QLectives – Socially Intelligent Systems for Quality
Project no. 231200

Instrument: Large-scale integrating project (IP)
Programme: FP7-ICT

Deliverable D.4.3.2
QMedia v2 - Short report

Submission date: 2011-02-17

Start date of project: 2009-03-01 Duration: 48 months

Organisation name of lead contractor for this deliverable: TUDelft

Project co-funded by the European Commission within the Seventh Framework Programme (2007-2013)		
Dissemination level		
PU	Public	**X**
PP	Restricted to other programme participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium (including the Commission Services)	
CO	Confidental, only for members of the consortium (including the Commission Services)	

Document information

1.1 Author(s)

Author	Organisation	E-mail
Nazareno Andrade	TU Delft	N.FerreiradeAndrade@tudelft.nl
Tamás Vinkó	TU Delft	T.Vinko@tudelft.nl
Johan Pouwelse	TU Delft	J.A.Pouwelse@tudelft.nl

1.2 Other contributors

Name	Organisation	E-mail
Adele Jia	TU Delft	a.j.lu@tudelft.nl
Niels Zeilemaker	TU Delft	n.zeilemaker@tudelft.nl
Giorgos Logiotatidis	TU Delft	g.t.logiotatidis@tudelft.nl
Boudewijn Schoon	TU Delft	p.b.schoon@tudelft.nl

1.3 Document history

Version#	Date	Change
V0.1	December, 2010	Starting version, template
V0.2	December, 2010	Complete first draft

1.4 Document data

Keywords	peer-to-peer, distributed mass media, decentralized channels, P2P widgets, graphical user interface
Editor address data	N.FerreiradeAndrade@tudelft.nl, T.Vinko@tudelft.nl
Delivery date	17 February, 2011

1.5 Distribution list

Date	Issue	E-mail
	Consortium members	QLECTIVES@list.surrey.ac.uk
	Project officer	Jose.FERNANDEZ-VILLACANAS@ec.europa.eu
	EC archive	INFSO-ICT-231200@ec.europa.eu

QLectives Consortium

This document is part of a research project funded by the ICT Programme of the Commission of the European Communities as grant number ICT-2009-231200.

University of Surrey (Coordinator)
Department of Sociology/Centre
for Research in Social Simulation
Guildford GU2 7XH
Surrey
United Kingdom
Contact person: Prof. Nigel Gilbert
E-mail: n.gilbert@surrey.ac.uk

University of Fribourg
Department of Physics
Fribourg 1700
Switzerland
Contact person: Prof. Yi-Cheng Zhang
E-mail: yi-cheng.zhang@unifr.ch

Technical University of Delft
Department of Software Technology
Delft, 2628 CN
Netherlands
Contact Person: Dr Johan Pouwelse
E-mail: j.a.pouwelse@tudelft.nl

University of Warsaw
Faculty of Psychology
Warsaw 00927
Poland
Contact Person: Prof. Andrzej Nowak
E-mail: nowak@fau.edu

ETH Zurich
Chair of Sociology, in particular
Modelling and Simulation
Zurich, CH-8092
Switzerland
Contact person: Prof. Dirk Helbing
E-mail: dhelbing@ethz.ch

Centre National de la Recherche Scientifique, CNRS
Paris 75006,
France
Contact person: Dr. Camille ROTH
E-mail: camille.roth@polytechnique.edu

University of Szeged
MTA-SZTE Research Group on
Artificial Intelligence
Szeged 6720, Hungary
Contact person: Dr Mark Jelasity
E-mail: jelasity@inf.u-szeged.hu

Institut für Rundfunktechnik GmbH
Munich 80939
Germany
Contact person: Dr. Christoph Dosch
E-mail: dosch@irt.de

QLectives introduction

QLectives is a project bringing together top social modelers, peer-to-peer engineers and physicists to design and deploy next generation self-organising socially intelligent information systems. The project aims to combine three recent trends within information systems:

- **Social networks** - in which people link to others over the Internet to gain value and facilitate collaboration
- **Peer production** - in which people collectively produce informational products and experiences without traditional hierarchies or market incentives
- **Peer-to-Peer systems** - in which software clients running on user machines distribute media and other information without a central server or administrative control

QLectives aims to bring these together to form Quality Collectives, i.e. functional decentralised communities that self-organise and self-maintain for the benefit of the people who comprise them. We aim to generate theory at the social level, design algorithms and deploy prototypes targeted towards two application domains:

- **QMedia** - an interactive peer-to-peer media distribution system (including live streaming), providing fully distributed social filtering and recommendation for quality
- **QScience** - a distributed platform for scientists allowing them to locate or form new communities and quality reviewing mechanisms, which are transparent and promote

The approach of the QLectives project is unique in that it brings together a highly inter-disciplinary team applied to specific real world problems. The project applies a scientific approach to research by formulating theories, applying them to real systems and then performing detailed measurements of system and user behaviour to validate or modify our theories if necessary. The two applications will be based on two existing user communities comprising several thousand people - so-called ”Living labs”, media sharing community tribler.org; and the scientific collaboration forum EconoPhysics.

Executive summary

This report accompanies and documents QMedia version 2.0. QMedia is a next-generation social media distribution platform and one of QLectives' living labs. Its vision is to provide an alternative to mass media through the concept of self-organising personalised collectives.

Version 2.0 of the QMedia software is an extension to and a purpose-built version of Tribler. Tribler is an open-source media-sharing client with a existing user base. Moreover, QMedia is based on the QLectives Platform version 2.0, described in QLectives Deliverable D.4.1.2.

This report concentrates on the changes brought into QMedia since version 1.0, released in Month 12 of QLectives. Such changes are the following:

- *A second version of QMedia's graphical user interface* that improves its performance and usability, and incorporates feedback from users and project partners;
- *Version 2.0 of QMedia's indirect reciprocation mechanism, Bartercast*, incorporating the input from Stream 2 of this project on the need for improved information dissemination;

A preliminary version of QMedia version 2.0 has been released in December 2010. This version of the software, named Tribler 5.3 was downloaded more than 15,000 times since its publication.

Contents

Chapter 1

Introduction

This report describes QMedia version 2.0, the second version of an experimental media-sharing distributed software built around the concept of quality collectives. The network of QMedia users serves as a living lab for QLectives, both as a testbed for developed algorithms and providing input for the design and evaluation of new algorithms.

QMedia's vision is to provide a decentralized, user-centric and social media-distribution platform. This platform can provide an alternative to mass media through self-organising personalised collectives. As planned in the project proposal, QMedia and the QLectives Platform are based on the previously-existent Tribler codebase. Tribler is an open-source software developed since 2006 by the Delft University of Technology in collaboration with several other universities and volunteers. Tribler serves as common codebase for two other FP7 projects, namely P2P-Next [1] and PetaMedia [2].

Both QMedia and the QLectives Platform are composed of core modules from Tribler extended and adapted to meet QLectives-specific requirements. We envision that such extensions and adaptations may lead to QMedia and the QLectives platform to become forks from the Tribler codebase in the future.

The contributions to QMedia documented in this deliverable are relative to QMedia version 1.0, and are the following:

A second version of QMedia's graphical user interface that improves its performance and usability, and incorporates feedback from users and project partners. This contribution is documented in Section 2.

Version 2.0 of QMedia's indirect reciprocation mechanism, Bartercast, incorporating the input from Stream 2 of this project on the need for improved information dissemination. This contribution is documented in Section 3.

Chapter 2

Version 2.0 of QMedia's Graphical User Interface

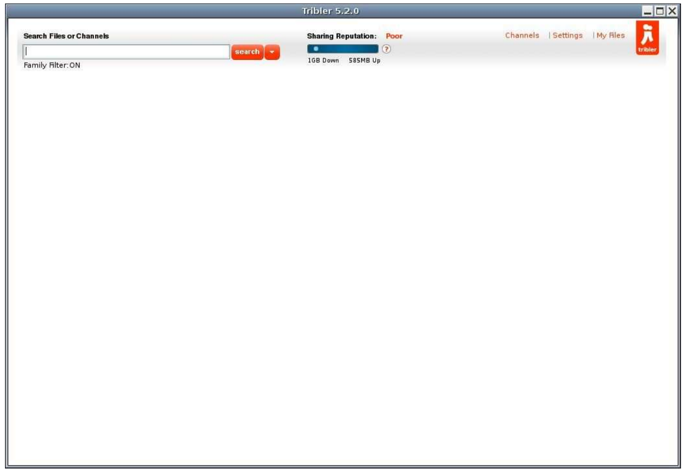
Version 1.0 of QMedia provided a major redesign of Tribler's user interface. This redesign was meant to better address QMedia's requirements of ease to use as well as its intended audience. QMedia 2.0 evolved and extended the graphical user interface (GUI) of QMedia 1.0 into a more efficient, user-friendly and feature-rich interface.

2.1 Issues identified with QMedia version 1.0's GUI

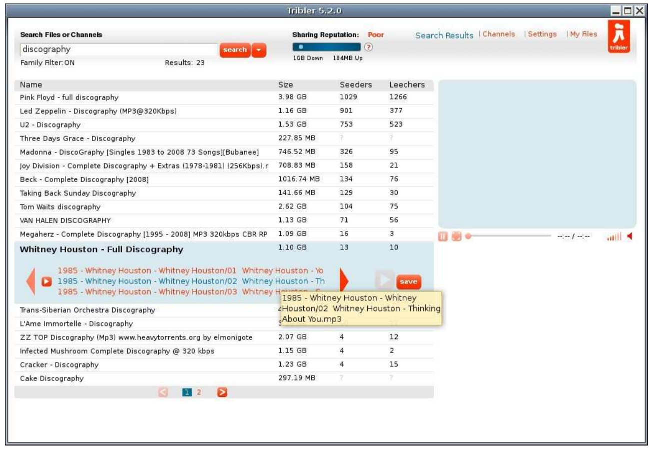
The overall approach used in QMedia platform v1, was to make the interface as simple as possible, relying on the search engine metaphor. The resulting interface, unarguably clean and simple, is shown in Figure 2.1(a).

Actual experience with usage of this GUI, however, has shown that it has a number of shortcomings. Feedback from users points to the following set of problems with this user interface:

- The 'Sharing Reputation' control as shown right from the search box, is confusing. Most users do not understand what this represents.
- Custom controls were used instead of the native ones included in users' operating system. However, although custom controls allow for a distinctive GUI, it also causes confusion to many users. Users reported, for example, not realizing that the 'links' on the top-right side of the user interface are ac-



(a) Start page



discography
Family Filter:ON Results: 23

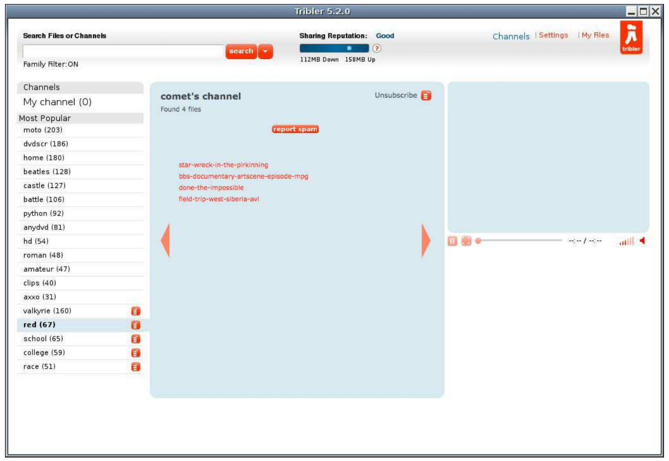
Name	Size	Seeders	Leechers
Pink Floyd - full discography	3.98 GB	1029	1266
Led Zeppelin - Discography (MP3@320Kbps)	1.16 GB	901	377
U2 - Discography	1.53 GB	753	523
Three Days Grace - Discography	227.85 MB	?	?
Madonna - DiscoGraphy [Singles 1983 to 2008 73 Songs][Bubanee]	746.52 MB	326	95
Joy Division - Complete Discography + Extras (1978-1981) (256Kbps).r	708.83 MB	158	21
Beck - Complete Discography [2008]	1016.74 MB	134	76
Taking Back Sunday Discography	141.66 MB	129	30
Tom Waits discography	2.62 GB	104	75
VAN HALEN DISCOGRAPHY	1.13 GB	71	56
Megaherz - Complete Discography [1995 - 2008] MP3 320kbps CBR RP	1.09 GB	16	3
Whitney Houston - Full Discography	1.10 GB	13	10
Trans-Siberian Orchestra Discography	[illegible]	[illegible]	[illegible]
L'Ame Immortelle - Discography	[illegible]	[illegible]	[illegible]
ZZ TOP Discography (Mp3) www.heavytorrents.org by elmonigote	2.07 GB	4	12
Infected Mushroom Complete Discography @ 320 kbps	1.15 GB	4	2
Cracker - Discography	1.23 GB	4	15
Cake Discography	297.19 MB	?	?

1985 - Whitney Houston - Whitney Houston/01 Whitney Houston - Yo
1985 - Whitney Houston - Whitney Houston/02 Whitney Houston - Th
1985 - Whitney Houston - Whitney Houston/03 Whitney H

1985 - Whitney Houston - Whitney Houston/02 Whitney Houston - Thinking About You.mp3

1 2

(b) Torrent details

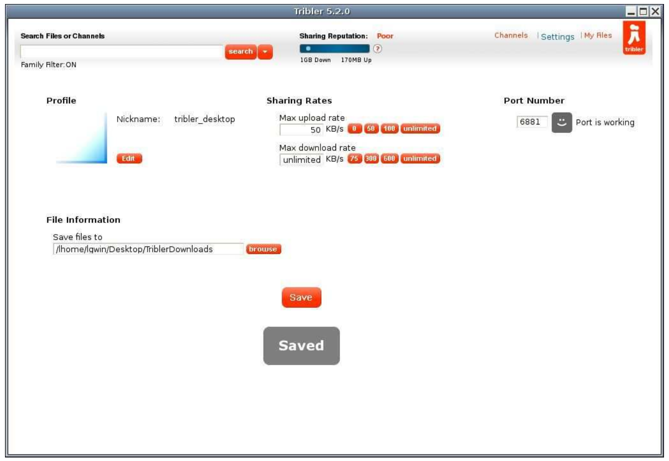


Channels
My channel (0)
Most Popular
moto (203)
dvdscr (186)
home (180)
beatles (128)
castle (127)
battle (106)
python (92)
anydvd (81)
hd (54)
roman (48)
amateur (47)
clips (40)
axxo (31)
valkyrie (160)
red (67)
school (65)
college (59)
race (51)

comet's channel
Found 4 files

star-wreck-in-the-pirkinning
bbs-documentary-artscene-episode-mpg
done-the-impossible
field-trip-west-siberia-avi

(c) Popular channels



Profile
Nickname: tribler_desktop

Sharing Rates
Max upload rate 50 KB/s
Max download rate unlimited KB/s

Port Number
6881 Port is working

File Information
Save files to
/home/lqwin/Desktop/TriblerDownloads

(d) Settings page

Figure 2.1: Overview of QMedia version 1.0 graphical user interface.

tually buttons. Similarly, users reported difficulty with the custom controls used to replace the search results list, file list and all buttons.

- Instead of re-using a single list, the channel view (as shown in Figure 2.1(c)) used a different list control. This required users to learn one unnecessary variant of another control.

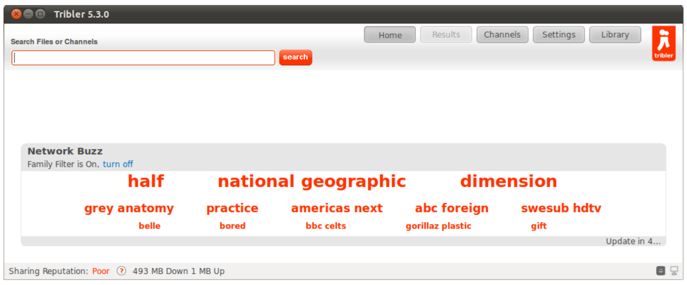
- The settings page as shown in Figure 2.1(d), did not show any structure. Different types of settings are shown together.

2.2 Improvements for QMedia version 2.0's GUI

The issues identified with QMedia version 1.0's GUI demanded significant changes both in the design and code of the GUI. Porting controls to use native components from different operating systems and to rely on a minimum common set of components required a rewrite of roughly 90% of all GUI-related code. An overview of the result of all modifications to the GUI is shown in Figure 2.2.

After this redesign, the start page now has a more prominent role in the interface. Users can now discover new content using the 'Network Buzz' control shown in Figure 2.2(a). The Network Buzz helps a user to understand what content is actually available in the network. Figure 2.2(a) also shows the buttons (top right corner), which replace the 'links' used in QMedia platform v1. The buttons are more easily identifiable, thus should ease navigation. The last change in the start page is the redesign of the 'Sharing Reputation' control, which is now shown in a status bar located at the bottom of the screen. This status bar additionally shows the connectability and network activity.

Figure 2.2(b) shows a search in progress. QMedia version 2.0 also highlights in yellow every new result found after the initial set that is first displayed. This helps a user to identify new items, which could otherwise be missed. Another new feature is that searching for files and channels is now combined. If a channel is found matching the search query, this is shown in the bottom of the table. We also opted for the removal of the seeders and leechers columns in the search result list. These have been replace by a popularity measure, which is represented by a colored and variably-sized bar – a greener, longer bar indicates a more popular, and hence healthier, swarm.



Search Files or Channels

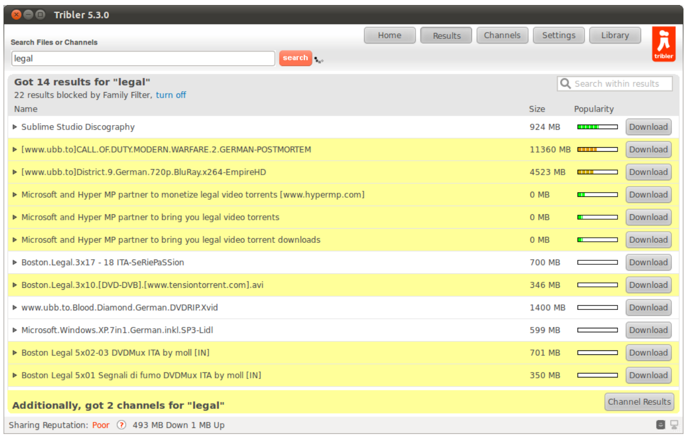
Network Buzz
Family Filter is On, turn off

half national geographic dimension
grey anatomy practice americas next abc foreign swesub hdtv
belle bored bbc celts gorillaz plastic gift

Update in 4...

Sharing Reputation: Poor ? 493 MB Down 1 MB Up

(a) Start page



Search Files or Channels
legal

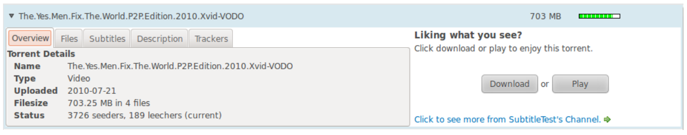
Got 14 results for "legal"
22 results blocked by Family Filter, turn off

Name	Size	Popularity	
Sublime Studio Discography	924 MB		Download
[www.ubb.to]CALL.OF.DUTY.MODERN.WARFARE.2.GERMAN-POSTMORTEM	11360 MB		Download
[www.ubb.to]District.9.German.720p.BluRay.x264-EmpireHD	4523 MB		Download
Microsoft and Hyper MP partner to monetize legal video torrents [www.hypermp.com]	0 MB		Download
Microsoft and Hyper MP partner to bring you legal video torrents	0 MB		Download
Microsoft and Hyper MP partner to bring you legal video torrent downloads	0 MB		Download
Boston.Legal.3x17 - 18 ITA-SeRiePaSSion	700 MB		Download
Boston.Legal.3x10.[DVD-DVB].[www.tensiontorrent.com].avi	346 MB		Download
www.ubb.to.Blood.Diamond.German.DVDRIP.Xvid	1400 MB		Download
Microsoft.Windows.XP.7in1.German.inkl.SP3-Lidl	599 MB		Download
Boston Legal 5x02-03 DVDMux ITA by moll [IN]	701 MB		Download
Boston Legal 5x01 Segnali di fumo DVDMux ITA by moll [IN]	350 MB		Download

Additionally, got 2 channels for "legal"

Sharing Reputation: Poor ? 493 MB Down 1 MB Up

(b) Search results



The.Yes.Men.Fix.The.World.P2P.Edition.2010.Xvid-VODO 703 MB

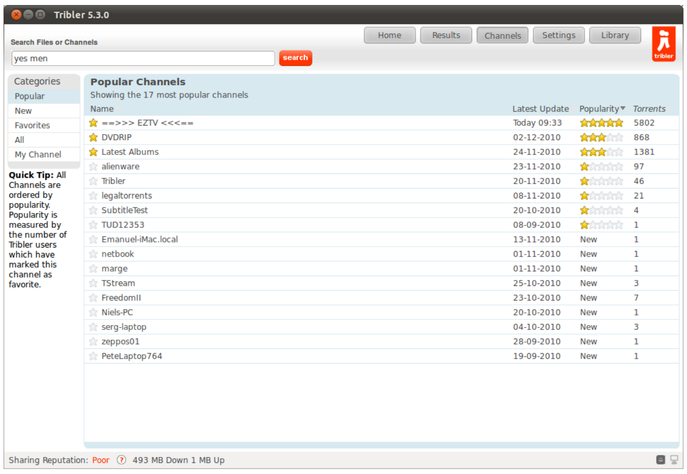
Torrent Details

Name	The.Yes.Men.Fix.The.World.P2P.Edition.2010.Xvid-VODO
Type	Video
Uploaded	2010-07-21
Filesize	703.25 MB in 4 files
Status	3726 seeders, 189 leechers (current)

Liking what you see?
Click download or play to enjoy this torrent.

Click to see more from SubtitleTest's Channel.

(c) Torrent details



Search Files or Channels
yes men

Quick Tip: All Channels are ordered by popularity. Popularity is measured by the number of Tribler users which have marked this channel as favorite.

Popular Channels
Showing the 17 most popular channels

Name	Latest Update	Popularity	*Torrents*
==>>> EZTV <<<==	Today 09:33	★★★★★	5802
DVDRIP	02-12-2010	★★★☆☆	868
Latest Albums	24-11-2010	★★★☆☆	1381
alienware	23-11-2010	★☆☆☆☆	97
Tribler	20-11-2010	★☆☆☆☆	46
legaltorrents	08-11-2010	★☆☆☆☆	21
SubtitleTest	20-10-2010	★☆☆☆☆	4
TUD12353	08-09-2010	★☆☆☆☆	1
Emanuel-iMac.local	13-11-2010	New	1
netbook	01-11-2010	New	1
marge	01-11-2010	New	1
TStream	25-10-2010	New	3
FreedomII	23-10-2010	New	7
Niels-PC	20-10-2010	New	1
serg-laptop	04-10-2010	New	3
zeppos01	28-09-2010	New	1
PeteLaptop764	19-09-2010	New	1

Sharing Reputation: Poor ? 493 MB Down 1 MB Up

(d) Popular channels

Figure 2.2: QMedia version 2.0's graphical user interface.

Expanding a torrent from any list (either in the search results, channel results or library) leads to the new torrent-details panel as shown in Figure 2.2(c). This panel hides most information from the non-advanced user using a tabbed control, while allowing an expert user to view all details of the torrent. Additionally, the subtitle tab allows a user to select a subtitle to be used in our integrated player.

Figure 2.2(d) shows the new channels panel. This panel now uses the same list control as the search results, but with different columns. On the left, there are different categories allowing users to filter the available channels. On the right, we display the actual contents for each category. For each channel, the name, latest update, popularity (the number of users who have marked it as one of their favorites), and number of torrent is shown. Clicking on any channel refreshes the list and shows the channel contents.

In version 2.0 of Qmedia, we opted to change the action of "subscribing to a channel" to "marking it as your favorite". After consulting with an user experience expert, subscribing was changed because it would suggest a user that he/she would start downloading all content in this channel: when you are subscribed to a magazine, you get the latest issues delivered to you. Marking as favorite with the star icon was then chosen as this is a well-known combination. This combination is used by browsers and gmail, for example. A "mark as spam" button is still provided in the interface.

A beta version of Qmedia version 2.0 was released to the general public on December 8th 2010. Since then, we've observed a peak in both new installs and sustained usage of the system. As shown in Figure 2.3, the redesign of the GUI caused a sizeable increase in both the number of channels and the number of users marking one as their favorite.

Overall, we've received positive feedback to the Qmedia version 2.0 release. We are currently discussing with users further improvements. Several users have suggested new features and are reporting found bugs in our online forum. We have also integrated a customer feedback page on our Website, where users can suggest features and changes to the software. We expect that opening as many communication channels as possible with our users should result in a higher quality platform with higher loyalty in its user base.

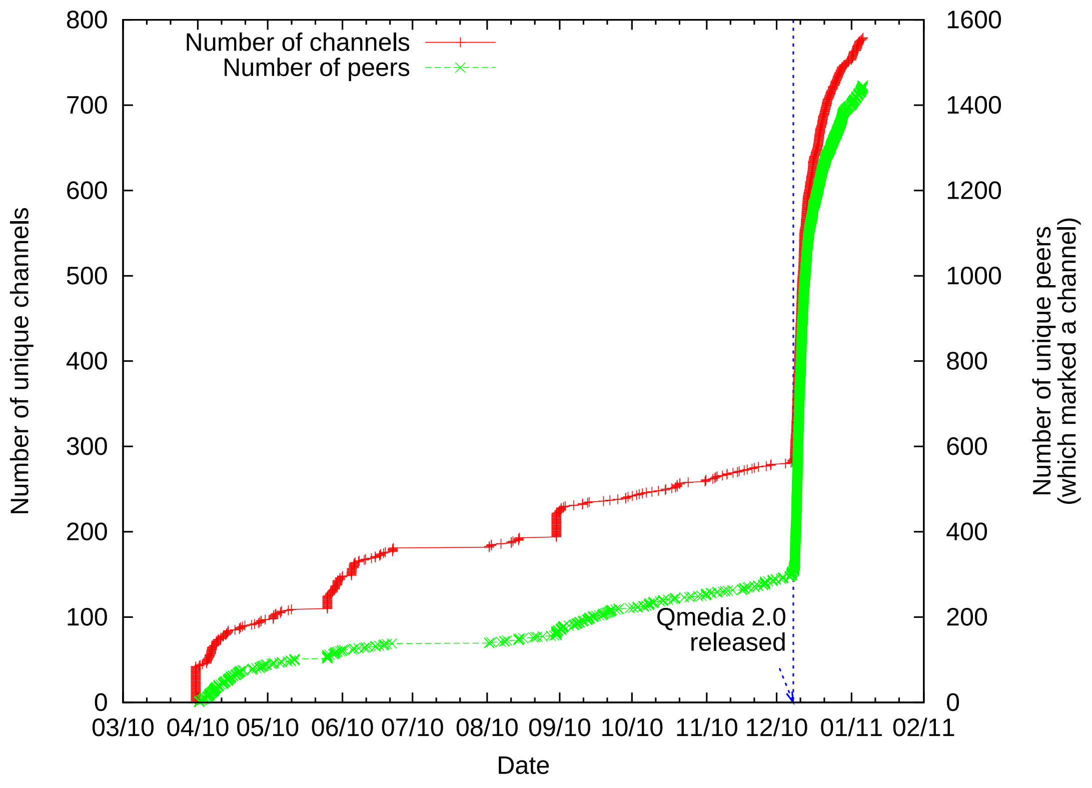


Figure 2.3: Overall popularity of the Channels feature in QMedia before and after the release of version 2.0.

Chapter 3

Bartercast version 2.0

Bartercast is QMedia's users' reputation mechanism. It aims at enabling the promotion of cooperation through indirect reciprocity.

Research on the coverage and accuracy of Bartercast as deployed in QMedia version 1.0 pointed to the need of improving its information dissemination system (see QLectives Deliverable 4.2.2 for a description of the methodology and results of this research). In summary, it was found necessary that for QMedia version 2.0, Bartercast had more aggressive data dissemination, and that peers in the network should disseminate third-party information.

These requirements, in turn, called for major changes in Bartercast code. The need for such changes was both a motivation for and the most immediate benefiter from the development Dispersy, the distributed permission module implemented in QLectives Platform version 2.0 and documented in the QLectives Deliverable 4.1.2. Dispersy was implemented with the new requirements of Bartercast in mind, and, in turn, Bartercast for QMedia 2.0, called Bartercast 2.0, was implemented on top of Dispersy.

As a result, Bartercast 2.0 is, in the lingo of the Dispersy Framework, a Dispersy Community, called the Barter Community.. The purpose of this community is to distribute enough information to allow nodes to determine the reputation of other nodes in the system.

3.1 Overview

The implementation of Bartercast in QMedia version 1.0 is built on top of Buddy-Cast (see QLectives Deliverable 4.3.1 for a description of BuddyCast). For QMedia version 2.0, we have used solely Dispersy as the dissemination platform and re-implemented Bartercast with important extensions and enhancements. The new Bartercast, referred henceforth as Barter Community, is designed to be secure and scalable, taking advantage of the flexible policies provided by Dispersy.

Communities in Dispersy must have at least a master member, which identify the community itself, and a message. The Barter community, being an open community that includes all QMedia's nodes, has a publicly known master member. All other members in the community, which are effectively all nodes in the QMedia network, are allowed to send the sole message that characterizes the Barter community. This message is named the *barter-record*, and is described in the following section.

3.2 The *barter-record* message

The barter community uses the *barter-record* message to distribute the data exchange activity between pairs of nodes. Each *barter-record* message contains one record about a pair. Each message contains two integers *amount-from* and *amount-to* which represent in megabytes the size of the exchanged data between the two nodes that have signed the message.

The message is attached to several policies: the AuthenticationPolicy, the DestinationPolicy and the DistributionPolicy and it gets defined by the following code:

```
def get_meta_messages(self):
    return [Message(self, u"barter-record",
                          MultiMemberAuthentication(2, self.on_signature_request),
                          PublicResolution(),
                          LastSyncDistribution(0, 10),
                          SimilarityDestination(1, 512, 205, 307, 384)
                          BarterRecordPayload())]
```

The policies and the parameters used in the code are discussed in the following.

The MultiMemberAuthentication Policy The barter-record message uses the MultimemberAuthentication policy defined in the Dispersy core. This policy requires a message to be signed by two or more nodes in a community before it is distributed to other nodes. Since a barter-record contains information about the activity between a pair of nodes, it has to be signed by both nodes to ensure that its contents are correct. Hence the MultiMemberAuthentication policy is used with the parameter of 2.

The LastSyncDistribution Policy The LastSyncDistribution policy is designed to limit the number of messages per node that Dispersy distributes at a given time. The Barter Community initializes this policy with the parameter 10, therefore the last 10 messages that every node generates are distributed at a given time.

To clarify, a node can collect into its barter community database more than 10 barter records generated by one other node, during its lifetime. However Dispersy will distribute at a given time only the last 10 records for each node.

The SimilarityDestination Policy To ensure that the barter community can scale to a large number of nodes the SimilarityDestination Policy is used to provide semantic clustering. This policy allows each node to define its interest in the community and receive only records that originate from nodes whose similarity match is above a specified threshold. Using this mechanism, the number of times a record is transferred is reduced and the total number of records stored into each node's database is limited.

The similarity is defined as a bloom filter. The current implementation is still not making use of the full potential of this feature, and populates this bloom filter randomly. For the near future, we envision populating the bloom filter with the following:

1. keys of items for which the application received positive feedback from the user;
2. system-specific keys that indicate the position of the user in the system, such as the first 2 or 3 bytes of his IP address, his country, his time zone, etc.;

3. a number of semi-random bits that regulate the size of similarity-formed groups in the system.

The SimilarityDestination for the Barter Community is initialized with a bloom filter of 512 bits with minimum fill of 205 bits (40%), maximum fill of 307 bits (60%) and similarity threshold 384 bits (75%).

Each barter record message that is created by Alice will be distributed to anyone within her sphere of influence, based on the similarity of Alice and the other nodes. If Bob is also within the sphere of influence of Alice, then Bob will assist in the propagation of Alice's records to other nodes within the sphere of influence of Alice. Note that Bob will not forward Alice's records outside of Alice's sphere of influence.

3.3 Generation of barter-records

We refer to an example to explain the generation of the barter-records:

As Alice and Bob interact, 10 MB is sent by Alice and 2 MB is sent by Bob. During these transactions, the most interested node, Alice, keeps requesting that Bob signs barter-records that contain the amount of MB that they have sent and received so far. If Bob does not agree with the values proposed by Alice and therefore does not return a signed record, Alice stops sending data.

Once a transaction is over and the final double-signed barter-record is owned by Alice, she can announce her contribution to the rest the community by distributing the record. Schematically we get the following transactions:

Alice	- transactions -	Bob
	– 10 MB –>	
	<– 2 MB —	
	...	
Sign Request	— R(Pa(A,B,7,1)) –>	
	<– R(Pb(Pa(A,B,7,1))) —	Sign Reply
	...	
Sign Request	— R(Pa(A,B,10,2)) –>	
	<– R(Pb(Pa(A,B,10,2))) —	Sign Reply

Continued on next page

Alice	- transactions -	Bob
	…	
Record	Pb(Pa(A,B,10,2))	

3.4 Experiments

To test the performance of Dispersy we conducted a number of experiments with the Barter Community. Our goal is to evaluate the dissemination performance of Dispersy. For these experiments we used the debugging mode of Dispersy, the Dispersy Scripts, along with a precalculated network activity scenario of a BitTorrent swarm.

The scenario describes the availability and the traffic exchange between 10 peers in different time steps. Each peer takes as input two files, the *bartercast.log* and the *availability.log*. The first file describes the traffic exchange transactions a peer had with other peers during its lifetime and it has the following format:

Timestep	Peer	Up	Down
1	2	10	20
2	3	1	3
3	2	12	22
…	…	…	…
10	6	7	10

The second file describes the availability of the peers during the experiment and has the following format:

Timestamp	Command
1	start
3	stop
…	…
5	start
10	stop

When a peer is *started* it is able to send and receive messages, whereas when it's *stopped* it can neither send or receive messages. Using this technique we can simulate the network churn.

Using a precalculated scenario we reduce the randomness in the experiments, hence we have a more controlled environment. We are able to repeat the exact same experiments multiple times and observe the differences between differences configurations, such as different synchronization intervals for Dispersy or churn rate of the network. The scenario was calculated using the ongoing work of Mircea Barcac on a sophisticated generator for BitTorrent scenarios.

The experiment is executed as an emulation, hence multiple Dispersy instances are executed and bind to network interfaces creating a real network. For this propose we developed scripts for the automated deployment of peers in multiple machines. The scripts ensure that a peer gets created and bound to a specific host and port combination. Every peer will execute its part of the scenario according to its configuration.

The current scenario consists of only ten peers, generating a total of 562 records over 450 seconds. Due to the size of the network there is no need to enable the SimilarityDestination policy, therefore just for this experiment we replace the SimilarityDestination policy with the CommunityDestination policy, so that every peers gets a copy of each and every one of the barter-records created. The CommunityDestination policy is initialized with *peer_count=0*, to disallow the initial spreading of messages during the creation time which is done separately to the synchronization mechanism. The other policies of the barter-records message are as defined in the first section. The dispersy-sync message is dispatched every 20 seconds with destination one of the other peers.

The results of the experiments are summarized in Figure 3.1. We observe that the records are spreading around the network. The synchronization mechanism that uses bloom filters to reduce the sending of duplicate information is performing well, reducing the number of dropped messages (i.e. messages that already exist in our database). The send and received traffic are always increasing since each peer sends out a dispersy-sync messages along with bloom filters, which other peers receive.

It is important to note that peer coverage does not converge to the total number of records available. This is due to the LastSyncDistribution policy which is initialized with the number 10, hence only the 10 last messages can be disseminated for each peer at a given time in the network. If a peer creates more than

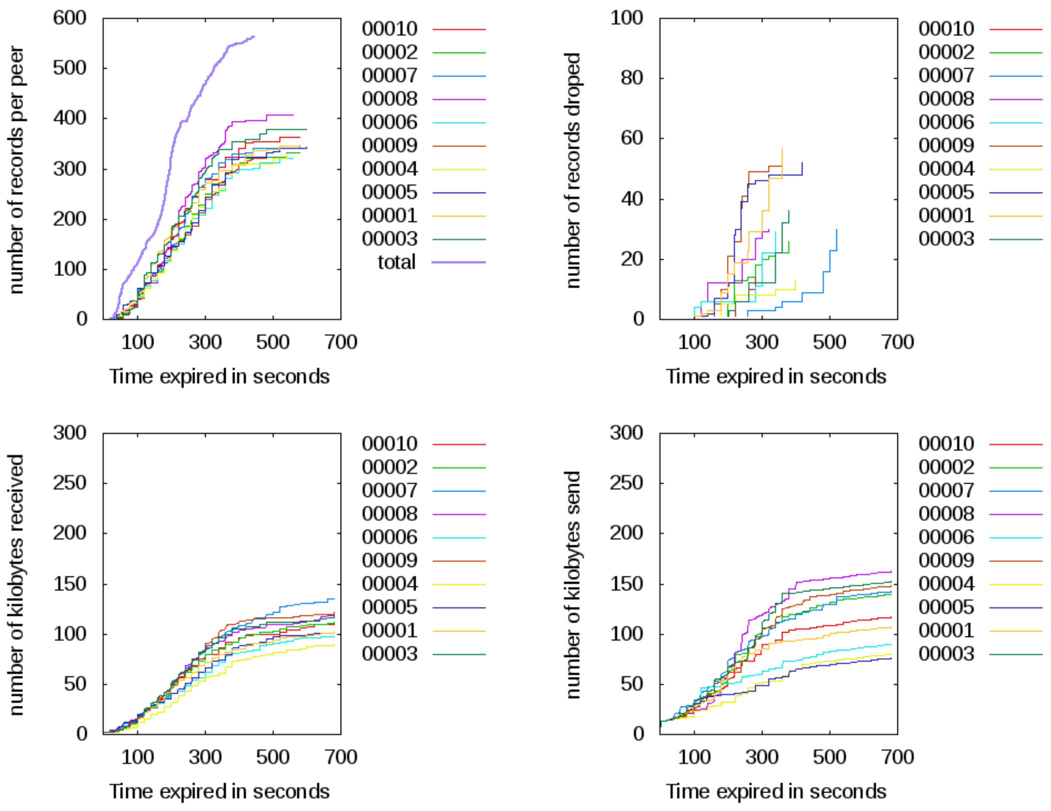


Figure 3.1: Results of emulation of 10 peers synchronizing every 20 seconds.

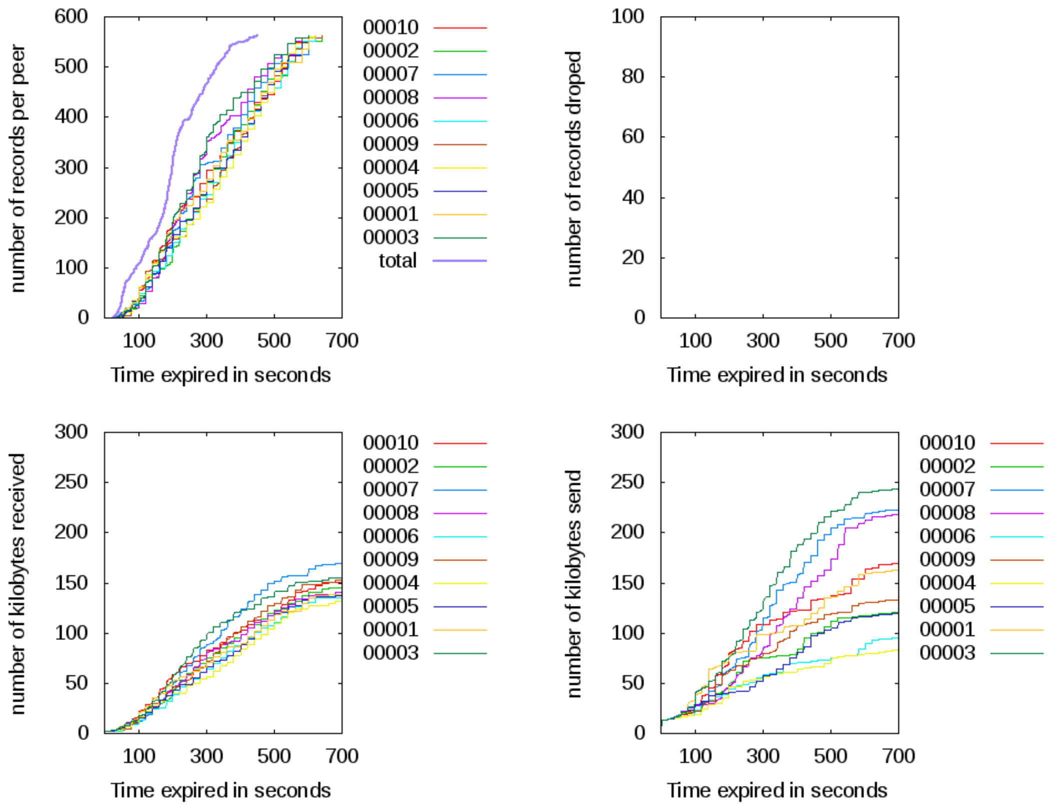


Figure 3.2: Results of emulation of 10 peers synchronizing every 20 seconds with LastSyncDistribution of 1000.

10 messages in a time span of 20 seconds, which is the configured interval of the dispersy-sync messages then some of the messages created will not be disseminated at all.

To verify the claim above, we executed the exact same experiment with LastSyncDistribution policy initialized with the number 1000. Since the total number of messages for this experiment is limited to 562, setting the limit of the LastSyncDistribution policy to 1000 guarantees that all the messages will be disseminated during the whole life of the experiment. Figure 3.2 confirms this assumption.

Since all messages are available, peers eventually converge to the total number of records available. From Figure 3.2(a), it is clear that peers increase the total number of records they possess about every 20 seconds which is the interval of the dispersy-sync message.

Due to the more frequent synchronization all peers follow more closely the blue line which represents the total number of records available in the network.

However more bandwidth is consumed for each peer due to the larger number of synchronizations.

The set of the experiments we conducted visualize the working performance of Dispersy. It shows that the parameters of each community, such as the Last-SyncDistribution policy initialization, must be specifically configured for community, depending on its needs (i.e. the number of records created, the required speed of dissemination and others) and that one configuration cannot fit all needs.

Chapter 4

Summary and Further Research Questions

This report documents the second release of QMedia, a media-sharing P2P system built on the QLectives Platform. Furthermore, this report describes the contributions of the QLectives project to the common code base that constitutes QMedia, Tribler and the QLectives Platform. The current version of the software is ready for deployment so as to serve as the QMedia Living Lab.

A preliminary version of this release, containing the GUI improvements but not yet Bartercast 2.0 was released in early December 2010. There were approximately 15,000 downloads of the software in the two first weeks after its release. This number is particularly promising from the perspective of the Living Labs' goals. The public release of QMedia version 2.0 with all the improvements described in this report is planned for the first quarter of 2011.

Bibliography

[1] P2P-Next. http://www.p2p-next.org/.

[2] PetaMedia. http://www.petamedia.eu/.