Grid Video Processing

Distributed Approach to Video Processing

Petr Holub
hopet@ics.muni.cz

CESNET z. s. p. o.

Laboratory of Advanced Networking Technologies
Institute of Computer Science and Faculty of Informatics
Masaryk University of Brno

Internet2 Fall Member Meeting 2005, Philadelphia, 2005–09–20
Outline

Motivation

Distributed Encoding Environment

Performance Evaluation

Future Work

Real-World™ Usage Example
Motivation

- Huge production of video content (e.g. lecture archives and live streams from lecture halls).
- Problems with centralized storage capacity (price/capacity/performance).
- Use existing computing and storage infrastructures.
- Create scalable tool from financial point of view.
Distributed Encoding Environment

- Uses PC clusters of MetaCenter project as computing environment.
- IBP storage infrastructure of DiDaS project as transient storage for processing.
Internet Backplane Protocol

- Distributed storage with soft-state/best-effort characteristics.
- Storage servers are called IBP depots.
  - Time-limited allocations.
  - Volatile allocations.
  - End2end services: encryption, checksums, compression.
- Support for redundancy.
  - Both performance and reliability.
- libxio library developed at Masaryk University/CESNET
  - Implements POSIX calls for both local files and IBP stored files.
  - Easy enabling of IBP in existing applications.
DEE Workflow

Outline
Motivation
Distributed Encoding Environment
Future Work
Real-World Usage Example

DEE Workflow Diagram:

- **DV upload** from editing computer
- **DV download** to IBP
- **DV chunks upload**
- **DV chunks download**
- **DV remux**
- **DV split**
- **many nodes** transcoding to raw AVI, encoding to RealMedia
- **single node** joining RM chunks
- **RM upload**
- **RM chunks upload**
- **RM chunks download**
- **RM upload**
- **RM chunks removal**
• Native IBP LoRS tools.
  • Command-line tools for Windows, Linux and MacOS, GUI for Windows.
  • Problems with files >2 GB :-(
• javalors tool
  • Uploading and downloading using several threads in parallel and GUI.
  • Written in Java, works on Windows, Linux (maybe even MacOS X).
• Sound/video re-multiplexing.
• Creating video chunks for parallel encoding.
- IBP-enabled transcode
- Image transformation by transcode
  - image reduction, de-noising, deinterlacing, color corrections, audio resampling
- Transcoding to target format, or
- transcoding to raw video and then encoding using external encoder.
  - RealProducer for Linux 9.x
• Permanently stored on IBP
  • Playing with IBP-enabled client tools or via implementation of IBP as local file-system.
  • Playing through IBP-enabled Apache HTTP.
• Uploading to streaming and downloading servers local storage.
Selected Properties of the DEE

<computersciencetheory>

- Scheduling model $\in PO$ class.
  - Classical model $\in NPO$ class when scheduling different jobs on uniform processors.
  - DEE uses uniform jobs on different processors, which belongs to $PO$.
  - Data depot scheduling can be made as $PO$ for network described as complete graph.
- Data location optimization with respect to computing infrastructure.
</computersciencetheory>
Performance Evaluation

- DV to RealMedia with/without remultiplexing
- Source: 6911 PAL DV frames (4:36s), 48 kHz audio
- Target: deinterlaced (HQ cubic blend), $384 \times 288$ (Lanczos), 44 kHz audio, bitrates: 28, 56, 128, 256, 384, 512, 768 kbps
- Testbed
  - $2 \times$ Pentium 4 @ 3.0 GHz, 2 GB + 2 GB swap, Intel PRO/1000 1 Gbps
  - DiDaS IBP infrastructure
DiDaS/MetaCenter Infrastructure Used for DEE
DEE Execution Profiles

DEE Execution Profile
Parallelism = 2
DEE Execution Profiles

DEE Execution Profile
Parallelism = 8
Parallelization Performance

DEE Parallelization Performance

\[ y = \frac{1}{0.0498x} + 3.05 \quad (RMS = 0.030) \]
\[ y = \frac{1}{0.0520x} + 2.52 \quad (RMS = 0.040) \]
Future Work/Work in Progress

- Integration with new grid middleware (schedulers).
- DEEng – we plan to reimplement DEE, new GUI, production quality code, etc.
Real-World™ Usage Example

- Lecture recording, processing, archiving and publishing at Masaryk University in Brno.
- Motivation
  - Continuous video feed from 3 lecturing halls Faculty of Informatics.
  - More to come, esp. from other faculties.
System Architecture

Cameras

Canopus ADVC-100 convertors

Caching PCs

IBP Storage

Streaming and Downloading Server

Processing Cluster

DEE
Providing Archived Content

- **Streaming**
  - **RealMedia format:**
    - 768, 512, and 256 kbps target audiences and SureStream, $360 \times 288 @ 15 - 25$ fps.
  - Experiments with 1.5 Mbps full PAL ($768 \times 576 @ 25$ fps).
  - RealServer with local disk array.

- **Downloading**
  - MPEG-4: $512 \times 384 @ 25$ fps @ 512 kbps + MP3 @ 128 kbps
  - Apache web server with local disk array.

- Currently a simple web interface.
- IP-based authorization limited to university IP address range.
Acknowledgments

- CESNET Development Foundation projects 017/2002 and 018/2002
- We would like to thank to Lukáš Hejtmánek for creating libxio, enabling IBP in many applications, and maintaining IBP infrastructure.
- We would like to thank to Luděk Matyska and Eva Hladká for supporting us from e-learning side.
Thank you for your attention!

Q/A?