



# WebRTC beyond Audio and Video

Silvia Pfeiffer  
Coviu

WebRTC Summit, 4<sup>th</sup> Nov  
8:30am – 9:05am



# A bit about me

W3C Recommendation



## HTML5

A vocabulary and associated APIs for HTML and XHTML

W3C Recommendation 28 October 2014

**This Version:**  
<http://www.w3.org/TR/2014/REC-html5-20141028/>

**Latest Published Version:**  
<http://www.w3.org/TR/html5/>

**Latest Version of HTML:**  
<http://www.w3.org/TR/html/>

**Latest Editor's Draft of HTML:**  
<http://www.w3.org/html/wg/drafts/html/master/>

**Previous Version:**  
<http://www.w3.org/TR/2014/PR-html5-20140916/>

**Previous Recommendation:**  
<http://www.w3.org/TR/1999/REC-html401-19991224/>

**Editors:**

**WHATWG:**  
Ian Hickson, Google, Inc.

**W3C:**  
Robin Berjon, W3C  
Steve Faulkner, The Paciello Group  
Travis Leithead, Microsoft Corporation  
Erika Doyle Navara, Microsoft Corporation  
Edward O'Connor, Apple Inc.  
Silvia Pfeiffer



Books



## WebVTT: The Web Video Text Tracks Format

Draft Community Group Report, 28 October 2015

**This version:**  
<http://w3c.github.io/webvtt/>

**Test Suite:**  
<http://github.com/w3c/webvtt-test-suite/>

**Editors:**  
Simon St Laurent (Opera Software AB)  
Clemens Nemeth (Apple Inc.)

**Former Editors:**  
Silvia Pfeiffer (W3C)  
Philip Legrain (Opera Software ASA)  
Clemens Nemeth

**Participating:**  
Opera Software ASA, Opera Software, Apple, Apple, Google

**Comments:**  
[github@w3c.org](mailto:github@w3c.org)  
Twitter

Network Working Group  
Request For Comments: 5242  
Category: Informational

S. Pfeiffer  
May 2015

The WebVTT (Web Video Text Tracks) Format Version 0

Status of This Memo:

This memo provides information for the Internet community. It does not specify an Internet standard of any kind. Distribution of this memo is unlimited.

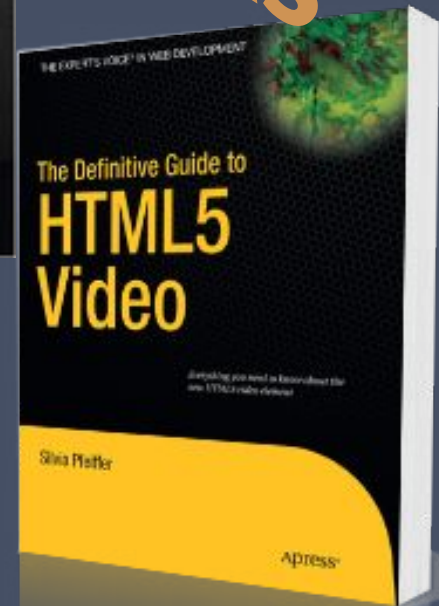
Network Working Group  
Request For Comments: 5244  
Obsoletes: 5242  
Category: Standards Track

S. Pfeiffer  
S. Montgomery  
September 2015

WebVTT Media Types

Status of This Memo:

This document specifies an Internet standards track protocol for the Internet community, and requests discussion and suggestions for improvements. Please refer to the current edition of the "Internet Official Protocol Standards" (RFC 5138) for the standardization state and status of this protocol. Distribution of this memo is unlimited.



# What is WebRTC to you?

*Strongly agree*

*Agree*

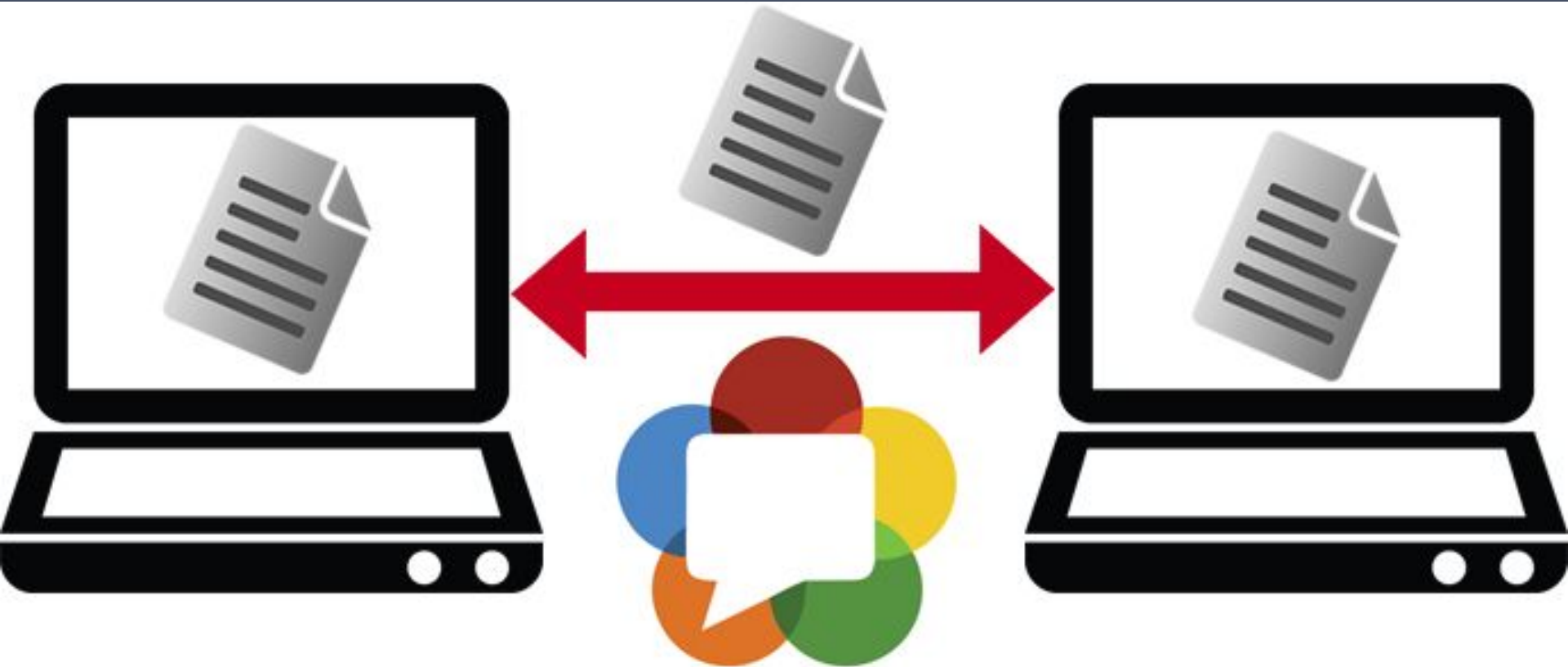
*Disagree*

*Strongly disagree*

# Is it ...

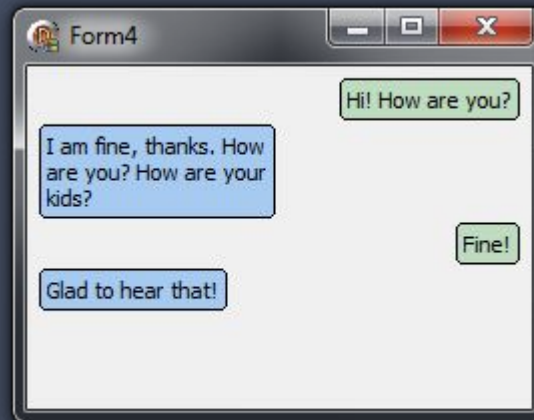
1. An SDK to develop voice and video mobile applications?
2. A means to include the browser in your unified communications solution?
3. An extension-free way to re-write your video conferencing application?
4. An opportunity for adding video/audio communications to your Web application?
5. A new Web API that Web developers need help with?

Are you even using  
the data channel?



# What does data sharing mean to you?

- File sharing ?
- Screensharing ?
- Text chat ?



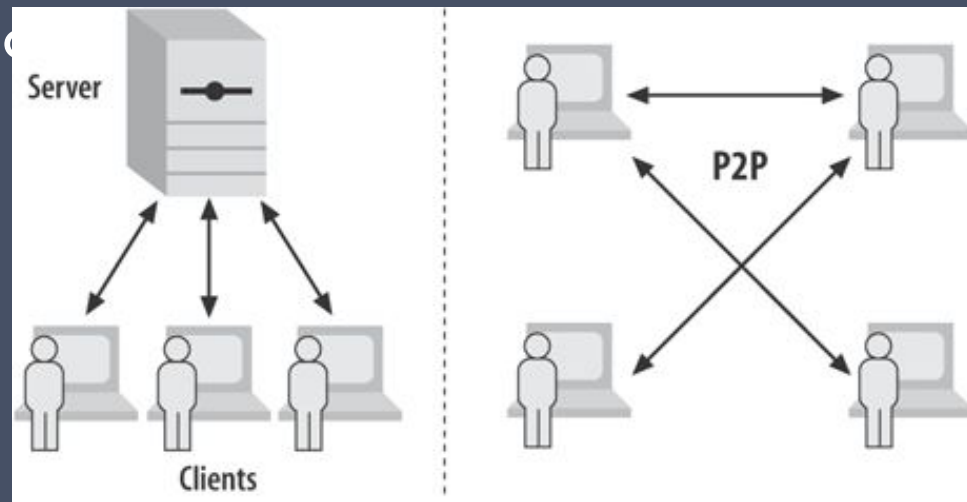
BACK TO  
BASICS






# What is WebRTC to the Web?

- Google's original motivation: video conferencing for the Web
- What it really is: A novel paradigm: peer-to-peer
  - Every browser is a (potential) server
  - Every browser can share files
  - Every browser can share code
  - Distributed computing for the Web





# Distributed Computing Characteristics

- 
- concurrency of components  
(decentralisation)
  - lack of a global clock
  - independent failure of components

- Decentralisation of data

# Decentralisation

- No central storage, efficiency
- High availability
- Privacy of data
- Scalability



- Decentralisation of computing

- Parallel computing
- Scalability



- Decentralisation of communication channels



# Challenges of Peer-to-Peer

- Decentralisation of data
  - No central storage: search is more difficult
  - Potential for lower availability
- Decentralisation of computing
  - Parallel computing complexity
  - Coherence
- Decentralisation of communication channels
  - Not scalable



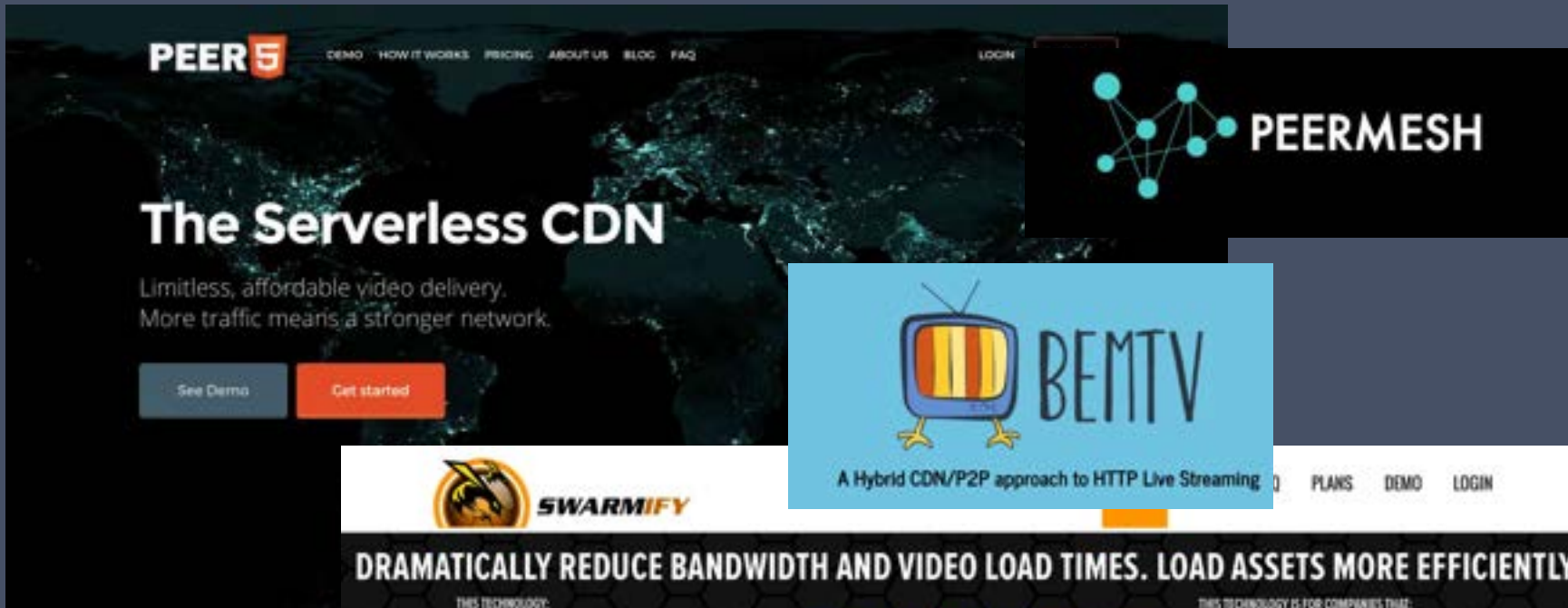
# P2P Applications of WebRTC



**WebRTC**



# I. Video & Data streaming CDNs




**PEER5** DEMO HOW IT WORKS PRICING ABOUT US BLOG FAQ LOGIN

## The Serverless CDN

Limitless, affordable video delivery.  
More traffic means a stronger network.

[See Demo](#) [Get started](#)



**BEMTV**  
A Hybrid CDN/P2P approach to HTTP Live Streaming

PLANS DEMO LOGIN



**SWARMIFY**

### DRAMATICALLY REDUCE BANDWIDTH AND VIDEO LOAD TIMES. LOAD ASSETS MORE EFFICIENTLY.

THIS TECHNOLOGY:

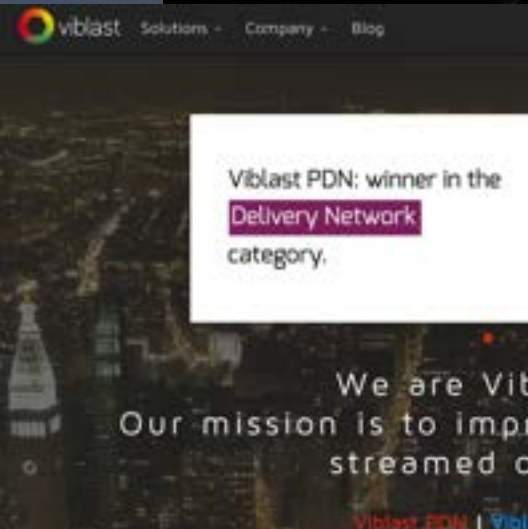
- REDUCES BANDWIDTH DRAMATICALLY
- ADDS TO ANY SITE ON TOP OF ANY CDN / HOST
- PROVIDES A COMP ADVANTAGE FOR VIDEO PROVIDERS
- PROVIDES INSTANT CACHE EXPIRATION
- CAN ADD TO ANY HTML5 VIDEO PLAYER IN MINUTES

**30 DAY UNLIMITED TRIAL**  
IMPLEMENT TODAY AND FEEL THE POWER OF THE SWARM

THIS TECHNOLOGY IS FOR COMPANIES THAT:

- STREAM VIDEO & WANT TO REDUCE EXPENSES
- WANT UNLIMITED POINTS OF PRESENCE (PoPs).
- EXPECT SECURE, FAST AND RELIABLE CDN QUALITY VIDEO
- CAN BENEFIT FROM HYPER-LOCAL CONNECTIVITY
- MANAGE LARGE VIDEO STREAMING OPERATIONS

OR USE THE SWARMIFY VIDEO PLAYER  
CAN ADD TO ANY PLAYER WITH ENTERPRISE INTEGRATION



**viblast** Solutions - Company - Blog

Viblast PDN: winner in the **Delivery Network** category.

We are Viblast.  
Our mission is to improve how video is streamed online.

[Viblast PDN](#) | [Viblast Player](#)

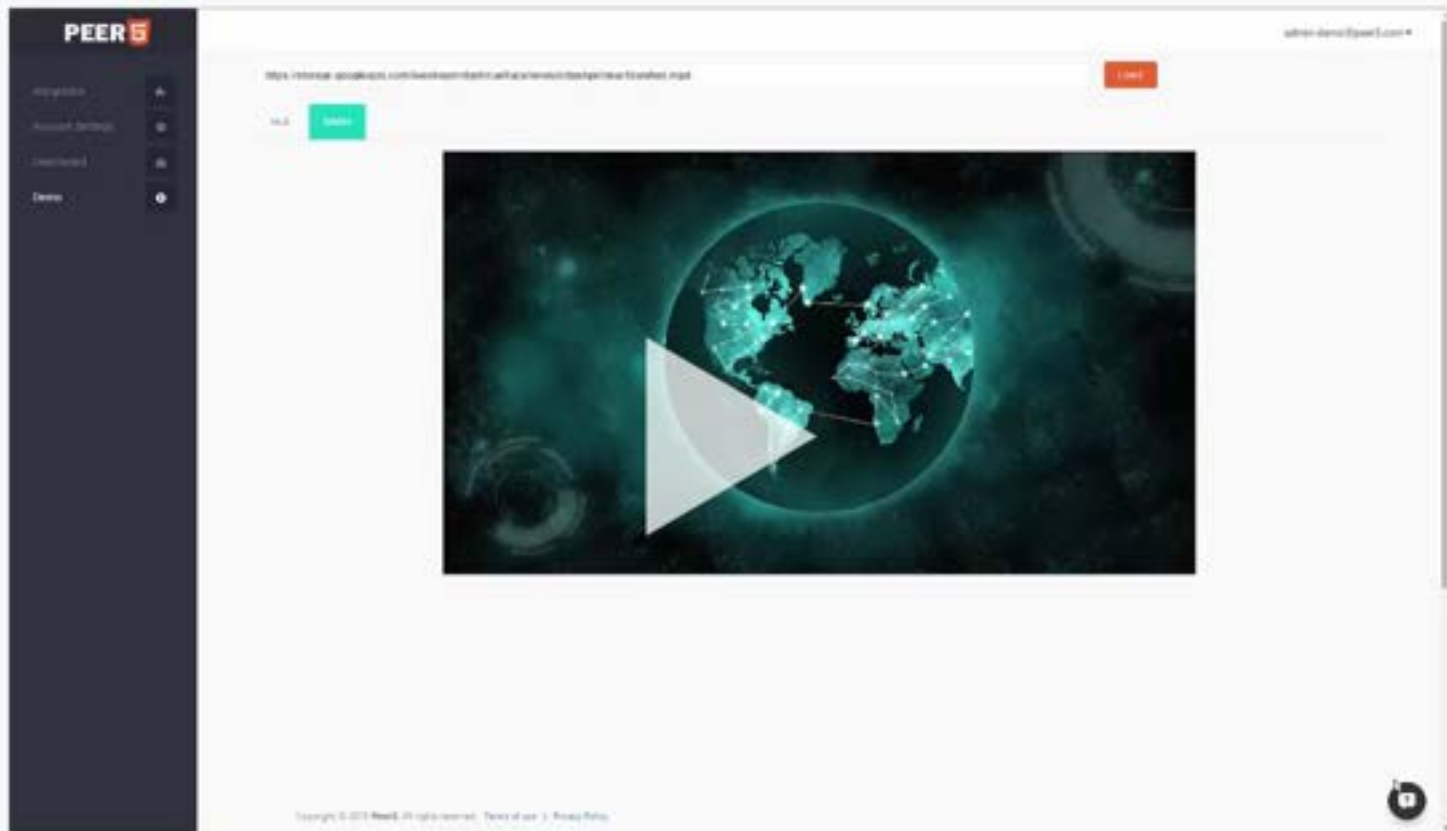


**STREAMROOT**

# How it works

- Provide video file on a server via HTML5 MSE
- Every user gets the video segment information from the server
- *Every user also turns into a server*
- All users share video file segments with each other via HTML5 MSE and WebRTC

## OUR VIDEO STREAMING CDN



Typical Live Streaming Video Results

Demo: <https://www.peer5.com/video?=-blg190215>

# P2P CDN goals

- Low-latency downloads
- Fast discovery: rely on server
- Robustness towards peak usage
- Less server infrastructure
- Less bandwidth use for publisher



# Challenges for P2P CDN approaches

Video consumer may object to

- Uncontrolled use of upstream bandwidth

QoS for video consumer may degrade

- Limited by upload speed of others

# 2. Webtorrent: Bittorrent for the Web

## Instant.io - Streaming file transfer over WebTorrent

Download/upload files using the [WebTorrent](#) protocol (BitTorrent over WebRTC). This is a beta.

### Start seeding

Drag-and-drop a file (or files) to begin sharing. Or choose a file:  No file chosen

### Start downloading

Download from a magnet link or info hash  \*

[WebTorrent](#) is powered by JavaScript and WebRTC. Works in Chrome, Firefox, and Opera (desktop and Android).

Code is available on [GitHub](#) under MIT License. Run `localStorage.debug = '*'` in the console and refresh to enable verbose logs.

# Webtorrent: How it works

- Provide file through browser (seed)
- Webtorrent creates the torrent file, i.e. a DHT of chunks of the file
- Share the magnet URI of the torrent file with others
- New users at the magnet URI start leeching and seeding

# Web/Bittorrent goals

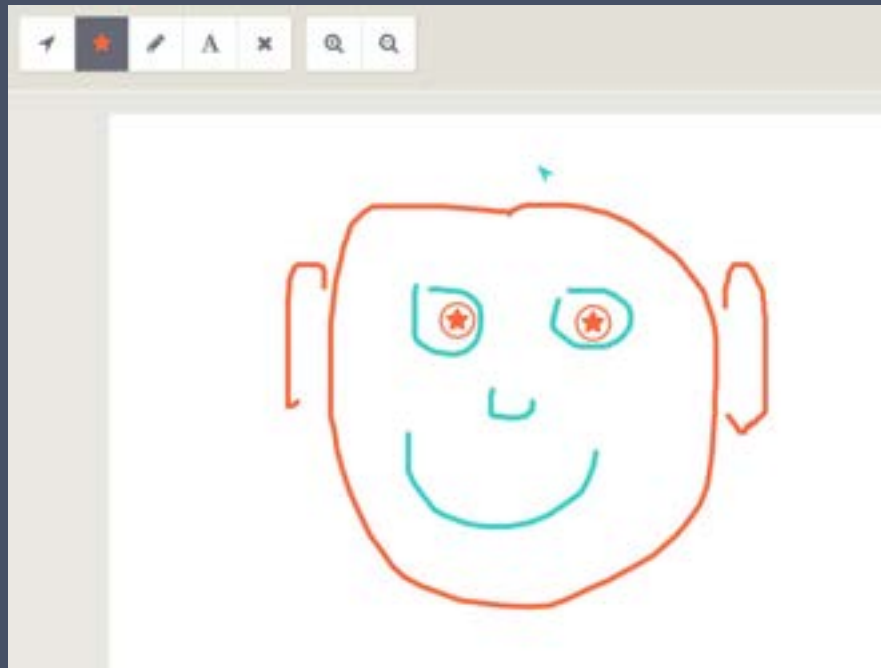
- No servers / pure P2P
- Sharing economy
- Parallel download instead of server limits
- Avoid server infrastructure
- Minimal bandwidth use for publisher



# Challenges for Webtorrent approaches

- User may object to uncontrolled use of upstream bandwidth (leads to freeriders)
- Need to leave browser open to continue seeding
- Distribution relies on a large number of participants
- Latency through distributed nodes
- Limited security – only publicly available content

# 3. P2P for Video Collaboration?



# • Traditional collaboration

- Edit token serialises annotations between endpoints
- Server keeps track of changes and holds ground truth
- Token request and release introduces
  - Document locking
  - Taking turns
  - Unresponsive UX

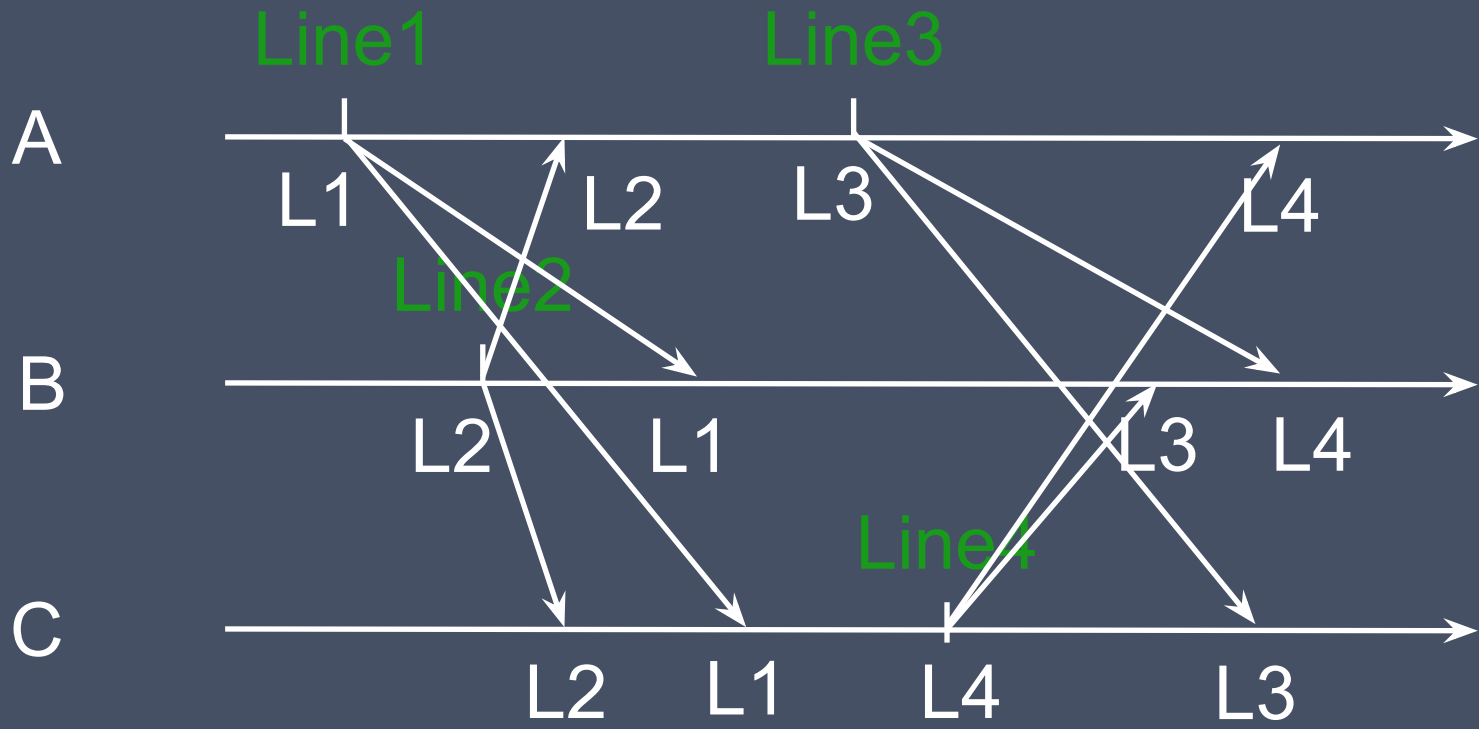
# Properties of P2P Collaboration

- Collaborate on annotations in parallel
- Unsynchronised clocks
- Reliable communication is not guaranteed
- Participants drop out and return, new ones join
- Eventually everyone needs to see the same state

# How to achieve consistency

- The order per endpoint matters: monotonically increasing system state
- Broadcast state update to everyone else (gossip protocol)
- The result between peers is commutative

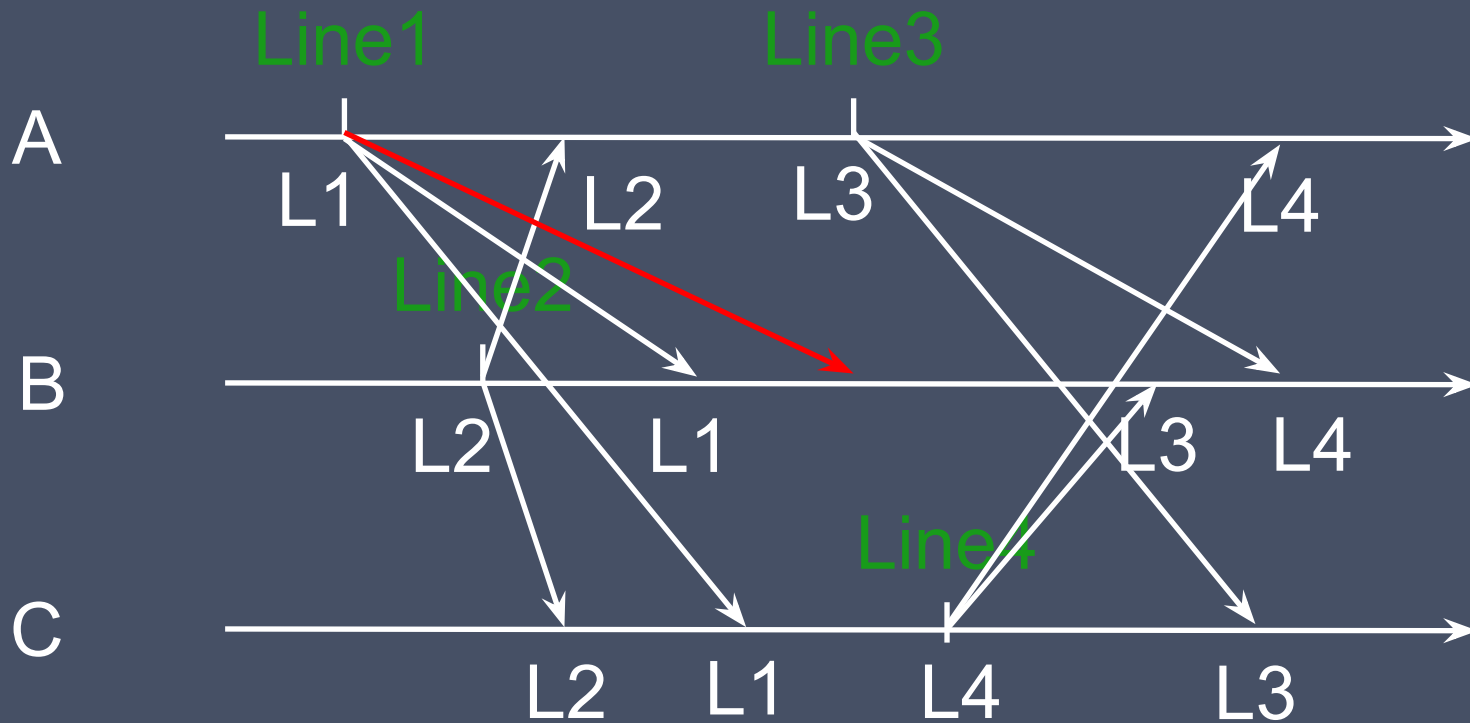
# CRDT = commutative replicated data type



Consistent shared state: {L1, L2, L3, L4}

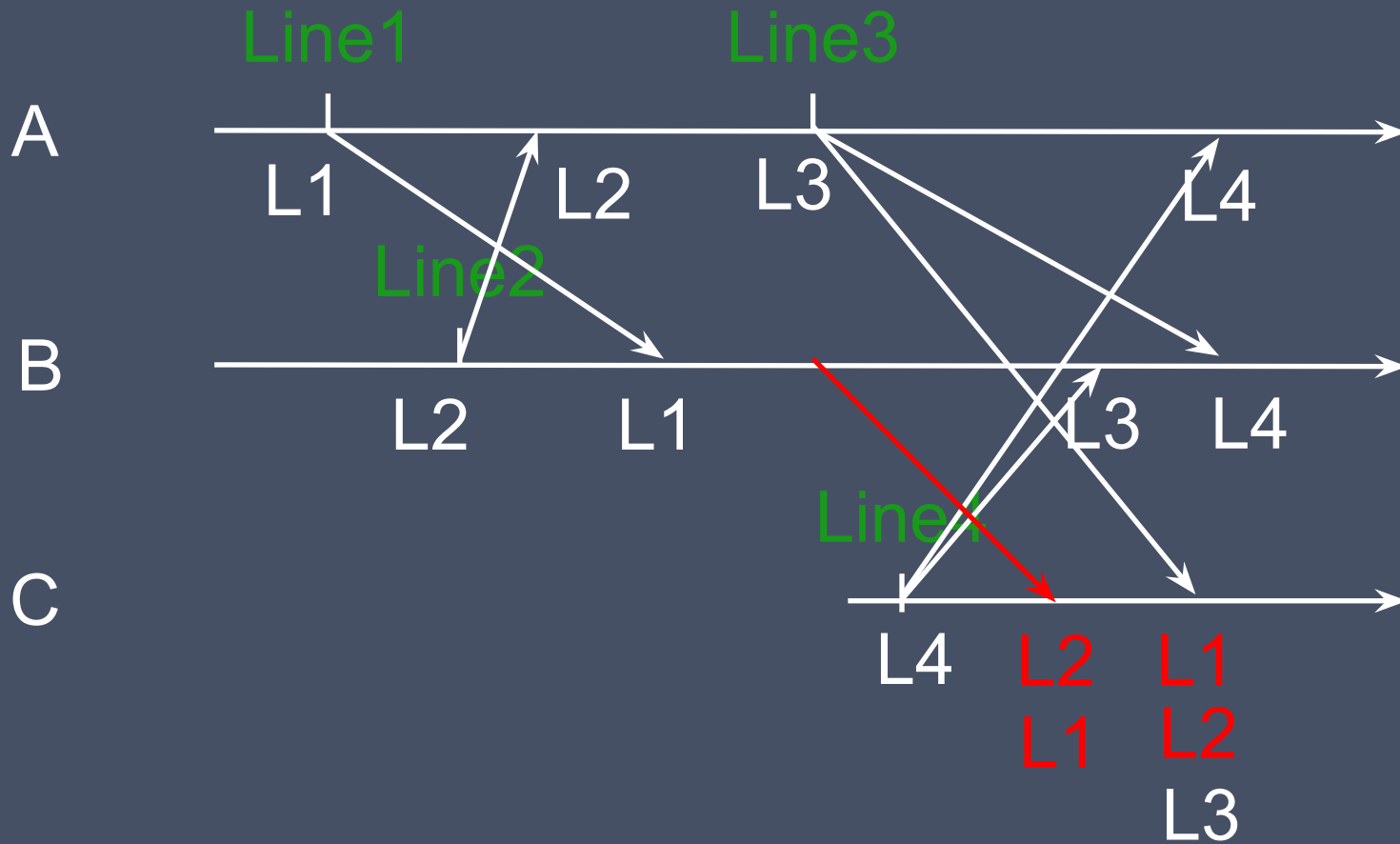


# Duplication detection



Consistent shared state: {L1, L2, L3, L4}

# Disconnect & Reconnect / Late connect



Consistent shared state:  $\{L1, L2, L3, L4\}$

# P2P Collaboration goals

- No servers / pure P2P
- Sharing in a safe environment: security and privacy
- Fastest way to share annotations
- Avoid server infrastructure
- Minimal bandwidth use for publisher

- P2P is a new paradigm on the Web:

# Summary

- File sharing
- Richer collaboration
- Advantages are clear:
  - Security & Privacy
  - Speed in parallelisation
  - Server-less introduces scalability
- Challenges:



# Silvia Pfeiffer

## Coviu

[silvia@coviu.com](mailto:silvia@coviu.com)

[@CoviuApp](#)  
[@silviapfeiffer](#)