Can I get data from the local My Movies database to a third party application?

Question:

Can I get data from the local My Movies database to a third party application?

Answer:

The My Movies Remote Server API is available to professionals in My Movies, and to users with 2500 points on their user accounts. Be aware that professionals supplying solutions to customers must be part of our installer program - purchasing points for a user account grants the user access to support, but a professional supplier not part of our installer program cannot receive support.

Before integrating the API into your application, you must contact us on support@mymovies.dk, to ensure that we can send you any notices of service or API changes or similar. If there are changes to API functions, they are made so that existing applications will be able to continue their operations.

Important:

With the release of My Movies 5, the Remote Server API have been deprecated for end-users, and is only be available for our professional customers through our installer program.

Notice:

The API is available in all installations of My Movies 5, and is for legacy usage also available in My Movies Home and Essentials Server Solution (previously My Movies for Windows Home Server 2011) 2.11 or later, or My Movies for Windows Media Center or My Movies Collection Management for Windows 4.03 or later on Windows 7 or 8. The API is not supported on Windows XP and Windows Vista.

Availability:

The API is made available through the tray application in My Movies for Media Center

and through the built-in service on My Movies for Windows Home Server 2011.

Once started, the API can be accessed through http://127.0.0.1:51414/. The port 51414 is opened in the Windows Firewall when My Movies is installed, however notice that it is registered to be opened on "Home Network" or "Work Networks" only, which means that it is not open if the machine is configured to be on a public network. Similar, if a third party firewall or security software is used on the Host PC, the port must be opened manually.

The remote server advertises itself through Apple Bonjour services, with the type id "_mymoviesdata._tcp", including the name of the machine and the port number of the service. The name of the machine can be changed through the API itself. Notice that Bonjour advertising is available only when Bonjour Printing services is installed on the host machine, a software that is not automatically installed with My Movies.

Commands can be sent to the API such as http://127.0.0.1:51414/?command=SendKey&key=Up&username=user&password=pass.

The following informations must be sent with all requests:

"command". This specifies the command you would like to call.

"username" (alternative to apikey). This is the users username as specified in your application settings, which must match the username used in My Movies on the Host PC. The requirement of this ensures that the local data API is not fully open.

"password" (alternative to apikey). This is the users password as specified in your application settings, which must match the password used in My Movies on the Host PC. The requirement of this ensures that the local data API is not fully open.

"apikey" (alternative to username/password). If you do not want the users to log into your remote application with their My Movies user account, you can as an alternative ask them to enter an API key. This is typically used by custom installer installations where a user account is optional. The API key, which is a Unique generated string can be found in "Collection Management", "Tools", "Options", "Remote Control Server" in My Movies for Windows, or in "Remote Control Server" settings in the My Movies Home and Essentials Server Solution software package.

The API either returns with an XML file or the actual file that was requested. In case of an xml file, it will contain a response node, such as:

<Response status="ok"></Response>

or

<Response status="failed">Message from server.</Response>

The server can return the following statuses:

"ok". The command was received and processed as expected. This is the default response. Depending on the command, the Response node may contain either XML or clear text inside for the result.

"failed". The command failed for one or more reasons. The reason is included as clear text inside the Response node.

There can be other return values on specific commands, which will be documented with the command.

The following API commands are available (the parameters mentioned are additional to the three mentioned above):

Ping - This command returns "pong", and can be used to check if the service is available.

Version - This command returns the version of the My Movies application on the Host PC. This can be used if API commands are added in later versions, and your application requires a specific version. The version is returned in the format "W.X.Y.Z", where "W" specifies the major version, "X" the first digit of the minor version, "Y" the second digit of the minor version, and "Z" specifies the Pre Release version, or build number on released versions. A "Z" version below 100 is a "Pre Release" version, with 1 being "Pre Release 1", 2 being "Pre Release 2" and so on. A "Z" version of "100" or higher indicates a final release version, with 100 being the initial final release, 102 being a "Build 2" and so on. As an example "4.0.2.100" specifies a final "4.02" version, where "4.0.3.2" specifies "4.03 Pre Release 2".

Rename (name) - Renames the instance name advertised through Bonjour, which

means that the default machine name can be changed to "Livingroom" or similar.

ResetName - Resets the instance name advertised through Bonjour to it's default value, which is the machine name.

GetTitleList (sortby, sortdirection, fieldlist, browsinginterface, maxparentallevel, maxparentaltvlevel, includeunrated, parentalcategoryrestrict, parentalcategoryexclude, parentalcategoryinclude, hidechilditems, filterby, filtervalue) - Returns a list of the titles that are in the users collection.

The optional parameter "sortby" (Added in 4.05 Build 5) can take values:

id

sorttitle

productionyear

releasedate

rating

runtime

parentalrating

collectionnumber

recordchanged

If the paramter "sortby" is not supplied, the XML will be sorted by the collection number.

The optional parameter "sortdirection" (Added in 4.05 Build 5) can take values:

asc

desc

The optional parameter "fieldlist" (Added in 4.05 Build 5) is a comma separated list of fields (eg. "id,rating,productionyear"), and can include fields:

productionyear

releasedate

rating

runtime

parentalrating

barcode

type

sorttitle

edition

aspectratio

videoformat

adult

imdbid

mediatype

bd3d

genres

categories

collectionnumber

The optional parameter "browsinginterface" (Added in 4.06 Pre Release 5) is a true/false value that informs the API if you are sending your query for a collection browsing interface. The software has an option to hide certain groups, like titles in wish-list from the browsing interface, which are applied if you send the value 'true' for this parameter. The default is 'false'.

The optional parameter "maxparentallevel" and "maxparentaltvlevel" (Added in 4.06 Pre Release 5) defines a maximum parental control level to return titles for as a numeric value from 0 to 8, where 8 (default) is the highest value. The "maxparentaltvlevel" applies to titles using TV rating values, while "maxparentallevel" applies to all other titles. The parental control levels maps to

different values in different countries - contact us for a detailed list.

The optional parameter "includeunrated" (Added in 4.06 Pre Release 5) is a true/false value, which determines if titles that either does not have a rating set, or is rated "Unrated" should be returned. The default value is 'true'.

The optional parameter "parentalcategoryrestrict" (Added in 4.06 Pre Release 5) is a numeric value containing a the id of a category to restrict the result set to. If supplied, only titles contained in this category will be returned, except if the "parentalcategoryinclude" option is supplied, in which case titles in this category will be returned as well. The intended use for this parameter is in relation to parental control functions, where devices in kids rooms might by default be limited to show only movies in a "Children" category.

The optional parameter "parentalcategoryexclude" (Added in 4.06 Pre Release 5) is a numeric value containing the id of a category to exclude from the result set, no matter if other parental control parameters applies. An example of the usage of this parameter could be if a collection contains adult titles, but the user do not wish to block "Unrated" or titles with high ratings in general, then they can configure parental controls to always block titles that is in an "Adult" category.

The optional parameter "parentalcategoryinclude" (Added in 4.06 Pre Release 5) is a numeric value containing the id of a category to include in the result set, no matter if other parental control parameters applies. An example of the usage of this parameter could be if parental controls is set to block "Unrated" titles, you can select to include titles in a "Children" category, even if titles in that category is listed as "Unrated".

The optional parameter "hidechilditems" (Added in 4.06 Pre Release 5) is a true/false value, which determines if titles in box set containers should be excluded from the result set. If you are building a browsing interface, it is recommended to pass this parameter as 'true' and instead let the user open the box set container titles to see content of it. If you pass this parameter as 'true', the result set will contain an additional node with the child title count, allowing you to determine which items that should be listed as box sets, and therefore could be opened by the user. Once the user opens a box set container, use the "filterby" and "filtervalue" parameters to filter by the selected box set. The default value is 'false'.

The optional parameters "filterby" and "filtervalue" (Added in 4.06 Pre Release 5) allows you to filter the result set by various parameters:

- If "filterby" is set to "boxset", you can include a numeric id of a parent title to filter by in the "filtervalue" parameter.
- If "filterby" is set to "category", you can include a numeric id of a category to filter by in the "filtervalue" parameter.
- If "filterby" is set to "genre", you can include a numeric id of a genre, or the string based internal name of a genre in the "filtervalue" parameter.
- If "filterby" is set to "filter", you can include a numeric id of a stored filter to filter by in the "filtervalue" parameter.
- If "filterby" is set to "online", only titles that are stored online (on storage such as harddrives or NAS devices) is returned.
- If "filterby" is set to "offline", only titles that are not pointed to online storage paths (such as harddrives or NAS devices) is returned.

The returned XML will always include id and recordchanged values. If the "fieldlist" parameter is not supplied, additionally localtitle and originaltitle values is automatically included.

GetCollection (listtype, owned, tvseriesondiscsasdiscs, sortby, sortdirection, fieldlist, browsinginterface, maxparentallevel, maxparentaltvlevel, includeunrated, parentalcategoryrestrict, parentalcategoryexclude, parentalcategoryinclude, hidechilditems, filterby, filtervalue) - (Added in 4.06 Pre Release 5) - Returns a list of elements in users collection. Designed for browsing interfaces.

The required parameter "listtype" can take values "Movies", "TVSeries" or "Media" (default), indicating if you want only movie items, tv series items or all media returned.

The optional parameter "owned" is a true/false values that specifies if you would like to show only TV Series that have episodes that the user has in their collection, in which case the "owned" parameter is provided as "true" (default). If you would like to return all TV Series, no matter if it contains episodes that the user owns, this parameter should be provided as "false".

The optional parameter "tvseriesondiscsasdiscs" can take a true/false value, which

defines if disc titles (DVD or Blu-ray) which contains known TV Series and TV Episode structures should be returned as disc titles, or if they should be returned as TV Series.

The optional parameter "sortby" can take values:

sorttitle

productionyear

releasedate

ratingbac

runtime

parentalrating

collectionnumber

watched

recordchanged

firstaired

episodeadded

added (Added in 5.00 Pre Release 10)

If the paramter "sortby" is not supplied, the XML will be sorted by the collection number.

The optional parameter "sortdirection" can take values:

asc

desc

The optional parameter "fieldlist" is a comma separated list of fields (eg. "id,rating,productionyear"), and can include fields:

title

Can I get data from the local My Movies database to a third party application?

localtitle

originaltitle

productionyear

releasedate

rating

runtime

parentalrating

barcode

sorttitle

edition

aspectratio

videoformat

adult

imdbid

bd3d

genres

categories

collectionnumber

watched

firstaired

episodeswatched

episodesunwatched (Added in 5.00 Pre Release 10)

lastepisodeadded

episodecount (Added in 5.00 Pre Release 10)

mediatype (Added in 5.00 Pre Release 10)

disctype

The optional parameter "browsinginterface" is a true/false value that informs the API if you are sending your query for a collection browsing interface. The software has an option to hide certain groups, like titles in wish-list from the browsing interface, which are applied if you send the value 'true' for this parameter. The default is 'false'.

The optional parameter "maxparentallevel" and "maxparentaltvlevel" defines a maximum parental control level to return titles for as a numeric value from 0 to 8, where 8 (default) is the highest value. The "maxparentaltvlevel" applies to titles using TV rating values, while "maxparentallevel" applies to all other titles. The parental control levels maps to different values in different countries - contact us for a detailed list.

The optional parameter "includeunrated" is a true/false value, which determines if titles that either does not have a rating set, or is rated "Unrated" should be returned. The default value is 'true'.

The optional parameter "parentalcategoryrestrict" is a numeric value containing a the id of a category to restrict the result set to. If supplied, only titles contained in this category will be returned, except if the "parentalcategoryinclude" option is supplied, in which case titles in this category will be returned as well. The intended use for this parameter is in relation to parental control functions, where devices in kids rooms might by default be limited to show only movies in a "Children" category.

The optional parameter "parentalcategoryexclude" is a numeric value containing the id of a category to exclude from the result set, no matter if other parental control parameters applies. An example of the usage of this parameter could be if a collection contains adult titles, but the user do not wish to block "Unrated" or titles with high ratings in general, then they can configure parental controls to always block titles that is in an "Adult" category.

The optional parameter "parentalcategoryinclude" is a numeric value containing the id of a category to include in the result set, no matter if other parental control parameters applies. An example of the usage of this parameter could be if parental controls is set to block "Unrated" titles, you can select to include titles in a "Children" category, even if titles in that category is listed as "Unrated".

The optional parameter "hidechilditems" is a true/false value, which determines if

titles in box set containers should be excluded from the result set. If you are building a browsing interface, it is recommended to pass this parameter as 'true' and instead let the user open the box set container titles to see content of it. If you pass this parameter as 'true', the result set will contain an additional node with the child title count, allowing you to determine which items that should be listed as box sets, and therefore could be opened by the user. Once the user opens a box set container, use the "filterby" and "filtervalue" parameters to filter by the selected box set. The default value is 'false'.

The optional parameters "filterby" and "filtervalue" allows you to filter the result set by various parameters:

If "filterby" is set to "boxset", you can include a numeric id of a parent title to filter by in the "filtervalue" parameter.

If "filterby" is set to "category", you can include a numeric id of a category to filter by in the "filtervalue" parameter.

If "filterby" is set to "genre", you can include a numeric id of a genre, or the string based internal name of a genre in the "filtervalue" parameter.

If "filterby" is set to "filter", you can include a numeric id of a stored filter to filter by in the "filtervalue" parameter.

If "filterby" is set to "online", only titles that are stored online (on storage such as harddrives or NAS devices) is returned.

If "filterby" is set to "offline", only titles that are not pointed to online storage paths (such as harddrives or NAS devices) is returned.

If "filterby" is set to "boxsets", only box set container titles is returned.

If "filterby" is set to "notwatched", only titles that have not been watched is returned.

If "filterby" is set to "title", you can include a string in the "filtervalue" parameter to search for. This is the recommended method for searching the collection.

The returned XML will always include id, type (DiscTitle, TV Series), mediaType (sub type for disc titles) and episodeCount values. If the "fieldlist" parameter is not supplied, additionally the title value is automatically included.

GetTVSeriesList (owned, sortby, sortdirection, fieldlist, browsinginterface, maxparentaltvlevel, includeunrated, parentalcategoryrestrict, parentalcategoryexclude, parentalcategoryinclude, filterby, filtervalue) - (Added in 4.06 Pre Release 5) - Returns a list of the TV Series that are in the users collection.

The optional parameter "owned" is a true/false values that specifies if you would like to show only series that have episodes that the user has in their collection, in which case the "owned" parameter is provided as "true" (default). If you would like to return all series, no matter if it contains episodes that the user owns, this parameter should be provided as "false".

The optional parameter "sortby" can take values:

guid

sorttitle

firstaired

rating

runtime

parentalrating

recordchanged

If the paramter "sortby" is not supplied, the XML will be sorted by sort title.

The optional parameter "sortdirection" can take values:

asc

desc

The optional parameter "fieldlist" is a comma separated list of fields (eg. "guid,rating,runtime"), and can include fields:

sorttitle

firstaired

rating

runtime

parentalrating

genres

categories

episodeswatched

The optional parameter "maxparentaltvlevel" defines a maximum parental control level to return titles for as a numeric value from 0 to 8, where 8 (default) is the highest value. The parental control levels maps to different values in different countries - contact us for a detailed list.

The optional parameter "includeunrated" is a true/false value, which determines if titles that either does not have a rating set, or is rated "Unrated" should be returned. The default value is 'true'.

The optional parameter "parentalcategoryrestrict" is a numeric value containing a the id of a category to restrict the result set to. If supplied, only titles contained in this category will be returned, except if the "parentalcategoryinclude" option is supplied, in which case titles in this category will be returned as well. The intended use for this parameter is in relation to parental control functions, where devices in kids rooms might by default be limited to show only movies in a "Children" category.

The optional parameter "parentalcategoryexclude" is a numeric value containing the id of a category to exclude from the result set, no matter if other parental control parameters applies. An example of the usage of this parameter could be if a collection contains adult titles, but the user do not wish to block "Unrated" or titles with high ratings in general, then they can configure parental controls to always block titles that is in an "Adult" category.

The optional parameter "parentalcategoryinclude" is a numeric value containing the id of a category to include in the result set, no matter if other parental control parameters applies. An example of the usage of this parameter could be if parental controls is set to block "Unrated" titles, you can select to include titles in a "Children" category, even if titles in that category is listed as "Unrated".

The optional parameters "filterby" and "filtervalue" allows you to filter the result set by various parameters:

- If "filterby" is set to "category", you can include a numeric id of a category to filter by in the "filtervalue" parameter.
- If "filterby" is set to "genre", you can include a numeric id of a genre, or the string based internal name of a genre in the "filtervalue" parameter.
- If "filterby" is set to "filter", you can include a numeric id of a stored filter to filter by in the "filtervalue" parameter.

The returned XML will always include guid, recordchanged, personalChaged, episodeCount, lastEpisodeChange and lastEpisodePersonalChange values. If the "fieldlist" parameter is not supplied, additionally the title value is automatically included.

GetTVSeasonList (series, owned) - (Added in 4.06 Pre Release 5) - Returns a list of the TV Seasons in a TV Series.

The required parameter "series" specifies the TV Series guid to return seasons for.

The optional parameter "owned" is a true/false values that specifies if you would like to show only seasons that have episodes that the user has in their collection, in which case the "owned" parameter is provided as "true" (default). If you would like to return all seasons, no matter if it contains episodes that the user owns, this parameter should be provided as "false".

The returned XML will include season number, episodeCount, lastEpisodeChange and lastEpisodePersonalChange values. There is not any name for a season, as it is to be listed in interfaces as "Season X" where X is the season number. If a browsing interface would like to display an "All Seasons" items, this must be added as a virtual item by the browsing interface itself.

GetTVEpisodeList (series, owned, sortby, sortdirection, fieldlist, filterby, filtervalue) - (Added in 4.06 Pre Release 5) - Returns a list of the TV Series that are in the users collection.

The required parameter "series" specifies the TV Series guid to return episodes for.

The optional parameter "owned" is a true/false values that specifies if you would like to show only series that have episodes that the user has in their collection, in which case the "owned" parameter is provided as "true" (default). If you would like to return all series, no matter if it contains episodes that the user owns, this parameter should be provided as "false".

The optional parameter "sortby" can take values:

guid

sorttitle

firstaired

recordchanged

seasonnumber

episodenumber (Sorts by Season Number, then Episode Number)

If the paramter "sortby" is not supplied, the XML will be sorted by sort title.

The optional parameter "sortdirection" can take values:

asc

desc

The optional parameter "fieldlist" is a comma separated list of fields (eg. "guid,rating,runtime"), and can include fields:

sorttitle

firstaired

rating

watched

added

The optional parameters "filterby" and "filtervalue" allows you to filter the result set by various parameters:

- If "filterby" is set to "seasonnumber", you can include a numeric id of a season to filter by in the "filtervalue" parameter.

The returned XML will always include guid, episodeNumber, seasonNumber, recordChanged and personalChaged, episodeCount, lastEpisodeChange and lastEpisodePersonalChange values. If the "fieldlist" parameter is not supplied, additionally the title value is automatically included.

GetLatestChange - (Legacy as of 4.06 Pre Release 5, replaced with GetCollectionChanges) - Returns the latest date a title was changed in or added to the users collection. It also returns a the number of titles in the users collection. Both entries should be used to determine changes, as a deleted title will not change the date of the last changed title.

GetCollectionChanges - (Added in 4.06 Pre Release 5) - Returns a list of latest change dates to all elements in the system, and also includes an element count. Both entries should be used to determine changes, as a deleted elements will not change the date of the last changed title.

GetGenres - (Added in 4.06 Pre Release 5) Returns a list of genres. The list only returns genres that are used on one or more titles in the collection.

GetCategories - (Added in 4.06 Pre Release 5) Returns a list of categories. The list only returns categories that are used on one or more titles in the collection.

GetFilters - (Added in 4.06 Pre Release 5) Returns a list of stored filters.

GetTitle (id, excludedisctitles) - Returns a document with the full overview of a title in the users collection. The optional parameter "excludedisctitles" (Added in 4.05 Build 5) with possible values true or false excludes titles and chapter structures of discs, to be able to load a smaller XML documents, as these can include much data.

GetTVSeries (guid) - Returns a document with the full overview of a tv series in the users collection.

GetMovie (guid) - Returns a document with the full overview of a movie in the users collection.

GetMovieCollection (guid) - Returns a document with the full overview of a movie collection in the users collection.

GetTitlePersonalData (id, excludedisctitles) (Added in 4.06 Build 3) - Returns a document with the personal data of a title in the users collection.

GetTitleAndPersonalData (id, excludedisctitles) (Added in 4.06 Build 3) - Returns a document that combines the GetTitle and GetTitlePersonalData calls.

GetTitleTrailers (id, country, includeenglish) - (Added in 4.06 Pre Release 5) Returns one or more trailers for the title. If the user have pointed to a local trailer in his collection, the path to this is supplied, however due to lack of credentials, this often cannot be used by non-Windows based environments. If the user has a "Trailer" or "Trailers" folder in the folder path of the title, a list trailer files for this path is returned, but also due to lack of credentials, often cannot be used by non-Windows based environments. Last, there will be an "OnlineTrailer" returned, which is recommended to use.

The parameter "id" takes an integer value of the title to return trailer(s) for.

The parameter "country" (optional) takes a string value of the country to return an online trailer for. The possible values are "Denmark", "Finland", "France", "Germany", "Italy", "Spain", "Sweden", "Switzerland", "The Netherlands", "United Kingdom". If the users country is either not known, or is not in the list of possible countries, it is recommended to use "United Kingdom".

The parameter "includeenglish" is a true (default) / false value which instructs to include English language trailers if trailers for the country specified with the "country" parameter was not found.

GetTitlePlayOptionsForDune (id) - Returns a list of play options that can be passed to the Dune players IP command API. The parameter "id" takes an integer value of the title to return the play path for.

GetEpisodePlayOptionsForDune (id) - (Added in 4.06 Pre Release 5) Returns a play option for the episode that can be passed to the Dune players IP command API. The parameter "id" takes the guid of the episode to return the play path for.

GetTitleFrontCover (id) - (Legacy as of 4.06 Pre Release 5, replaced with GetImage) - Returns the front cover image for the title with the id.

GetTitleBackCover (id) - (Legacy as of 4.06 Pre Release 5, replaced with GetImage) - Returns the back cover image for the title with the id.

GetTitleFrontThumbnail (id) - (Legacy as of 4.06 Pre Release 5, replaced with GetImage) - Returns the front cover thumbnail image for the title with the id.

GetTitleBackThumbnail (id) - (Legacy as of 4.06 Pre Release 5, replaced with GetImage) - Returns the back cover thumbnail image for the title with the id.

GetTitleBackdrop (id) - (Legacy as of 4.06 Pre Release 5, replaced by GetImage) - Returns the backdrop image for the title with the id.

GetImage (id, type, filterby, filtervalue) - (Added in 4.06 Pre Release 5) - Returns the an image for the element with the guid specified.

The required parameter "id" identifier is an integer for Disc Title (DVD, Blu-ray and HD DVD) profiles, and a guid string for Movie, TV Series, and TV Episode profiles.

The required parameter "type" can contain the following values:

frontcover (Full size)

frontcoverthumb (Maximum 100x140)

frontcoverbigthumb (Maximum 200x280)

frontcovermedium (Maximum 343x480)

backcover (Full size)

backdrop (Disc Titles)

moviebackdrop (Movie) - (Added in 5.13 Pre Release 5)

movieposter (Movie) - (Added in 5.13 Pre Release 5)

movieposterthumb (Movie - Maximum 100x140) - (Added in 5.13 Pre Release 5)

movieposterbigthumb (Movie - Maximum 200x280) - (Added in 5.13 Pre Release 5)

moviepostermedium (Movie - Maximum 343x480) - (Added in 5.13 Pre Release 5)

tvposter (TV Series, TV Seasons) - (Added in 5.13 Pre Release 5)

tvposterthumb (TV Series, TV Seasons - Maximum 100x140) - (Added in 5.13 Pre Release 5)

tvposterbigthumb (TV Series, TV Seasons - Maximum 200x280) - (Added in 5.13 Pre Release 5)

tvpostermedium (TV Series, TV Seasons - Maximum 343x480) - (Added in 5.13 Pre Release 5)

tvbanner (TV Series, TV Seasons) - (Added in 5.13 Pre Release 5)

tvbackdrop (TV Series) - (Added in 5.13 Pre Release 5)

tvimage (TV Episodes) - (Added in 5.13 Pre Release 5)

Legacy as of 5.13 Pre Release 4:

~~poster (TV Series, TV Seasons)~~

~~posterthumb (TV Series, TV Seasons - Maximum 100x140)~~

~~posterbigthumb (TV Series, TV Seasons - Maximum 200x280)~~

~~postermedium (TV Series, TV Seasons - Maximum 343x480)~~

~~banner (TV Series, TV Seasons)~~

~~backdrop (TV Series)~~

~~image (TV Episodes)~~

The optional parameters "filterby" and "filtervalue" allows you to filter the result set by various parameters:

- If "filterby" is set to "seasonnumber", you can include a numeric id of a season to filter by in the "filtervalue" parameter.

AddElementToCategory (id, category) - (Added in 4.06 Pre Release 5) - Adds an element to a category. If the element is a Disc Title, please pass parameter as an integer value for the "id" parameter. If the element is a TV Series, please pass parameter as a string value for the "id" parameter.

ElementWatched (id) - (Added in 4.06 Pre Release 5) - Marks an element as

watched. If the element is a Disc Title, please pass parameter as an integer value for the "id" parameter. If the element is a TV Episode, please pass parameter as a string value for the "id" parameter.

ClearElementWatched (id) - (Added in 4.06 Pre Release 5) - Clears the watched marking of an element. If the element is a Disc Title, please pass parameter as an integer value for the "id" parameter. If the element is a TV Series, please pass parameter as a string value for the "id" parameter.

RemoveElement (id, removefiles) - (Added in 4.06 Pre Release 5) - Removes the element from the system. If the element is a Disc Title, please pass parameter as an integer value for the "id" parameter. If the element is a TV Series or TV Episode, please pass parameter as a string value for the "id" parameter. The parameter "removefiles" is a true/false (default) value to define if files connected to the element should be removed. Warning! The "removefiles" parameter should be used with care, and must include a specific warning to the user that this will remove the files.