A Massively Scalable Persistent Content Distribution System

The Sixth IASTED International Conference on
Communications, Internet, and Information Technology
~CIIT 2007~

July 2 – 4, 2007
Banff, Alberta, Canada
• Delivery and Control Techniques in Content Networking (DelCo) project
  • More information from our web site: http://delco.cs.tut.fi/
• Project members
  • Tampere University of Technology, Institute of Communications Engineering
  • Prof. Jarmo Harju
  • Jani Peltotalo
  • Sami Peltotalo
  • Alex Jantunen
  • Lassi Väätämöinen
  • TeliaSonera
    • Rami Lehtonen
  • Nokia Research Center
    • Rod Walsh
Motivation

- The distribution of digital content is increasingly becoming the primary task of the IP networks
  - This has led to intensive research activities concentrating on multicast, IP Datacast (IPDC), Content Distribution Networks (CDN) and peer-to-peer (P2P) overlay 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
  - This demand is even extending during the next few years, because of the advances in media consumption over the Internet and increased quality of the media files
- Hence, there is a need and possibility for a trusted and controlled mass media delivery system with more optimal use of the network capacity
Combining multicast and peer-to-peer delivery techniques

- We present a content distribution system, called Delco, which combines IP multicast and peer-to-peer overlay network techniques in a way that has not been previously attempted.
  - Simultaneous multicast and peer-to-peer delivery (multicast-only and P2P-only delivery also possible).
  - The primary advantage of using IP multicast is to reduce the workload of servers and the distribution network elements.
  - Reliability is achieved by:
    - Repeat transmissions and forward error correction in the forward multicast path.
    - Unicast repair over a P2P network:
      - Enables receivers to complete the multicast delivery from each other and promises much reduced server load.

![Diagram of content distribution system](image-url)
Requirements

• The content distribution network should be as independent as possible from the access networks and end systems
• The content distribution mechanisms should be service independent and existing technologies and metadata descriptions should be used as much as possible
• The content distribution system should cope with the flash-crowd effect (sudden popularity of a single file) and scale to very large audiences
• The content distribution network should be resistant to attacks against users (in forms of viruses etc.), content modification and availability of content
The system architecture is specified to support any multicast and peer-to-peer technique

- FLUTE and BitTorrent in our implementation
- Aligned content segmentation used for optimal network traffic

Service types

- Download
  - The exact content is known before the service transmission starts
  - An example content of a download service is a movie or a movie package (several movies)
- Channel
  - A semi-automatic download, where a user accepts the content delivery and the content is automatically downloaded at the receiver
  - A channel service could be, for example, a football channel where news and game clips are delivered after they have been produced

An XML metadata file (file extension .delco) is used to define a content service and its delivery mechanisms

- SDP file defines multicast delivery session
- Torrent file is used for P2P delivery
- RSS feeds are used for channel services
- Includes also information for reporting
At the start of a mass media content release, there are generally many customers requesting the same content. Hence a service provider can utilise IP multicast/datacast delivery which serves very large user groups without overloading server and network resources.
The DelCo content distribution system provides:
- reliability at reasonable network cost
  - receivers repair missing blocks from each other using P2P during the multicast transmission period or afterwards
- persistence via P2P techniques
  - operator’s back-end seeds will assist if needed
- reduced server workload.
Architecture of the server implementation ...
... Architecture of the server implementation

- **Server Application entity**
  - Service metadata management and database insertion/retrieval
  - Generates metadata files ‘automatically’ (SDP, .torrent and .delco files)
  - Metadata file delivery from local file system using web server
    - Starts, manages and ends data transfer
      - Multicast and back-end P2P delivery
- **Service entity**
  - Fetches active services from database and shows those to client
  - Receives and handles reports from clients
- **Tracker**
  - BitTorrent usage
Server user environment ...
... Server user environment

- Local web server and BitTorrent tracker are started and managed as separate processes
- Service management is done using a Server Application entity
- Web browser for service report monitoring
  - HTTP request from the web browser is sent to a report script which makes database query and comprises a report page
  - Web-based implementation enables remote monitoring
Architecture of the Server Application entity …
... Architecture of the Server Application entity

- Two main parts:
  1. DataEngine
     - Service metadata generation and management
     - Composition of service metadata files
  2. ServerEngine
     - Service start up and run-time management
- In addition database access using DatabaseAPI
  - Service metadata database insertion (retrieval in server start up)
  - Server-end reporting
Architecture of the client implementation

- User interface is operation system dependent component which can be changed if needed without modifications to other components
  - Implemented using WxWidget
- Delco Middleware manages services in general level
- GenericDownload offers generic interface to retrieve data using different download components (Multicast, P2P, point-to-point)
  - Only GenericDownload is dependent of the used download components and program libraries
  - It is possible to change or add new download components without modifications to Delco Middleware
- Boost is used for threading and file system operations for portability
Client user environment

- Services are browsed and selected using a web browser.
- Service metadata (.delco file) is started into a client application which joins FLUTE and/or BitTorrent sessions defined in the metadata.
- User can monitor and manage active services.
- Content is consumed on the fly using file system (stream service) or when whole data is downloaded.
- With some media players it is also possible to use socket to deliver data to the player on the fly.
Detailed architecture of the client implementation...
Detailed architecture of the client implementation

- DownloadManager manages one service which can contain several Item_sessions (one session per file)
- Item_session downloads data pieces using generic download interfaces (Multicast, Peer2Peer)
- PieceManager is aware of downloaded data pieces per each file

- It is possible to register new download components to GenericDownload on the fly

- Download components are using call back mechanism whenever writing or reading data from hard disk
  - Call back can be caught in the Delco Middleware and needed actions can be done
  - PieceManager gets information about progress of the download
  - By monitoring user can control different components in a controlled way
Conclusions

- We have presented a mass media content distribution system based on IP multicast and P2P delivery techniques.
- We have 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.
  - Large scale laboratory and field testing of simultaneous multicast and P2P download would 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.
  - A detailed performance analysis of the Delco system would yield a better understanding of how to optimize peercasting systems.
  - The scale of the benefit of implementing custom designed multicast and P2P components as an alternative to reusing existing generic code.
Thank You!

Questions?