

1 Tracker Enhancements

1.1 Support for a Configurable Number of Removable Media Caches

Tracker currently stores information about removable media devices in its database. The database stores information about each removable device's UID, mount point, and mount status. Tracker monitors the HAL for changes in mounts, for new removable media devices that are attached to the system, and for changes to files on the file system.

The function Tracker calls when a removable device is mounted will be modified to:

1. Check if the removable device that was mounted is a new removable device with no cache and current number of removable media caches is equal to the maximum number of removable caches in the configuration file. The configuration setting will allow the user to configure tracker to perform no caching, to have unlimited caches, or store an integer number of caches between 1 and 500.
2. If step 1 is true, query the database to find the removable media cache that has been unmounted for the greatest amount of time and remove it.
3. Continue with the current procedure to index the removable media.

Note that if Tracker is configured to allow an unlimited number of caches, the number of removable devices that can be indexed will still be constrained by the physical size of the storage device Tracker's database is written to.

1.2 Support for a Mechanism to Delete a Removable Media Cache After a Specified Timeout

There is an existing function that looks for all volumes that have not been mounted for more than three days and removes all of the thumbnail information from the database. The function is run when the Tracker daemon process is started and is then run once a day the Tracker daemon is continuously running. This code will be modified in two ways:

- Modify Tracker to add a configuration setting to determine when to clean up a cache instead of being hard-coded for three days. The user will be able to configure tracker to never auto clean the caches or specify an integer number of days that a device must be unmounted for before it is removed.
- Modify Tracker to remove all of the database entries for the cache.

1.3 Support for configuration API

Tracker already has the D-Bus methods `SetIntOption(option, value)` and `SetBoolOption(option, value)` for controlling the configuration. The two new options to support the additions described in sections 1 and 1.2 will be added to the `SetIntOption` method.

Additionally the following functions will be added to expose all configuration options in the D-Bus interface and allow the current configuration to be queried:

Interface	Input	Return	Purpose/Description
AddListOption	string option – The name of the config option to add the value to string value – The value to add	N/A	Adds the specified value to an option list in memory. For example, it can be used to add a path to the “WatchDirectoryRoots” option.
RemoveListOption	string option – The name of the config option to remove the value from string value – The value to remove	N/A	Removes the specified value from an option list in memory. For example, it can be used to remove a directory path from the “WatchDirectoryRoots” option.
SetListOption	string option – The name of the config option to set array value – The array of values to set	N/A	Sets the specified list option to the specified array of values in memory.
SetStringOption	string option – The name of the config option to set string value – The value to set	N/A	Sets the specified string option to the specified string in memory.
GetBoolOption	string option – The name of the config option to get	A bool equal to the current value of the config option	Gets the current setting in memory for the specified boolean option.
GetIntOption	string option – The name of the config option to get	An int equal to the current value of the config option	Gets the current setting in memory for the specified integer option.
GetListOption	string option – The name of the config option to get	An array equal to the current value of the config option	Gets the current setting in memory for the specified list option.
GetStringOption	string option – The name of the config option to get	A string equal to the current value of the config option	Gets the current setting in memory for the specified string option.
SaveConfig	N/A	N/A	Saves the current Tracker configuration to the configuration file on disk. It should be noted that all of the option mutator functions make changes to the configuration only in memory. To make changes to configuration permanent SaveConfig must be called.

1.4 Support for API to manage removable media caches

The methods listed in the following table will be added to the Tracker Daemon’s D-Bus API.

Interface	Input	Return	Purpose/Description
GetRemovableDeviceCaches	N/A	A dictionary that maps the device's UDI to an array of ID, Mount Path, and Enabled status.	This function will list all of the removable devices it has currently cached.
ClearRemovableDeviceCache	int id – The database ID of the cache to delete	N/A	This function will remove a specific removable device cache if it is not enabled (mounted).
ClearAllRemovableDeviceCaches	N/A	N/A	This function will remove all removable device caches that are not mounted.

A new command line tool will also be created to use the new D-Bus methods above to allow for easier testing. The tool will have the following usage:

- tracker-cache list – Lists all of the caches currently stored by tracker.
- tracker-cache delete CACHE_ID – Deletes the cache with the specified CACHE_ID if it is not enabled.
- tracker-cache delete all – Deletes all of the disabled caches.