

Storisphere 2: Return of the killer video-editing app

or “Son of Storisphere”

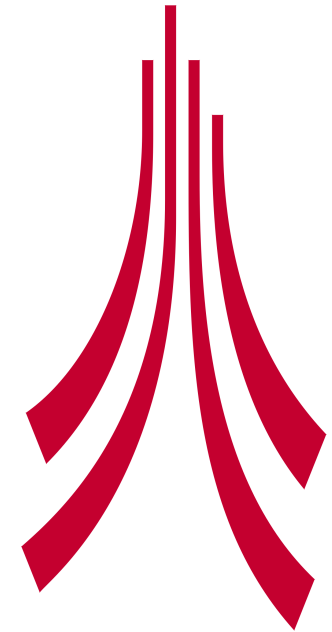
or “Storisphere: The Next Generation”

or “The Revenge – This time it's personal”

Steven Simpson

David Hutchison

Mu Mu, James Brown, Craig Bojko,
Jamie Jellicoe, Ross Wilson



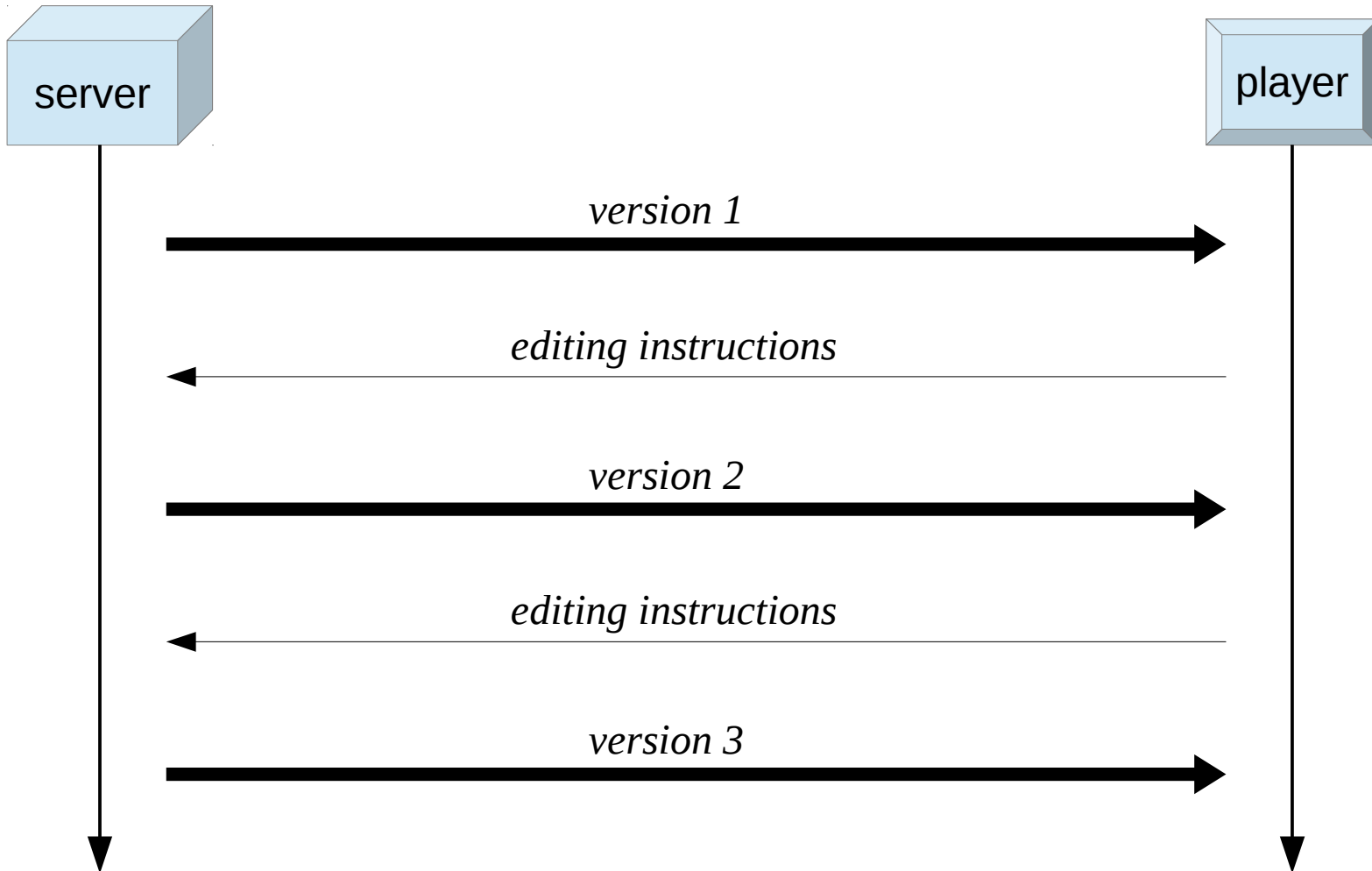
Lancaster University

School of Computing and Communications

Storisphere aims

- Support collaborative storytelling
 - Like SourceForge/GitHub, but for video
 - Aimed at 'hyperlocal TV', community production
 - Support audio/video, stills, and audio commentary
- Edit on slim client devices
 - Web front-end
 - Minimal configuration
 - Server does the hard work
 - Cache and decode on client for playback

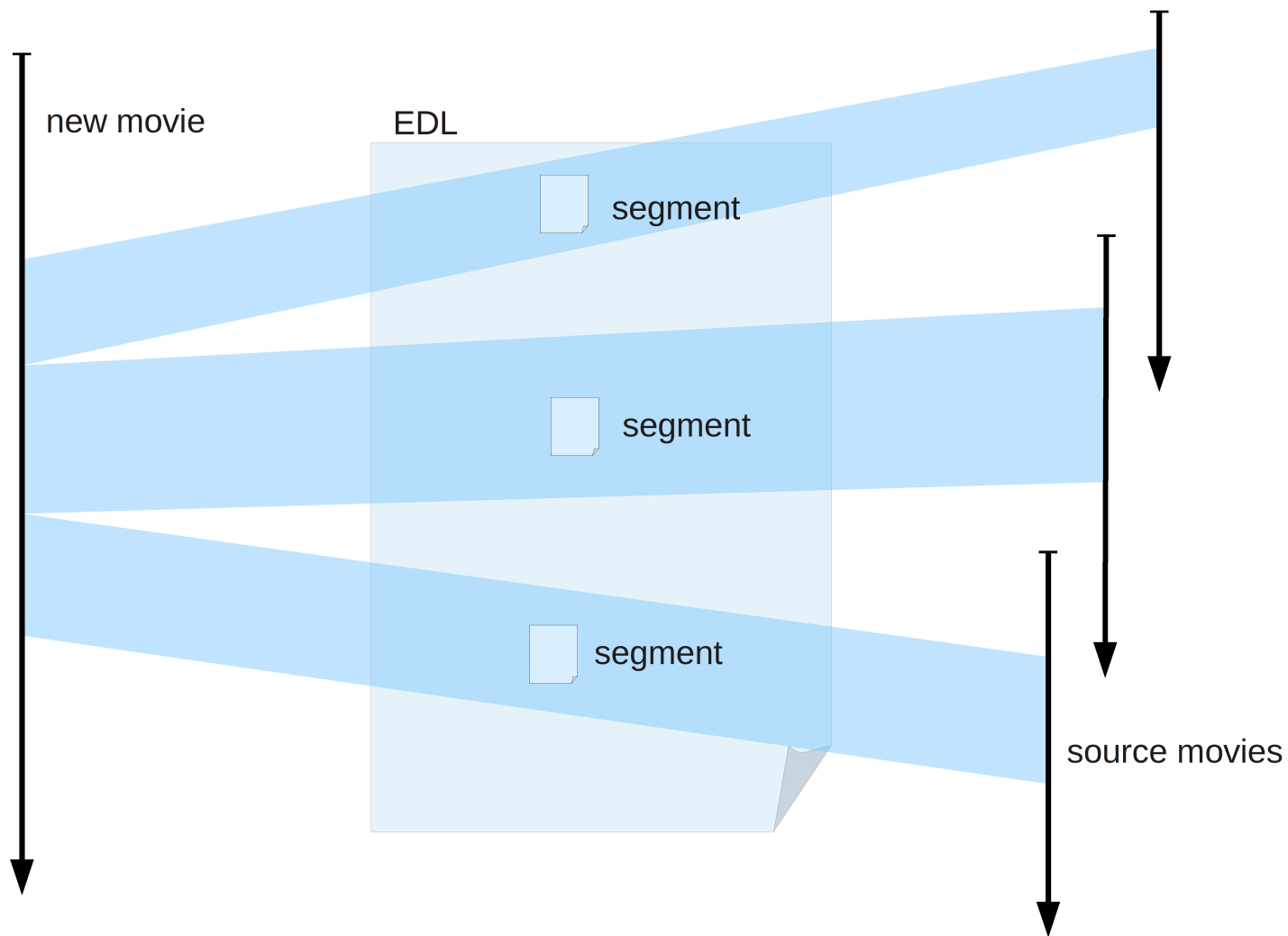
Editing on slim clients



Technique

- On ingest of new rush, split video/audio into independently decodable chunks
 - Chunk boundaries must fall on GOP boundaries
 - Also store at different quality levels (video resolutions, audio frequencies)
 - Create rush EDL (edit-decision list)
- On playback of an EDL, work out which chunks are needed
 - Deliver chunk addresses with re-combination instructions
 - MPEG-4
 - Conformation

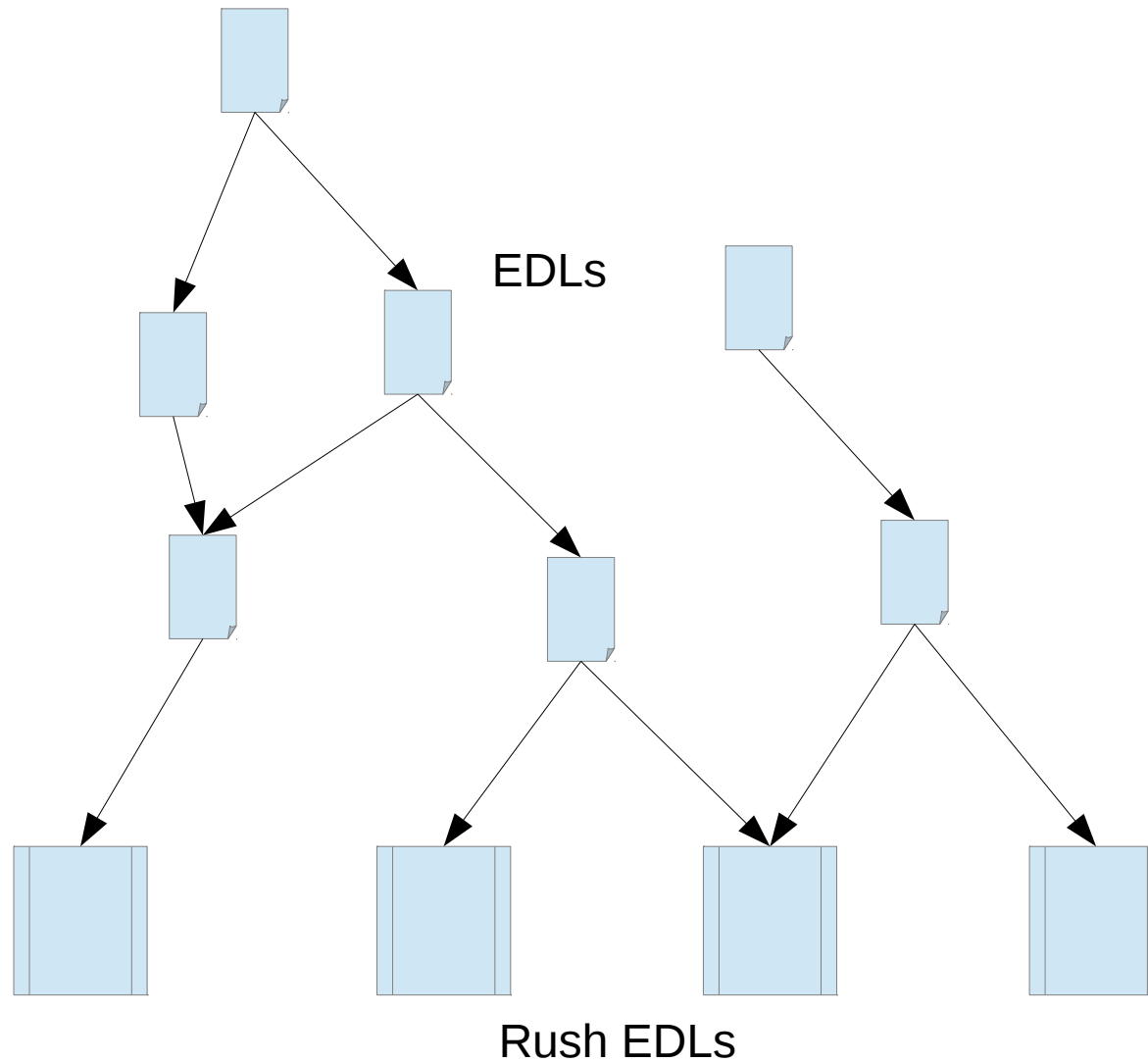
EDL Composition



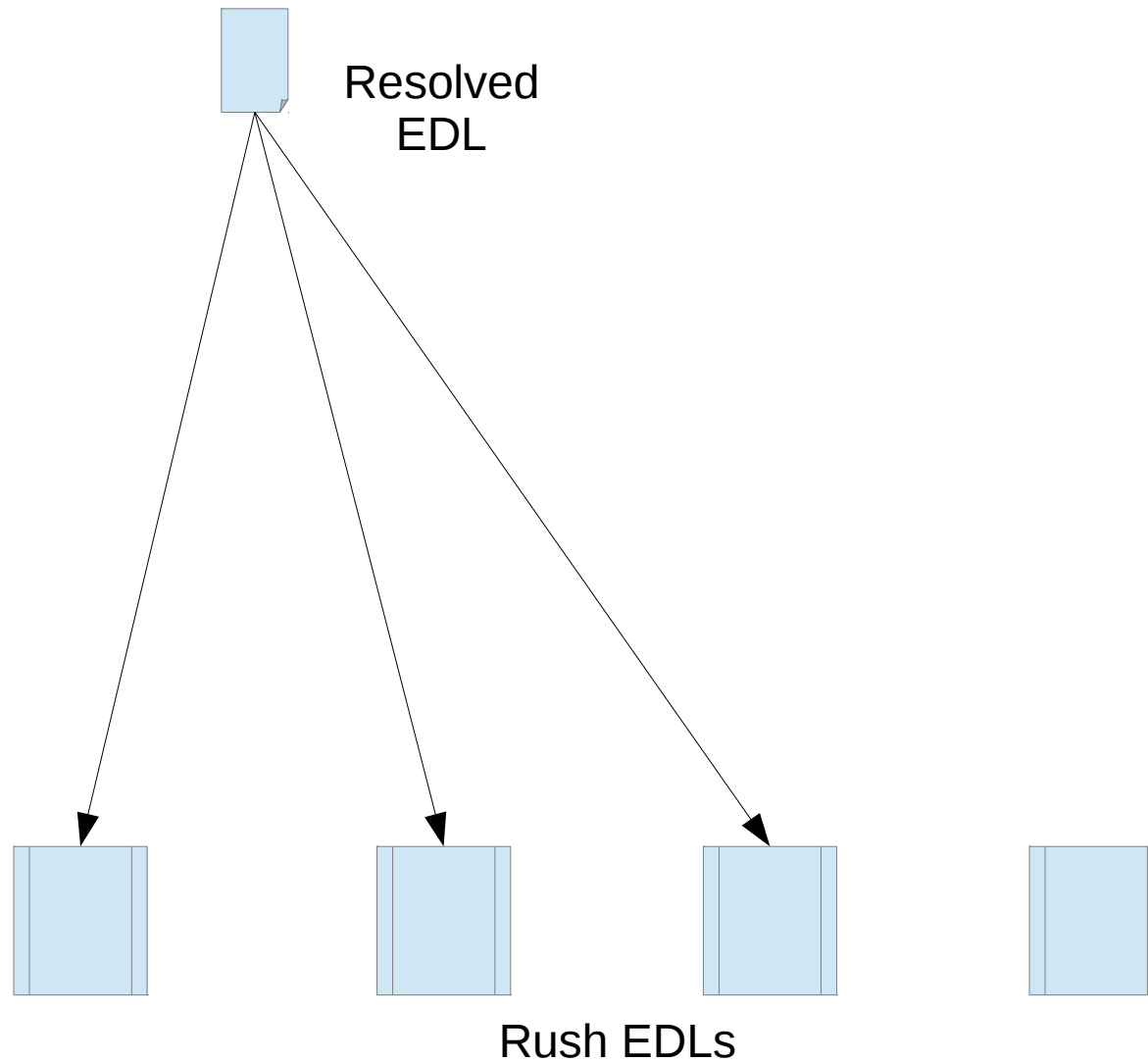
Benefits

- Form new EDL using parts of different existing EDLs
 - Hierarchies of EDLs; needs resolution process
- Late choice of quality level
 - By referencing an EDL for a virtual rush at 'no particular level'
 - Editing and previewing is independent of quality level
- After ingest, no further transcoding
 - No transcoding overheads
 - Chunks are re-usable

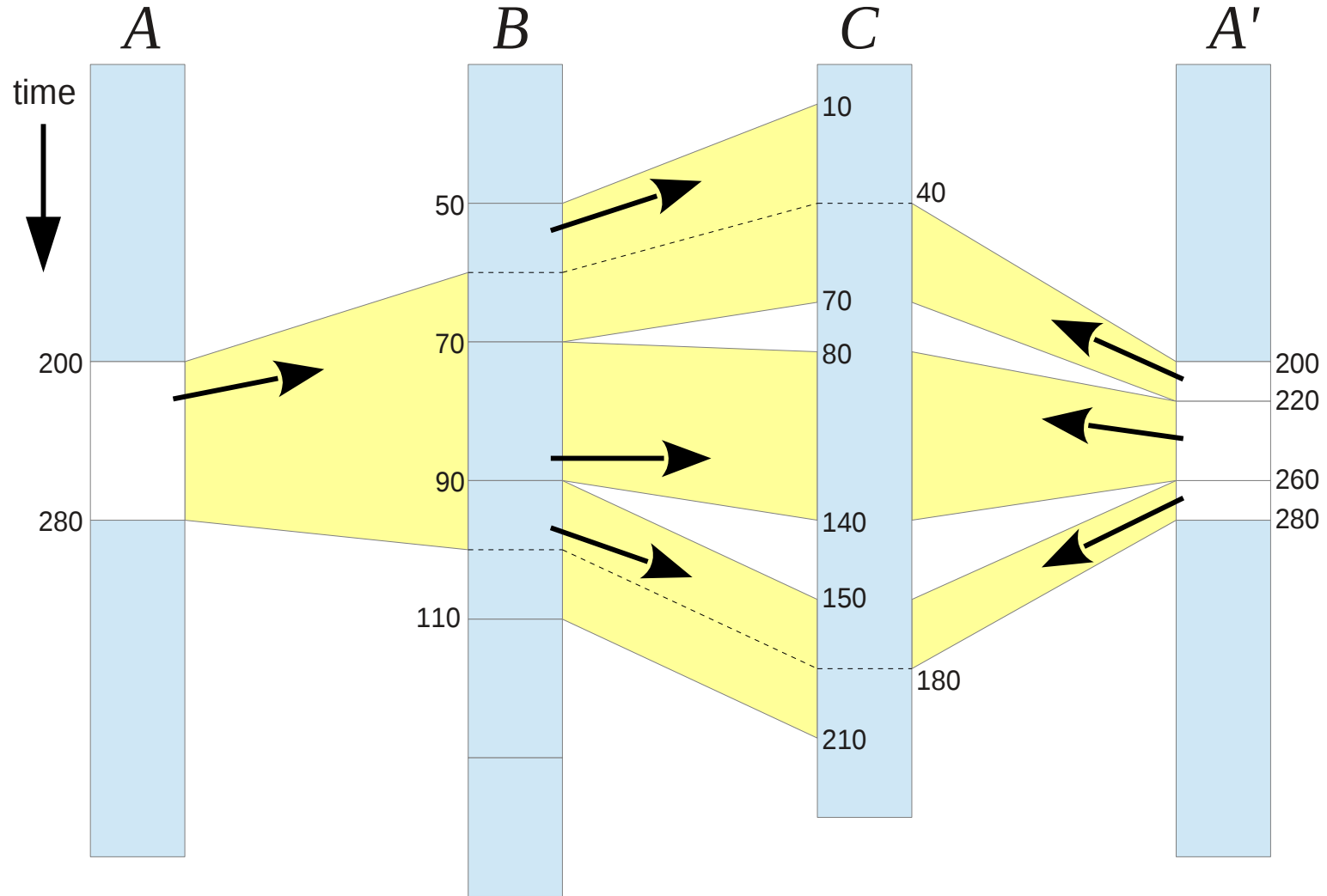
Recursive EDLs



EDL resolution



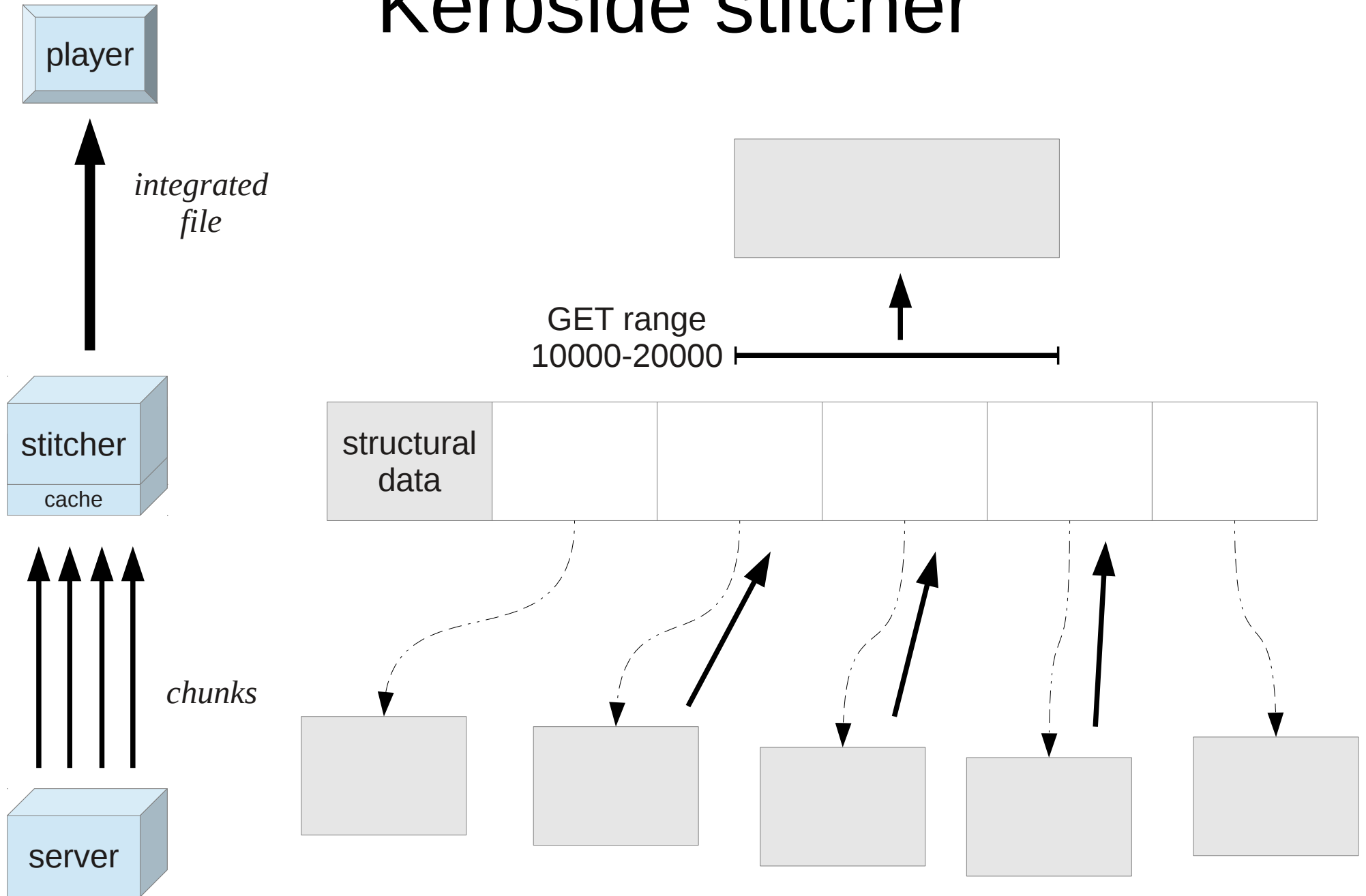
EDL resolution



First steps

- MARS EDL resolution and conformation engine
 - Conformation to MPEG-4 without further transcoding
- Mediaplex time-indexed meta-data for search
 - Ingest of user-generated and on-air content
- Storiboard GUI for collaboration
- Kerbside Stitcher for improving compatibility with existing players
 - Work around limited MPEG-4 implementations

Kerbside stitcher



Previously on Storisphere: TNG

- Time and place added to meta-data
 - Location-/time-aware search, and tagging (provided by partner)
- Stitcher discovery
 - Use side-effects of NAT to auto-associate user with nearest stitcher
- Pre-caching in the stitcher
 - Combine stitcher discovery with user/group associations to suggest stories to fetch during idle hours
- Stitcher deployment on home gateway
 - Gateway provided by partner
 - Embedded environment
 - Java 5, OSGi
 - Limited resources
 - Lots of optimizations applied

Home Gateway

- Limited memory and power
 - Don't decode uninteresting boxes
 - Assume absolute URLs
 - Avoid extraneous serialization steps and buffering
- Slow storage (USB stick)
 - Reduce number of files written to
 - Keep in RAM
- OSGi
 - Drop Oracle embedded HTTP server
 - Switch to HTTP servlet API



- Java 5 (Skelmir VM)
 - Downgrade from 7
 - No run-time autoboxing optimizations?
 - Expose underlying primitive type arrays
 - Limited FP?
 - Reduce FP operations
 - Stick to integer operations

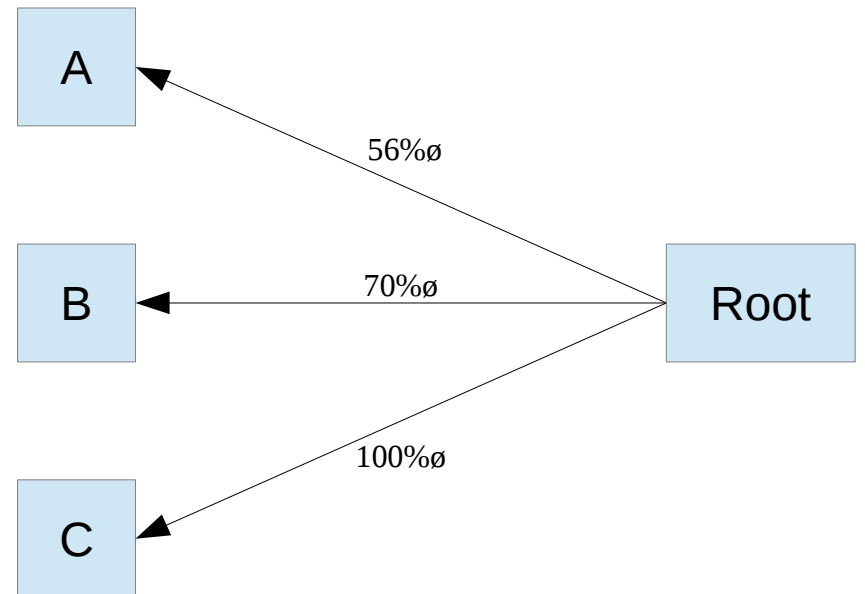
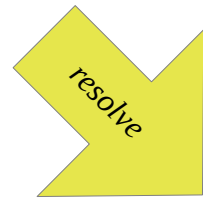
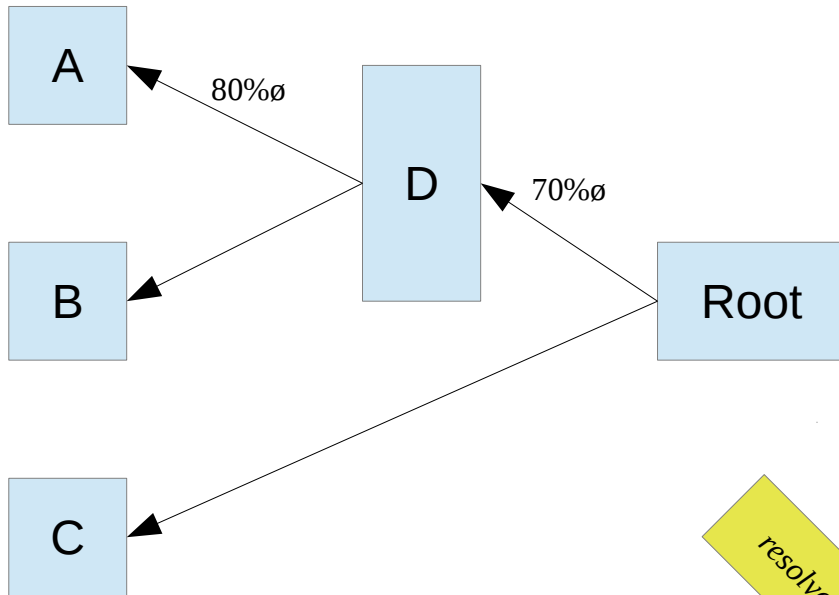
Next steps

- Migrate from MPEG-4 to MPEG-DASH
 - Exploit DASH-JS player using Media Source Extensions
 - DASH replaces MPEG-4 as the re-composition format
 - Edit lists expressed as an extension
 - Room for volume control and video fade-in/-out?
- Migrate from MPEG-4 to direct Media Source Extensions
 - Cut out the intermediate (DASH) step
 - Custom format (EDL itself?)
 - Handle effects directly
- Server independence
 - Like the Web, but with video
 - Interaction between content-dedicated sites and private users

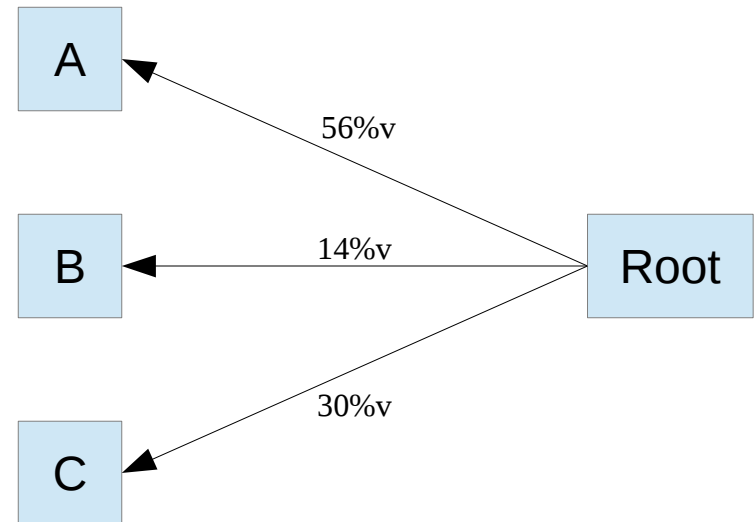
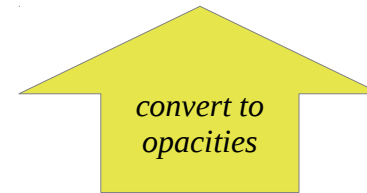
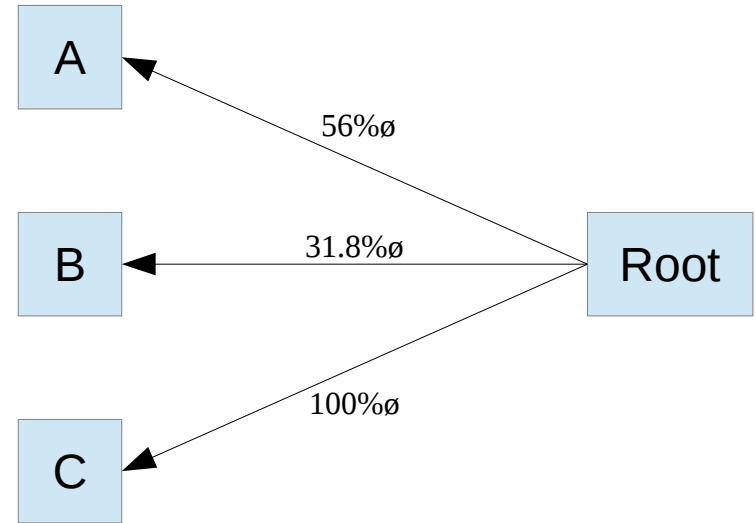
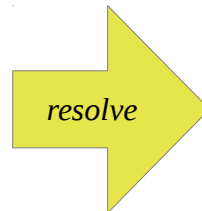
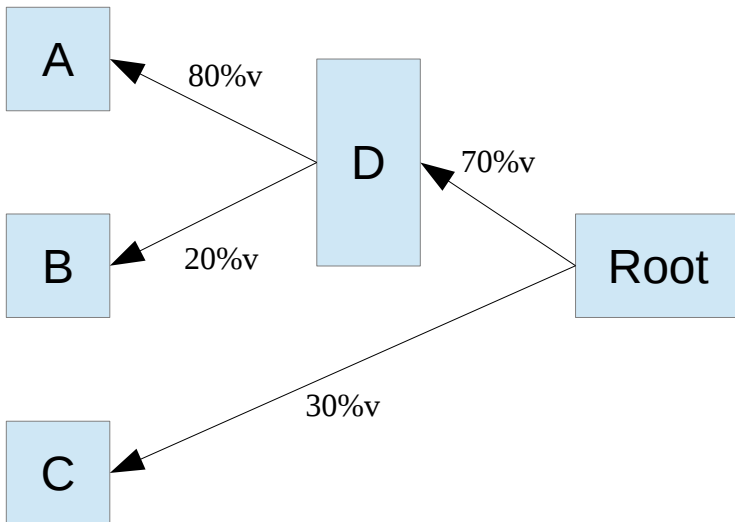
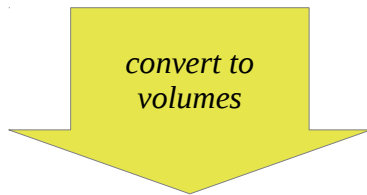
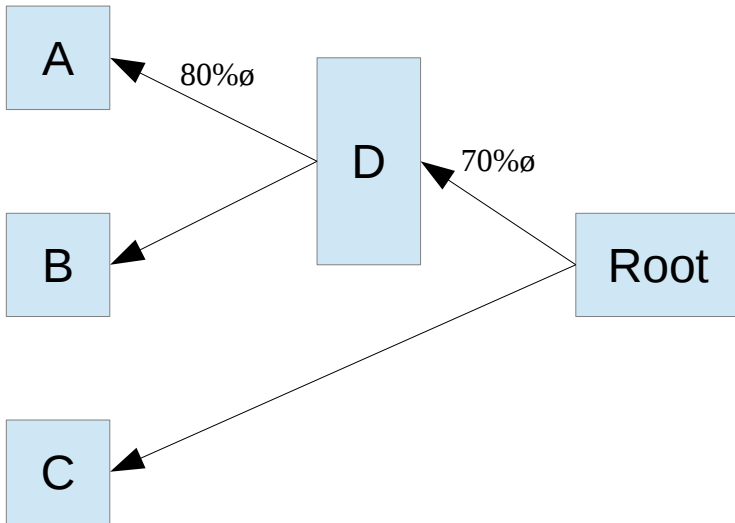
Fading in and out

- MPEG-4 options
 - Audio, yes
 - but only by using multiple tracks, each at a fixed volume
 - Video, no
- JavaScript/Media Source Extensions
 - Audio, ?
 - Video, yes?
 - CSS opacity?

Video opacity



Video opacity



Video opacity

- Store as volumes, not opacities
 - The editor's job
- Resolve multiplicatively
 - Only works with volumes

Links

<http://one.lancs.ac.uk/>

Storisphere: collaborative video editing system
(formerly "ONE")

Acknowledgements

steer 

<http://www.fp7-steer.eu/>



<http://www.firm-innovation.net/>

Storisphere, and especially its MARS component, have developed from the Open Narratives Environment (ONE), conceived by colleagues from BBC Research and Development at MediaCityUK (Michael Sparks and Adrian Woolard), subsequently taken forward by Adam Lindsay and others at Lancaster University. We greatly acknowledge their contribution to the work documented herein.