

Clearspace Web Services Developer Guide

This document describes various technical aspects of the Clearspace Web Services implementation. Each service and object is described and several examples of the client APIs are available.

Contents

- [Overview](#)
- [Web Service Style](#)
- [Authentication](#)
- [Objects](#)
- [Services](#)
- [Fault Codes](#)
- [Client API Examples](#)
- [Client API Dependencies](#)

Overview

Clearspace Web Services implement the [SOAP protocol](#). WSDL description files are available as part of the distribution or by requesting the following URL from your Jive Clearspace instance:
<http://myhost.com/forums/rpc/soap/>.

Web Service Style

Clearspace Web Services use Document-Literal style binding with wrapped parameter format. This style was chosen for a number of reasons: it is WS-I compliant, the message is easily validateable, and it works well with .NET web services. There is also a general push towards Document-Literal style web services.

For a general description about web services styles see:

<http://www.ibm.com/developerworks/webservices/library/ws-whichwsdl/>

Authentication

Clearspace Web Services uses the [Username Token Profile V1.0](#) (pdf) specification. The Username Token Profile specification is part of the [OASIS Web Services Security \(WS-Security\)](#) specification.

A header containing the username and password must be passed into every request.

```
<wsdl:Envelope xmlns:soap="..." xmlns:wssse="..." >
  <wsdl:Header>
    <wssse:Security>
      <wssse:UsernameToken>
        <wssse:Username>admin</wssse:Username>
        <wssse:Password>password</wssse:Password>
      </wssse:UsernameToken>
    </wssse:Security>
  </wsdl:Header>
</wsdl:Envelope>
```

```

    </wsse:Security>
  </wsdl:Header>
</wsdl:Envelope>

```

- The namespace for WSDL prefix is <http://schemas.xmlsoap.org/wsdl/>
- The namespace for WSSE prefix is <http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd>

Objects

Object	Description
Community	A container for threads and a hierarchy of other communities.
Document	A Document object encapsulates a document in the Clearspace system.
ForumMessage	A ForumMessage encapsulates message data.
ForumThread	A ForumThread is a container for a hierarchy of ForumMessages.
Group	Organizes users into a group for easier permissions management.
JiveObject	Super-class of other jive webservice objects.
User	Provides information about and services for users of the system.
Watch	A watch is a way for a user to track updates to an object.

Community

Field	Type	Description
getName	String	Returns the name of the community.
getDescription	String	Returns the description of the community.
getCreationDate	Date	Returns the Date that the community was created.
getModificationDate	Date	Returns the Date this community was last modified.
getLocale	Locale	Returns the locale for this community.
getFinalLocale	Locale	Returns the locale for this community.
getCommunityCount	int	Returns the number of community in the community.
getRecursiveCommunityCount	int	Returns the number of

		communities in the community and all sub-communities.
getThreadCount	int	Returns number of thread in the community and all sub-communities.
getMessageCount	int	Returns the number of messages in the community.
getLatestMessageID	long	Returns the most recently created or edited message in this community (including all sub-communities).
getParentCommunityID	long	Returns ID of the parent community of this community.
getCommunityDepth	int	Returns the depth of a this community relative to the root.

Document

Field	Type	Description
getDocumentID	String	Returns the document ID corresponding to the document.
getCommunityID	long	
getDocumentTypeID	long	Return the type of the document.
getLanguage	String	Return the ISO-639 code of the language associated with the document, or null if the document has no language specified (thus implying that it uses the global default language).
getUserID	long	Returns the user that authored the document.
getAuthorCount	int	Returns the count of authors which have been added to this document.
getCreationDate	Date	A convenience method which returns the date the document was originally created.
getExpirationDate	Date	Returns the date that the document is set to expire on.
getSubject	String	Returns the title of the document.
getUnfilteredSubject	String	Returns the document title, bypassing any active filters.

getBody	String	Returns the textual body of the document.
getUnfilteredBody	String	Returns the textual body of the document, bypassing any active filters.
getAttachmentCount	int	
getImageCount	int	Returns the number of images the document has.
getCommentStatus	int	Returns the comment status of this document.
getModificationDate	Date	A convenience method which returns the date the document was last modified as determined by the version with a com.

ForumMessage

Field	Type	Description
getID	long	Returns the id of this message.
getCommunityIndex	int	Returns the index of this message in the community.
getCreationDate	Date	Returns the date this message was created.
getModificationDate	Date	Returns the date this message was last modified.
getSubject	String	Returns the message subject.
getUnfilteredSubject	String	Returns the message subject, bypassing any active filters.
getBody	String	Returns the message body.
getUnfilteredBody	String	Returns the message body, bypassing any active filters.
getUserID	long	Returns the ID User that authored this message.
getParentMessageID	long	Returns the ID of the parent message of this message or <tt>0</tt> if this message is the root message of a thread.
getAttachmentCount	int	Returns the number of attachments this message has.
getModerationValue	int	Returns the number of moderation points this message has.
getForumThreadID	long	Returns the ID of the thread this

		message belongs to.
getCommunityID	long	Returns the ID of the community this message belongs to.

ForumThread

Field	Type	Description
getName	String	Returns the subject of the root message of this thread.
getCreationDate	Date	Returns the date that this thread was created.
getModificationDate	Date	Returns the date that this thread was last modified.
getModerationValue	int	Returns the number of moderation points this thread has.
getCommunityID	long	Returns the ID of the community this thread belongs to.
getRootMessageID	long	Returns the ID of the root message this thread.
getLatestMessageID	long	Returns the ID of the most recently created messages in this thread.
getMessageCount	int	Returns the number of messages in the thread.

Group

Field	Type	Description
getName	String	Returns the name of the group.
getDescription	String	Returns the description of the group.
getCreationDate	Date	Returns the date that the group was created.
getModificationDate	Date	Returns the date that the group was last modified.
getMemberCount	int	Returns the number of users that belong to this group.

JiveObject

Field	Type	Description
getID	long	Retrieve the unique identifier for the jive object, or -1 if the object

		hasn't yet been added/inserted into the system.
getObjectType	int	Return the object type of the jive object.

User

Field	Type	Description
getID	long	Returns the user's id.
getUsername	String	Returns the user's username.
getName	String	Returns the user's name.
getPassword	String	
getPasswordHash	String	Returns the user's password in hashed form.
getEmail	String	Returns the user's email address.
getCreationDate	Date	Returns the date that the user was created.
getModificationDate	Date	Returns the date that the user was last modified.

Watch

Field	Type	Description
getUserID	long	Returns the id of the user who is associated with this watch.
getObjectID	long	Returns the object ID of the object being watched.
getObjectType	int	Returns the object type of the object being watched.
getWatchType	int	Returns the type of the watch.

Services

Service	Description
AddressBookService	Provides ability to interace with the Private Message addressbook.
AttachmentManagerService	A web service for managing attachment settings.
CommunityService	Provides the ability to manipulate communities.
DocumentService	This service provides methods to load and manipulate documents.
ForumService	Provides the ability to manipulate forum

	messages.
GroupService	Provides a the ability for managing groups and group membership.
PermissionService	Provides a webservice for managing permissions on users and groups.
PrivateMessageService	Provides the ability to manipulate private messages.
SearchService	Provides the ability to search for content.
SystemPropertiesService	Provides a web service for managing Jive System Properties.
UserService	Provides a webservice for managing user's, avatar's, and status levels.
WatchService	A service for manipulating a user's watches on objects.

Operations

AddressBookService

Provides ability to interace with the Private Message addressbook.

Field	Type	Description
getRoster	User	Retrieves a list of users contained within the specified users addressbook.
addUser	void	Adds a specified username to users addressbook.
addUsers	void	Adds a list of users to the Private Message addressbook of the specified user.
removeUser	void	Removes the specified username from a users Private Message addressbook.
removeUsers	void	Removes the specified list of users from a users Private Message addressbook.

AttachmentManagerService

A web service for managing attachment settings.

Field	Type	Description
isAttachmentsEnabled	boolean	Returns true if attachments are enabled, false otherwise.

setAttachmentsEnabled	void	Sets whether attachments are enabled, false otherwise.
getMaxAttachmentSize	int	Returns the maximum size of an individual attachment in kilobytes.
setMaxAttachmentSize	void	Sets the maximum size of an individual attachment in kilobytes.
getMaxAttachmentsPerDocument	int	Returns the maximum number of attachments per document.
setMaxAttachmentsPerDocument	void	Sets the maximum number of attachments per document.
isValidType	boolean	Returns true if the content type is valid based on the current settings of the <code>allowAllByDefault</code> flag and the allowed and disallowed types list.
addAllowedType	void	Adds a content type to the list of explicitly allowed types.
removeAllowedType	void	Removes a content type from the list of explicitly allowed types.
getAllowedTypes	String	Returns a List of explicitly allowed types.
addDisallowedType	void	Adds a content type to the list of explicitly disallowed types.
removeDisallowedType	void	Removes a content type from the list of explicitly disallowed types.
getDisallowedTypes	String	Returns a List of explicitly disallowed types.
isAllowAllByDefault	boolean	Returns true if in the "allow all content types by default" mode.
setAllowAllByDefault	void	Sets the default allowed content types mode.
isImagePreviewEnabled	boolean	Returns true if image preview support is enabled.
setImagePreviewEnabled	void	Toggles whether image preview support is enabled.
getImagePreviewMaxSize	int	Returns the max dimension of generated thumbnails (ie, the max value for the width or height).
setImagePreviewMaxSize	void	Sets the max dimension of generated thumbnails (ie, the

		max value for the width or height).
isImagePreviewRatioEnabled	boolean	Returns true if the aspect ratio of thumbnails should be preserved.
setImagePreviewRatioEnabled	void	Toggles whether the aspect ratio of thumbnails should be preserved.
getMaxAttachmentsPerMessage	int	Returns the maximum number of attachments per message.
setMaxAttachmentsPerMessage	void	Sets the maximum number of attachments per message.

CommunityService

Provides the ability to manipulate communities.

Field	Type	Description
getCommunity	Community	
createCommunity	Community	Creates a new Community as a sub-community of the root community using the name and description.
createSubCommunity	Community	Creates a new Community as a sub-community off of the specified community.
deleteCommunity	void	
updateCommunity	void	
getProperties	Property	Returns all the extended properties for the community with the specified id.
getProperty	String	Returns a specific extended property for the community with the specified property name and community ID.
setProperty	void	Set an extended property for the community with the given community id.
deleteProperty	void	Delete a property with the given name from the community with the given id.
getRecursiveCommunities	Community	Returns all of the communities in the system recursively.
getRecursiveCommunitiesByCommunityID	Community	Returns all of the communities under the specified community recursively.

getRecursiveCommunityCount	int	Returns a count of all the communities in the system.
getRecursiveCommunityCountByCommunityID	int	

DocumentService

This service provides methods to load and manipulate documents.

Field	Type	Description
createDocument	Document	Create a new document.
createBinaryDocument	Document	Creates a new binary document.
getBinaryDocumentContent	DataSource	Returns the content of the binary document.
getPopularDocuments	List	Returns the top x documents across all communities according to ratings, document views & time passed since the document was created.
getPopularDocuments	List	Returns the top x documents across all communities according to ratings, document views & time passed since the document was created in the languages specified.
getUserApprovalDocuments	List	Returns a list of documents that a user needs to approve.
moveDocument	void	Moves a document from it's current community to another.
deleteDocument	void	Deletes a document.
isCommentsEnabled	boolean	Returns true if the comments feature is turned on.
setCommentsEnabled	void	Enables or disables the comments feature system wide.
isTrackbacksEnabled	boolean	Returns true if the trackbacks feature is turned on.
setTrackbacksEnabled	void	Enables or disables the trackbacks feature system wide.
getDocumentByInternalDocID	Document	Returns the document with the specified docID.
getDocumentByInternalDocIDAndVersion	Document	Returns the revision of document with the specified docID and version number.
getDocumentByDocumentID	Document	Returns a document from the community based on its

		document ID.
getDocumentByDocumentIDAndVersion	Document	Returns the revision of the document with the specified docID and version number.
getApprovalStatusForDocument	Collection	Returns a collection of ApprovalStatus objects.
addDocumentApproverOnDocument	void	Adds a user as a document approver for this document.
addDocumentApproverOnCommunity	void	Add the user as a document approver for this entire community.
getDocumentApproversOnDocument	Collection	Returns all of the users that must approve new documents before they reach a PUBLISHED state when it is in a PENDING_APPROVAL state.
getDocumentApproversOnCommunity	Collection	Returns all of the users that must approve new documents before they reach a PUBLISHED state when it is in a PENDING_APPROVAL state.
getAuthors	Collection	Returns the users who are allowed to edit the document.
addAuthor	void	Adds the user with the supplied id as an author of this document of the specified document.
removeAuthor	void	Removes the user with the supplied id as an author of this document.

ForumService

Provides the ability to manipulate forum messages.

Field	Type	Description
getForumMessage	ForumMessage	Returns a ForumMessage by its ID.
deleteMessage	void	Delete the message with the following id.
deleteMessageAndChildren	void	Delete the message with the specified id.
getMessageProperties	Property	Returns all the extended properties for the message with the specified ID.

getMessageProperty	String	Returns a specific extended property for the message with the specified property name and message ID.
setMessageProperty	void	Set an extended for the property with the given message id.
deleteMessageProperty	void	Delete a property with the given name from the message with the given ID.
getUnfilteredMessageProperties	Property	Returns the properties without applying filters to them first.
getAttachmentsByMessageID	DataSource	Returns an array of attachments that are attached to the specified message.
addAttachmentToMessage	void	Adds an attachment to the message with the specified ID.
getMessageIDsByCommunityIDAndFilter	Filter	Returns all of the message IDs for all of the message that are in the community with the specified ID, filtered by the the specified result filter.
getMessageIDsByCommunityID	long	Returns all of the message IDs for all of the message that are in the community.
getMessageCountsByCommunityIDAndFilter	AndFilter	Returns the number of messages that are in the forum with the specified id after the specified filter has been applied.
getMessageCountsByCommunityID	int	Returns the number of messages that are in the community.
getMessageCountByCommunityIDAndFilter	AndFilter	Returns the number of messages that are in the community with the specified id after the specified filter has been applied.
getMessageCountByCommunityID	int	Returns the number of messages that are in the community.
getMessageIDsByThreadIDAndFilter	long	Returns all of the message IDs for all of the message that are in the thread after the specified filter has been applied.
getMessageIDsByThreadID	long	Returns the IDs of the messages that are in the thread.
getMessageCountByThreadIDAndFilter	Filter	Returns the number of messages that are in the thread with the

		specified id after the specified filter has been applied.
getMessageCountByThreadID	int	Returns the number of messages that are in the thread.
getMessagesByCommunityIDAndFilter	ForumMessage	Returns all of the messages that are in the community with the specified ID, filtered by the the specified result filter.
getMessagesByCommunityID	ForumMessage	Returns all of the messages that are in the community.
getMessagesByThreadIDAndFilter	ForumMessage	Returns all of the messages that are in the thread after the specified filter has been applied.
getMessagesByThreadID	ForumMessage	Returns the messages that are in the thread.
getRootMessage	ForumMessage	Returns the root of a thread.
hasParent	boolean	Returns true if the <code>child</code> message has a parent message.
getParent	ForumMessage	Returns the parent of the <code>child</code> ForumMessage.
getChild	ForumMessage	Returns the child of parent at index index in the parent's child array.
getChildren	ForumMessage	Returns an array of all the child messages of the parent.
getRecursiveChildren	ForumMessage	Returns an array for all child messages (and sub-children, etc) of the parent.
isLeaf	boolean	Returns true if node is a leaf.
getRecursiveChildCount	int	Returns the total number of recursive children of a parent.
getIndexOfChild	int	Returns the index of child in parent.
getMessageDepth	int	Returns the depth of a message in the message tree hierarchy.
getChildCount	int	Returns the number of children of parent.
getRecursiveMessages	ForumMessage	Returns an array for all messages in the thread in depth-first order.

updateForumMessage	void	
getForumThread	ForumThread	Returns a ForumThread by its ID.
getPopularThreads	ForumThread	Returns an array of thread IDs for all the popular threads in the system.
deleteThread	void	Deletes a thread with the specified ID.
moveThread	void	Moves the thread with the specified ID to the community with the specified ID.
getThreadProperties	Property	Returns all the extended properties for the thread with the specified ID.
getThreadProperty	String	Returns a specific extended property for the thread with the specified property name and thread ID.
setThreadProperty	void	Set an extended for the property for the thread with the given ID.
deleteThreadProperty	void	Delete a property with the given name from the thread with the given ID.
getThreadsByCommunityID	ForumThread	Returns all of the IDs for threads a community.
getThreadsByCommunityIDAndFilter	ForumThread	Returns all of the IDs for threads a community has filtered by the specified result filter.
getThreadCountByCommunityID	int	Returns the number of threads in the specified community.
getThreadCountByCommunityIDAndFilter	int	Returns the number of threads in the specified community after being filtered by the specified filter.
getPopularThreadsByCommunityID	ForumThread	Return an array of popular threads by community.
createThread	ForumThread	Creates a new thread.
createMessage	ForumMessage	Creates a new message on the given thread.
createReplyMessage	ForumMessage	Creates a new message that is a direct response to a given message.

GroupService

Provides a the ability for managing groups and group membership.

Field	Type	Description
getGroup	Group	Returns a Group by its ID.
getGroupByName	Group	Returns a Group by its name.
getGroupMembers	User	Returns an array of all userIDs for all the members of a particular group.
getGroupCount	int	Returns a count of all groups in the system.
getGroups	Group	Returns an array of all the group IDs for all the groups in the system.
getUserGroups	Group	Returns an array of all the group IDs that a user belongs too.
getGroupAdmins	User	Returns an array of all the user IDs that administer this group.
getAdministratorCount	int	Returns the count of how many administrators there are for the group with the specified ID.
createGroup	Group	Creates a new group.
deleteGroup	void	Delete the group with the specified id.
updateGroup	void	Update the following group in the system.
getProperties	Property	Returns an array of all the extended properties for a group.
setProperty	void	Set a new extended property on the specified group.
deleteProperty	void	Deletes an extended property from the specified group.
addMemberToGroup	void	Add the user with the specified userID to the group with the specified groupID.
addAdministratorToGroup	void	Make the user with the specified userID an administrator of the group with the specified groupID.
removeMemberFromGroup	void	Remove the user with the specified id from the group with the specified id.
removeAdministratorFromGroup	void	Remove the user with the specified ID as an administrator from the group with the specified

		ID.
--	--	-----

PermissionService

Provides a webservice for managing permissions on users and groups.

Field	Type	Description
addPermissionToUser	void	Add the specified permission to the user with the specified id.
removePermissionFromUser	void	Remove the specified permission from the user with the specified id.
addPermissionToGroup	void	Add the specified permission to the group with the specified id.
removePermissionFromGroup	void	Remove the specified permission from the group with the specified id.
addCommunityPermissionToUser	void	Add the specified permission on the specified community to the user with the specified id.
removeCommunityPermissionFromUser	void	Remove the specified permission on the specified community from the user with the specified id.
isAuthorized	boolean	Returns true if the current user has globally has the specified permission.
isAuthorizedOnCommunity	boolean	Returns true if the current user has the permission specified on the specified community.
registeredUserHasPermission	boolean	Returns true if registered users have a particular permission globally.
registeredUserHasPermissionOnCommunity	boolean	Returns true if registered users have a particular permission on the community with the specified ID.
anonymousUserHasPermission	boolean	Returns true if the anonymous users have a particular permission globally.
anonymousUserHasPermissionOnCommunity	boolean	Returns true if the anonymous users have a particular permission on the community with the specified ID.
usersWithPermissionCount	int	Returns a count of the users that have a particular permission.
usersWithPermission	long	Returns all the userID's of users

		with a particular permission.
usersWithPermissionCountOnCommunity	int	Returns a count of the users that have a particular permission on the specified community.
usersWithPermissionOnCommunity	long	Returns all the userID's of users with a particular permission on the specified community.
removeCommunityPermissionFromGroup	Group	Remove the specified permission on the specified community from the group with the specified id.
addCommunityPermissionToGroup	void	Add the specified permission on the specified community to the group with the specified id.
isUserAuthorized	boolean	Check to see if the specified user has the particular permission system wide.
isUserAuthorizedOnCommunity	boolean	Check to see if the specified user has the particular permission on the specified community.

PrivateMessageService

Provides the ability to manipulate private messages.

Field	Type	Description
isPrivateMessagesEnabled	boolean	Returns <code><tt>>true</tt></code> if the feature is enabled, <code><tt>>false</tt></code> otherwise.
getMessages	PrivateMessage	Returns all the messages in the folder sorted by date descending.
deleteMessage	void	Deletes a private message from the folder by moving it to the trash folder.
moveMessage	void	Moves a private message to another folder.
getMessageCount	int	Returns the total number of private messages a user has in their mailbox.
getMessageCountForFolder	int	Returns the message count on a specific folder.
getUnreadMessageCount	int	Returns the total number of unread private messages a user has in their mailbox.
getUnreadMessageCountForFolder	int	Returns the total number of unread private messages a user

		has in a specific folder.
getFolders	PrivateMessageFolder	Returns an Iterator of PrivateMessageFolder objects for the folders the user has.
getFolder	PrivateMessageFolder	Returns the specified folder for a user.
createFolder	PrivateMessageFolder	Creates a new folder.
deleteFolder	void	Deletes a folder.
getMessage	PrivateMessage	Returns the specified private message.
createMessage	PrivateMessage	Creates a new private message.
saveMessageAsDraft	void	Saves a message as a draft by storing it in the sender's <tt>Drafts</tt> folder.
sendMessage	void	Sends a private message to another user.

SearchService

Provides the ability to search for content.

Field	Type	Description
search	List	Provides the ability to create complex search queries with the ability to change sorting, filtering, etc.
quickSearch	List	Provides the ability to do quick search queries based on the provided string.
countSearchResults	int	Returns the number of possible results for the specified query.
countQuickSearchResults	int	Returns the number of possible results for the specified query.
searchByCommunities	List	Provides the ability to create complex search queries with the ability to change sorting, filtering, etc.
countSearchResultsByCommunityID	int	Returns the number of possible results for the specified query by category.
quickMessageSearchByCommunityID	List	Provides the ability to do quick search queries based on the provided string.
countQuickMessageSearchResultsByCommunityID	int	Returns the number of possible

		results for the specified query.
--	--	----------------------------------

SystemPropertiesService

Provides a web service for managing Jive System Properties.

Field	Type	Description
getProperties	Property	Obtains all Jive System Properties.
saveProperty	void	Saves a name/value pair as a Jive System Property.
deleteProperty	void	Deletes a Jive System Property.

UserService

Provides a webservice for managing user's, avatar's, and status levels.

Field	Type	Description
getUser	User	Returns a user by its id.
getUserByUsername	User	Returns a user by its username.
updateUser	void	Used to update user information in the system.
createUser	User	Create a new user.
deleteUser	void	Delete the user with the specified id.
getUserProperties	Property	Return all extended properties for the user with the specified id.
setUserProperty	void	Set an extended property for a user.
deleteUserProperty	void	Delete an extended property from a user.
getUserCount	int	Returns the number of users in the system.
getUsersBounded	User	Returns the IDs of users beginning at startIndex and until the number results equals numResults.
getUsers	User	Returns the IDs of the first 1000 users.
getUserByEmailAddress	User	Returns a user object corresponding to the email address given.

WatchService

A service for manipulating a user's watches on objects.

Field	Type	Description
getDeleteDays	int	Returns the number of days that a watched object can remain inactive before watches on that object are deleted.
setDeleteDays	void	Sets the number of days that a watched object can remain inactive before watches on that object are deleted.
createUserWatch	Watch	Create a watch on a user for the specified user.
createThreadWatch	Watch	Create a watch on a thread for the specified user.
createCommunityWatch	Watch	Create a watch on a community for the specified user.
getCommunityWatchCount	int	Return the count of all community watches in a particular communityID for the given userID.
getTotalWatchCount	int	Returns a count of all watches that a userID has of a particular type.
getWatchList	Watch	Returns an array of Watch objects for a particular object type that the given user is watching.
getCommunityWatches	long	Returns an array of IDs for all the community objects a user is watching in a community.
getUserWatch	Watch	Returns a watch on a particular user, or null if there isn't a watch.
getThreadWatch	Watch	Returns a watch on a particular thread, or null if there isn't a watch.
getCommunityWatch	Watch	Returns a watch on a particular community, or null if there isn't a watch.
isCommunityWatched	boolean	Returns true if the user is watching the specified community.
isThreadWatched	boolean	Returns true if the user is

		watching the specified thread.
isUserWatched	boolean	Returns true if the user is watching the specified user.
deleteWatch	void	Delete the specified watch.
deleteWatches	void	Deletes all watches that a user has.
getWatchUsers	User	Returns all the users who are watching this objectType and objectID.

#Services

Permission Values

Value	Description
0x0000000000000000	No Permssion
0x1000000000000000	Permission to see the online status of a particular user.
0x2000000000000000	Permission to administer a particular user.
0x4000000000000000	Permission to administer a particular group.
0x8000000000000000	Permission to administer the entire sytem.
0x10000000000000000	Custom permission 1. Custom permissions are values reserved for use by end-users to implement custom permission behavior.
0x20000000000000000	Custom permission 2.
0x40000000000000000	Custom permission 3.
0x80000000000000000	Custom permission 4.
0x001	Permission to read a forum.
0x002	Permission to create new messages.
0x004	Permission to create new threads.
0x008	Permission to add attachments to messages.
0x010	Permission to create polls.
0x020	Permission to create private messages.
0x040	Permission to rate messages.
0x080	Permission to moderate threads.
0x100	Permission to administer a particular forum.
0x200	Permission to administer a category.
0x400	Permission to vote in a poll.
0x800	Permission to create attachments in private

	messages.
0x1000	Permission to create, delete, and schedule announcements.
0x2000	Permission to create and delete avatars.

[Back to Services](#)

Client API Examples

Jive Software provides a Java and .NET client APIs. For both languages, the entry point to the client API is the class *com.jivesoftware.forum.webservices.ServiceLocator*. The *ServiceLocator* handles authentication and provides getters for all of the Services.

Below are a few example of how to use the client APIs to connect to the Jive Clearspace service and perform various tasks. Note, most connections use the HTTPS protocol. We recommend this approach as it's the most secure. The system can be configured to use plain HTTP connections. For descriptions of the various service objects below, see the #Services.

Also note, some Java examples use JDK 1.5 syntax (foreach loop, Generics). This is just for example purposes as the client API will work on JDK 1.4 or better.

Acquire the Root Forum Category

```
ServiceLocator locator = new ServiceLocator("https://myforums.com", "admin", "password");
CommunityService communityService = locator.getCommunityService();
Community community = communityService.getCommunity(1);
```

```
ServiceLocator locator = new ServiceLocator("https://myforums.com", "admin", "password");
CommunityService communityService = locator.getCommunityService();
Community community = communityService.getCommunity(1);
```

Create a New Thread

```
ServiceLocator locator = new ServiceLocator("https://myforums.com", "admin", "password");
CommunityService communityService = locator.getCommunityService();
ForumService forumService = locator.getForumService();
UserService userService = locator.getUserService();

// We want this thread to appear in the community with ID 33
Community community = communityService.getCommunity(33);

// Get the user we want to post the message as
User user = userService.getUser("myUsername");

// Create the thread
ForumThread thread = forumService.createThread("My Test Message", "My message body",
community.getID(), user.getID());
```

```
ServiceLocator locator = new ServiceLocator("https://myforums.com", "admin", "password");
CommunityService communityService = locator.getCommunityService();
ForumService forumService = locator.getForumService();
```

```

UserService userService = locator.userService;

// We want this thread to appear in the community with ID 33
Community community = communityService.GetCommunity(33);

// Get the user we want to post the message as
User user = userService.GetUser("myUsername");

// Create the thread
ForumThread thread = forumService.CreateThread("My Test Message", "My message body",
community.ID, user.ID);

```

Add a Reply to a Message

```

// The message to reply to
long messageID = 1221;
// The user we're posting the message as
long userID = 12123;

ServiceLocator locator = new ServiceLocator("https://myforums.com", "admin", "password");
ForumService contentService = locator.getContentService();
// Create the reply
ForumMessage reply = forumService.createReplyMessage("my subject", "my response", message,
userID);

```

```

// The message to reply to
long messageID = 1221;
// The user we're posting the message as
long userID = 12123;

ServiceLocator locator = new ServiceLocator("https://myforums.com", "admin", "password");
ForumService forumService = locator.contentService;
// Create the reply
ForumMessage reply = forumService.CreateReplyMessage("my subject", "my response", message,
userID);

```

Add an Attachment to a Message

```

// The message we're going add an attachment to
long messageID = 12123;

// Create a FileDataSource for the attachment. Do this with a DataSource object
// (part of the Java Activation Framework)
javax.activation.DataSource myData = new
javax.activation.FileDataSource("/home/me/myfile.doc");

ServiceLocator locator = new ServiceLocator("https://myforums.com", "admin", "password");
ForumService svc = locator.getForumService();
svc.addAttachmenToMessage("myfile", myData, messageID);

```

```

// The message we're going add an attachment to
long messageID = 12123;
bytes[] myData = File.ReadAllBytes("c:\myFile.txt");

ServiceLocator locator = new ServiceLocator("https://myforums.com", "admin", "password");
ForumMessageService svc = locator.forumMessageService;
svc.AddAttachmenToMessage("myfile", myData, messageID);

```

Create a New User

```

ServiceLocator locator = new ServiceLocator("https://myforums.com", "admin", "password");

```

```
UserService svc = locator.getUserService();

User user = svc.createUser("johnsmith", "password", "johns@foo.com");
```

```
ServiceLocator locator = new ServiceLocator("https://myforums.com", "admin", "password");
UserService svc = locator.userService;

User user = svc.CreateUser("johnsmith", "password", "johns@foo.com");
```

Get First 1000 Messages in a Thread

```
ServiceLocator locator = new ServiceLocator("https://myforums.com", "admin", "password");
ForumService forumService = locator.GetForumService();

long myThreadID = 333L;
ArrayList<ForumMessage> messages = new ArrayList<ForumMessage>();
long[] messageIDs = forumService.getMessageIDsByThreadID(myThreadID);

for (long messageID : messageIDs) {
    ForumMessage current = messageService.getForumMessage(messageID);
    messages.add(current);
}
```

```
ServiceLocator locator = new ServiceLocator("https://myforums.com", "admin", "password");
ForumService svc = locator.forumService;

long myThreadID = 333L;
List<ForumMessage> messages = new List<ForumMessage>();
long[] messageIDs = svc.GetMessageIDsByThreadID(myThreadID);

for (long messageID in messageIDs) {
    ForumMessage current = svc.GetForumMessage(messageID);
    messages.Add(current);
}
```

Execute a Search

```
ServiceLocator locator = new ServiceLocator("https://myforums.com", "admin", "password");
SearchService service = locator.getSearchService();

Query query = new Query();
query.setQueryString("database java");
query.setAfterDate(new GregorianCalendar(2006, 1, 1).getTime()); // Anything later than Jan 1, 2006
query.setSortField(Query.DATE);
query.setSortOrder(Query.ASCENDING);

// Only grab the first 100 entries
long[] messageIDs = service.messageSearch(query, 1, 100);
```

```
ServiceLocator locator = new ServiceLocator("https://myforums.com", "admin", "password");
SearchService service = locator.searchService;

Query query = new Query();
query.queryString = "database java";
query.AfterDate = new DateTime(2006, 1, 1); // Anything later than Jan 1, 2006
query.SortField = Query.DATE;
query.SetSortOrder = Query.ASCENDING;

// Only grab the first 100 entries
long[] messageIDs = service.MessageSearch(query, 1, 100);
```

Grant a Specific Group Read Permission to a Specific Forum

```
long myForumID = 3333L;

ServiceLocator locator = new ServiceLocator("https://myforums.com", "admin", "password");
GroupService groupService = locator.getGroupService();
PermissionService permissionService = locator.getPermissionService();

Group group = groupService.getGroup("myGroup");
permissionService.addForumPermissionToGroup(com.jivesoftware.forum.ForumPermissions.READ_FORUM,
true, group.getID(), myForumID);
```

```
long myForumID = 3333L;

ServiceLocator locator = new ServiceLocator("https://myforums.com", "admin", "password");
GroupService groupService = locator.groupService;
PermissionService permissionService = locator.permissionService;

Group group = groupService.GetGroup("myGroup");
permissionService.AddForumPermissionToGroup(Permissions.READ_FORUM, true, group.ID, myForumID);
```

Client API Dependencies

The client APIs have the following dependencies, which are included in the distributions.

The Java client API dependencies.

The java client includes one jar which contains all the dependencies in the jar (jiveforums-webservices-5.1.jar) and one that only contains the core libraries itself. The java client is implemented using the XFire web services library (<http://xfire.codehaus.org>). The libraries needed for the xfire client and its dependencies are:

Library File	Description
xfire-all.jar	XFire libraries
xfire-ws-security.jar	XFire Security extension
activation.jar	Sun Java Activation Framework
commons-codec.jar	Apache Commons Codec library
commons-httpclient.jar	Apache Commons HTTP Client library
commons-logging.jar	Apache Commons Logging, A generic logging interface.
javamail.jar	Sun Mail Framework, used for attachments.
jaxen.jar	Jaxen, a XPath engine.
jdom.jar	JDom, a library for mainulating xml as simple java objects.
stax-api.jar	API for XML pull style parsing.
stax-utils.jar	Utils for stax.
sun-jaxws-api.jar	Sun Jax web services API
sun-saaj-api.jar	Sun Soap Attachments API

sun-saaj-impl.jar	The sun implementation of SAAJ
wss4j.jar	The Apache implementation of the WS-Security spec.
wstx-asl.jar	Woodstox a fast STAX implementation from Codehaus.
XmlSchema.jar	XML Schema support.
xmlsec.jar	XML Security.

The .NET client API dependencies.

Library File	Description
Microsoft.Web.Services2.dll	Microsoft web services extensions, used for WS-Security support.
Castle.DynamicProxy.dll	Dynamic Proxy library. Used for exception handling.