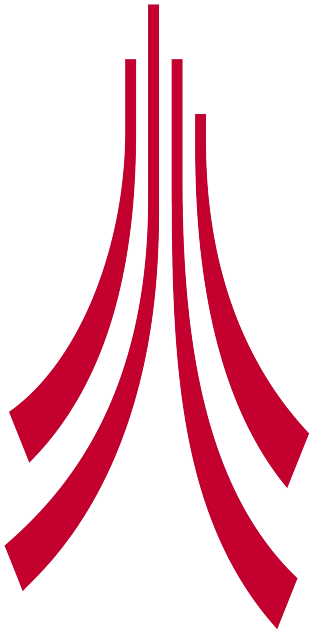
On-line Video-Editing Challenges in Storisphere

Steven Simpson
David Hutchison
Mu Mu, James Brown, Craig Bojko,
Jamie Jellicoe, Ross Wilson

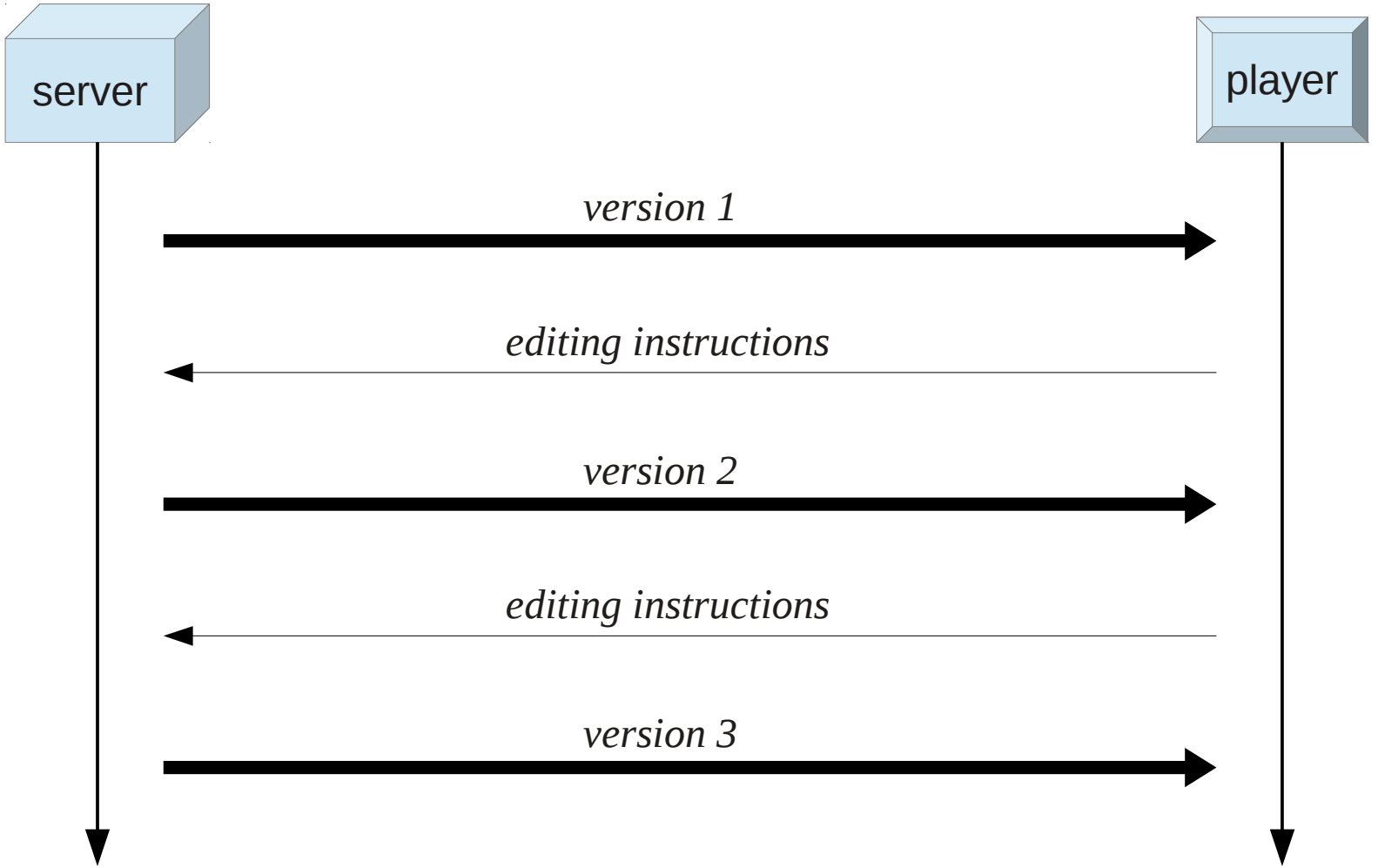


Lancaster University
School of Computing and Communications

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

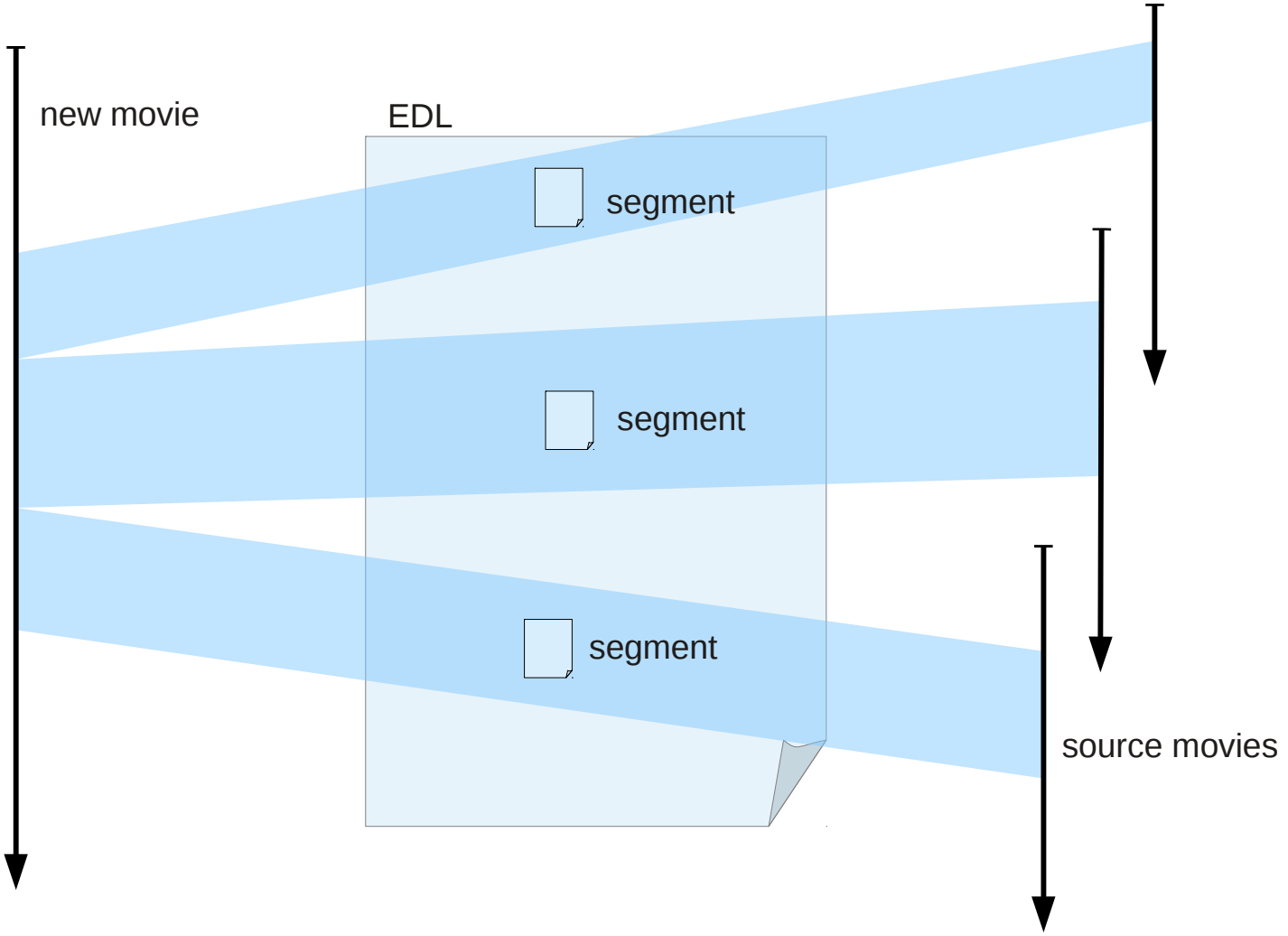
Editing on slim clients



Content preparation

- Separate video and audio tracks.
- Split each track into independent chunks.
  - On GOP boundaries
  - Using closed GOPs
  - Publish as static files
- Write a rush EDL (edit-decision list) to recombine them into the original.
  - One EDL segment per chunk

EDL Composition



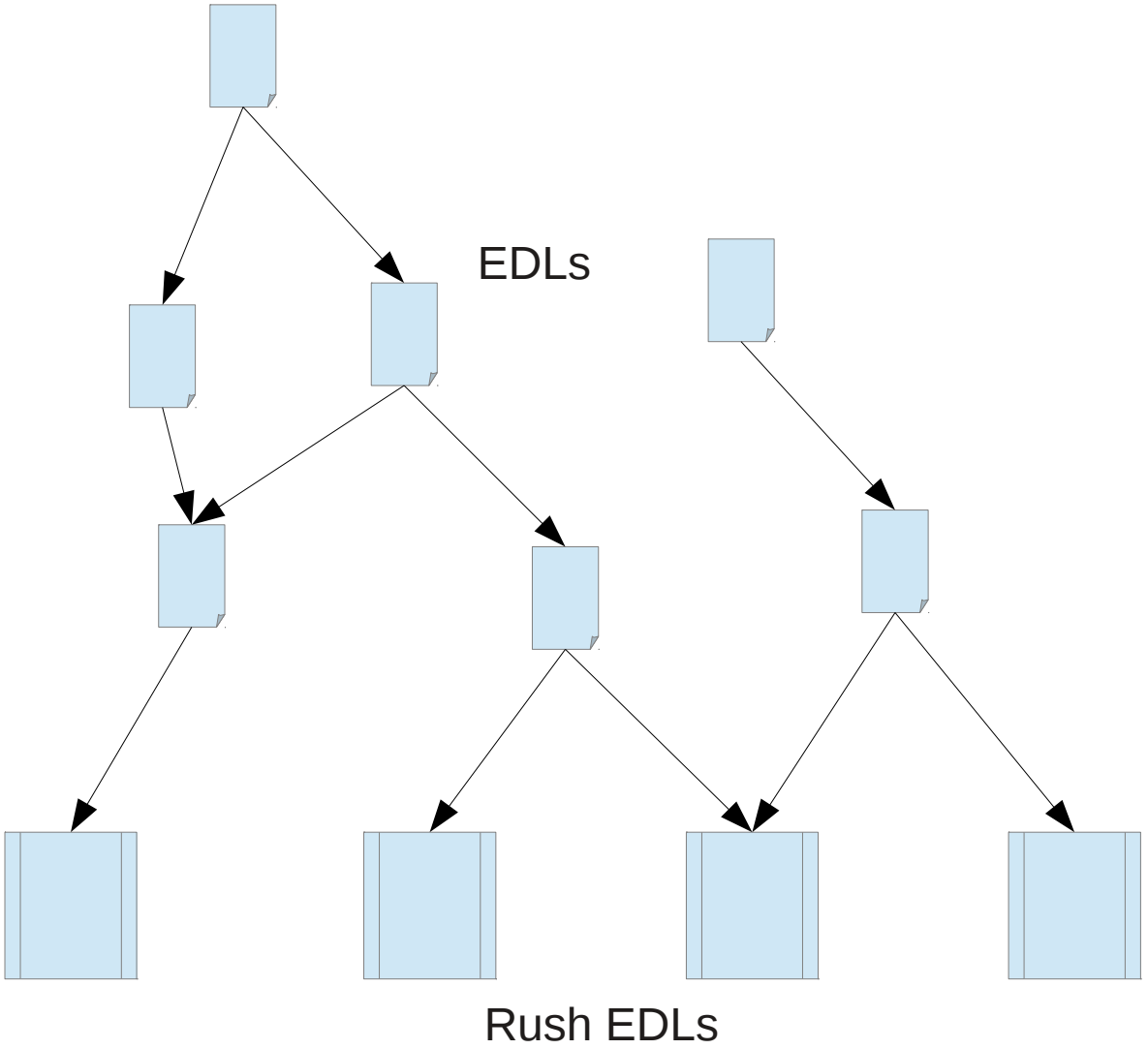
Content retrieval

- Convert chunk-describing segments into MPEG-4 structural data.
- Add URI references to chunk files.
- Send to player.
- Player parses structural data and fetches chunks.
- Cache chunks for re-use!

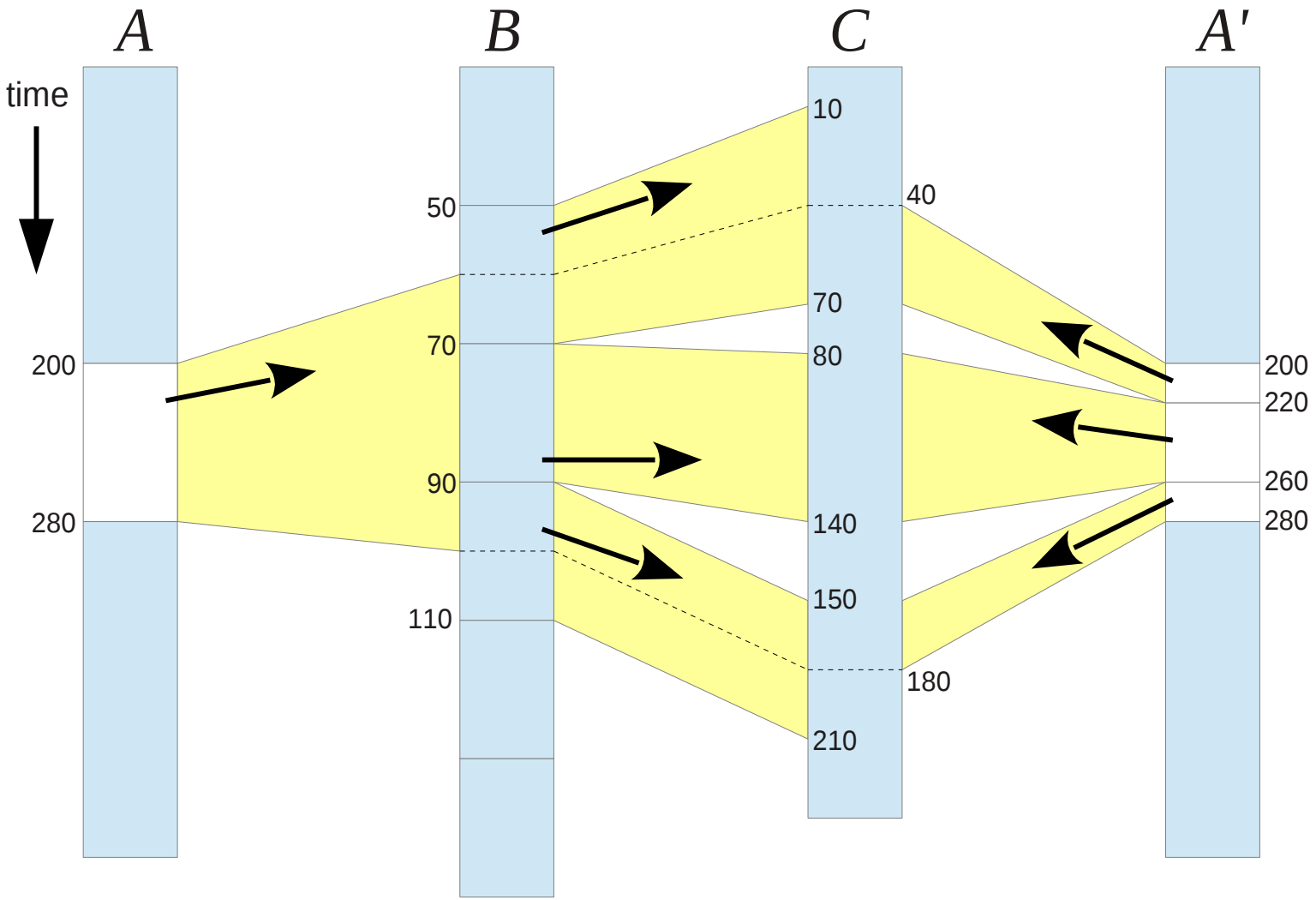
Benefits of EDLs

- All derived content expressed as small text documents
  - Combine segments from separate rushes
  - Once ingested, no need for further transcoding
- EDL hierarchy
  - Can reference rush EDLs or other derived content
  - Build up advanced stories in stages
  - Track origins
  - Defer choice of resolution until point of playback

Recursive EDLs



EDL resolution



Challenges

- EDLs
  - Rational numbers
    - Propagation of prime factors
- MPEG-4 format
  - Limited effects
    - No visual combination
    - Fixed audio volume
  - 32- or 64-bit limits
- Player
  - Missing functionality
    - MPEG-4 edit lists
    - External chunks
    - 64-bit integers
  - Inefficient implementation
    - No playback until all chunks fetched
  - Faulty implementation
    - Chunks abandoned
    - Mixed audio frequencies
    - Lingering stills
    - Mixed aspect ratios

Solutions: Rational numbers

- Limited set of frame rates
  - Numerator of rate becomes denominator of frame durations: 30000/1001=29.97Hz => 30000 MPEG-4 timescale
- Record ideal 'cut' positions.
  - Frame boundaries
  - Editor forbids cutting at arbitrary positions

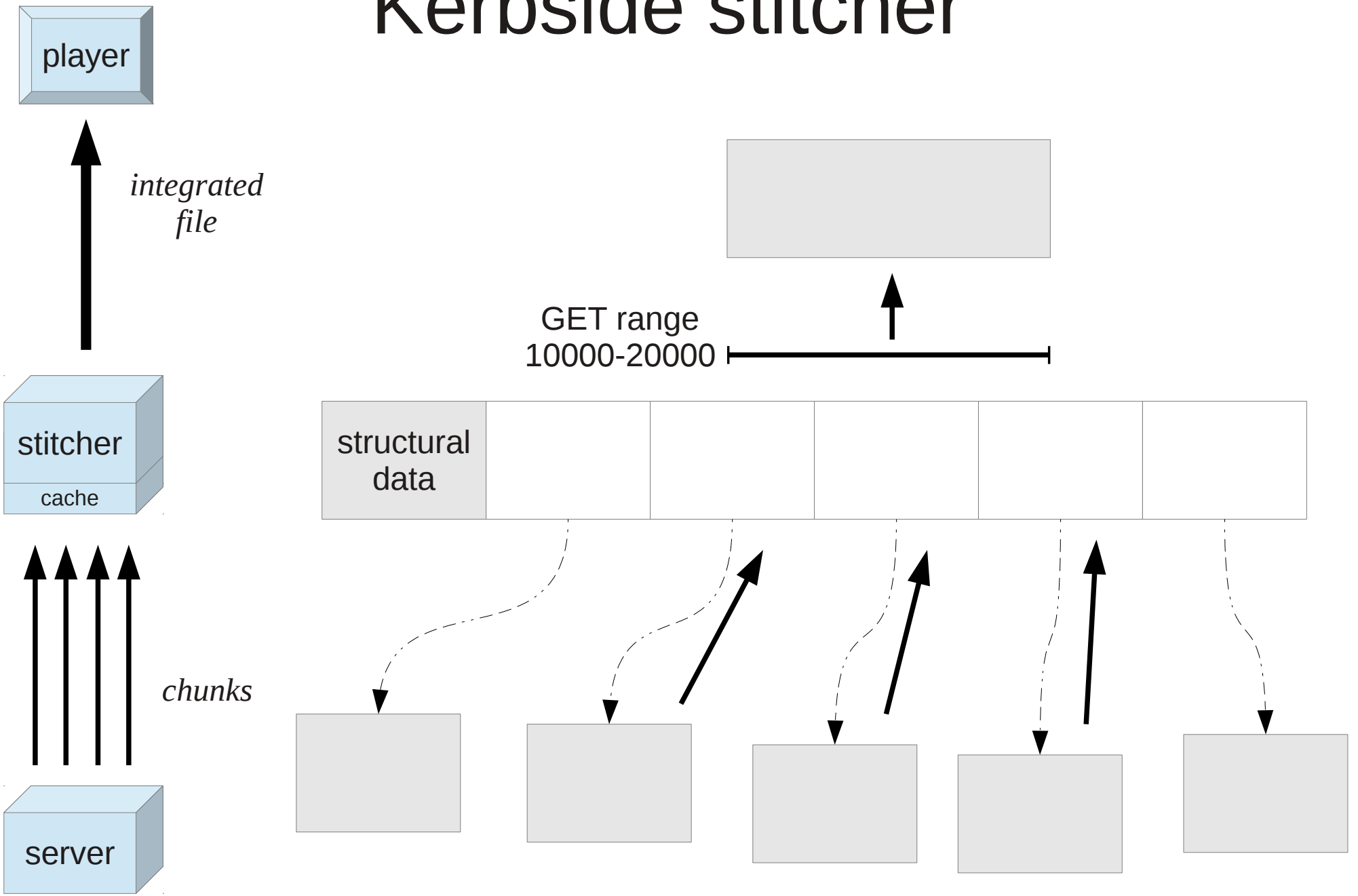
Solutions: MPEG-4 format

- Don't do effects!
  - Audio overlays okay, though
- Lose some accuracy in generation of MPEG-4 edit list.
  - Shouldn't be easily perceptible

Solutions: Player

- Force use of QuickTime
  - Good support for edit lists
- Use a kerbside stitcher
  - Gets round bad/missing implementations of external chunk fetching
- Ingest at only one audio quality (44.1kHz)
- JIT translation of stills into 'very slow' edits
- Put up with aspect-ratio problems

Kerbside stitcher



Better solution: Our own player

- Requirements
  - Needs to be in JavaScript for maximum portability
  - Needs access to native decoder
    - Byte-based, not file/URI-based
- Benefits
  - Not limited by MPEG-4
  - Could do our own effects
  - Build the stitcher into the player

Links

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

Storisphere: collaborative video editing system

(formerly “ONE”)

Acknowledgements

FIRM
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.