UltraGrid: from point-to-point uncompressed HD to flexible multi-party high-end collaborative environment

Jiří Matela (matela@ics.muni.cz)
Masaryk University

EVL, UIC, Chicago, 2008–09–03
Laboratory of Advanced Networking Technologies

- Founded in 2002
- Directed by Luděk Matyska and Eva Hladká
Major Research Areas

- Multimedia distribution and processing
  - algorithms for real-time distributed processing
  - high-end (HD, post-HD) interactive multimedia transmission
- collaboration with industry
Major Research Areas

• Grid technologies
  • information services/monitoring (software development)
    • Logging and Bookkeeping Service for EGEE
  • infrastructure management (theoretical, practical)
  • scheduling (theoretical, practical)

• 68 000 CPUs
• 70PB of storage
• 300 000 jobs per day
Major Research Areas

• Virtualization
  • Grid environments
  • network virtualization
• Collaborative environments
  • collaboration with social sciences and psychology
• Active networks
  • user-programmable networks
• Security
  • authentication, authorization
  • frameworks for large scale collaborative/distributed environments
Collaboration

- EU projects
  - infrastructure: DataGrid, EGEE (I, II, ...)
  - software development & computer science: GridLab, CoreGrid (NoE)
  - support actions: Ithanet, EuroCareCF
  - design study: EGI-DS
- Also number of national projects
Collaboration

• Other EU collaboration
  • major partners e.g., INFN (IT), PSNC (PL), Koç University (TR)
• U.S. partners (e.g.)
  • Center for Computation & Technology, LSU
  • Electronic Visualization Lab, UIC
  • iCAIR, Northwestern University
  • Argonne National Laboratories
  • ResearchChannel, University of Washington
  • Dept. of Medicine, University of Michigan

• Asia partners
  • Academia Sinica
UltraGrid

- real-time transmission of high-resolution video
High-resolution

- HD, 2K, 4K, 6K resolutions
Data bandwidth

What is usually understood under uncompressed HD? (1920 × 1080, 1.485 Gbps, transmitted over SDI, SMPTE 292M)

- HD bandwidth calculation:

\[
\text{total resolution} \times \text{bit/point} \times \text{fps} \times \frac{2}{3} = 1.485.000.000 \text{ bps}
\]

- Resolution: includes 1920 × 1080 of effective resolution, but also adds up blanking lines, totaling 2200 × 1125.

- Color depth: 10 bits/point/color plane \(\Rightarrow\) 30 bits/point
  - Computers are usually unable to render more than 8 bits/color plane.

- Frame rate: 24p, 25p, 29.97p, 30p, 50i, 59.94i, 60i

- Sampling: usually 4:2:2
Data bandwidth

What is usually understood under uncompressed HD?
(1920 × 1080, 1.485 Gbps, transmitted over SDI, SMPTE 292M)

• HD bandwidth calculation:

\[
\text{2200} \times \text{1125} \times \frac{30}{\text{total resolution}} \times \frac{30}{\text{bit/point}} \times \frac{2/3}{\text{fps}} = 1.485.000.000 \text{ bps}
\]

• Resolution: includes 1920 × 1080 of effective resolution, but also adds up blanking lines, totaling 2200 × 1125.

• Color depth: 10 bits/point/color plane \(\Rightarrow\) 30 bits/point
  
  • Computers are usually unable to render more than 8 bits/color plane.

• Frame rate: 24p, 25p, 29.97p, 30p, 50i, 59.94i, 60i

• Sampling: usually 4:2:2
Data bandwidth

What is usually understood under uncompressed HD?
(1920 × 1080, 1.485 Gbps, transmitted over SDI, SMPTE 292M)

- HD bandwidth calculation:

\[
2200 \times 1125 \times 30 \times 30 \times \frac{2}{3} = 1.485.000.000 \text{ bps}
\]

- Resolution: includes 1920 × 1080 of effective resolution, but also adds up blanking lines, totaling 2200 × 1125.

- Color depth: 10 bits/point/color plane \(\Rightarrow\) 30 bits/point
  - Computers are usually unable to render more than 8 bits/color plane.

- Frame rate: 24p, 25p, 29.97p, 30p, 50i, 59.94i, 60i

- Sampling: usually 4:2:2
Data bandwidth

What is usually understood under uncompressed HD?
(1920 × 1080, 1.485 Gbps, transmitted over SDI, SMPTE 292M)

- HD bandwidth calculation:

\[
\text{total resolution} \times \text{bit/point} \times \text{fps} \times \frac{2}{3} = \text{1.485.000.000 bps}
\]

- Resolution: includes 1920 × 1080 of effective resolution, but also adds up blanking lines, totaling 2200 × 1125.

- Color depth: 10 bits/point/color plane → 30 bits/point

  - Computers are usually unable to render more than 8 bits/color plane.

- Frame rate: 24p, 25p, 29.97p, 30p, 50i, 59.94i, 60i

- Sampling: usually 4:2:2
Data bandwidth

What is usually understood under uncompressed HD? (1920 × 1080, 1.485 Gbps, transmitted over SDI, SMPTE 292M)

- HD bandwidth calculation:

\[
\text{2200} \times \text{1125} \times \text{30} \times \text{30} \times \frac{2}{3} = 1.485.000.000 \text{ bps}
\]

- Resolution: includes 1920 × 1080 of effective resolution, but also adds up blanking lines, totaling 2200 × 1125.

- Color depth: 10 bits/point/color plane \(\rightarrow\) 30 bits/point
  - Computers are usually unable to render more than 8 bits/color plane.

- Frame rate: 24p, 25p, 29.97p, 30p, 50i, 59.94i, 60i

- Sampling: usually 4:2:2
Data bandwidth – continuation

- HD – 1.16 Gbps
- 2K – 1.24 Gbps
- 4K – 4.94 Gbps
- 4K (4096×3112) – 7.12 Gbps
- 6K – 15.94 Gbps
UltraGrid – real-time transmission and latency

- End-to-end (including camera, network and display)
- Frame is shot by video camera, captured, transmitted and displayed
- Uncompressed HD: 85 ms
  - Centaurus II capture card
  - Linux
  - 10GE Myrinet card
- DXT-Compressed HD: 95 ms
  - At least 4 CPU cores
  - Otherwise same configuration
- E.g. professional digital camera has shutter lag 40ms
  - time between you pressing the shutter release button and the camera actually starts taking the shot
UltraGrid – usage example

- partnership with a movie industry: CinePost
- experimental use of UltraGrid for remote cutting and color adjustment
UltraGrid – usage example

- partnership with a movie industry: CinePost
- experimental use of UltraGrid for remote cutting and color adjustment
CoUniverse: Motivation

- Orchestration of large number of components
  - data: producers, consumers, distributors
    - starting, stopping, (re)configuring, monitoring
  - underlying infrastructures: networks, λ-services, computing elements
    - reservations, allocations, monitoring
    - handling alternative resources

- Ever changing environment
  - monitoring, adaptation, managing alternatives
CoUniverse: Motivation

- Real-time multimedia applications
  - bandwidth of data streams comparable to capacity of links
    - automagic additivity assumption no longer works
  - many application can’t automatically adapt to networking conditions
    - either need to be told explicitly what to do
    - or use an alternative application
  - encapsulation of applications, that can’t be modified themselves
CoUniverse: Architecture

- Universe
  - collaborative space of limited size
  - equivalent of “venue” in other systems, though with slightly different motivations (size of scheduling, allocations, monitoring, etc.)

- Multiverse
  - information service
  - registration and lookup of universes
CoUniverse: Architecture

- Control plane vs. data plane
  - optimized for different purposes
  - control plane has robustness and resilience as primary focus
    - based on peer-to-peer overlay network with aggressive monitoring and rerouting
  - data plane has performance (bandwidth, latency) as primary focus
    - uses native network including some specialized features like multicast (application-level, network-level, optical-level), dedicated circuits (λ-services, SONET circuits)
CoUniverse: Architecture

- Components
  - network composed of network nodes and network links
  - applications organized into application groups
    - encapsulation of non-modifiable applications
    - integration of applications that can be modified
  - application group controller (AGC)
    - steers application groups
    - dynamically elected, any node can take this role (conceptually, though there might be some policy-based limitations)
    - takes care of stream scheduling, plan preparation and distribution
    - reacts to changes in the Universe (on any level)
CoUniverse: Implementation

- Java-based prototype implementation
  - JXTA 2.4 for control plane
- Scheduler implementation
  - implemented constraint-based scheduler, that works fine for smaller communities (uses Choco solver)
  - implemented simple scheduler for application groups, that don’t use bandwidth comparable to link capacities
  - working on a scheduler using combination of heuristics and constraint-based verification
- Application modules
  - UltraGrid + various videoconferencing applications
  - generic application wrapper (e.g., microscope image streaming applications, etc.)
CoUniverse: Implementation

- Monitoring
  - network node monitoring, application monitoring, network link monitoring (on application level, not ping)
  - currently working on more advanced monitoring (we don't want magic-closed MonALISA)
CoUniverse: Implementation

- Network visualization
  - visualization of the resulting plan, active streams, nodes, applications
  - integration of data from monitoring in progress
- https://www.sitola.cz/CoUniverse/
CoUniverse: Demos

- GLIF 2007
- SC|07
- planned demonstration Internet2 Fall MM 2008, SC|08
JPEG 2000

- Superior low bit-rate performance
  - Offers superior performance at very low bit-rates (0.25 b/pixel)
- Lossless and lossy compression
- Progressive transmission by pixel accuracy and resolution
  - Compressed stream can be organized by pixel accuracy
    - Resolution as original, more data received more quality image displayed
  - Compressed stream can be organized by resolution accuracy
    - Quality as original, more data received bigger resolution image displayed
JPEG 2000

- Half data image example – somebody cut the wire
JPEG 2000

- Half data image example – somebody cut the wire
JPEG 2000 – implementation

- 3 basic steps
  - RGB <-> YUV color space conversion (optional)
    - YUV 4:2:2 sampling saves 1/3 of bandwidth
  - Discrete Wavelet Transform – DWT
    - DWT is the mechanism behind the progressive resolution transmission capability
  - Bit plane coding
My implementation on GPU using CUDA

Measured on HD image using GeForce G280 GPU

- RGB <-> YUV color space conversion (optional)
  - 0.5ms using CUDA
  - 6ms SSE2 assembler instructions using 128bit registers
- Discrete Wavelet Transform – DWT
  - 2ms using CUDA – unoptimized version, can be improved
  - 255ms on CPU, using C – highly unoptimized version
- Bit plane coding
  - not implemented
Thank you for your attention!

Q?/A!

matela@ics.muni.cz