction as DMTransactionMBS)

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** This delegate method is called after the DMTransaction objects abort method is called and stops a currently-running asynchronous transaction.

transactionHadError(Transaction as DMTransactionMBS)

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** This delegate method is called when a transaction encounters an error condition.

Notes: If the transaction does complete successfully, then transactionSuccessful: is called instead. Takes one parameter, which is a reference to the DMTransaction object that experienced the error.

transactionSuccessful(Transaction as DMTransactionMBS)

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** This delegate method is called when a transaction successfully completes.

Notes: If the transaction does not complete successfully, then transactionHadError: is called instead. Takes one parameter, which is a reference to the DMTransaction object that completed.

2.8 class DMiDiskSessionMBS

class DMiDiskSessionMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** The class for an iDisk Session.

Notes: Please note: All server-side paths passed as arguments to DMiDiskFileManager category methods should start with a forward slash, omitting the iDisk account name at the beginning of the path for example, if uploading a local file to a users Public folder, the path passed to the createFileAtPath method would be of the format /Public/<filename>.

2.8.1 Methods

account as DMAccountMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** The account used for this session.

basicAttributesAtPath(path as string) as DMTransactionMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Performs a lightweight access against a single file or collection (directory) resource referenced by thePath.

Notes:

Use this method when only basic information such as the last modified date or content length is needed, as it is lighter weight than an extendedAttributesAtPath call. If a resource exists at thePath and the transaction is successful, the returned DMTransaction objects result method returns an NSDictionary. That NSDictionary contains resource attributes whose values, if present and applicable, can be retrieved using the following NSString key constants:

Key	Corresponding Value Type
kDMContentLength	Number (bytes)
kDMLastModified	Date

[Note: kDMContentLength is not returned for collections.] Returns a DMTransaction object that can be used to get the result of the transaction, status, payload, and many other pieces of information about the actual transaction.

Constructor(c as DMAccountMBS) as DMiDiskSessionMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** This method creates an iDisk session instance that can be used to perform I/O transactions with the .Mac iDisk servers.

Notes: It takes an DMSecondaryUserMBS object. The session will be used to access the iDisk owned by the .Mac member with the given credentials.

See also:

- 2.8.1 Constructor(c as DMAccountMBS, memberName as string) as DMiDiskSessionMBS 41
- 2.8.1 Constructor(c as DMSecondaryUserMBS) as DMiDiskSessionMBS 41
- 2.8.1 Constructor(c as DMSecondaryUserMBS, memberName as string) as DMiDiskSessionMBS 41

Constructor(c as DMAccountMBS, memberName as string) as DMiDiskSessionMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** This method creates an iDisk session instance that can be used to perform I/O transactions with the .Mac iDisk servers.

Notes: It takes an DMSecondaryUserMBS object and also takes the name of the .Mac member whose iDisk is to be accessed using the provided credentials.

See also:

- 2.8.1 Constructor(c as DMAccountMBS) as DMiDiskSessionMBS 40
- 2.8.1 Constructor(c as DMSecondaryUserMBS) as DMiDiskSessionMBS 41
- 2.8.1 Constructor(c as DMSecondaryUserMBS, memberName as string) as DMiDiskSessionMBS 41

Constructor(c as DMSecondaryUserMBS) as DMiDiskSessionMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** This method creates an iDisk session instance that can be used to perform I/O transactions with the .Mac iDisk servers.

Notes: It takes an DMSecondaryUserMBS object. The session will be used to access the iDisk owned by the .Mac member with the given credentials.

See also:

- 2.8.1 Constructor(c as DMAccountMBS) as DMiDiskSessionMBS 40
- 2.8.1 Constructor(c as DMAccountMBS, memberName as string) as DMiDiskSessionMBS 41
- 2.8.1 Constructor(c as DMSecondaryUserMBS, memberName as string) as DMiDiskSessionMBS 41

Constructor(c as DMSecondaryUserMBS, memberName as string) as DMiDiskSessionMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** This method creates an iDisk session instance that can be used to perform I/O transactions with the .Mac iDisk servers.

Notes: It takes an DMSecondaryUserMBS object and also takes the name of the .Mac member whose iDisk is to be accessed using the provided credentials.

See also:

- 2.8.1 Constructor(c as DMAccountMBS) as DMiDiskSessionMBS 40
- 2.8.1 Constructor(c as DMAccountMBS, memberName as string) as DMiDiskSessionMBS 41
- 2.8.1 Constructor(c as DMSecondaryUserMBS) as DMiDiskSessionMBS 41

contentsAtPath(path as string) as memoryblock

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Returns the contents of the remote file specified in path as a string.

Notes: If path specifies a directory, or if some other error occurs, this method returns nil. [Note: Only the data fork of the given file is returned.]

copyPath(source as string, destination as string) as boolean

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** This method will create a copy of the source resource at the given destination path, but will not overwrite an existing resource.

Notes:

Each path argument can be either a local or a remote path passing a local source path and remote destination path leads to an upload of a local resource, while passing a remote source path and local destination path leads to a download of a remote resource. If two remote paths are provided as arguments, this method issues a single, server-side COPY request. This method has usage similar to that of the corresponding method of NSFileManager. However, it does not support the usage of a callback handler and if a non-nil handler argument is passed to this method, it will be ignored. In the case of an upload or download, only the data forks of files and directories are copied. In the case of an upload, local symlinks are not preserved when a symlink is encountered, the resource it references will be uploaded. [Note: If two local paths are provided as arguments, this method calls NSFileManagers copyPath method with whatever handler argument was provided.]

For the plugin the handle parameter is always nil.

copyResourceAtPath(sourcePath as string, destinationPath as string) as DMTransactionMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Copies the resource identified by sourcePath if the account who owns the credentials used has write access to destinationPath.

Notes: This method will overwrite the resource at destinationPath if that resource is not locked by another client. Returns a DMTransaction object that can be used to get the result of the transaction, status, payload, and many other pieces of information about the actual transaction.

createDirectoryAtPath(path as string) as boolean

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** This method will create a new directory at the given remote path.

Notes: It has the same usage as the corresponding method of `NSFileManager`. However, it does not support the setting of attributes and if a non-nil attributes argument is passed to this method, it will be ignored.

createFilePath(path as string, contentData as memoryblock) as boolean

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** This method creates a remote file at path that contains contents, behavior which matches the style of the corresponding method of `NSFileManager`.

Notes: However, it does not support the setting of attributes and if a non-nil attributes argument is passed to this method, it will be ignored.

credentials as variant

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Returns the `DMCredential`-conforming object containing the credentials used to access this sessions `iDisk`.

Notes: Returns a `DMAccountMBS` or a `DMSecondaryUserMBS` object.

defaultLockDuration as double

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** The default lock duration set for this `DMiDiskSession` instance.

Notes:

Returns the default lock duration set for this `DMiDiskSession` instance. If no default lock duration has been set for this instance using the `setDefaultLockDuration:` method, an `NSTimeInterval` representing 5 minutes, given in seconds, is returned.

This method sets the default lock duration, in seconds, used by all lock-related `DMiDiskSession` methods that do not specify a transaction-specific lock duration of their own. If this method is not called, the default lock duration is 5 minutes, given in seconds.

(Read and Write computed property)

deleteResourceAtPath(path as string) as DMTransactionMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Deletes the resource referenced by thePath if it is not locked by another client and the account who owns the credentials used has write access to thePath.

Notes:

If the delete request fails because the specified resource or its parent is locked, the returned DMTransaction objects result method will return an Array of Strings representing the exact uris which are currently locked. [Note: Unlike on most file systems, a non-empty collection (directory) can be deleted. Issuing a delete targeted at a collection will delete the collection and all of its contents without warning (assuming you have the correct access privileges).] Returns a DMTransaction object that can be used to get the result of the transaction, status, payload, and many other pieces of information about the actual transaction.

directoryContentsAtPath(Path as string) as string()

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Searches the contents of the remote directory specified by path and returns an array of strings identifying the directories and files (including symbolic links) contained in path.

Notes: The search is shallow and therefore does not return the contents of any subdirectories. Returns nil if the directory specified at path does not exist or there is some other error accessing it. Returns an empty array if the directory exists but has no contents.

extendedAttributesAtPath(path as string) as DMTransactionMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** This method provides the attributes of the file or collection (directory) resource specified by thePath.

Notes:

If a resource exists at thePath and the request is successful, the returned DMTransaction objects result method returns an NSDictionary. That NSDictionary contains resource attributes whose values, if present and applicable, can be retrieved using the following NSString key constants:

[Note: kDMContentLength is not returned for collections.] Returns a DMTransaction object that can be used to get the result of the transaction, status, payload, and many other pieces of information about the actual transaction.

Key	Corresponding Value Type
kDMContentLength	Number (bytes)
kDMLastModified	Date
kDMURI	String
kDMDisplayName	String
kDMContentType	String
kDMIsCollection	Number (Boolean)
kDMIsLocked	Number (Boolean)
kDMLockOwner	String
kDMLockToken	String
kDMLockTimeout	Number (seconds)

fileExistsAtPath(path as string) as boolean

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Returns YES if the remote file specified in path exists, or NO if it does not.

See also:

- 2.8.1 fileExistsAtPath(path as string, byref isDirectory as boolean) as boolean 45

fileExistsAtPath(path as string, byref isDirectory as boolean) as boolean

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Returns whether the remote file specified in path exists.

Notes: If you want to determine if path is a directory, specify the address of a Boolean variable for isDirectory, which will be set to YES if path is a directory, and to NO otherwise.

See also:

- 2.8.1 fileExistsAtPath(path as string) as boolean 45

getDataAtPath(path as string) as DMTransactionMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Loads data from a given path.

Notes: If the transaction is successful, the returned DMTransaction objects result method returns a string containing the desired resource. [Note: If the transaction is not yet complete, the result method will return the data that has been downloaded so far.] The returned DMTransaction object can be used to get the result of the transaction, status, payload, and many other pieces of information about the actual transaction.

See also:

- 2.8.1 `getDataAtPath(sourcePath as string, IfModifiedSince as date)` as `DMTransactionMBS` 46
- 2.8.1 `getDataAtPath(sourcePath as string, IfModifiedSince as date, ByteStart as double, ByteLength as double)` as `DMTransactionMBS` 46

`getDataAtPath(sourcePath as string, IfModifiedSince as date)` as `DMTransactionMBS`

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Loads data from a given path.

Notes: If the transaction is successful, the returned `DMTransaction` objects result method returns a string containing the desired resource. [Note: If the transaction is asynchronous and is not yet complete, the result method will return the data that has been downloaded so far.] The returned `DMTransaction` object can be used to get the result of the transaction, status, payload, and many other pieces of information about the actual transaction.

See also:

- 2.8.1 `getDataAtPath(path as string)` as `DMTransactionMBS` 45
- 2.8.1 `getDataAtPath(sourcePath as string, IfModifiedSince as date, ByteStart as double, ByteLength as double)` as `DMTransactionMBS` 46

`getDataAtPath(sourcePath as string, IfModifiedSince as date, ByteStart as double, ByteLength as double)` as `DMTransactionMBS`

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Loads data from a given path.

Notes: If the transaction is successful, the returned `DMTransaction` objects result method returns a string containing the desired resource. The `NSRange` object, passed as the `byteRange` argument, should have its location set as the index of the first byte to be retrieved and its length should be set as the total number of bytes to be retrieved, including the first byte. You can also pass `nil` for `lastModifiedDate` if you are not interested in when the server resource was last modified. [Note: If the transaction is asynchronous and is not yet complete, the result method will return the data that has been downloaded so far.] The returned `DMTransaction` object can be used to get the result of the transaction, status, payload, and many other pieces of information about the actual transaction.

See also:

- 2.8.1 `getDataAtPath(path as string)` as `DMTransactionMBS` 45
- 2.8.1 `getDataAtPath(sourcePath as string, IfModifiedSince as date)` as `DMTransactionMBS` 46

iDiskSessionWithCredentials(c as DMAccountMBS) as DMiDiskSessionMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** This method creates an iDisk session instance that can be used to perform I/O transactions with the .Mac iDisk servers.

Notes: It takes an DMAccountMBS object. The returned session will be used to access the iDisk owned by the .Mac member with the given credentials.

See also:

- 2.8.1 iDiskSessionWithCredentials(c as DMAccountMBS, memberName as string) as DMiDiskSessionMBS 47
- 2.8.1 iDiskSessionWithCredentials(c as DMSecondaryUserMBS) as DMiDiskSessionMBS 47
- 2.8.1 iDiskSessionWithCredentials(c as DMSecondaryUserMBS, memberName as string) as DMiDiskSessionMBS 48

iDiskSessionWithCredentials(c as DMAccountMBS, memberName as string) as DMiDiskSessionMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** This method creates an iDisk session instance that can be used to perform I/O transactions with the .Mac iDisk servers.

Notes: It takes an DMAccountMBS object and also takes the name of the .Mac member whose iDisk is to be accessed using the provided credentials.

See also:

- 2.8.1 iDiskSessionWithCredentials(c as DMAccountMBS) as DMiDiskSessionMBS 47
- 2.8.1 iDiskSessionWithCredentials(c as DMSecondaryUserMBS) as DMiDiskSessionMBS 47
- 2.8.1 iDiskSessionWithCredentials(c as DMSecondaryUserMBS, memberName as string) as DMiDiskSessionMBS 48

iDiskSessionWithCredentials(c as DMSecondaryUserMBS) as DMiDiskSessionMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** This method creates an iDisk session instance that can be used to perform I/O transactions with the .Mac iDisk servers.

Notes: It takes a DMSecondaryUserMBS object. The returned session will be used to access the iDisk owned by the .Mac member with the given credentials.

See also:

- 2.8.1 iDiskSessionWithCredentials(c as DMAccountMBS) as DMiDiskSessionMBS 47

- 2.8.1 `iDiskSessionWithCredentials(c as DMAccountMBS, memberName as string)` as `DMiDiskSessionMBS` 47
- 2.8.1 `iDiskSessionWithCredentials(c as DMSecondaryUserMBS, memberName as string)` as `DMiDiskSessionMBS` 48

`iDiskSessionWithCredentials(c as DMSecondaryUserMBS, memberName as string)` as `DMiDiskSessionMBS`

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** This method creates an `iDisk` session instance that can be used to perform I/O transactions with the `.Mac` `iDisk` servers.

Notes: It takes an `DMSecondaryUserMBS` object and also takes the name of the `.Mac` member whose `iDisk` is to be accessed using the provided credentials.

See also:

- 2.8.1 `iDiskSessionWithCredentials(c as DMAccountMBS)` as `DMiDiskSessionMBS` 47
- 2.8.1 `iDiskSessionWithCredentials(c as DMAccountMBS, memberName as string)` as `DMiDiskSessionMBS` 47
- 2.8.1 `iDiskSessionWithCredentials(c as DMSecondaryUserMBS)` as `DMiDiskSessionMBS` 47

`isSynchronous` as `boolean`

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Whether transactions are performed synchronous (`true`) or asynchronous (`false`).

Notes:

The default mode is asynchronous.

This property is used to set the synchronicity state for generated transactions. Passing `true` will cause the object's transaction methods to block until they have either completed their transaction or failed with an error. Passing `false` will cause the methods to that return a `DMTransaction` object to return immediately, using any previously set delegate object to signal completion or error. It is safe to call this method to switch between asynchronous and synchronous modes as needed—since an in-flight `DMTransaction` object uses the settings that were in place at the time it was created, calling this method will not affect any transaction already in progress. [Note: The default mode is asynchronous.]

(Read and Write computed property)

kDMModify as string

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** One of the privileges constants.

Notes: Includes modify access to a given entity. Does not include create or delete access.

kDMRead as string

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** One of the privileges constants.

Notes: Includes read and list access to a given entity.

kDMReadChildren as string

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** One of the privileges constants.

Notes: Includes read and list access to children of a given entity only. Does not include read and list access to the entity itself.

kDMWrite as string

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** One of the privileges constants.

Notes: Includes create, modify and delete access to a given entity.

listCollectionAtPath(path as string) as DMTransactionMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Lists the collection at the given path.

lockResourceAtPath(path as string) as DMTransactionMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Locks the resource referenced by thePath so that only this session can execute write operations (e.g., PUT, DELETE) on that resource.

Notes:

The iDisk server reserves the right to grant a lock duration/timeout that differs from the requested lock duration when this method is successful, the returned DMTransaction objects result method will return an NSDictionary containing an NSNumber value (accessible using the kDMLockTimeout key) which is the lock timeout that was granted by the server. The lock will expire after the time specified by the granted lock timeout (in seconds) has passed. If the lock request fails because the specified resource or its parent is already locked, the returned DMTransaction objects result method will return an NSArray of NSStrings representing the exact uris which are currently locked. To maintain a lock, issue a relock request on the same resource before the original lock has expired. [Note: The lock created is a cooperative, write lock.]

Returns a DMTransaction object that can be used to get the result of the transaction, status, payload, and many other pieces of information about the actual transaction.

See also:

- 2.8.1 lockResourceAtPath(Path as string, lockDuration as double) as DMTransactionMBS 50

lockResourceAtPath(Path as string, lockDuration as double) as DMTransactionMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Locks the resource referenced by thePath so that only this session can execute write operations (e.g., PUT, DELETE) on that resource.

Notes:

The iDisk server reserves the right to grant a lock duration/timeout that differs from the requested lock duration when this method is successful, the returned DMTransaction objects result method will return an NSDictionary containing an NSNumber value (accessible using the kDMLockTimeout key) which is the lock timeout that was granted by the server. The lock will expire after the time specified by the granted lock timeout (in seconds) has passed. If the lock request fails because the specified resource or its parent is already locked, the returned DMTransaction objects result method will return an NSArray of NSStrings representing the exact uris which are currently locked. To maintain a lock, issue a relock request on the same resource before the original lock has expired. [Note: The lock created is a cooperative, write lock.]

Returns a DMTransaction object that can be used to get the result of the transaction, status, payload, and many other pieces of information about the actual transaction.

See also:

- 2.8.1 lockResourceAtPath(path as string) as DMTransactionMBS 50

makeCollectionAtPath(path as string) as DMTransactionMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** This method creates a collection (directory) at the location specified in thePath.

Notes: If any of the levels above the one being created does not exist, then the create will fail. Returns a DMTransaction object that can be used to get the result of the transaction, status, payload, and many other pieces of information about the actual transaction.

movePath(source as string, destination as string) as boolean

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** This method will move the source resource to the given destination path, but will not overwrite an existing resource.

Notes:

Each path argument can be either a local or a remote path passing a local source path and remote destination path leads to an upload of a local resource, while passing a remote source path and local destination path leads to a download of a remote resource. If two remote paths are provided as arguments, this method issues a single, server-side MOVE request. This method has usage similar to that of the corresponding method of NSFileManager. However, it does not support the usage of a callback handler and if a non-nil handler argument is passed to this method, it will be ignored. In the case of an upload or download, only the data forks of files and directories are moved. In the case of an upload, local symlinks are not preserved when a symlink is encountered, the resource it references will be uploaded. [Note: If two local paths are provided as arguments, this method calls NSManagers movePath method with whatever handler argument was provided.]

For the plugin the handle parameter is always nil.

moveResourceAtPath(sourcePath as string, destinationPath as string) as DMTransactionMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Moves the resource identified by sourcePath if it is not locked by another client and the account who owns the credentials used has write access to destinationPath.

Notes:

This method will overwrite the resource at destinationPath if that resource is not locked by another client. Returns a DMTransaction object that can be used to get the result of the transaction, status, payload, and many other pieces of information about the actual transaction.

owner as string

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Returns the name of the .Mac member who owns the iDisk being accessed by this session.

putData(data as memoryblock, destinationPath as string) as DMTransactionMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** This method places the contents of the data argument into the resource at the location specified by the destination path.

Notes:

[Note: Call will fail if any sublevels of destinationPath (excluding the object itself) do not exist, and the call may return nil if an invalid data argument is passed.]

Returns a DMTransaction object that can be used to get the result of the transaction, status, payload, and many other pieces of information about the actual transaction.

See also:

- 2.8.1 putData(data as memoryblock, destinationPath as string, ByteStart as double, ByteLength as double) as DMTransactionMBS 52

putData(data as memoryblock, destinationPath as string, ByteStart as double, ByteLength as double) as DMTransactionMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** This method places the contents of the data argument into the resource at the location specified by the byteRange argument.

Notes:

The NSRange object, passed as the byteRange argument, should have its location set as the index of the first byte to be written to and its length should be set as the total number of bytes, including the first byte. If the end of the specified range is beyond the end of the file, then the length of the resource is extended to accommodate all of the bytes. [Note: Call will fail if any sublevels of destinationPath (excluding the object itself) do not exist, and the call may return nil if an invalid data argument is passed.]

Returns a DMTransaction object that can be used to get the result of the transaction, status, payload, and many other pieces of information about the actual transaction.

See also:

- 2.8.1 putData(data as memoryblock, destinationPath as string) as DMTransactionMBS 52

putLocalFileAtPath(localPath as string, destinationPath as string) as DMTransactionMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** This method places the contents of a local file into a new server-side file at the given destination path.

Notes:

If the localPath parameter refers to a symlink, the symlink will not be preserved rather, the resource it references will be uploaded. [Note: Call will fail if all sublevels of destinationPath (excluding the object itself) do not already exist, and the call may return nil if an invalid localPath argument is passed.]

Returns a DMTransaction object that can be used to get the result of the transaction, status, payload, and many other pieces of information about the actual transaction.

quotaAttributes as DMTransactionMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Use this method to get the users iDisk quota value along with the amount of storage space that is currently used.

Notes:

The returned DMTransaction objects result method returns an Dictionary containing two Number values, which can be retrieved using the following key constants:

Key Constant	Corresponding Value Type
kDMiDiskQuotaInBytes	Number (bytes)
kDMiDiskSpaceUsedInBytes	Number (bytes)

relockResourceAtPath(path as string) as DMTransactionMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Refreshes the lock on the resource referenced by thePath.

Notes:

The iDisk server reserves the right to grant a lock duration/timeout that differs from the requested lock duration when this method is successful, the returned DMTransaction objects result method will return an Dictionary containing an NSNumber value (accessible using the kDMLockTimeout key) which is the lock timeout that was granted by the server. Once refreshed, the lock will expire after the time specified by the granted lock timeout (in seconds) has passed. Attempting to relock a resource that is not locked will always fail. If the relock request fails because the specified resource or its parent was originally locked by another client, the returned DMTransaction objects result method will return an array of Strings representing the exact uris which are currently locked. [Note: The lock created is a cooperative, write lock.]

Returns a DMTransaction object that can be used to get the result of the transaction, status, payload, and

many other pieces of information about the actual transaction.
See also:

- 2.8.1 `relockResourceAtPath(Path as string, lockDuration as double) as DMTransactionMBS` 54

`relockResourceAtPath(Path as string, lockDuration as double) as DMTransactionMBS`

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Refreshes the lock on the resource referenced by thePath.

Notes:

The iDisk server reserves the right to grant a lock duration/timeout that differs from the requested lock duration when this method is successful, the returned DMTransaction objects result method will return an NSDictionary containing an NSNumber value (accessible using the `kDMLockTimeout` key) which is the lock timeout that was granted by the server. Once refreshed, the lock will expire after the time specified by the granted lock timeout (in seconds) has passed. Attempting to relock a resource that is not locked will always fail. If the relock request fails because the specified resource or its parent was originally locked by another client, the returned DMTransaction objects result method will return an NSArray of NSStrings representing the exact uris which are currently locked. [Note: The lock created is a cooperative, write lock.]

Returns a DMTransaction object that can be used to get the result of the transaction, status, payload, and many other pieces of information about the actual transaction.

See also:

- 2.8.1 `relockResourceAtPath(path as string) as DMTransactionMBS` 53

`removeFileAtPath(path as string) as boolean`

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** This method will remove the resource at the given remote path and has the same usage as the corresponding method of NSFileManager.

Notes: However, it does not support the usage of a callback handler and if a non-nil handler argument is passed to this method, it will be ignored.

`resourceExistsAtPath(path as string) as boolean`

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** A method which always operates synchronously, `resourceExistsAtPath` returns true if the resource specified by thePath exists on the iDisk server, and returns false otherwise.

Notes: [Note: To check for the existence of a resource asynchronously, use the `basicAttributesAtPath`:

method.]

transactionDelegate as DMTransactionDelegateMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** The transaction delegate in use for all the transactions made by this object.

Notes:

This method is used to set the delegate object which will receive status messages from asynchronous transactions generated by the object that conforms to this protocol. The object passed in this method's delegate parameter must conform to the informal DMTransactionDelegate protocol. Transaction delegates receive their messages on the run loop originally used to create the given DMTransaction. It is safe to call this method to change the delegate as needed—since an in-flight DMTransaction object uses the settings that were in place at the time it was created, calling this method will not affect any transaction already in progress. A transaction delegate object does not receive callbacks for transactions executed synchronously. (Read and Write computed property)

unlockResourceAtPath(path as string) as DMTransactionMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Releases any lock on the resource referenced by thePath.

Notes: Attempting to unlock a resource that is not locked will fail. Returns a DMTransaction object that can be used to get the result of the transaction, status, payload, and many other pieces of information about the actual transaction.

2.8.2 Properties

Handle as Integer

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** The internal object reference.

Notes: (Read and Write property)

2.8.3 Constants

kDMContentLength = "DAV:getcontentlength"

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of the constants used as key in the dictionary you get from DMiDiskSessionMBS.extendedAttributesAtPath or DMiDiskSessionMBS.basicAttributesAtPath.

Notes: Value is a number.

kDMContentType = "DAV:getcontenttype"

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of the constants used as key in the dictionary you get from DMiDiskSession.extendedAttributesAtPath.

Notes: Value is a string.

kDMDateCreated = "DAV:creationdate"

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of the DotMacKit constants.

kDMDebugLogging = "DMDebugLogging"

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of the DotMacKit constants.

kDMDisplayName = "DAV:displayname"

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of the DotMacKit constants.

kDMErrorLogging = "DMErrorLogging"

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of the DotMacKit constants.

kDMGroup = "Group"

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of the principal type constants.

Notes: Type for principal representing a .Mac member-owned group.

kDMiDiskQuotaInBytes = "DAV:quota"

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of the dictionary keys in the dictionary returned from DMiDiskSession.quotaAttributes

Notes: A number with the bytes.

kDMiDiskSpaceUsedInBytes = "DAV:quotaused"

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of the dictionary keys in the dictionary returned from DMiDiskSession.quotaAttributes

Notes: A number with the bytes.

kDMIIsCollection = "IsCollection"

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of the constants used as key in the dictionary you get from DMiDiskSession.extendedAttributesAtPath.

Notes: Value is a boolean.

kDMIIsLocked = "IsLocked"

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of the constants used as key in the dictionary you get from `DMiDiskSession.extendedAttributesAtPath`.

Notes: Value is a boolean.

kDMLastModified = "DAV:getlastmodified"

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of the constants used as key in the dictionary you get from `DMiDiskSessionMBS.extendedAttributesAtPath` or `DMiDiskSessionMBS.basicAttributesAtPath`.

Notes: Value is a date.

kDMLockOwner = "LockOwner"

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of the constants used as key in the dictionary you get from `DMiDiskSession.extendedAttributesAtPath`.

Notes: Value is a string.

kDMLockTimeout = "LockTimeout"

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of the constants used as key in the dictionary you get from `DMiDiskSession.extendedAttributesAtPath`.

Notes: Value is a number (seconds).

kDMLockToken = "LockToken"

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of the constants used as key in the dictionary you get from `DMiDiskSession.extendedAttributesAtPath`.

Notes: Value is a string.

kDMLoggingLevel = "com.apple.DotMacKit.LoggingLevel"

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of the DotMacKit constants.

kDMMemberAccount = ".Mac Member"

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of the principal type constants.

Notes: Type for principal representing .Mac member.

kDMPrincipalName = "PrincipalName"

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of the constants for principalPropertiesFromPrincipalID.

Notes: The name of the given principal.

kDMPrincipalOwner = "PrincipalOwner"

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of the constants for principalPropertiesFromPrincipalID.

Notes: The name of the .Mac member who owns the principal—this is the same as the principal name if the principal represents a DMMemberAccount.

kDMPrincipalType = "PrincipalType"

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of the constants for principalPropertiesFromPrincipalID.

Notes: The type of the given principal, which is one of the Principal Type Constants listed below.

kDMSecondaryUser = "Secondary User"

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of the principal type constants.

Notes: Type for principal representing a .Mac member-owned secondary user.

kDMURI = "URI"

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of the constants used as key in the dictionary you get from `DMiDiskSession.extendedAttributesAtPath`.

Notes: Value is a string.

2.9 class DMAccountMBS

class DMAccountMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** The class for an account.

2.9.1 Methods

applicationID as string

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** The application ID currently set for this `DMMemberAccount` object is returned.

Notes: (Read and Write computed property)

applicationName as string

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** The human-readable name of the calling application is returned if set.

Notes:

This method sets the human-readable name of the calling application. Once set, it will be included in the user-agent header sent with all http requests.

(Read and Write computed property)

name as string

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** A username string is returned.

owner as string

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** The name of the .Mac member who owns the credential is returned.

password as string

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** A password string is returned.

principalID as string

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** This method is implemented by DMMemberAccount, DMSecondaryUser or DMGroup and returns the appropriate principal ID string.

principalIDWithName(name as string, memberName as string) as string

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Returns a principal ID string identifying the principal with the given name and owner (the owner is the name of the .Mac member that owns the given principal).

Notes: This class method is useful when the caller needs to obtain a principle ID representing a DMMemberAccount, DMSecondaryUser or DMGroup without creating an instance of one of those classes.

2.9.2 Properties

Handle as Integer

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** The internal object reference.

Notes: (Read and Write property)

2.10 class DMGroupMBS

class DMGroupMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** The class for a group.

2.10.1 Methods

addMembers(members() as string) as DMTransactionMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Adds members to the given .Mac member-owned group.

Notes: If the group already exists, then the identified principals are added as members. If the group does not already exist, this call fails. The members argument is an NSArray of principal ID strings that identify principals corresponding to DMMemberAccounts, DMSecondaryUsers and/or other DMGroup objects. Returns a DMTransaction object whose -isSuccessful method can be used to test for success.

isSynchronous as boolean

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Whether transactions are performed synchronous (true) or asynchronous (false).

Notes:

The default mode is asynchronous.

This property is used to set the synchronicity state for generated transactions. Passing true will cause the object's transaction methods to block until they have either completed their transaction or failed with an error. Passing false will cause the methods to that return a DMTransaction object to return immediately, using any previously set delegate object to signal completion or error. It is safe to call this method to switch between asynchronous and synchronous modes as needed—since an in-flight DMTransaction object uses the settings that were in place at the time it was created, calling this method will not affect any transaction already in progress. [Note: The default mode is asynchronous.]

(Read and Write computed property)

members as DMTransactionMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Lists all members of the given .Mac member-owned group.

Notes: Returns a DMTransaction object whose isSuccessful method can be used to test for success and whose -result method, upon success, returns an Array of principal ID strings that identify principals corresponding to DMMemberAccounts, DMSecondaryUsers and/or other DMGroup objects.

name as string

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Returns the name of the .Mac member-owned group represented by this DMGroup object.

owner as string

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Returns the name of the .Mac member who owns this group.

principalID as string

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** This method is implemented by DMMemberAccount, DMSecondaryUser or DMGroup and returns the appropriate principal ID string.

principalIDWithName(name as string, memberName as string) as string

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Returns a principal ID string identifying the principal with the given name and owner (the owner is the name of the .Mac member that owns the given principal).

Notes: This class method is useful when the caller needs to obtain a principle ID representing a DMMemberAccount, DMSecondaryUser or DMGroup without creating an instance of one of those classes.

removeMembers(members() as string) as DMTransactionMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Removes members from the given .Mac member-owned group.

Notes: The members argument is an NSArray of principal ID strings that identify principals corresponding to DMMemberAccounts, DMSecondaryUsers and/or other DMGroup objects. Returns a DMTransaction object whose -isSuccessful method can be used to test for success.

transactionDelegate as DMTransactionDelegateMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** The transaction delegate in use for all the transactions made by this object.

Notes:

This method is used to set the delegate object which will receive status messages from asynchronous transactions generated by the object that conforms to this protocol. The object passed in this method's delegate parameter must conform to the informal DMTransactionDelegate protocol. Transaction delegates receive their messages on the run loop originally used to create the given DMTransaction. It is safe to call this method to change the delegate as needed—since an in-flight DMTransaction object uses the settings that were in place at the time it was created, calling this method will not affect any transaction already in progress. A transaction delegate object does not receive callbacks for transactions executed synchronously. (Read and Write computed property)

2.10.2 Properties

Handle as Integer

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** The internal object reference.

Notes: (Read and Write property)

2.11 class DMMessageMBS

class DMMessageMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** The class for a message.

Notes: Once published to a DMTopic, a DMMessage object becomes immutable. Use DMMessage.isPublished method to determine whether a message has been published to .Mac.

2.11.1 Methods

Constructor(messageString as string)

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Creates a newly initialized DMMessage instance containing the provided message string.

Notes: This is a convenience method for creating a DMMessage containing just the provided message string and no payload object or properties dictionary. The message string cannot exceed 1K in size.

expirationDate as date

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** If this DMMessage has already been published, this method returns an NSDate that marks the time this DMMessage expires (with precision in seconds).

Notes: Otherwise, this method returns nil. If the expiration time returned for this DMMessage is in the past, it means that this message is awaiting clean-up.

isPublished as boolean

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Tells whether the DMMessage has already been published to .Mac.

Notes: A DMMessage can be published to .Mac only once and it becomes immutable once published. An

attempt to publish a DMMessage more than once will fail. Returns a BOOL value.

isSynchronous as boolean

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Whether transactions are performed synchronous (true) or asynchronous (false).

Notes:

The default mode is asynchronous.

This property is used to set the synchronicity state for generated transactions. Passing true will cause the object's transaction methods to block until they have either completed their transaction or failed with an error. Passing false will cause the methods to that return a DMTransaction object to return immediately, using any previously set delegate object to signal completion or error. It is safe to call this method to switch between asynchronous and synchronous modes as needed—since an in-flight DMTransaction object uses the settings that were in place at the time it was created, calling this method will not affect any transaction already in progress. [Note: The default mode is asynchronous.]

(Read and Write computed property)

message(messageString as string) as DMMessageMBS

Plugin Version: 11.2 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Creates a new message with the given text.

messageID as string

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Returns this DMMessage's globally unique identifier string.

messageString as string

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Returns the optional message string set for the DMMessage.

payload as DMTransactionMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Retrieves the optional payload object set for the DMMessage.

Notes: Upon success, the returned DMTransaction objects result method will return this messages payload object, an object that conforms to the NSCoder protocol. If auto-arrival of payloads is enabled for this messages parent topic, then the payload object is already available locally by the time this new message is received and the caller can call the returned DMTransactions -result method right away. However, if auto-arrival is disabled, this DMMessages payload is not downloaded until this -payload method is called. If this DMMessages transaction processing is set to asynchronous, then the status of the download can be observed by setting this DMMessages transaction delegate or through polling the returned DMTransaction object. [Note: If this DMMessages transaction delegate is not set, then it will inherit any transaction delegate that is explicitly set for its parent DMTopic or that has been inherited by its parent DMTopic from the session object used to create it.] If auto-arrival is disabled and this DMMessages transaction processing is set to synchronous, then this method will not return until the payload is completely downloaded. In all cases, the payload is retrieved with a call to the returned DMTransaction objects -result method.

payloadType as string

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Returns the type corresponding to the payload for this DMMessage.

Notes: The payload type is an NSString that represents the file type that would be used if the payload object were serialized to disk. Returns a type string, or nil if no payload object is set for this DMMessage.

properties as dictionary

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Returns the optional user-defined properties dictionary set for this DMMessage.

publicationDate as date

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Returns the time when this DMMessage was published to .Mac, which is an NSDate object with precision in seconds.

Notes: If this DMMessage has not yet been published, this method returns nil.

publisher as string

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Returns the name of the .Mac member who published this DMMessage to .Mac.

Notes: Returns "" if this DMMessage has not yet been published.

setMessageString(messageString as string) as integer

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Sets the optional message string for the DMMessage.

Notes: The message string cannot exceed 1K in size. An integer status constant is returned. [Note: Calling this method on an already-published message will result in kDMResourceExists being returned; otherwise, this method returns kDMSuccess. The -isPublished method can be used to determine whether this DMMessage has already been published.]

setProperty(dic as dictionary) as integer

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Sets the optional properties dictionary for this DMMessage.

Notes: This dictionary cannot exceed 100K in size. The dictionary can contain NSString keys and values only and the 100K limit includes both keys and values. [Note: Calling this method on an already-published message will result in kDMResourceExists being returned; otherwise, this method returns kDMSuccess. The isPublished method can be used to determine whether this DMMessage has already been published.]

setTimeToLive(timeToLive as double) as integer

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** This method can be used to set the time-to-live for this DMMessage prior to publication.

Notes:

The default time-to-live is 7 days, and the time-to-live cannot be made longer than 30 days. Takes an NSTimeInterval (precision in seconds). [Note: Calling this method on an already-published message will result in kDMResourceExists being returned; otherwise, this method returns kDMSuccess. The -isPublished method can be used to determine whether this DMMessage has already been published.]

(Read and Write computed property)

timeToLive as double

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** The time to live for this message.

Notes:

Returns an `NSTimeInterval` (with precision in seconds) representing the default time-to- live of 7 days if the default was not overridden by a `DMMessagePublisher` setting or by a call to this `DMMessages setTimeToLive` method.

(Read and Write computed property)

topic as DMTopicMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Returns the topic that contains this `DMMessage`.

Notes: Returns nil if this `DMMessage` has not yet been published to a topic.

transactionDelegate as DMTransactionDelegateMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** The transaction delegate in use for all the transactions made by this object.

Notes:

This method is used to set the delegate object which will receive status messages from asynchronous transactions generated by the object that conforms to this protocol. The object passed in this method's delegate parameter must conform to the informal `DMTransactionDelegate` protocol. Transaction delegates receive their messages on the run loop originally used to create the given `DMTransaction`. It is safe to call this method to change the delegate as needed—since an in-flight `DMTransaction` object uses the settings that were in place at the time it was created, calling this method will not affect any transaction already in progress. A transaction delegate object does not receive callbacks for transactions executed synchronously. (Read and Write computed property)

2.11.2 Properties

Handle as Integer

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** The internal object reference.

Notes: (Read and Write property)

2.11.3 Constants

kDMIInsufficientStorage=102

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of the DotMacKit status constants.

Notes: The .Mac account does not have sufficient disk space to accommodate the given operation.

kDMInvalidCredentials=101

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of the DotMacKit status constants.

Notes: The .Mac credentials set for this object do not have access to a .Mac account.

kDMNetworkError=103

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of the DotMacKit status constants.

Notes: Unable to interact with .Mac because of a networking error or lack of network availability.

kDMResourceExists=105

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of the DotMacKit status constants.

Notes: Indicates an error due to the pre-existence of a given resource.

kDMResourceNotFound=104

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of the DotMacKit status constants.

Notes: A required resource was not found.

kDMServiceBusy=106

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of the DotMacKit status constants.

Notes: The given .Mac service was too busy to perform the requested operation and the operation should be retried later.

kDMSuccess=1

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of the DotMacKit status constants.

Notes: The operation was successful.

2.12 class DMMemberAccountMBS

class DMMemberAccountMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** The class for a member account.

Notes: Subclass of the DMAccountMBS class.

2.12.1 Methods

accountFromPreferencesWithApplicationID(creatorCode as string) as DMMemberAccountMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Returns an initialized instance of DMMemberAccount keyed to a specific applications identification string.

Notes:

The instance returned will have the .Mac name and password from the system pre-applied. Pass the unique 4-character application identifier obtained from <http://developer.apple.com/datatype/creatorcode.html>.

accountUpgradeURL as string

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** This method provides access to the same localized URL that the upgradeAccount method loads in the default browser.

accountWithName(name as string, password as string, creatorCode as string) as DMMemberAccountMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Returns an initialized instance of DMMemberAccount keyed to a specific applications identification string.

Notes:

The instance returned will use the provided name and password, rather than any credentials stored in the .Mac preferences on the system. (To have the credentials stored on the system pre-applied, use the accountFromPreferencesWithApplicationID: method instead.) Pass the unique 4- character application identifier obtained from <http://developer.apple.com/datatype/creatorcode.html>.

canUpgradeAccount as DMTransactionMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Whether this account can be upgraded.

Notes: The returned DMTransaction objects result method returns a Boolean value. This value will be YES if the DMMemberAccount objects credentials are associated with an existing account that can be upgraded, such as a trial or email-only account, or if the account has expired and needs to be renewed. Otherwise, the

resulting value will be NO. [Note: If the account credentials do not exist or some other error occurs, the returned DMTransactions errorType method can be used to identify the error.]

Constructor(creatorCode as string)

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Creates an instance of DMMemberAccount keyed to a specific applications identification string.

Notes:

The instance returned will have the .Mac name and password from the system pre-applied. Pass the unique application identifying string obtained from

<http://developer.apple.com/datatype/creatorcode.html>

See also:

- 2.12.1 Constructor(name as string, password as string, creatorCode as string) 73

Constructor(name as string, password as string, creatorCode as string)

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Creates an initialized instance of DMMemberAccount keyed to a specific applications identification string.

Notes:

The instance returned will use the provided name and password, rather than any credentials stored in the .Mac preferences on the system. (To have the credentials stored on the system pre-applied, use the `initWithPreferencesWithApplicationID:` method instead.) Pass the unique application identifying string obtained from

<http://developer.apple.com/datatype/creatorcode.html>

See also:

- 2.12.1 Constructor(creatorCode as string) 73

createGroupWithName(name as string, members() as string) as DMTransactionMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** This method can be used to create a new DMGroup object with the provided name and members if a group of the same name does not already exist for the owners account on .Mac.

Notes: The members argument is an Array of principal ID strings identifying principals corresponding

to `DMMemberAccounts`, `DMSecondaryUsers` and/or `DMGroup` objects. Returns a `DMTransaction` object whose `isSuccessful` method can be used to test for success.

`createSecondaryUserWithName(name as string, password as string) as DMTransactionMBS`

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Adds a secondary user to .Mac if a secondary user of the same name does not already exist for this owning member account.

Notes: Returns a `DMTransaction` object whose `isSuccessful` method can be used to test for success.

`daysLeftUntilExpiration as DMTransactionMBS`

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** The returned `DMTransaction` objects result method returns a number representing the number of full days left until the given account expires.

Notes: [Note: If the account credentials do not exist or some other error occurs, the returned `DMTransactions` `errorType` method can be used to identify the error.]

`groupNamed(name as string) as DMGroupMBS`

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Use this method to obtain a `DMGroup` object representing a known group that already exists.

`groupNames as DMTransactionMBS`

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Lists all groups owned by this .Mac member.

Notes: Returns a `DMTransaction` object whose `isSuccessful` method can be used to test for success and whose result method returns an `Array of String` group names upon success.

`isSynchronous as boolean`

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Whether transactions are performed synchronous (true) or asynchronous (false).

Notes:

The default mode is asynchronous.

This property is used to set the synchronicity state for generated transactions. Passing true will cause the object's transaction methods to block until they have either completed their transaction or failed with an error. Passing false will cause the methods to that return a DMTransaction object to return immediately, using any previously set delegate object to signal completion or error. It is safe to call this method to switch between asynchronous and synchronous modes as needed—since an in-flight DMTransaction object uses the settings that were in place at the time it was created, calling this method will not affect any transaction already in progress. [Note: The default mode is asynchronous.]
(Read and Write computed property)

matchesSystemCredentials as boolean

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Returns true if the name and password currently set for this DMMemberAccount object match the .Mac credentials stored on the system.

removeGroupNamed(name as string) as DMTransactionMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Removes the DM-Group with the given name from .Mac.

Notes: Returns a DMTransaction object whose isSuccessful method can be used to test for success.

removeSecondaryUserNamed(name as string) as DMTransactionMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Removes the secondary user with the given name from .Mac.

Notes: Returns a DMTransaction object whose isSuccessful method can be used to test for success.

resetPassword(password as string, SecondaryUserNamed as string) as DMTransactionMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Resets the password of the secondary user with the given name.

Notes: Returns a DMTransaction object whose isSuccessful method can be used to test for success.

secondaryUserNames as DMTransactionMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Lists all secondary users owned by this .Mac member.

Notes: Returns a DMTransaction object whose isSuccessful method can be used to test for success and whose result method returns an Array of Strings on success.

servicesAvailableForAccount as DMTransactionMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** The returned DMTransaction objects result method returns an array of string entries corresponding to available services for the .Mac account represented by this DMMemberAccount instance.

Notes:

If a given .Mac service is enabled for this account, then the array will contain a string equal to one of these String constants:

Service Constant	Meaning
kDMiDiskService	The account has iDisk storage associated with it.
kDMEmailService	The account has an email address along with the associated email storage.
16kDMWebHostingService	The account has web hosting capabilities through its iDisks Sites folder.
kDMiSyncService	The account has access to .Macs iSync- related functionality.

[Note: If the account credentials do not exist or some other error occurs, the returned DMTransactions `errorType` method can be used to identify the error.]

signUpNewMember

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** This method will launch the users default web browser and open the .Mac new member sign-up page, which is localized based on system preferences.

Notes: This method can be used when the user of the given application is not yet a .Mac member and cannot fully enjoy the applications features without a .Mac account.

signUpNewMemberWithApplicationID(appID as string)

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** This method will launch the users default web browser and open the .Mac new member sign-up page, which is localized based on system preferences.

Notes:

This method performs the same function as the signUpNewMember method, and additionally ensures that the owner of the provided application ID (if a member of the .Mac Affiliate Program) earns a commission whenever a user of their application becomes a paying .Mac member. For more information on the .Mac Affiliate Program, please visit <http://www.mac.com/1/affiliates/faq.html>.

signUpURL as string

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** This method provides access to the same localized URL that the signUpNewMember method loads in the default browser.

Notes: [Note: The URL returned by this method should not be hard-coded within an application, as it may change in the future.]

signUpURLWithApplicationID(appID as string) as string

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Returns the URL for the .Mac sign-up page, with commission tracking enabled if the owner of the provided application ID is a member of the .Mac Affiliate Program.

Notes:

For more information on the .Mac Affiliate Program, please visit this website:
<http://www.mac.com/1/affiliates/faq.html>

[Note: The URL returned by this method should not be hard-coded within an application, as it may change in the future.]

transactionDelegate as DMTransactionDelegateMBS

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** The transaction delegate in use for all the transactions made by this object.

Notes:

This method is used to set the delegate object which will receive status messages from asynchronous transactions generated by the object that conforms to this protocol. The object passed in this method's delegate parameter must conform to the informal DMTransactionDelegate protocol. Transaction delegates receive their messages on the run loop originally used to create the given DMTransaction. It is safe to call this method to change the delegate as needed—since an in-flight DMTransaction object uses the settings that were in place at the time it was created, calling this method will not affect any transaction already in progress. A transaction delegate object does not receive callbacks for transactions executed synchronously. (Read and Write computed property)

upgradeAccount

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** This method will launch the users default web browser and open the appropriate .Mac membership upgrade page, which is localized based on system preferences.

validateCredentials as integer

Plugin Version: 9.7 Console & Web: Yes Mac: Yes, Win: No, Linux: No, . **Function:** Use this method to check that the given account contains valid .Mac credentials.

Notes: The returned integer will correspond to one of the Status Constants defined in the Constants section of this document.

2.12.2 Constants

kDMBackupService = "Backup"

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of constants returned in the DMMemberAccountMBS.servicesAvailableForAccount method.

kDMEmailService = "Email"

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of constants returned in the DMMemberAccountMBS.servicesAvailableForAccount method.

Notes: The account has an email address along with the associated email storage.

kDMiDiskService = "iDisk"

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of constants returned in the DMMemberAccountMBS.servicesAvailableForAccount method.

Notes: The account has iDisk storage associated with it.

kDMInsufficientStorage=102

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of the DotMacKit status constants.

Notes: The .Mac account does not have sufficient disk space to accommodate the given operation.

kDMInvalidCredentials=101

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of the DotMacKit status constants.

Notes: The .Mac credentials set for this object do not have access to a .Mac account.

kDMInvalidParameter=107

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of the DotMacKit status constants.

Notes: A user-provided parameter was invalid.

kDMiSyncService = "iSync"

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of constants returned in the `DMMemberAccountMBS.servicesAvailableForAccount` method.

Notes: The account has access to .Mac's iSync- related functionality.

kDMNetworkError=103

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of the DotMacKit status constants.

Notes: Unable to interact with .Mac because of a networking error or lack of network availability.

kDMResourceExists=105

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of the DotMacKit status constants.

Notes: Indicates an error due to the pre-existence of a given resource.

kDMResourceNotFound=104

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of the DotMacKit status constants.

Notes: A required resource was not found.

kDMServiceBusy=106

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of the DotMacKit status constants.

Notes: The given .Mac service was too busy to perform the requested operation and the operation should be retried later.

kDMSuccess=1

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of the DotMacKit status constants.

Notes: The operation was successful.

kDMUndefined=-1

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of the DotMacKit status constants.

Notes: Used by individual methods as noted in this documents method descriptions.

kDMUnknownError=100

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of the DotMacKit status constants.

Notes: Unable to complete operation because of an unknown error.

kDMWebHostingService = "WebHosting"

Plugin Version: 9.7 Console & Web: No Mac: Yes, Win: No, Linux: No, . **Function:** One of constants returned in the DMMemberAccountMBS.servicesAvailableForAccount method.

Notes: The account has web hosting capabilities through its iDisks Sites folder.

Chapter 3

List of all classes

• DMAccountMBS	60
• DMGroupMBS	62
• DMiDiskSessionMBS	39
• DMMemberAccountMBS	71
• DMMessageMBS	65
• DMMessagePublisherMBS	18
• DMMessageSubscriberMBS	11
• DMSecondaryUserMBS	15
• DMTopicMBS	26
• DMTransactionDelegateMBS	38
• DMTransactionMBS	31

Chapter 4

List of all global methods

- 2.4 LoadDotMacKitFrameworkMBS(path as folderitem) as boolean

25