User Guide and Settings of the Delco Client

Presented GUI of the application is not a complete one. It is more like a research GUI to demonstrate how the underlying middleware works with the configured download libraries and what options you can give to the middleware. Currently you can't save the settings made during the session as we don't want to make decision how settings are wanted to be saved (at the user level, computer level or to the file system, Windows registry etc.). Delco application is implemented to Windows XP and Linux environments.

You can assign delco extension to be launched with the application. Web server should also use x-application/delco MIME type for the delco files in order to make browsers handle them correctly. Optionally you can always save the delco file to the file system and start it manually.

1 Main Window

Here in Figure 1 we see the main window of the application. It shows all active services inside download and channel sections. You can control individual services by selecting and clicking them with the right mouse button and choosing preferred action (info, delete, settings). Menu bar has File, Tools and Help menus. From File –> Open you can manually open delco files. You can also close the application there. Tools menu has same actions for controlling selected services but has also global preferences settings. Changing global settings affects only to new started services. If you want to change settings for a running services use service specific settings for that described previously.
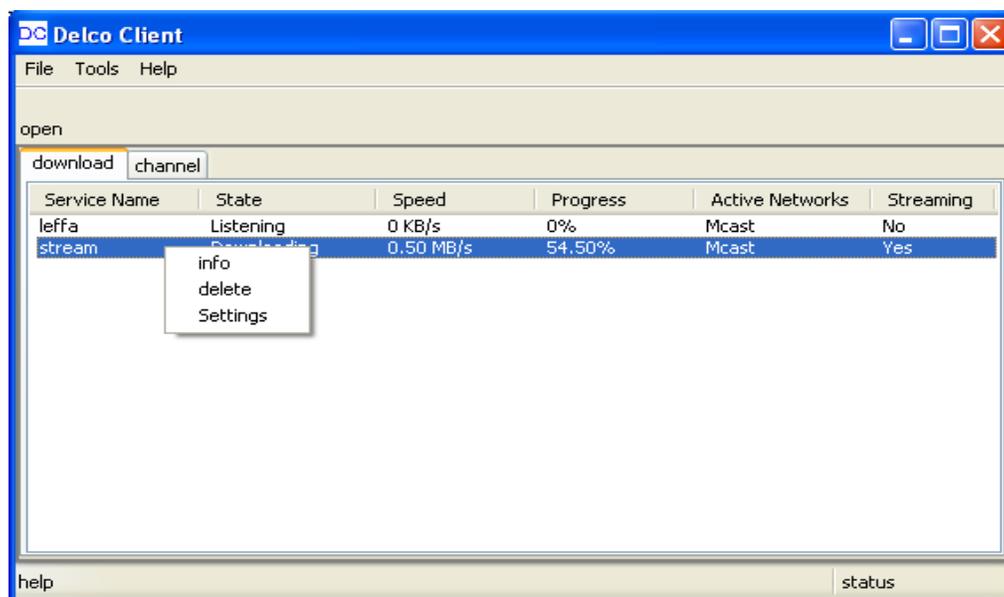

Figure 1: Main window

2 Info Frame

Figure 2 shows status of the stream service which has only one item (file). On the top of the list there is information from the Delco file (report server, item and pointers). After that it shows size of the file and

blocking info (number of pieces and piece size). Number of completed pieces tells how much we have totally completed, completed with P2P/multicast tells how much P2P/multicast protocol has completed. Total download/upload tells the payload (without overhead) value of that. Bytes from P2P/multicast tell the total payload including failed and redundant bytes. Failed bytes are bytes that either failed piece hash check or were received in a position where those couldn't be utilized. Redundant bytes are bytes that were received more than once. P2P/Multicast opened tells whether P2P/multicast component has opened an item (or a file) for writing.

Figure 2: Info frame of the service

3 Settings Dialog

These instructions apply for service specific settings and also for global settings. Remember that global settings affects only to new started services. There are some settings that need more clarification. Figure 3 shows the settings for the downloading. Here you can allow usage of the different protocols, by default they are all checked. Remember that multicast is prioritized in a way that client waits first for a multicast transmission. Currently client doesn't change to peer-to-peer mode automatically if multicast transmission never starts. If there is not going to be an active multicast session you should manually uncheck the multicast option in order to start downloading with peer-to-peer. You can however later resume to the multicast usage.

Peer-to-peer specific rate/slots settings affect at the item level for example if service has two items each item gets definitions set here. More aggressive peer-to-peer usage means that when multicast reception starts in the middle of the file, peer-to-peer component is freed to download the beginning part of the file simultaneously. Otherwise client waits for another round of multicast transmission. By default if multicast reception misses some intermediate pieces they are downloaded with peer-to-peer, however with lazy peer-to-peer usage you can prevent it.
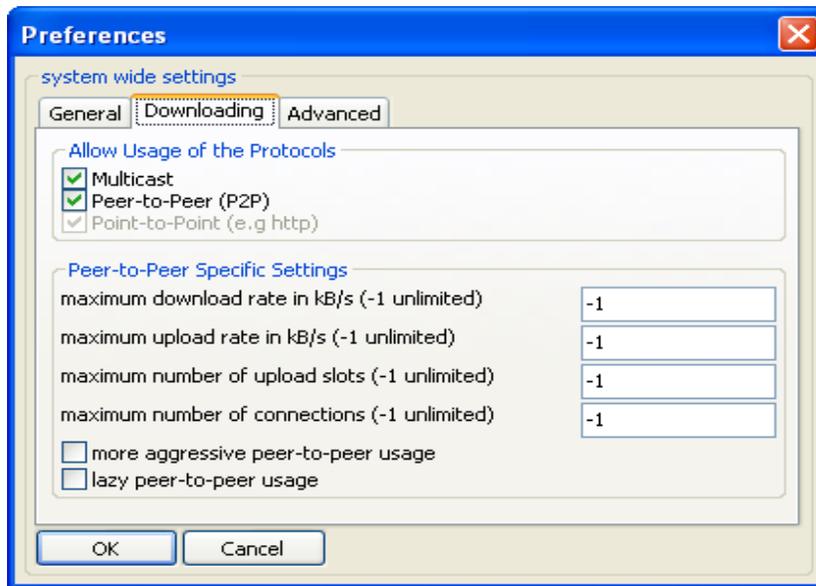
Figure 3: Preferences downloading tab

With streaming endpoint you can define streaming parameters. By default fileuri is used which means that data of stream type of download service is written to the file system (reception prioritized to be in order) where media player can play it. Simulate packet loss makes some random data to be writing in a piece to make piece hash check fail. Currently you can make only multicast data corrupted. Note that if piece hashes are not available (item has only multicast pointer) and piece verification can't be made by other means, failures are not detected. While starting a new service existing items/files can be resumed with file resuming checked however piece hashes must be available to the items/files. Delco reporting can be also shut down if report server is not working or for some other reason.
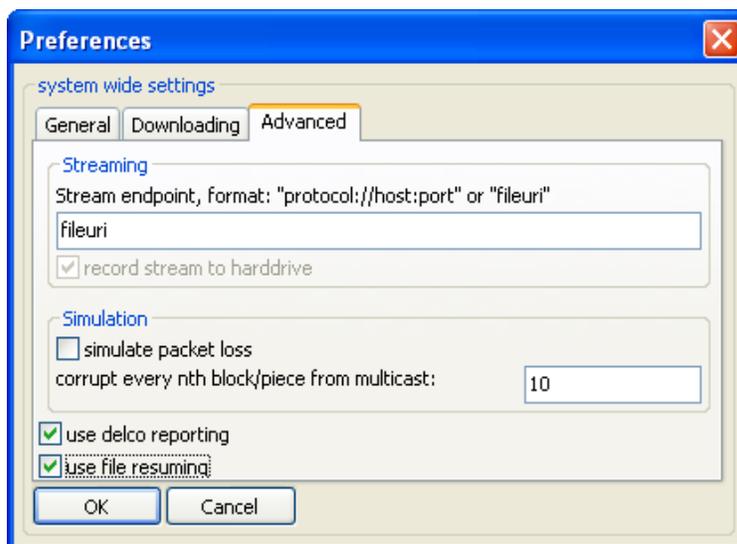

Figure 4: Preferences advanced tab