# A Massively Scalable Persistent Content Distribution System

Jani Peltotalo, Sami Peltotalo, Alex Jantunen, Lassi Väätämöinen, Jarmo Harju
Tampere University of Technology, Institute of Communications Engineering
P.O.Box 553, 33101 Tampere, Finland
{jani.peltotalo, sami.peltotalo, alex.jantunen, lassi.vaatamoinen, jarmo.harju}@tut.fi

Rami Lehtonen
TeliaSonera Ltd.
Hatanpään valtatie 18, 33100 Tampere, Finland
rami.lehtonen@teliasonera.com

Rod Walsh
Nokia Research Center
Visiokatu 1, 33720 Tampere, Finland
rod.walsh@nokia.com

## Abstract

This paper proposes a novel form of peercasting system as an improved solution for IP-based mass media content delivery. Proposal aims to improve existing homogenous and heterogeneous systems for mass media distribution to very large user bases, with needs for timely and reliable delivery, and content persistence. Thus IP multicast has great advantages for delivery with controlled last-mile elements, both for mobile and fixed usage. However, persistence while the number of still-receiving users dwindle, and reliability at reasonable network cost, are better served by Peer-to-Peer techniques. The work primarily considers discrete media delivery, which is useful both along side streaming media and in standalone applications. The feasibility of such a combined multicast and P2P system is shown and the initial prototype implementation of this proposal, *Delco*, is introduced.

**Keywords:** Multicast, Peer-to-Peer, Peercasting, Mass media

## 1 Introduction

The distribution of digital content (music, videos, photos, games etc.) is increasingly becoming the primary task of the IP networks. Coupled with the problems of current networks related to security, capacity demands, heterogeneity of the networks and rights management issues have created a huge demand for advanced solutions in the digital content distribution space.

There are a lot of research activities and studies dealing with different content distribution technologies including multicast, IP Datacast (IPDC), Peer-to-Peer (P2P) and Content Delivery Networks (CDN). A few research groups have also studied how to integrate P2P and CDN technologies. However, all the above mentioned content distribution technologies have their own limitations; multicast is currently unutilised, CDN servers are expensive to deploy and maintain, and in current P2P systems there exist severe problems around security issues like Denial of Service (DoS) attacks and viruses.

In this paper we present a content distribution system, called Delco, which is a new class of peercasting system combining multicast and P2P techniques in a way that has not been previously attempted. Existing peercasting systems (for example PeerCast and TVAnts) have focused on P2P delivery of live video. Recently P2P for video sharing, such as Veoh or the Venice project, may open this up. However, the broadcast medium has provided the thickest pipe to deliver mass media content, and systems such as MBMS and DVB-H give promises to deliver this also to mobile users in the near future.

## 2 Delco Content Distribution System

The main motivation for our work is to create better enabling tools for content providers, service providers, network operators and end-users to help the distribution and use of digital content. The advantage of using multicast is to reduce server and the distribution network workload. Similarly the use of P2P helps to reduce server load, which is a weak point when using the traditional client-server model for mass media delivery.

Since multicast commonly uses an unreliable transport protocol (UDP - the User Datagram Protocol), reliability must be achieved by other means. Repeat transmissions and Forward Error Correction (FEC) are two options to achieve the reliability in the main forwarding path. For the return path there are multiple options such as P2P repair and traditional client-server repair. The latter could be for example a HTTP-based file repair scheme. However, if there are a lot of receivers that do not have a complete file after multicast transmission, there becomes a huge load on the HTTP server(s). The former enables ”receivers” to complete the multicast delivery from each other and promises much reduced server load. Figure 1 shows an overall architecture of a combined multicast and P2P content distribution system.

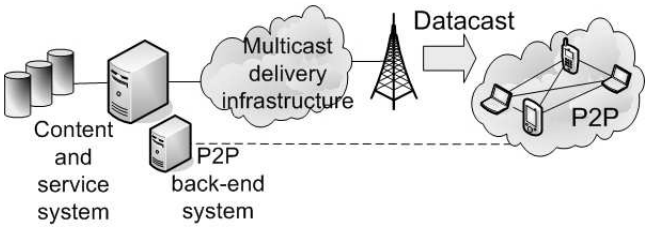


Figure 1: Combined multicast and P2P content distribution system

The main requirements for the development of our content distribution system were:

1. The content distribution system should cope with the flash crowd effect and scale to very large audiences.
2. The content distribution and Digital Rights Management (DRM) should be separated, so that the use of various technologies for the delivery as well as for the DRM is allowed.

3. The content distribution network should be resistant to attacks against users (in forms of viruses etc.), content modification and availability of content.

4. The content distribution network should be as independent as possible from the access networks and end-systems.

5. The content delivery mechanisms should be service independent and existing technologies and metadata descriptions should be used as much as possible.

The Delco system architecture is specified to support any multicast and peer-to-peer technique. For instance application layer multicast and Chord systems could be applied. However, we have implemented the system using File Delivery over Unidirectional Transport (FLUTE) [Paila et al. 2004] for the multicast delivery, and BitTorrent [BitTorrent 2003] for the P2P delivery. In the system a special metadata file (a Delco file) is used to describe the content of a service. A web server is used to make this metadata available to the customers, who can also make content searches via the web server.

The Delco system provides three different content delivery modes: multicast-only, combined multicast and P2P, and P2P-only. At the start of the mass media content release there are generally many customers requesting the same content, so a service provider can utilise multicast delivery (in multicast-only and combined multicast and P2P modes), which serves very large user groups without overloading server and network resources. In the multicast-only mode reliability may be improved with redundant data (repeated transmissions and FEC data). In the combined multicast and P2P mode clients whose access networks support multicast can start seeding to non-multicast clients after they have received some data through the multicast delivery. In this mode the service provider must have P2P seed(s) to backup the delivery. In the P2P-only mode a service provider must also have one or more P2P seeds to initiate and backup the delivery.

In the Delco system two different service types can be used: download and channel. In the former the exact content of a service is known before the service transmission starts, and in the latter it is not. The channel service may also be described as a semi-automatic download, where a user accepts the content delivery as for example in a software security update service and the content is automatically downloaded at the receiver.

Our current metadata file has file extension .delco and it uses XML syntax. A Delco file can describe multiple services, but the default is to describe just one. A service can include multiple content items, each having a unique content ID. A content pointer is used to map a content ID and delivery mechanism of the content. The following example Delco file describes a download service with two content items. The content is available via combined multicast and P2P delivery.

```
<?xml version="1.0" encoding="iso-8859-1"?>
<delco>
 <service id="1" name="Movie Package">
  <contentItem id="1" name="Movie 1">
   <contentPointer type="flute" value="http://www.example.com/sdp/movie1_flute.sdp"/>
   <contentPointer type="torrent" value="http://www.example.com/torrent/movie1.torrent"/>
  </contentItem>
  <contentItem id="2" name="Movie 2">
   <contentPointer type="flute" value="http://www.example.com/sdp/movie2_flute.sdp"/>
   <contentPointer type="torrent" value="http://www.example.com/torrent/movie2.torrent"/>
  </contentItem>
 </service>
</delco>
```

By blocking the content into pieces and using hash check for those it is possible to provide reliable simultaneous multicast and P2P delivery. With the help of piece hashes erroneous/missing pieces can be detected and repaired in a way that whole content can be eventually received. Content blocking can be smoothly used with FLUTE and BitTorrent because block sizes are adjustable in both delivery techniques. When the FLUTE's and the BitTorrent's block sizes are equal or the FLUTE's block size is a multiple of the BitTorrent's all data that has been received by the FLUTE can be utilised.

Functional Delco client consists of two parts: a user interface and a Delco library. The Delco library consists of three parts, a Delco client engine, a service data engine and a generic download component. The service data engine is used to construct service data objects from the Delco files. The Delco client engine uses service data objects to create own session for each service. A generic download component defines adapter interfaces for different downloading protocols (peer-to-peer, multicast and point-to-point). With the help of adapter interface it is possible to implement new downloading plug-ins, which maps the adapter to the existing library specific functions.

The Delco server system consists of a Delco specific control interface/data transfer system, a WWW server and a BitTorrent tracker. The control interface is for administrative use (service creation and content injection), the WWW server is used for service data distribution and the BitTorrent tracker is required for BitTorrent usage. Currently the control interface and the WWW server locate on same host for easier implementation, but it is also possible to distribute the functionality to different host entities. Current server system architecture is designed with a reusability in mind, so that the user interface and data transfer parts (multicast and P2P protocols) could be replaced independent of the current implementation.

## 3 Conclusions

This paper presents a mass media content distribution system based on IP multicast and P2P delivery techniques. The strengths of both IP multicast and P2P overlay have been leveraged to simultaneously scale server and distribution network capacities to serving greater number of receiving hosts for mass media content. The paper demonstrated the feasibility of such a system, even using existing software components which may be found among the open source community.

Several issues still remain unsolved or only partly solved. Laboratory and field testing of simultaneous multicast and P2P download with free transitions between P2P and multicast delivery, will highlight system bottlenecks and usability issues. Some security aspects, such as known DoS attacks against P2P networks (like poisoning peer list database of a BitTorrent tracker) are also worthy of further investigation. As is a detailed performance analysis of the Delco system that would yield a better understanding of how to optimise peercasting systems and also the scale of the benefit of implementing custom-designed multicast and P2P components an alternative to reusing existing generic code.

It is already evident that peercasting is an exciting and potentially fundamental step in the deployment of massively scalable mass media content distribution systems. It includes the promise of improved broadband, mobile datacast and triple/quad-play systems through better optimised underlying technology usage. Future work is expected to reveal to full extent of that improvement and its impact.

## References


BITTORRENT. BitTorrent protocol specification [online]. 2003 [cited 7 September 2006]. Available from: http://www.bittorrent.org/protocol.html.

PAILA, T., LUBY, M., LEHTONEN, R., ROCA, V., AND WALSH, R., 2004. FLUTE - File Delivery over Unidirectional Transport. RFC 3926 (Experimental), October.